

# **eTime Xpress Web Services API**

(For version 8.000.010)



**Celayix Software**  
Workforce Management Solutions

---

## TABLE OF CONTENTS

---

Introduction.....	6
Check Login.....	7
Employee data.....	8
Read Employees.....	8
Add/Update Employees.....	13
Delete Employees.....	17
Employee Qualification data.....	19
Read Employee Qualifications.....	19
Add/Update Employee Qualifications.....	24
Delete Employee Qualifications.....	27
Employee Call Log data.....	29
Read Employee Call Log Records.....	29
Add/Update Employee Call Log Records.....	33
Delete Employee Call Log Records.....	36
Customer data.....	38
Read Customers.....	38
Add/Update Customers.....	43
Delete Customers.....	47
Site data.....	49
Read Sites.....	49
Add/Update Sites.....	54
Delete Sites.....	58
Service Type data.....	60
Read Service Types.....	60
Add/Update Service Types.....	65
Delete Service Types.....	68
Location data.....	70
Read Locations.....	70
Add/Update Locations.....	75
Delete Locations.....	78

---

## TABLE OF CONTENTS

---

Shifts data.....	80
Read Shifts.....	80
Add/Update Shifts.....	83
Delete Shifts.....	88
Shifts Distribution data.....	91
Read Shifts Distributions.....	91
Codes Table data.....	94
Read Codes.....	94
Add/Update Codes.....	100
Delete Codes.....	103
Employee Time Off data.....	105
Read Employee Time Off Records.....	105
Add/Update Employee Time Off Records.....	108
Delete Employee Time Off Records.....	113
Volunteer data.....	115
Read Volunteer Records.....	115
Add/Update Volunteer Records.....	118
Delete Volunteer Records.....	122
Employee Availability data.....	125
Get Employee Availability.....	125
Employee Web Xpress Account data.....	132
Read Employee Web Xpress Accounts.....	132
Add/Update Employee Web Xpress Accounts.....	137
Delete Employee Web Xpress Accounts.....	140
Users data.....	142
Read Users.....	142
Add/Update Users.....	147
Delete Users.....	150
User Security Company/Branch Access data.....	152
Read User Security Company/Branch Access.....	152

---

## TABLE OF CONTENTS

---

Add/Update User Security Company/Branch Access.....	157
Delete User Security Company/Branch Access.....	160
User Security Customer/Site Access data.....	162
Read User Security Customer/Site Access.....	162
Add/Update User Security Customer/Site Access.....	167
Delete User Security Customer/Site Access.....	170
User Security Employee Access data.....	172
Read User Security Employee Access.....	172
Add/Update User Security Employee Access.....	177
Delete User Security Employee Access.....	180
User Security Groups data.....	182
Read User Security Groups.....	182
Add/Update User Security Groups.....	187
Delete User Security Groups.....	190
Security Groups data.....	192
Read Security Groups.....	192
Add/Update Security Groups.....	197
Delete Security Groups.....	200
Security Group Company/Branch Access data.....	202
Read Security Group Company/Branch Access.....	202
Add/Update Security Group Company/Branch Access.....	207
Delete Security Group Company/Branch Access.....	210
Security Group Customer/Site Access data.....	212
Read Security Group Customer/Site Access.....	212
Add/Update Security Group Customer/Site Access.....	217
Delete Security Group Customer/Site Access.....	220
Security Group Employee Access data.....	222
Read Security Group Employee Access.....	222
Add/Update Security Group Employee Access.....	227
Delete Security Group Employee Access.....	230

---

## TABLE OF CONTENTS

---

Pay Rates data.....	232
Read Pay Rates.....	232
Add/Update Pay Rates.....	235
Delete Pay Rates.....	239
Bill Rates data.....	242
Read Bill Rates.....	242
Add/Update Bill Rates.....	245
Delete Bill Rates.....	249
Shift Allowances data.....	252
Read Shift Allowances.....	252
Add/Update Shift Allowances.....	254
Delete Shift Allowances.....	257
Shifts Tasks data.....	259
Read Shifts Tasks.....	259
Shifts Events data.....	262
Read Shifts Events.....	262

---

## Introduction

---

The eTime Xpress Web Services API is a full featured interface to the eTime Xpress business logic. A developer can build applications that read, create, update and delete eTime information objects such as employees, customers, sites, services and shifts as well as execute eTime Xpress functions like employee availability.

This document is intended for developers and assumes you are familiar with software development, web services and eTime Xpress.

eTime Xpress Web Services API supports the following W3C specifications:

- WSDL - WSDL 1.1 is the de facto standard according to the W3C
- SOAP - SOAP 1.1 HTTP Binding is the de facto standard according to the W3C
- XML Schema - W3C 2001 XML Schema Recommendation

Additional eTime Xpress Web Services API documents:

- \* eTime Xpress Web Services API Datasets - Detailed information about the data used in the operations.
- \* eTime Xpress Web Services API Examples - SOAP request and response examples.

Steps to get started:

1. Review the documentation.
2. Get the most recent version of eTimeWS.wsdl file for your eTime Xpress version and generate Web references.
3. Execute the Check Login operation with correct username and password to get session id.
4. Use the generated session id to execute operations.
5. For each operation submit a synchronous request to the Web service, await the response, and process the results.

---

## Check Login

---

**Operation:** apSrv  
INPUT-OUTPUT dsContext

**Description:** This operation creates a new session in the system and returns its unique identifier (logs in the user in the system).

**NOTE:**

- If the specified user credentials (pcUserID and pcPassword) are not valid, the returned value of the pcSessionID parameter will be empty.
- The obtained value of the pcSessionID parameter should be used in all other Web Service operations as a required parameter.

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:**

See eTime Xpress Web Services API Examples - Check Login.

**User Security:** N/A

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	<b>Required</b>
pcPassword	Password string. <b>Values:</b> <any string>	<b>Required</b>
<b>OUTPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	
ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

**Errors:**

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1450	User ID and password combination does not match. Level: OPERATION
RUN_ERROR,1455	User ID is Inactive or Void. Unable to logon. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,RESET	Internal error used for login. Level: OPERATION

# Read Employees

**Operation:** apSrvEmpGet  
 INPUT-OUTPUT dsContext, OUTPUT dsEmployee

**Description:** This operation retrieves information about existing employees in the system.

ENTITY TABLE: dsEmployee.ttEmployee  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsEmployee Dataset  
 KEY FIELDS: cid, eid

Fields returned by this operation can be of the following three types:  
 - DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
 - CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
 - SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Employees.

**User Security:** Employee Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcStatus	Status Record Flag. Values: A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: employee.eid > 100 Values: <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by employee.emshort Values: <any string> Default: <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. Values: * - All DATA fields will be	Optional



# Read Employees

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields  <b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.  <b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.  <b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.  <b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.  <b>Values:</b> &lt;any positive integer value&gt;  <b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttEmployee.eid value that defines start record for a batch.            NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.  <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.  <b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.            NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key</p>	<b>Optional</b>

---

# Read Employees

---

## Parameters:

Name	Description	Type
------	-------------	------

---

### INPUT PARAMETERS

search to be performed, the `pcBatchSearchKey` parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:

1) **FIRST CHARACTER SEARCH:** The operation will try to find a first record (after the current record) whose `pcBatchSearchKey` field's value starts from a specified character. In this case the `pcBatchSearchKey` parameter value format is: `<TAB> + <TAB> + <Character to search> + <Current record key>`, where `<Current record key>` is the business object ID value for the current record and has the same format as `pcBatchKey`.

2) **MULTIPLE CHARACTERS SEARCH:** The operation will try to find a first record whose `pcBatchSearchKey` field value matches as many symbols as possible in the specified string. In this case the `pcBatchSearchKey` parameter value format is: `<String to search>`. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the `pcBatchSearchKey` field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.

**NOTE:** In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the `plBatchKeySearchOK` parameter, and the Batch Key value of the returned record will be returned in the `plBatchOneLetterSearchKey` parameter.

**Values:** `<any string>`  
**Default:** `<empty string>`

---

`pcBatchRepositionKeys`

A double `<TAB>`-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the `pcBatchKey` parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: `<pcBatchKey1> + <TAB> + <TAB> + <pcBatchKey2 > + ... + <TAB> + <TAB>+ <pcBatchKeyN>`. If `<pcBatchKey1>` fails, then the operation will try using `<pcBatchKey2>`, and so on, until the corresponding business object is found. **NOTE:** If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty `pcBatchKey` parameter value will be used by the present operation.

**Values:** `<any string>`  
**Default:** `N/A`

**Optional**

---

### OUTPUT PARAMETERS

`ExecuteStatus`

Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output `dsContext` dataset.

**Values:** See Errors part below

`pcIgnoredInServiceFieldsList`

A comma-separated list of fields that are restricted for the current user (`pcUserID`). Corresponding fields in the table will have default values.

**Values:** `<any string that corresponds to the description>`

`plBatchLast`

Last Record flag, which indicates whether the last record was reached in the current batch operation.

**Values:** YES - The last record was reached;  
NO - The last record was NOT reached.

---

# Read Employees

---

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for	

---

## Read Employees

---

**Errors:**

ID	Description
assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

# Add/Update Employees

**Operation:** apSrvEmpSave  
 INPUT-OUTPUT dsContext, INPUT-OUTPUT dsEmployee

**Description:** This operation creates new and/or updates information about existing employees in the system.

ENTITY TABLE: dsEmployee.ttEmployee  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsEmployee Dataset  
 KEY FIELDS: cid, eid  
 MANDATORY FIELDS: cid, bid, eid  
 IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
 - "A" - the ADD record mode  
 - "U" - the UPDATE record mode  
 - <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new employees (ADD record mode), the following conditions should be met:  
 - If Company settings Automatic Employee ID Assignment = NO, then the values of the KEY FIELDS should be unique in the system. If Automatic Employee ID Assignment = YES, then the system will assign automatically generated value to the eid field of new records, and ignore value passed in the input dataset.  
 - The emshort field of new records should be unique within the current Company.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Employees.

**User Security:** Employee Web Services --> ADD  
 Employee Web Services --> UPDATE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company).	Required

# Add/Update Employees

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Values:</b> 1..999	
<b>pcUpdateRequestFieldsList</b>	<p>"" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined.</p> <p><b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be updated;            &lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcUpdateExceptFieldsList</b>	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plImportEnableUpdate</b>	<p>A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field).</p> <p><b>Values:</b> YES - Allows to update existing records;            NO - Does not allow to update existing records.</p> <p><b>Default:</b> YES</p>	<b>Optional</b>
<b>plUseBeforeImage</b>	<p>Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset.</p> <p><b>Values:</b> YES - Enables the concurrency mode;            NO - Does not enable the concurrency mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>pcConcurrencyFieldList</b>	<p>A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>

## OUTPUT PARAMETERS

<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>
----------------------	---

## Errors:

ID	Description
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	List of context fields that are required for service and are not defined. Level: OPERATION
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION
<b>RUN_ERROR,1005</b>	Invalid entry. Level: RECORD
<b>RUN_ERROR,1015</b>	

---

## Add/Update Employees

---

### Errors:

ID	Description
	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1030	Start date cannot be greater than the End date. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1135	The software is not configured use. Contact customer support for assistance. Level: OPERATION
RUN_ERROR,1225	No records found. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1710	[%Employee%] ID cannot be 0. Level: RECORD
RUN_ERROR,1715	Records with a status of Archived cannot be created. Level: RECORD
RUN_ERROR,1720	This [%employee%] is archived. Status cannot be changed. Level: RECORD
RUN_ERROR,1725	Only void records can be archived. Level: RECORD
RUN_ERROR,1730,<Employee license count>	[%Employee%] license count of &1 has been exceeded unable to add new [%employee%]. Level: RECORD
RUN_ERROR,1735,<Employee license count>	Cannot restore. [%Employee%] license count of &1 has been exceeded. Level: RECORD
RUN_ERROR,1740,<Wait period in days>	Cannot restore. If the total number of employee records in the database exceed the licensed employee count, archived records cannot be restored for a period of &1 days. Level: RECORD
RUN_ERROR,1745,<Employee ID> +<TAB>+ <Employee Full Name> +<TAB> + <Company ID and Name> + <TAB> +<Branch ID and Name>	[%Employee%] ID &1 is already being used by &2 in [%company%] &3/[%branch%] &4. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD

---

## Add/Update Employees

---

### Errors:

ID	Description
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION



# Delete Employees

**Operation:** apSrvEmpSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsEmployee

**Description:** This operation deletes existing employees from the system.

ENTITY TABLE: dsEmployee.ttEmployee

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsEmployee Dataset

KEY FIELDS: cid, eid

MANDATORY FIELDS: cid, bid, eid

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it. If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Employees.

**User Security:** Employee Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

**OUTPUT PARAMETERS**

<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
----------------------	--

**Errors:**

ID	Description
----	-------------

---

## Delete Employees

---

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1195	Shifts for this [%employee%] exist in the schedule. Unable to delete. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1540	Record is currently being used in the system. Cannot delete. Level: RECORD
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Employee Qualifications

**Operation:** apSrvQualifGet  
INPUT-OUTPUT dsContext, OUTPUT dsQualif

**Description:** This operation retrieves information about existing employee qualifications from the system.

ENTITY TABLE: dsQualif.ttQualif  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsQualif Dataset  
KEY FIELDS: cid, eid, qualcd

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Employee Qualifications.

**User Security:** Employee Qualifications Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcStatus	Employee's Status Record Flag. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: qualif.effdate = TODAY <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by qualif.effdate <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be	Optional

# Read Employee Qualifications

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>read excluding pcReadExceptFieldsList fields;                      &lt;comma-separated list&gt; - List of DATA fields that will be read;                      &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields  <b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.  <b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.  <b>Values:</b> YES - Enables the Batch mode;                      NO - Does not enable the Batch mode.  <b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.  <b>Values:</b> &lt;any positive integer value&gt;  <b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is (ttQualif.eid + &lt;TAB&gt; + ttQualif.QualCd) value that defines start record for a batch.                      NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.  <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;                      PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;                      FIRST - First batch;                      LAST - Last batch.  <b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.                      NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key</p>	<b>Optional</b>

---

# Read Employee Qualifications

---

## Parameters:

Name	Description	Type
------	-------------	------

---

### INPUT PARAMETERS

search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:

1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: <TAB> + <TAB> + <Character to search> + <Current record key>, where <Current record key> is the business object ID value for the current record and has the same format as pcBatchKey.

2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: <String to search>. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.

NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.

**Values:** <any string>  
**Default:** <empty string>

---

**pcBatchRepositionKeys**

A double <TAB>-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: <pcBatchKey1> + <TAB> + <TAB> + <pcBatchKey2 > + ... + <TAB> + <TAB>+ <pcBatchKeyN>. If <pcBatchKey1> fails, then the operation will try using <pcBatchKey2>, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.

**Values:** <any string>  
**Default:** N/A

**Optional**

---

### OUTPUT PARAMETERS

**ExecuteStatus**

Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.

**Values:** See Errors part below

**pcIgnoredInServiceFieldsList**

A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.

**Values:** <any string that corresponds to the description>

**plBatchLast**

Last Record flag, which indicates whether the last record was reached in the current batch operation.

**Values:** YES - The last record was reached;  
NO - The last record was NOT reached.

# Read Employee Qualifications

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for	

---

## Read Employee Qualifications

---

**Errors:**

ID	Description
assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Add/Update Employee Qualifications

**Operation:** apSrvQualifSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsQualif

**Description:** This operation creates new and/or updates information about existing employee qualifications in the system.

ENTITY TABLE: dsQualif.ttQualif  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsQualif Dataset  
KEY FIELDS: cid, eid, qualcd  
MANDATORY FIELDS: cid, bid, eid, qualcd  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
- "A" - the ADD record mode  
- "U" - the UPDATE record mode  
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new employee qualifications (ADD record mode), the following condition should be met:  
- The values of the KEY FIELDS should be unique in the system.

ERROR HANDLING:  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Employee Qualifications.

**User Security:** Employee Qualifications Web Services --> ADD  
Employee Qualifications Web Services --> UPDATE

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined.	Optional



# Add/Update Employee Qualifications

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p><b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be updated;            &lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
pcUpdateExceptFieldsList	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	Optional
plImportEnableUpdate	<p>A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field).</p> <p><b>Values:</b> YES - Allows to update existing records;            NO - Does not allow to update existing records.  <b>Default:</b> YES</p>	Optional
plUseBeforeImage	<p>Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset.</p> <p><b>Values:</b> YES - Enables the concurrency mode;            NO - Does not enable the concurrency mode.  <b>Default:</b> NO</p>	Optional
pcConcurrencyFieldList	<p>A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	Optional

## OUTPUT PARAMETERS

ExecuteStatus	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>
---------------	---

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD

---

## Add/Update Employee Qualifications

---

### Errors:

ID	Description
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1555	Effective date cannot be greater than the expiry date. Level: RECORD
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete Employee Qualifications

**Operation:** apSrvQualifSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsQualif

**Description:** This operation deletes existing employee qualifications from the system.

ENTITY TABLE: dsQualif.ttQualif  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsQualif Dataset  
KEY FIELDS: cid, eid, qualcd  
MANDATORY FIELDS: cid, bid, eid, qualcd

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Employee Qualifications.

**User Security:** Employee Qualifications Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

**Errors:**

ID	Description
----	-------------

---

## Delete Employee Qualifications

---

### Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
<b>NO_CONTEXT_DATA</b> ,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION	
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION	
<b>RUN_ERROR,1025</b>	Record is currently locked. Level: RECORD	
<b>RUN_ERROR,1065</b>	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD	
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level: OPERATION, RECORD	
<b>RUN_ERROR,1435</b>	This function is not licensed. Level: OPERATION	
<b>RUN_ERROR,1685</b>	Invalid [%Company%] / [%Branch%] combination. Level: RECORD	
<b>RUN_ERROR,1700</b>	Invalid [%Employee%] ID. Level: RECORD	
<b>RUN_ERROR,2185</b>	Record has been changed by another user or system process. Refresh recommended. Level: RECORD	
<b>RUN_ERROR,INVALID_CONTEXT_DATA</b> ,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION	
<b>RUN_ERROR,LOCKED_CONTEXT_DATA</b>	Context record is currently locked. Level: OPERATION	
<b>RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)</b>	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION	
<b>RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)</b>	Invalid Session ID. Level: OPERATION	
<b>RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)</b>	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION	

# Read Employee Call Log Records

**Operation:** apSrvCallLogGet  
 INPUT-OUTPUT dsContext, OUTPUT dsCallLog

**Description:** This operation retrieves information about existing employee call log records from the system.

ENTITY TABLE: dsCallLog.ttCallLog  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsCallLog Dataset  
 KEY FIELDS: callid

Fields returned by this operation can be of the following three types:

- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and
- SYSTEM FIELDS, required for system purposes.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.

EXAMPLE: See eTime Xpress Web Services API Examples - Read Employee Call Log Records.

**User Security:** Call Log Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: calllog.called <> "CD" Values: <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by calllog.called Values: <any string> Default: <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. Values: * - All DATA fields will be read excluding pcReadExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be read; <empty string> - All DATA fields will be read excluding pcReadExceptFieldsList fields Default: <empty string>	Optional

# Read Employee Call Log Records

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcReadExceptFieldsList</b>	A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>plBatch</b>	Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit<>0. <b>Values:</b> YES - Enables the Batch mode; NO - Does not enable the Batch mode. <b>Default:</b> NO	<b>Optional</b>
<b>piBatchLimit</b>	The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one. <b>Values:</b> <any positive integer value> <b>Default:</b> 0	<b>Optional</b>
<b>pcBatchKey</b>	A Batch Key is ttCalllog.callid value that defines start record for a batch. NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description). <b>Values:</b> <any string> <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcBatchDirection</b>	Direction of the batch. <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST; PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST; FIRST - First batch; LAST - Last batch. <b>Default:</b> NEXT	<b>Optional</b>
<b>pcBatchSearchKey</b>	A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: <table>.<field>, where <field> - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter. NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch. <b>Values:</b> <any string> <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcBatchSearchKeyValue</b>	A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used: 1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified	<b>Optional</b>

# Read Employee Call Log Records

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	
pcBatchRepositionKeys	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2 &gt; + ... + &lt;TAB&gt; + &lt;TAB&gt;+ &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> N/A</p>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
pcIgnoredInServiceFieldsList	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
plBatchLast	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p> <p><b>Values:</b> YES - The last record was reached;  NO - The last record was NOT reached.</p>	
plBatchKeySearchOK	<p>Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful.</p> <p><b>Values:</b> YES - The record was found;  NO - The record was NOT found.</p>	

# Read Employee Call Log Records

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
<code>plBatchReturnedExtraData</code>	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if <code>pcBatchDirection = "NEXT"</code> and there are less than <code>piBatchLimit-1</code> records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
<code>plBatchOneLetterSearchKey</code>	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
<code>NO_CONTEXT_DATA,</code> <comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
<code>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</code>	More than one service is set to be run by one call. Level: OPERATION
<code>RUN_ERROR,1060</code>	Record is not available. Level: RECORD
<code>RUN_ERROR,1240</code>	You do not have security privileges to access this function. Level: OPERATION, RECORD
<code>RUN_ERROR,1435</code>	This function is not licensed. Level: OPERATION
<code>RUN_ERROR,1965</code>	There are wrong fields in the requested list: &1. Level: RECORD
<code>RUN_ERROR,1980,</code> <query string>	Invalid QUERY string: &1. Level: OPERATION
<code>RUN_ERROR,INVALID_CONTEXT_DATA,</code> <Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
<code>RUN_ERROR,LOCKED_CONTEXT_DATA</code>	Context record is currently locked. Level: OPERATION
<code>RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)</code>	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
<code>RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)</code>	Invalid Session ID. Level: OPERATION
<code>RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)</code>	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION



## Add/Update Employee Call Log Records

**Operation:** apSrvCallLogSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsCallLog

**Description:** This operation creates new and/or updates information about existing employee call log records in the system.

ENTITY TABLE: dsCallLog.ttCallLog  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsCallLog Dataset  
 KEY FIELDS: callid  
 MANDATORY FIELDS: cid, bid, callid  
 IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
 - "A" - the ADD record mode  
 - "U" - the UPDATE record mode  
 - <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: The system will assign automatically generated values to the Call Log ID (callid) field of new records.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Employee Call Log Records.

**User Security:** Call Log Web Services --> ADD  
 Call Log Web Services --> UPDATE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. Values: * - All DATA fields will be	Optional

# Add/Update Employee Call Log Records

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
plImportEnableUpdate	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records. <b>Default:</b> YES	Optional
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

## OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	

---

## Add/Update Employee Call Log Records

---

### Errors:

ID	Description
	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete Employee Call Log Records

**Operation:** apSrvCallLogSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsCallLog

**Description:** This operation deletes existing employee call log records from the system.

ENTITY TABLE: dsCallLog.ttCallLog  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsCallLog Dataset  
 KEY FIELDS: callid  
 MANDATORY FIELDS: cid, bid, callid

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete Employee Call Log Records.

**User Security:** Call Log Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

**Errors:**

ID	Description
----	-------------

---

## Delete Employee Call Log Records

---

**Parameters:**

<b>Name</b>	<b>Description</b>	<b>Type</b>
<b>OUTPUT PARAMETERS</b>		
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	List of context fields that are required for service and are not defined. Level: OPERATION	
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION	
<b>RUN_ERROR,1025</b>	Record is currently locked. Level: RECORD	
<b>RUN_ERROR,1065</b>	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD	
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level: OPERATION, RECORD	
<b>RUN_ERROR,1435</b>	This function is not licensed. Level: OPERATION	
<b>RUN_ERROR,1685</b>	Invalid [%Company%] / [%Branch%] combination. Level: RECORD	
<b>RUN_ERROR,1700</b>	Invalid [%Employee%] ID. Level: RECORD	
<b>RUN_ERROR,2185</b>	Record has been changed by another user or system process. Refresh recommended. Level: RECORD	
<b>RUN_ERROR,INVALID_CONTEXT_DATA,&lt;Context Parameter Name&gt; + &lt;TAB&gt; + &lt;Comma-separated list of available values&gt;</b>	The &l parameter's value must be one of the following: &2. Level: OPERATION	
<b>RUN_ERROR,LOCKED_CONTEXT_DATA</b>	Context record is currently locked. Level: OPERATION	
<b>RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)</b>	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION	
<b>RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)</b>	Invalid Session ID. Level: OPERATION	
<b>RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)</b>	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION	

# Read Customers

**Operation:** apSrvCustGet  
 INPUT-OUTPUT dsContext, OUTPUT dsCustomer

**Description:** This operation retrieves information about existing customers from the system.

ENTITY TABLE: dsCustomer.ttCustomer  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsCustomer Dataset  
 KEY FIELDS: cid, bid, uid

Fields returned by this operation can be of the following three types:  
 - DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
 - CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
 - SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Customers.

**User Security:** Customer Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcStatus	Status Record Flag. Values: A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: customer.uid > 100 Values: <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by customer.csname Values: <any string> Default: <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. Values: * - All DATA fields will be	Optional

# Read Customers

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>read excluding pcReadExceptFieldsList fields;                      &lt;comma-separated list&gt; - List of DATA fields that will be read;                      &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields  <b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.  <b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.  <b>Values:</b> YES - Enables the Batch mode;                      NO - Does not enable the Batch mode.  <b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.  <b>Values:</b> &lt;any positive integer value&gt;  <b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttCustomer.uid value that defines start record for a batch.                      NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.  <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;                      PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;                      FIRST - First batch;                      LAST - Last batch.  <b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.                      NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key</p>	<b>Optional</b>

---

## Read Customers

---

### Parameters:

Name	Description	Type
------	-------------	------

---

#### INPUT PARAMETERS

search to be performed, the `pcBatchSearchKey` parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:

1) **FIRST CHARACTER SEARCH:** The operation will try to find a first record (after the current record) whose `pcBatchSearchKey` field's value starts from a specified character. In this case the `pcBatchSearchKeyValue` parameter value format is: `<TAB> + <TAB> + <Character to search> + <Current record key>`, where `<Current record key>` is the business object ID value for the current record and has the same format as `pcBatchKey`.

2) **MULTIPLE CHARACTERS SEARCH:** The operation will try to find a first record whose `pcBatchSearchKey` field value matches as many symbols as possible in the specified string. In this case the `pcBatchSearchKeyValue` parameter value format is: `<String to search>`. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the `pcBatchSearchKey` field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.

**NOTE:** In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the `plBatchKeySearchOK` parameter, and the Batch Key value of the returned record will be returned in the `plBatchOneLetterSearchKey` parameter.

**Values:** `<any string>`  
**Default:** `<empty string>`

---

`pcBatchRepositionKeys`

A double `<TAB>`-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the `pcBatchKey` parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: `<pcBatchKey1> + <TAB> + <TAB> + <pcBatchKey2 > + ... + <TAB> + <TAB>+ <pcBatchKeyN>`. If `<pcBatchKey1>` fails, then the operation will try using `<pcBatchKey2>`, and so on, until the corresponding business object is found. **NOTE:** If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty `pcBatchKey` parameter value will be used by the present operation.

**Values:** `<any string>`  
**Default:** `N/A`

**Optional**

---

#### OUTPUT PARAMETERS

`ExecuteStatus`

Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output `dsContext` dataset.

**Values:** See Errors part below

`pcIgnoredInServiceFieldsList`

A comma-separated list of fields that are restricted for the current user (`pcUserID`). Corresponding fields in the table will have default values.

**Values:** `<any string that corresponds to the description>`

`plBatchLast`

Last Record flag, which indicates whether the last record was reached in the current batch operation.

**Values:** YES - The last record was reached;  
NO - The last record was NOT reached.



---

# Read Customers

---

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for	

---

## Read Customers

---

**Errors:**

ID	Description
assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Add/Update Customers

---

**Operation:** apSrvCustSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsCustomer

**Description:** This operation creates new and/or updates information about existing customers in the system.

ENTITY TABLE: dsCustomer.ttCustomer  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsCustomer Dataset  
KEY FIELDS: cid, bid, uid  
MANDATORY FIELDS: cid, bid, uid  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:

- "A" - the ADD record mode
- "U" - the UPDATE record mode
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new customers (ADD record mode), the following conditions should be met:

- If Company settings Automatic Customer ID Assignment = NO, then the values of the KEY FIELDS should be unique in the system. If Automatic Customer ID Assignment = YES, then the system will assign automatically generated value to the uid field of new records, and ignore value passed in the input dataset.
- The csshort field of new records should be unique within the current Company.

Setting the value of the Bill address same as above field to YES will not copy values from the Contact Address to the corresponding Bill Address fields. This field can be used only as an indicator that this assignment was performed before.

### ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it. If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Customers.

**User Security:** Customer Web Services --> ADD  
Customer Web Services --> UPDATE

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required

# Add/Update Customers

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcUserID</b>	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	<b>Required</b>
<b>piCompanyID</b>	Unique Company ID. <b>Values:</b> 1..999	<b>Required</b>
<b>piBranchID</b>	Branch ID (unique within the Company). <b>Values:</b> 1..999	<b>Required</b>
<b>pcUpdateRequestFieldsList</b>	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcUpdateExceptFieldsList</b>	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>plImportEnableUpdate</b>	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records. <b>Default:</b> YES	<b>Optional</b>
<b>plUseBeforeImage</b>	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	<b>Optional</b>
<b>pcConcurrencyFieldList</b>	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	
<b>plSiteRefreshRequired</b>	Flag that indicates that site information is changed during update. <b>Values:</b> YES - Site's data were changed. Refresh Site data is recommended; <others values> - No changes.	<b>Optional</b>

## Errors:

ID	Description
----	-------------

# Add/Update Customers

**Parameters:**

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION	
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION	
RUN_ERROR,1005	Invalid entry. Level: RECORD	
RUN_ERROR,1015	Required entry. Level: RECORD	
RUN_ERROR,1025	Record is currently locked. Level: RECORD	
RUN_ERROR,1030	Start date cannot be greater than the End date. Level: RECORD	
RUN_ERROR,1050	Since date cannot be greater than the End date. Level: RECORD	
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD	
RUN_ERROR,1225	No records found. Level: OPERATION	
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD	
RUN_ERROR,1435	This function is not licensed. Level: OPERATION	
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD	
RUN_ERROR,1705	Invalid [%Customer%] ID. Level: RECORD	
RUN_ERROR,1755,	<Fields Label> + <TAB> + <Max field's size> The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD	
RUN_ERROR,1760,	<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value> The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD	
RUN_ERROR,1765,	<Fields Label> + <TAB> + <Comma-separated list of available values> The &1 field's value must be one of the following: &2. Level: RECORD	
RUN_ERROR,1870	[%Customer%] ID cannot be 0. Level: RECORD	
RUN_ERROR,1875,	<Customer ID> +<TAB>+ <Customer Short Name> +<TAB> + <Company ID and Name> + <TAB> +<Branch ID and Name> [%Customer%] ID &1 is already being used by &2 in [%company%] &3/[%branch%] &4. Level: RECORD	
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD	

---

## Add/Update Customers

---

### Errors:

ID	Description
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

# Delete Customers

**Operation:** apSrvCustSave  
 INPUT-OUTPUT dsContext, INPUT-OUTPUT dsCustomer

**Description:** This operation deletes existing customers from the system.

ENTITY TABLE: dsCustomer.ttCustomer

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsCustomer Dataset

KEY FIELDS: cid, bid, uid

MANDATORY FIELDS: cid, bid, uid

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Customers.

**User Security:** Customer Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

**OUTPUT PARAMETERS**

<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
----------------------	--

**Errors:**

ID	Description
----	-------------

---

## Delete Customers

---

### Errors:

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1270	There are [%sites%] defined for this [%customer%]. Unable to delete. Level: RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1540	Record is currently being used in the system. Cannot delete. Level: RECORD
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1705	Invalid [%Customer%] ID. Level: RECORD
RUN_ERROR,2180,<WEBX Defaults table>	Web Xpress &l are configured to use this [%customer%]. Unable to delete. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION



# Read Sites

**Operation:** apSrvSiteGet  
 INPUT-OUTPUT dsContext, OUTPUT dsSite

**Description:** This operation retrieves information about existing sites from the system.

ENTITY TABLE: dsSite.ttSite  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsSite Dataset  
 KEY FIELDS: cid, bid, uid, sid

Fields returned by this operation can be of the following three types:  
 - DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
 - CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
 - SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Sites.

**User Security:** Site Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	<b>Required</b>
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	<b>Required</b>
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	<b>Required</b>
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	<b>Required</b>
pcStatus	Status Record Flag. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	<b>Required</b>
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: site.sid > 100 <b>Values:</b> <any string>	<b>Required</b>
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by site.psname <b>Values:</b> <any string> <b>Default:</b> <empty string>	<b>Optional</b>
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields;	<b>Optional</b>

## Read Sites

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>&lt;comma-separated list&gt; - List of DATA fields that will be read;</p> <p>&lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>piCustomerID</b>	<p>Customer ID. It is unique for Company ID/Branch ID.</p> <p><b>Values:</b> 1..999999</p> <p><b>Default:</b> N/A</p>	<b>Optional</b>
<b>piSiteID</b>	<p>Site ID. It is unique for Company ID/Branch ID/Customer ID.</p> <p><b>Values:</b> 1..9999</p> <p><b>Default:</b> N/A</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.</p> <p><b>Values:</b> YES - Enables the Batch mode; NO - Does not enable the Batch mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.</p> <p><b>Values:</b> &lt;any positive integer value&gt;</p> <p><b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is (ttSite.uid + &lt;TAB&gt; + ttSite.sid) value that defines start record for a batch.</p> <p>NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.</p> <p><b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST; PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST; FIRST - First batch; LAST - Last batch.</p> <p><b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.</p> <p>NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the</p>	<b>Optional</b>

---

## Read Sites

---

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> &lt;empty string&gt;</p>	
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchRepositionKeys</b>	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2 &gt; + ... + &lt;TAB&gt; + &lt;TAB&gt;+ &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> N/A</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
<b>pcIgnoredInServiceFieldsList</b>	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the</p>	

# Read Sites

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
	table will have default values. <b>Values:</b> <any string that corresponds to the description>	
plBatchLast	Last Record flag, which indicates whether the last record was reached in the current batch operation. <b>Values:</b> YES - The last record was reached; NO - The last record was NOT reached.	
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &l. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user.	

---

## Read Sites

---

### Errors:

ID	Description
(#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Add/Update Sites

---

**Operation:** apSrvSiteSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsSite

**Description:** This operation creates new and/or updates information about existing sites in the system.

ENTITY TABLE: dsSite.ttSite  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsSite Dataset  
KEY FIELDS: cid, bid, uid, sid  
MANDATORY FIELDS: cid, bid, uid, sid  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
- "A" - the ADD record mode  
- "U" - the UPDATE record mode  
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new sites (ADD record mode), the following conditions should be met:

- If Company settings Automatic Site ID Assignment = NO, then the values of the KEY FIELDS should be unique in the system. If Automatic Site ID Assignment = YES, then the system will assign automatically generated value to the sid field of new records, and ignore value passed in the input dataset.
- The psshort field of new records should be unique within the current Company

Setting the value of the Bill address same as above field to YES will not copy values from the Contact Address to the corresponding Bill Address fields. This field can be used only as an indicator that this assignment was performed before.

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it. If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Sites.

**User Security:** Site Web Services --> ADD  
Site Web Services --> UPDATE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required

## Add/Update Sites

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	Optional
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
plImportEnableUpdate	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records. <b>Default:</b> YES	Optional
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

### OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	

---

## Add/Update Sites

---

### Errors:

ID	Description
	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1030	Start date cannot be greater than the End date. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1225	No records found. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1665	Invalid [%Customer%] / [%Site%] combination. Level: RECORD
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1880	Invalid [%Customer%] Status / [%Site%] Status combination. Level: RECORD
RUN_ERROR,1885,<Customer ID> + <TAB> + <Company Id and Name>	[%Customer%] ID &1 is not valid for [%Company%]. Level: RECORD
RUN_ERROR,1890	[%Site%] ID cannot be 0. Level: RECORD
RUN_ERROR,1895,<Site ID> + <TAB> + <Site Name>	[%Site%] ID &1 is already being used by &2. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level:



---

## Add/Update Sites

---

### Errors:

ID	Description
	OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Delete Sites

---

**Operation:** apSrvSiteSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsSite

**Description:** This operation deletes existing sites from the system.

ENTITY TABLE: dsSite.ttSite

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsSite Dataset

KEY FIELDS: cid, bid, uid, sid

MANDATORY FIELDS: cid, bid, uid, sid

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it. If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Sites.

**User Security:** Site Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

**OUTPUT PARAMETERS**

<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
----------------------	--

**Errors:**

ID	Description
----	-------------

---

---

## Delete Sites

---

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1265	There are shifts scheduled at this [%site%]. Unable to delete. Level: RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1540	Record is currently being used in the system. Cannot delete. Level: RECORD
RUN_ERROR,1665	Invalid [%Customer%] / [%Site%] combination. Level: RECORD
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,2175,<WEBX Defaults table>	Web Xpress &l are configured to use this [%site%]. Unable to delete. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Service Types

**Operation:** apSrvSrvTypeGet  
INPUT-OUTPUT dsContext, OUTPUT dsSrvType

**Description:** This operation retrieves information about existing service types from the system.

ENTITY TABLE: dsSrvType.ttSrvType

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsSrvType Dataset

KEY FIELDS: srvtycd

Fields returned by this operation can be of the following three types:

- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and
- SYSTEM FIELDS, required for system purposes.

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.

EXAMPLE: See eTime Xpress Web Services API Examples - Read Service Types.

**User Security:** Service Type Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcStatus	Status Record Flag. Values: A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: srvtype.srvtycd <> "A" Values: <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by srvtype.srvtycd Values: <any string> Default: <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. Values: * - All DATA fields will be	Optional

# Read Service Types

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields  <b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.  <b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.  <b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.  <b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.  <b>Values:</b> &lt;any positive integer value&gt;  <b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttSrvType.srvtycd value that defines start record for a batch.            NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.  <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.  <b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.            NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key</p>	<b>Optional</b>

# Read Service Types

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	
pcBatchRepositionKeys	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2 &gt; + ... + &lt;TAB&gt; + &lt;TAB&gt;+ &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> N/A</p>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
pcIgnoredInServiceFieldsList	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
plBatchLast	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p> <p><b>Values:</b> YES - The last record was reached;  NO - The last record was NOT reached.</p>	

# Read Service Types

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &l. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for	

---

## Read Service Types

---

**Errors:**

ID	Description
assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION



## Add/Update Service Types

**Operation:** apSrvSrvTypeSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsSrvType

**Description:** This operation creates new and/or updates information about existing service types in the system.

ENTITY TABLE: dsSrvType.ttSrvType  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsSrvType Dataset  
KEY FIELDS: srvtycd  
MANDATORY FIELDS: cid, bid, srvtycd  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:

- "A" - the ADD record mode
- "U" - the UPDATE record mode
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new service types (ADD record mode), the following condition should be met:

- The srvtycd field of new records should be unique within the current Company/Branch

The system will assign automatically generated values to the Service Type ID (srvtyid) fields of new records.

### ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Service Types.

**User Security:** Service Type Web Services --> ADD  
Service Type Web Services --> UPDATE

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that	Optional

## Add/Update Service Types

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined.</p> <p><b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be updated;            &lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
pcUpdateExceptFieldsList	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	Optional
plImportEnableUpdate	<p>A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field).</p> <p><b>Values:</b> YES - Allows to update existing records;            NO - Does not allow to update existing records.</p> <p><b>Default:</b> YES</p>	Optional
plUseBeforeImage	<p>Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset.</p> <p><b>Values:</b> YES - Enables the concurrency mode;            NO - Does not enable the concurrency mode.</p> <p><b>Default:</b> NO</p>	Optional
pcConcurrencyFieldList	<p>A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	Optional

### OUTPUT PARAMETERS

ExecuteStatus	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>
---------------	---

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	<p>List of context fields that are required for service and are not defined.            Level: OPERATION</p>
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	<p>More than one service is set to be run by one call. Level: OPERATION</p>
RUN_ERROR,1005	<p>Invalid entry. Level: RECORD</p>
RUN_ERROR,1020	<p>Record already exists. Level: RECORD</p>
RUN_ERROR,1025	

---

## Add/Update Service Types

---

### Errors:

ID	Description
	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1905	[%Service%] ID cannot be blank. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete Service Types

**Operation:** apSrvSrvTypeSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsSrvType

**Description:** This operation deletes existing service types from the system.

ENTITY TABLE: dsSrvType.ttSrvType

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsSrvType Dataset

KEY FIELDS: srvtycd

MANDATORY FIELDS: cid, bid, srvtycd

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it. If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Service Types.

**User Security:** Service Type Web Services --> DELETE

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

### OUTPUT PARAMETERS

<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
----------------------	--

### Errors:

ID	Description
----	-------------

---

## Delete Service Types

---

### Errors:

ID	Description
NO_CONTEXT_DATA,	<b>&lt;comma separated list of missed context data&gt;</b> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1900,	<b>&lt;Service Type ID&gt; + &lt;TAB&gt; + &lt;Table description&gt; + &lt;TAB&gt; + &lt;Service Record status&gt;</b> [%Service%] code &1 is being used. Error: &2 record found. Record can not be deleted. Note: Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,	<b>&lt;Context Parameter Name&gt; + &lt;TAB&gt; + &lt;Comma-separated list of available values&gt;</b> The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

# Read Locations

**Operation:** apSrvLocationGet  
 INPUT-OUTPUT dsContext, OUTPUT dsLocation

**Description:** This operation retrieves information about existing locations from the system.

ENTITY TABLE: dsLocation.ttLocation  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsLocation Dataset  
 KEY FIELDS: LocationObj

Fields returned by this operation can be of the following three types:  
 - DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
 - CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
 - SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Locations.

**User Security:** Location Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcStatus	Status Record Flag. Values: A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: Location.LocationName BEGINS "A" Values: <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by Location.LocationName Values: <any string> Default: <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. Values: * - All DATA fields will be	Optional

# Read Locations

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields  <b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.  <b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.  <b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.  <b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.  <b>Values:</b> &lt;any positive integer value&gt;  <b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttLocation.LocationObj value that defines start record for a batch.            NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.  <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.  <b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.            NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key</p>	<b>Optional</b>

---

## Read Locations

---

### Parameters:

Name	Description	Type
------	-------------	------

---

#### INPUT PARAMETERS

search to be performed, the `pcBatchSearchKey` parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:

1) **FIRST CHARACTER SEARCH:** The operation will try to find a first record (after the current record) whose `pcBatchSearchKey` field's value starts from a specified character. In this case the `pcBatchSearchKeyValue` parameter value format is: `<TAB> + <TAB> + <Character to search> + <Current record key>`, where `<Current record key>` is the business object ID value for the current record and has the same format as `pcBatchKey`.

2) **MULTIPLE CHARACTERS SEARCH:** The operation will try to find a first record whose `pcBatchSearchKey` field value matches as many symbols as possible in the specified string. In this case the `pcBatchSearchKeyValue` parameter value format is: `<String to search>`. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the `pcBatchSearchKey` field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.

**NOTE:** In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the `plBatchKeySearchOK` parameter, and the Batch Key value of the returned record will be returned in the `plBatchOneLetterSearchKey` parameter.

**Values:** `<any string>`  
**Default:** `<empty string>`

---

`pcBatchRepositionKeys`

A double `<TAB>`-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the `pcBatchKey` parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: `<pcBatchKey1> + <TAB> + <TAB> + <pcBatchKey2 > + ... + <TAB> + <TAB>+ <pcBatchKeyN>`. If `<pcBatchKey1>` fails, then the operation will try using `<pcBatchKey2>`, and so on, until the corresponding business object is found. **NOTE:** If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty `pcBatchKey` parameter value will be used by the present operation.

**Values:** `<any string>`  
**Default:** `N/A`

**Optional**

---

#### OUTPUT PARAMETERS

`ExecuteStatus`

Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output `dsContext` dataset.

**Values:** See Errors part below

`pcIgnoredInServiceFieldsList`

A comma-separated list of fields that are restricted for the current user (`pcUserID`). Corresponding fields in the table will have default values.

**Values:** `<any string that corresponds to the description>`

`plBatchLast`

Last Record flag, which indicates whether the last record was reached in the current batch operation.

**Values:** YES - The last record was reached;  
NO - The last record was NOT reached.



---

## Read Locations

---

### Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for	

---

## Read Locations

---

### Errors:

ID	Description
assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Add/Update Locations

---

**Operation:** apSrvLocationSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsLocation

**Description:** This operation creates new and/or updates information about existing locations in the system.

ENTITY TABLE: dsLocation.ttLocation  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsLocation Dataset  
KEY FIELDS: LocationObj  
MANDATORY FIELDS: cid, bid, LocationObj  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:

- "A" - the ADD record mode
- "U" - the UPDATE record mode
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new locations (ADD record mode), the following conditions should be met:

- The values of the KEY FIELD (LocationObj) should be unique in the system. The system will assign automatically generated values to the LocationObj field of new records, overwriting values passed in the input dataset.
- The Location name (code) field of new records should be unique within the current Company.

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it. If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Locations.

**User Security:** Location Web Services --> ADD  
Location Web Services --> UPDATE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required

## Add/Update Locations

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcUpdateRequestFieldsList</b>	<p>"" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined.</p> <p><b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be updated;            &lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcUpdateExceptFieldsList</b>	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plImportEnableUpdate</b>	<p>A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field).</p> <p><b>Values:</b> YES - Allows to update existing records;            NO - Does not allow to update existing records.</p> <p><b>Default:</b> YES</p>	<b>Optional</b>
<b>plUseBeforeImage</b>	<p>Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset.</p> <p><b>Values:</b> YES - Enables the concurrency mode;            NO - Does not enable the concurrency mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>pcConcurrencyFieldList</b>	<p>A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>

### OUTPUT PARAMETERS

<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>
----------------------	---

### Errors:

ID	Description
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	<p>List of context fields that are required for service and are not defined.            Level: OPERATION</p>
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	<p>More than one service is set to be run by one call. Level: OPERATION</p>
<b>RUN_ERROR,1005</b>	<p>Invalid entry. Level: RECORD</p>
<b>RUN_ERROR,1020</b>	<p>Record already exists. Level: RECORD</p>

---

## Add/Update Locations

---

### Errors:

ID	Description
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1930	Invalid [%location%] code. Level: RECORD
RUN_ERROR,1935	[%Branch%] rules have been defined using this [%location%] record. Unable to update record status. Level: RECORD
RUN_ERROR,1940	This [%location%] was created in [%company%] &1 / [%branch%] &2 and cannot be updated in current [%company%] &3 / [%branch%] &4. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

# Delete Locations

**Operation:** apSrvLocationSave  
 INPUT-OUTPUT dsContext, INPUT-OUTPUT dsLocation

**Description:** This operation deletes existing locations from the system.

ENTITY TABLE: dsLocation.ttLocation

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsLocation Dataset

KEY FIELDS: LocationObj

MANDATORY FIELDS: cid, bid, LocationObj

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Locations.

**User Security:** Location Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

**OUTPUT PARAMETERS**

<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
----------------------	--

**Errors:**

ID	Description
----	-------------

---

## Delete Locations

---

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1910	Shifts exist in the schedule using this [%location%] record. Unable to delete. Level: RECORD
RUN_ERROR,1915	Travel time information has been defined for this [%location%] record. Unable to delete. Level: RECORD
RUN_ERROR,1920	[%Branch%] rules have been defined using this [%location%] record. Unable to delete. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Shifts

**Operation:** apSrvShiftGet  
INPUT-OUTPUT dsContext, OUTPUT dsShifts

**Description:** This operation retrieves information about existing shifts from the system.

ENTITY TABLE: dsShifts.ttShifts  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShifts Dataset  
KEY FIELDS: shiftid

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Shifts.

**User Security:** Shifts Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pdFromDate	Shift's Scheduled From Date. <b>Values:</b> <any valid date string that corresponds to the eTime Xpress Datatime data type>. For example, MM/DD/YYYY	Required
pdToDate	Shift's Scheduled To Date. <b>Values:</b> <any valid date string that corresponds to the eTime Xpress Datatime data type>. For example, MM/DD/YYYY	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: shifts.cid < 1 <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by shifts.shdate <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults	Optional



# Read Shifts

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined.</p> <p><b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plGetMidnightDifference</b>	<p>Performs additional calculations related with Midnight Cut Off. Additional Midnight cutoff calculations process the portion of the shift occurring after midnight separately filling-in the following fields:            ttShifts.MidDiffScheduled,ttShifts.MidDiffWorked,ttShifts.MidDiffOverlap1,ttShifts.MidDiffOverlap2,ttShifts.MidDiffWorkedOverlap1,ttShifts.MidDiffWorkedOverlap2.            See more fields description in eTime Xpress Web Services API Datasets - dsShifts Dataset.</p> <p><b>Values:</b> YES - Performs additional calculations related with Midnight Cutoff;            NO - Does not perform additional calculations related with Midnight Cutoff.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>pcEmpStat</b>	<p>Employee status.</p> <p><b>Values:</b> A - Active records;            I - Inactive records;            V - Void records;            &lt;empty string&gt; - All records.</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcCustStat</b>	<p>Customer status.</p> <p><b>Values:</b> A - Active records;            I - Inactive records;            V - Void records;            &lt;empty string&gt; - All records.</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcSiteStat</b>	<p>Site status.</p> <p><b>Values:</b> A - Active records;            I - Inactive records;            V - Void records;            &lt;empty string&gt; - All records.</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
<b>pcIgnoredInServiceFieldsList</b>	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	

## Errors:

ID	Description
----	-------------

---

## Read Shifts

---

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Add/Update Shifts

**Operation:** apSrvShiftSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsShifts

**Description:** This operation creates new and/or updates information about existing shifts in the system.

ENTITY TABLE: dsShifts.ttShifts  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShifts Dataset  
KEY FIELDS: shiftid  
MANDATORY FIELDS: cid, bid, shiftid

The cChkNew field is used as a record mode flag and can have the following values:

- "A" - the ADD record mode
- "U" - the UPDATE record mode
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: The system will assign automatically generated values to the Shift ID (shiftid) fields of new records.  
This operation updates only fields supported by ETime Shift Import functionality.

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Shifts.

**User Security:** Shifts Web Services --> ADD  
Shifts Web Services --> UPDATE  
Shifts Web Services --> UPDATE CONTROL CENTER  
Shifts Web Services --> UPDATE TIME AND ATTENDANCE  
Shifts Web Services --> UPDATE BILLING  
Shifts Web Services --> UPDATE PAYROLL

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcGlobalIgnoreList	A string containing zero or more Message Type Letters,	Optional

## Add/Update Shifts

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>listed below. The corresponding Message Type functionality checks will be ignored by the present operation for all shifts.</p> <p>D - Employee is Double Booked to a Shift            T - Employee has Time Off            R - Shift is restricted            Q - Employee Failed Shift Qualifications            W - Employee has that Day Off            P - Max Pay Rate Exceeded            C - Shift Exceeds Counter            E - Employee Budget Exceeded            S - Site Budget Exceeded            L - Location Travel Time            H - Employee Minimum Rest Period Between Shifts            N - Shift is part of a rotation            F - Shift confirmed            O - Shift worked            Y - Shift Self-Scheduled            Z - Shift Published for Self-Scheduling            X - Employee has Requested that Time Off            1 - Employee Daily Budget Exceeded            2 - Employee Weekly Budget Exceeded            3 - Employee Monthly Budget Exceeded            4 - Customer Daily Budget Exceeded            5 - Customer Weekly Budget Exceeded            6 - Customer Monthly Budget Exceeded            7 - Site Daily Budget Exceeded            8 - Site Weekly Budget Exceeded            9 - Site Monthly Budget Exceeded</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	
<b>pcUpdateRequestFieldsList</b>	<p>"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined.</p> <p><b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be updated;            &lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcUpdateExceptFieldsList</b>	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plImportEnableUpdate</b>	<p>A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field).</p> <p><b>Values:</b> YES - Allows to update existing records;            NO - Does not allow to update existing records.</p> <p><b>Default:</b> YES</p>	<b>Optional</b>
<b>plUseBeforeImage</b>	<p>Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset.</p> <p><b>Values:</b> YES - Enables the concurrency mode;            NO - Does not enable the concurrency mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>pcConcurrencyFieldList</b>	<p>A comma-separated list of DATA FIELDS that will be used</p>	<b>Optional</b>

## Add/Update Shifts

### Parameters:

Name	Description	Type
------	-------------	------

#### INPUT PARAMETERS

for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS.  
**Values:** <any string that corresponds to the description>  
**Default:** <empty string>

#### OUTPUT PARAMETERS

<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
----------------------	--

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1030	Start date cannot be greater than the End date. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1130	OPEN shift cannot be set to worked. Level: RECORD
RUN_ERROR,1145	Invalid entry. Enter hours as 00 to 23 and minutes as 00 to 59. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1665	Invalid [%Customer%] / [%Site%] combination. Level: RECORD
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1705	Invalid [%Customer%] ID. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD

---

## Add/Update Shifts

---

### Errors:

ID	Description
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1815	Invalid [%Service%]. Level: RECORD
RUN_ERROR,1820	Invalid [%Location%] ID. Level: RECORD
RUN_ERROR,1825	Invalid [%Shift Tag%] Set. Level: RECORD
RUN_ERROR,1830	Unknown Error. Level: RECORD
RUN_ERROR,1835, <detail error string from shift' business-logic validation routine>	<detail error string from shift' business-logic validation routine> Level: RECORD
RUN_ERROR,1840,<Shift's Company ID> + <TAB> + <Shift's Branch ID> + <TAB> + <Current Company ID> + <TAB> + <Current Branch ID>	This shift was created in [%company%] &1 / [%branch%] &2 and cannot be updated in current [%company%] &3 / [%branch%] &4. Level: RECORD
RUN_ERROR,1845	OPEN shift cannot be set to Tentative. Level: RECORD
RUN_ERROR,1850	Invalid Publishing Type. Level: RECORD
RUN_ERROR,1855, <Start or End>	Shift is published for Self-Scheduling. Invalid Self-Scheduling &1 (Date/Time). Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2105	Safe check time cannot be set on an OPEN shift. Level: RECORD
RUN_ERROR,2110,<Fields Value> + <TAB> + <Fields Label> + <TAB> + <Fields Label>	When a shift is &3, the &2 must be set to &1. Level: RECORD
RUN_ERROR,2115,<Fields Label> + <TAB> + <Fields Label>	&1 must be specified to set a shift as &2. Level: RECORD
RUN_ERROR,2120,<Fields Label(s)>	The &1 cannot be set due to insufficient licensing or user security rights. Level: RECORD
RUN_ERROR,2125,<Fields Label> + <TAB> + <Fields Label>	Shift cannot be set as &1 without first being &2. Level: RECORD
RUN_ERROR,2130	This shift is not worked. Unable to lock for billing or payroll. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	

---

## Add/Update Shifts

---

### Errors:

ID	Description
	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Delete Shifts

---

**Operation:** apSrvShiftSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsShifts

**Description:** This operation deletes existing shifts from the system.

ENTITY TABLE: dsShifts.ttShifts

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShifts Dataset

KEY FIELDS: shiftid

MANDATORY FIELDS: cid, bid, shiftid

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.

Use cValmess and cObjectName fields in returned table to detect

RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Shifts.

**User Security:** Shifts Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcGlobalIgnoreList	A string containing zero or more Message Type Letters, listed below. The corresponding Message Type functionality checks will be ignored by the present operation for all shifts. D - Employee is Double Booked to a Shift N - Shift is part of a rotation F - Shift confirmed O - Shift worked Y - Shift Self-Scheduled Z - Shift Published for Self-Scheduling <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional



---

## Delete Shifts

---

### Parameters:

Name	Description	Type
------	-------------	------

---

#### INPUT PARAMETERS

#### OUTPUT PARAMETERS

<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
----------------------	--

### Errors:

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1330	This shift is locked for billing or payroll. Unable to delete. Level: RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1665	Invalid [%Customer%] / [%Site%] combination. Level: RECORD
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1785,	<Error description from the Process Child records program when record is deleted> &1. Unable to delete. Level: RECORD
RUN_ERROR,1790	This shift is worked. Unable to delete. Level: RECORD
RUN_ERROR,1795	Shift is part of a rotation. Level: RECORD
RUN_ERROR,1800	Shift has been confirmed. Level: RECORD
RUN_ERROR,1805	Shift has been self-scheduled. Level: RECORD
RUN_ERROR,1810	Shift has been published for self-scheduling. Level: RECORD
RUN_ERROR,2185	

---

## Delete Shifts

---

### Errors:

ID	Description
	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

# Read Shifts Distributions

**Operation:** apSrvShiftDistrGet  
 INPUT-OUTPUT dsContext, OUTPUT dsShiftDistribution

**Description:** This operation retrieves information about existing shifts distributions records from the system.

ENTITY TABLE: dsShiftsDistribution.ttShifts  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShiftsDistribution Dataset  
 KEY FIELDS: shiftid

ENTITY TABLE: dsShiftsDistribution.ttShiftDis  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShiftsDistribution Dataset  
 KEY FIELDS: shiftid

DATA RELATIONSHIP: dsShiftsDistribution.ttShifts.shiftid - 1:1 -> dsShiftsDistribution.ttShiftDis.shiftid

Fields returned by this operation can be of the following three types:  
 - DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
 - CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
 - SYSTEM FIELDS, required for system purposes.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.

EXAMPLE: See eTime Xpress Web Services API Examples - Read Shifts Distributions.

**User Security:** Shifts Web Services --> READ  
 Shift Distributions Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pdFromDate	Shift's Scheduled From Date. Values: <any valid date string that corresponds to the eTime Xpress Datatime data type>. For example, MM/DD/YYYY	Required
pdToDate	Shift's Scheduled To Date. Values: <any valid date string that corresponds to the eTime Xpress Datatime data type>. For example, MM/DD/YYYY	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: shifts.cid < 1 Values: <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT	Optional

# Read Shifts Distributions

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by shifts.shdate <b>Values:</b> <any string> <b>Default:</b> <empty string>	
<b>pcReadRequestFieldsList</b>	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be read; <empty string> - All DATA fields will be read excluding pcReadExceptFieldsList fields <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcReadExceptFieldsList</b>	A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>plGetMidnightDifference</b>	Performs additional calculations related with Midnight Cut Off. Additional Midnight cutoff calculations process the portion of the shift occurring after midnight separately filling-in the following fields: ttShifts.MidDiffScheduled,ttShifts.MidDiffWorked,ttShifts.MidDiffOverlap1,ttShifts.MidDiffOverlap2,ttShifts.MidDiffWorkedOverlap1,ttShifts.MidDiffWorkedOverlap2. See more fields description in eTime Xpress Web Services API Datasets - dsShifts Dataset. <b>Values:</b> YES - Performs additional calculations related with Midnight Cutoff; NO - Does not perform additional calculations related with Midnight Cutoff. <b>Default:</b> NO	<b>Optional</b>

**OUTPUT PARAMETERS**

<b>pcIgnoredInServiceFieldsList</b>	A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values. <b>Values:</b> <any string that corresponds to the description>
<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below

**Errors:**

ID	Description
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	List of context fields that are required for service and are not defined. Level: OPERATION
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION
<b>RUN_ERROR,1240</b>	

---

## Read Shifts Distributions

---

### Errors:

ID	Description
	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Codes

**Operation:** apSrvCodesTableGet  
INPUT-OUTPUT dsContext, OUTPUT dsCodesTable

**Description:** This operation retrieves information about existing codes from the system.

ENTITY TABLE: dsCodesTable.ttCodesTable  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsCodesTable Dataset  
KEY FIELDS: code-type, cds

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Codes.

**User Security:** Code Table Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: codes_table.cds <> "CD" <b>Values:</b> <any string>	Required
pcCodeType	Code Type. There are available following Code Types: ALLOWANCES - Shift Allowance Code BILLALLOWRPT - Billing Allowance Report* BILLHOURTYPES - Bill Hour Types* BILLPERIOD - Billing Cycle Code BILLREASONCODES - Bill Reason Code BUDGETHOURS - Budget Period* CALLTYPE - Call Status Code COMPLAINTCODE - Complaint Code COUNTRY - Country Code CUSTNOTETYPES - Customer Notes Code DEPARTMENTS - Departments EMPBAC - Ethnic Background Code EMPMARITALSTAT - Marital Status Code EMPNOTETYPES - Employee Notes Code EMPPAYFREQ - Payroll Cycle Code EMPPHONETYPE - Phone Code EMPPOS - Position Code EMPWEIGHTTYPE - Employee Weight Code EXPTIMETYPE - Export Time Types* EXREFBRN - External Branch Ref. EXREFCOM - External Company Ref. EXREFCUS - External Cust Ref.	Required

---

## Read Codes

---

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
EXREFCUSREF01	- External Cust Ref. 01 Lookup	
EXREFCUSREF02	- External Cust Ref. 02 Lookup	
EXREFCUSREF03	- External Cust Ref. 03 Lookup	
EXREFCUSREF04	- External Cust Ref. 04 Lookup	
EXREFCUSREF05	- External Cust Ref. 05 Lookup	
EXREFCUSREF06	- External Cust Ref. 06 Lookup	
EXREFCUSREF07	- External Cust Ref. 07 Lookup	
EXREFCUSREF08	- External Cust Ref. 08 Lookup	
EXREFCUSREF09	- External Cust Ref. 09 Lookup	
EXREFCUSREF10	- External Cust Ref. 10 Lookup	
EXREFEMP	- External Emp Ref.	
EXREFEMPREF01	- External Emp Ref. 01 Lookup	
EXREFEMPREF02	- External Emp Ref. 02 Lookup	
EXREFEMPREF03	- External Emp Ref. 03 Lookup	
EXREFEMPREF04	- External Emp Ref. 04 Lookup	
EXREFEMPREF05	- External Emp Ref. 05 Lookup	
EXREFEMPREF06	- External Emp Ref. 06 Lookup	
EXREFEMPREF07	- External Emp Ref. 07 Lookup	
EXREFEMPREF08	- External Emp Ref. 08 Lookup	
EXREFEMPREF09	- External Emp Ref. 09 Lookup	
EXREFEMPREF10	- External Emp Ref. 10 Lookup	
EXREFLOC	- External Loc Ref.	
EXREFLOCREF01	- External Loc Ref. 01 Lookup	
EXREFLOCREF02	- External Loc Ref. 02 Lookup	
EXREFLOCREF03	- External Loc Ref. 03 Lookup	
EXREFLOCREF04	- External Loc Ref. 04 Lookup	
EXREFLOCREF05	- External Loc Ref. 05 Lookup	
EXREFLOCREF06	- External Loc Ref. 06 Lookup	
EXREFLOCREF07	- External Loc Ref. 07 Lookup	
EXREFLOCREF08	- External Loc Ref. 08 Lookup	
EXREFLOCREF09	- External Loc Ref. 09 Lookup	
EXREFLOCREF10	- External Loc Ref. 10 Lookup	
EXREFSCH	- External Shift Ref. Labels	
EXREFSCHREF1	- External Shift Ref. 1 Lookup	
EXREFSCHREF2	- External Shift Ref. 2 Lookup	
EXREFSCHREF3	- External Shift Ref. 3 Lookup	
EXREFSCHREF4	- External Shift Ref. 4 Lookup	
EXREFSCHREF5	- External Shift Ref. 5 Lookup	
EXREFSIT	- External Site Ref.	
EXREFSITREF01	- External Site Ref. 01 Lookup	
EXREFSITREF02	- External Site Ref. 02 Lookup	
EXREFSITREF03	- External Site Ref. 03 Lookup	
EXREFSITREF04	- External Site Ref. 04 Lookup	
EXREFSITREF05	- External Site Ref. 05 Lookup	
EXREFSITREF06	- External Site Ref. 06 Lookup	
EXREFSITREF07	- External Site Ref. 07 Lookup	
EXREFSITREF08	- External Site Ref. 08 Lookup	
EXREFSITREF09	- External Site Ref. 09 Lookup	
EXREFSITREF10	- External Site Ref. 10 Lookup	
EXREFSRV	- External Srv Ref	
EXREFSRVREF01	- External Srv Ref. 01 Lookup	
EXREFSRVREF02	- External Srv Ref. 02 Lookup	
EXREFSRVREF03	- External Srv Ref. 03 Lookup	
EXREFSRVREF04	- External Srv Ref. 04 Lookup	
EXREFSRVREF05	- External Srv Ref. 05 Lookup	
EXREFSRVREF06	- External Srv Ref. 06 Lookup	
EXREFSRVREF07	- External Srv Ref. 07 Lookup	
EXREFSRVREF08	- External Srv Ref. 08 Lookup	
EXREFSRVREF09	- External Srv Ref. 09 Lookup	
EXREFSRVREF10	- External Srv Ref. 10 Lookup	
GLACCTCATEGORY	- G/L Account Category Code*	
GLACCTSTATUS	- G/L Account Status Code*	
GLACCTTYPE	- G/L Account Type Code*	
HOURTYPES	- Hour Types*	
HPTACODES	- HandPunch TA Codes*	
INCIDENTCODE	- Incident Code	
INVENTORYCODE	- Inventory Code	
MESSAGETYPES	- Message Box Types*	
ORDERCODE	- Sort Order Code*	
OTHRSCODE	- Overtime Threshold Code*	
PAYREASONCODES	- Pay Reason Code	

# Read Codes

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	PLANWEEKDESC - Shift Planner Week Description* PUBLISHTYPE - Self-Schedule Publishing Types* QBXMLVERSION - QuickBooks XML Version* QUALIFCODE - Qualifications Code RECORDSTATUS - Record Status Codes* RESTRICTCODE - Restriction Code SEX - Sex Code* SITENOTETYPES - Site Notes Code STATDAYS - Statutory Day Code STATPROV - State/Province Code TAXID - Tax Code* TERMINATION - Termination Code THRESHOLDPRD - OT Threshold Periods* THRESHOLDTYPE - OT Threshold Types* TIMEENTRYSTATUS - Time Entry Status* TIMEIMPORT - Time Import Codes* VOLUNTEERCODE - Volunteer Code WEEKDAY - Weekday*	
	* - this code type has SYSTEM type and can be used in Read Codes operation only. <b>Values:</b> <any string that corresponds to the description>	
<b>pcSort</b>	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by codes_table.cds <b>Values:</b> <any string> <b>Default:</b> <empty string>	<b>Optional</b>
<b>plNoCodeType</b>	No Code Type flag. <b>Values:</b> YES - Ignore code types in operation's bussiness logic. <any other values> or not passed to the server - Use code type in operation's bussiness logic. <b>Default:</b> NO	<b>Optional</b>
<b>pcReadRequestFieldsList</b>	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be read; <empty string> - All DATA fields will be read excluding pcReadExceptFieldsList fields <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcReadExceptFieldsList</b>	A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>plBatch</b>	Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit<>0. <b>Values:</b> YES - Enables the Batch mode; NO - Does not enable the Batch mode. <b>Default:</b> NO	<b>Optional</b>
<b>piBatchLimit</b>	The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the	<b>Optional</b>



---

## Read Codes

---

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	operation can return up to two records, including the current one. <b>Values:</b> <any positive integer value> <b>Default:</b> 0	
<b>pcBatchKey</b>	A Batch Key is ttCodesTable.cds value that defines start record for a batch. NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description). <b>Values:</b> <any string> <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcBatchDirection</b>	Direction of the batch. <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST; PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST; FIRST - First batch; LAST - Last batch. <b>Default:</b> NEXT	<b>Optional</b>
<b>pcBatchSearchKey</b>	A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: <table>.<field>, where <field> - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter. NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch. <b>Values:</b> <any string> <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcBatchSearchKeyValue</b>	A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used: 1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: <TAB> + <TAB> + <Character to search> + <Current record key>, where <Current record key> is the business object ID value for the current record and has the same format as pcBatchKey. 2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: <String to search>. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the	<b>Optional</b>

---

## Read Codes

---

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> &lt;empty string&gt;</p>	
pcBatchRepositionKeys	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2 &gt; + ... + &lt;TAB&gt; + &lt;TAB&gt;+ &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> N/A</p>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
pcIgnoredInServiceFieldsList	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
plBatchLast	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p> <p><b>Values:</b> YES - The last record was reached; NO - The last record was NOT reached.</p>	
plBatchKeySearchOK	<p>Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful.</p> <p><b>Values:</b> YES - The record was found; NO - The record was NOT found.</p>	
plBatchReturnedExtraData	<p>Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record.</p> <p><b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.</p>	
plBatchOneLetterSearchKey	<p>A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string.</p> <p><b>Values:</b> &lt;any string&gt;</p>	

### Errors:

---

## Read Codes

---

### Parameters:

Name	Description	Type
------	-------------	------

---

#### OUTPUT PARAMETERS

ID	Description
<b>NO_CONTEXT_DATA</b> ,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION
<b>RUN_ERROR,1060</b>	Record is not available. Level: RECORD
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level: OPERATION, RECORD
<b>RUN_ERROR,1435</b>	This function is not licensed. Level: OPERATION
<b>RUN_ERROR,1965</b>	There are wrong fields in the requested list: &1. Level: RECORD
<b>RUN_ERROR,1980</b> , <query string>	Invalid QUERY string: &1. Level: OPERATION
<b>RUN_ERROR,INVALID_CONTEXT_DATA</b> ,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
<b>RUN_ERROR,LOCKED_CONTEXT_DATA</b>	Context record is currently locked. Level: OPERATION
<b>RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)</b>	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
<b>RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)</b>	Invalid Session ID. Level: OPERATION
<b>RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)</b>	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Add/Update Codes

**Operation:** apSrvCodesTableSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsCodesTable

**Description:** This operation creates new and/or updates information about existing codes in the system.

ENTITY TABLE: dsCodesTable.ttCodesTable  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsCodesTable Dataset  
KEY FIELDS: code-type, cds  
MANDATORY FIELDS: code-type, cds  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:

- "A" - the ADD record mode
- "U" - the UPDATE record mode
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new codes (ADD record mode), the following condition should be met:

- The values of the KEY FIELDS should be unique in the system.

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Codes.

**User Security:** Code Table Web Services --> ADD  
Code Table Web Services --> UPDATE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined.	Optional

## Add/Update Codes

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p><b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be updated;            &lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
pcUpdateExceptFieldsList	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	Optional
plImportEnableUpdate	<p>A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field).</p> <p><b>Values:</b> YES - Allows to update existing records;            NO - Does not allow to update existing records.  <b>Default:</b> YES</p>	Optional
plUseBeforeImage	<p>Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset.</p> <p><b>Values:</b> YES - Enables the concurrency mode;            NO - Does not enable the concurrency mode.  <b>Default:</b> NO</p>	Optional
pcConcurrencyFieldList	<p>A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	Optional

### OUTPUT PARAMETERS

ExecuteStatus	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>
---------------	---

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1065	

---

## Add/Update Codes

---

### Errors:

ID	Description
	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2005,<code size>	Code length cannot exceed &1. Level: RECORD
RUN_ERROR,2010,<code-type>	Invalid code type &1. Level: RECORD
RUN_ERROR,2015	Cannot add/update/delete system code. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete Codes

**Operation:** apSrvCodesTableSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsCodesTable

**Description:** This operation deletes existing codes from the system.

ENTITY TABLE: dsCodesTable.ttCodesTable

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsCodesTable Dataset

KEY FIELDS: code-type, cds

MANDATORY FIELDS: code-type, cds

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it. If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Codes.

**User Security:** Code Table Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

**OUTPUT PARAMETERS**

<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
----------------------	--

**Errors:**

ID	Description
----	-------------

---

## Delete Codes

---

### Errors:

ID	Description
NO_CONTEXT_DATA,	<b>&lt;comma separated list of missed context data&gt;</b> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR,2000,	<b>&lt;code&gt; + &lt;TAB&gt; + &lt;description&gt; + &lt;TAB&gt; + &lt;Rule ID&gt;+ &lt;TAB&gt; + &lt;Rule Title&gt;</b> Cannot delete code. (&1 - &2) Code found in rule #&3 &4. Level: RECORD
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1540	Record is currently being used in the system. Cannot delete. Level: RECORD
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1995,	<b>&lt;code&gt; + &lt;TAB&gt; + &lt;description&gt; + &lt;TAB&gt; + &lt;employee id&gt;</b> Cannot delete code. (&1 - &2) Code found in employee #&3 qualifications. Level: RECORD
RUN_ERROR,2015	Cannot add/update/delete system code. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,	<b>&lt;Context Parameter Name&gt; + &lt;TAB&gt; + &lt;Comma-separated list of available values&gt;</b> The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION



## Read Employee Time Off Records

**Operation:** apSrvTimeOffGet  
 INPUT-OUTPUT dsContext, OUTPUT dsEmpTmOff

**Description:** This operation retrieves information about existing employee time off records from the system.

ENTITY TABLE: dsEmpTmOff.ttEmpTmOff  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsEmpTmOff Dataset  
 KEY FIELDS: cid, bid, tmoffid

Fields returned by this operation can be of the following three types:

- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and
- SYSTEM FIELDS, required for system purposes.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.

EXAMPLE: See eTime Xpress Web Services API Examples - Read Employee Time Off Records.

**User Security:** Employee Time Off Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcStatus	Status Record Flag. Values: A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: emptmoff.tmoffid > 100 Values: <any string>	Required
pcToffStatus	Employee Time Off code status. Values: A - Active records; I - Inactive records; V - Void records; <empty string> - All records. Default: <empty string>	Optional
pcEmpStat	Employee status. Values: A - Active records; I - Inactive records; V - Void records; <empty string> - All records. Default: <empty string>	Optional
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system.	Optional

# Read Employee Time Off Records

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by emptmoff.tmoftid <b>Values:</b> <any string> <b>Default:</b> <empty string>	
<b>pcReadRequestFieldsList</b>	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be read; <empty string> - All DATA fields will be read excluding pcReadExceptFieldsList fields <b>Default:</b> <empty string>	Optional
<b>pcReadExceptFieldsList</b>	A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

## OUTPUT PARAMETERS

<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
<b>pcIgnoredInServiceFieldsList</b>	A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values. <b>Values:</b> <any string that corresponds to the description>

## Errors:

ID	Description
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	List of context fields that are required for service and are not defined. Level: OPERATION
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level: OPERATION, RECORD
<b>RUN_ERROR,1435</b>	This function is not licensed. Level: OPERATION
<b>RUN_ERROR,1965</b>	There are wrong fields in the requested list: &1. Level: RECORD
<b>RUN_ERROR,1980, &lt;query string&gt;</b>	Invalid QUERY string: &1. Level: OPERATION
<b>RUN_ERROR,INVALID_CONTEXT_DATA,&lt;Context Parameter Name&gt; + &lt;TAB&gt; + &lt;Comma-separated list of available values&gt;</b>	The &1 parameter's value must be one of the following: &2. Level: OPERATION

---

## Read Employee Time Off Records

---

**Errors:**

ID	Description
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Add/Update Employee Time Off Records

**Operation:** apSrvTimeOffSave  
 INPUT-OUTPUT dsContext, INPUT-OUTPUT dsEmpTmOff

**Description:** This operation creates new and/or updates information about existing employee time off records in the system.

ENTITY TABLE: dsEmpTmOff.ttEmpTmOff  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsEmpTmOff Dataset  
 KEY FIELDS: cid, bid, tmoffid  
 MANDATORY FIELDS: cid, bid, tmoffid  
 IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
 - "A" - the ADD record mode  
 - "U" - the UPDATE record mode  
 - <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: The system will assign automatically generated values to the Time Off Record ID (ttEmpTmOff) field of new records.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Employee Time Off Records.

**User Security:** Employee Time Off Web Services --> ADD  
 Employee Time Off Web Services --> UPDATE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. Values: * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated;	Optional

# Add/Update Employee Time Off Records

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>&lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcUpdateExceptFieldsList</b>	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plImportEnableUpdate</b>	<p>A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field).</p> <p><b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records.</p> <p><b>Default:</b> YES</p>	<b>Optional</b>
<b>plAllowRewriteWebTime</b>	<p>Allows to update Time Off records created in WebX and imported into eTime (webtmoffid &lt;&gt; 0). Once such a record is modified, the webtmoffid field will be set to zero.</p> <p><b>Values:</b> YES - Allows to update; NO - Does not allow to update.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>plRunAsOneTransaction</b>	<p>Process All Time Off records from dataset in one transaction.</p> <p>When adding multiple records in one transaction, it is possible to ignore errors of specific types using the plIgnoreUniqueError, plIgnoreEmployeeError, plIgnoreTimeOffCodeError, plIgnoreTimeOffCodeStatusError, and plIgnoreWebTimeOffPartError parameters. These parameters will not allow creating invalid records: if corresponding errors occur, the operation will roll back changes for the invalid record and return warning for it. All remaining records will be processed without rolling back the entire transaction.</p> <p><b>Values:</b> YES - If any of the records cannot be added/updated, the changes for all records in the transaction will be rolled back; NO - If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>plIgnoreUniqueError</b>	<p>Returns a warning instead of the Unique Time Off record error.</p> <p>This error occurs if another Time Off record exists in the system with the same combination of Company/Branch/Employee/Date/Time Start/Time End/Time Off Code fields values. See more information in the description of the plRunAsOneTransaction parameter.</p> <p><b>Values:</b> YES - Returns "IGNORE_UNIQUE" warning in the ttEmptmoff.cValmess field; NO - Returns error in the ttEmptmoff.cValmess field.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>plIgnoreEmployeeError</b>	<p>Returns a warning instead of the employee related errors.</p> <p>These errors occur if employee validation failed. See more information in the description of the plRunAsOneTransaction parameter.</p> <p><b>Values:</b> YES - Returns "IGNORE_EMPLOYEE_NOT_ACTIVE" / "IGNORE_EMPLOYEE" warnings in the ttEmptmoff.cValmess field; NO - Returns error in the ttEmptmoff.cValmess</p>	<b>Optional</b>

## Add/Update Employee Time Off Records

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	field. <b>Default:</b> NO	
<b>plIgnoreTimeOffCodeStatusError</b>	Returns a warning instead of the Time Off Code Status related errors. These errors occur if Time Off Code Status validation failed. See more information in the description of the plRunAsOneTransaction parameter. <b>Values:</b> YES - Returns "IGNORE_TIMEOFFCODESTATUS" warning in the ttEmptmoff.cValmess field; NO - Returns error in the ttEmptmoff.cValmess field. <b>Default:</b> NO	<b>Optional</b>
<b>plChangeIdWhenUpdate</b>	Assigns a new value to Time Off Record identifier (timeoffid) each time when a record is updated. <b>Values:</b> YES - Assigns a new value; NO - Does not assign current value. <b>Default:</b> YES	<b>Optional</b>
<b>plIgnoreWebTimeOffPartError</b>	Returns a warning instead of the Part of Time Off request error. These errors occur if Part of Time Off request validation failed. See more information in the description of the plRunAsOneTransaction parameter.  NOTE: if plAllowRewriteWebTime =YES, the record imported from WebX will be updated by the present operation and the value of the present parameter will be ignored. <b>Values:</b> YES - Returns "IGNORE_WEB_TIME_OFF_PART" warning in the ttEmptmoff.cValmess field; NO - Returns error in the ttEmptmoff.cValmess field. <b>Default:</b> NO	<b>Optional</b>
<b>plCheckShiftConflict</b>	Returns a warning instead of the employee related errors. These errors occur if employee validation failed. See more information in the description of the plRunAsOneTransaction parameter. Specifies whether the present operation should check for conflicts with existing shifts. Such conflicts may occur between new/modified Time Off records and existing shifts for the same employee. If plCheckShiftConflict=YES and there are no conflicting shifts, then the Time Off records will be added/updated. If plCheckShiftConflict=YES and one or more scheduled shifts exist for a time range of a new/modified Time Off record, then the following algorithm is used: If the field ttEmpTmOff.ShowErrorMessage = yes, then the current Time Off record will NOT be added/updated and the "CLIENT_ASK" warning message will be returned. If the field ttEmpTmOff.ShowErrorMessage = no, then this Time Off record will be added/updated. If the field ttEmpTmOff.SetOpenShift=yes, then the conflicting shift(s) will be set open. If the field ttEmpTmOff.SetOpenShift=no, then the conflicting shift(s) will not be set open. If plCheckShiftConflict=NO, then the check will not be performed and all specified Time Off records will be always added/updated. The operation will return CheckShiftConflict warnings for all added records in the cValmess fields of the ttEmptmoff table. <b>Values:</b> YES - Checks for conflicts; NO - Does not check for conflicts. <b>Default:</b> YES	<b>Optional</b>
<b>plIgnoreTimeOffCodeError</b>	Returns a warning instead of the Time Off Code related errors. These errors occur if Time Off Code validation failed. See more information in the description of the plRunAsOneTransaction parameter.	<b>Optional</b>

# Add/Update Employee Time Off Records

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Values:</b> YES - Returns "IGNORE_TIMEOFFCODE" warning in the ttEmptmoff.cValmess field; NO - Returns error in the ttEmptmoff.cValmess field. <b>Default:</b> NO	
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

## OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1145	Invalid entry. Enter hours as 00 to 23 and minutes as 00 to 59. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1615	Inactive\Void Entry. Level: RECORD

---

## Add/Update Employee Time Off Records

---

### Errors:

ID	Description
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1775	Wrong archive key. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,6005	Time Off Type &1 is Invalid. Level: RECORD
RUN_ERROR,6010	This record is a part of a time off request. Level: RECORD
RUN_ERROR,6015	Shift conflict with time off Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION



## Delete Employee Time Off Records

**Operation:** apSrvTimeOffSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsEmpTmOff

**Description:** This operation deletes existing employee time off records from the system.

ENTITY TABLE: dsEmpTmOff.ttEmpTmOff  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsEmpTmOff Dataset  
 KEY FIELDS: cid, bid, tmoffid  
 MANDATORY FIELDS: cid, bid, tmoffid

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete Employee Time Off Records.

**User Security:** Employee Time Off Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	<b>Required</b>
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	<b>Required</b>
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	<b>Required</b>
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	<b>Required</b>
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	<b>Required</b>
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	<b>Optional</b>
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

**Errors:**

ID	Description
----	-------------

---

## Delete Employee Time Off Records

---

### Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	List of context fields that are required for service and are not defined. Level: OPERATION	
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION	
<b>RUN_ERROR,1025</b>	Record is currently locked. Level: RECORD	
<b>RUN_ERROR,1065</b>	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD	
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level: OPERATION, RECORD	
<b>RUN_ERROR,1435</b>	This function is not licensed. Level: OPERATION	
<b>RUN_ERROR,1685</b>	Invalid [%Company%] / [%Branch%] combination. Level: RECORD	
<b>RUN_ERROR,1700</b>	Invalid [%Employee%] ID. Level: RECORD	
<b>RUN_ERROR,2185</b>	Record has been changed by another user or system process. Refresh recommended. Level: RECORD	
<b>RUN_ERROR,6015</b>	Shift conflict with time off Level: RECORD	
<b>RUN_ERROR,INVALID_CONTEXT_DATA,&lt;Context Parameter Name&gt; + &lt;TAB&gt; + &lt;Comma-separated list of available values&gt;</b>	The &l parameter's value must be one of the following: &2. Level: OPERATION	
<b>RUN_ERROR,LOCKED_CONTEXT_DATA</b>	Context record is currently locked. Level: OPERATION	
<b>RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)</b>	Invalid Session ID. Level: OPERATION	
<b>RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)</b>	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION	

# Read Volunteer Records

**Operation:** apSrvVolunshGet  
 INPUT-OUTPUT dsContext, OUTPUT dsVolunsh

**Description:** This operation retrieves information about existing volunteer records from the system.

ENTITY TABLE: dsVolunsh.ttVolunsh  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsVolunsh Dataset  
 KEY FIELDS: volunid

Fields returned by this operation can be of the following three types:  
 - DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
 - CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
 - SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Volunteer Records.

**User Security:** Volunteer Scheduling Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcStatus	Status Record Flag. Values: A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: volunsh.volcode <> "CD" Values: <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by volunsh.volcode Values: <any string> Default: <empty string>	Optional
pcEmployeeStatus	Employee status. Values: A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	Optional

# Read Volunteer Records

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Default:</b> <empty string>	
pcCustomerStatus	Customer status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcSiteStatus	Site status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	
pcIgnoredInServiceFieldsList	A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values. <b>Values:</b> <any string that corresponds to the description>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980,<query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level:

---

## Read Volunteer Records

---

**Errors:**

ID	Description
----	-------------

---

OPERATION

RUN\_ERROR,Service Session failure: Invalid Session ID. (#1670)

Invalid Session ID. Level: OPERATION

RUN\_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)

User ID is not valid. Contact your system administrator for assistance.  
Level: OPERATION

## Add/Update Volunteer Records

**Operation:** apSrvVolunshSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsVolunsh

**Description:** This operation creates new and/or updates information about existing volunteer records in the system.

ENTITY TABLE: dsVolunsh.ttVolunsh  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsVolunsh Dataset  
KEY FIELDS: volunid  
MANDATORY FIELDS: cid, bid, volunid  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
- "A" - the ADD record mode  
- "U" - the UPDATE record mode  
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new volunteer records (ADD record mode), the following condition should be met:  
- The Volunteer Shift ID (volunid) field should be set to 0. The system will assign automatically generated values to the volunid field of new records.

ERROR HANDLING:  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Volunteer Records.

**User Security:** Volunteer Scheduling Web Services --> ADD  
Volunteer Scheduling Web Services --> UPDATE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both	Optional

# Add/Update Volunteer Records

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
plImportEnableUpdate	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records. <b>Default:</b> YES	Optional
pcEmpStat	Employee status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcCustStat	Customer status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcSiteStat	Site status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

# Add/Update Volunteer Records

## Parameters:

Name	Description	Type
------	-------------	------

## OUTPUT PARAMETERS

## Errors:

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1145	Invalid entry. Enter hours as 00 to 23 and minutes as 00 to 59. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1705	Invalid [%Customer%] ID. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &l field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &l field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &l field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1815	Invalid [%Service%]. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,5095	Record ID cannot be found. Unable to update. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of	



---

## Add/Update Volunteer Records

---

### Errors:

ID	Description
	available values> The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

# Delete Volunteer Records

**Operation:** apSrvVolunshSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsVolunsh

**Description:** This operation deletes existing volunteer records from the system.

ENTITY TABLE: dsVolunsh.ttVolunsh  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsVolunsh Dataset  
KEY FIELDS: volunid  
MANDATORY FIELDS: cid, bid, volunid

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete Volunteer Records.

**User Security:** Volunteer Scheduling Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcEmpStat	Employee status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcCustStat	Customer status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcSiteStat	Site status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
plDeleteRotation	Proceeds Delete rotation. <b>Values:</b> YES - Proceeds deletes rotation; <any other values> or not passed to the server - Does not proceed delete rotation. <b>Default:</b> NO	Optional
plUseBeforeImage	Enables the concurrency mode. NOTE: All concurrency-related	Optional

# Delete Volunteer Records

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1705	Invalid [%Customer%] ID. Level: RECORD
RUN_ERROR,1815	Invalid [%Service%]. Level: RECORD
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION

---

## Delete Volunteer Records

---

**Errors:**

ID	Description
----	-------------

---

RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	
---	--

[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION	
---	--

---

RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	
--	--

Invalid Session ID. Level: OPERATION	
--------------------------------------	--

---

RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	
--	--

User ID is not valid. Contact your system administrator for assistance. Level: OPERATION	
--	--

---

---

## Get Employee Availability

---

**Operation:** apSrvAvailEmp  
INPUT-OUTPUT dsContext, INPUT dsShifts, OUTPUT dsAvailability

**Description:** This operation determines availability of existing employees. It returns employees which can be assigned to certain shifts.

ENTITY TABLE: dsShifts.ttShifts  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShifts Dataset  
KEY FIELDS: shiftid

ENTITY TABLE: dsAvailability.ttEmployee  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsAvailability Dataset  
KEY FIELDS: cid, eid

ENTITY TABLE: dsAvailability.ttAvailEmp  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsAvailability Dataset  
KEY FIELDS: eid

DATA RELATIONSHIP: dsAvailability.ttAvailEmp.eid -- 1:1 --> dsAvailability.ttEmployee.eid

Fields returned by this operation can be of the following three types:

- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and
- SYSTEM FIELDS, required for system purposes.

Based on the contents of the dsShifts.ttShifts table, the operation will have different behaviour:

- There are no records in the dsShifts.ttShifts table. The operation will create and use a temporary shift for availability check, where:

```
ttShifts.shdate = <pdShiftDate>
ttShifts.tmstart = <piShiftStart>
ttShifts.tmend = <piShiftEnd>
ttShifts.breaks = <peShiftBreak>
ttShifts.emsrtycd = <pcShiftServiceType>
ttShifts.cid = <piCompanyID>
ttShifts.bid = <piBranchID>
ttShifts.uid = <piShiftCustomerID>
ttShifts.sid = <piShiftSiteID>.
```

- There is one record in the dsShifts.ttShifts table. The operation will use information from all of its fields, except:

```
ttShifts.shdate = <pdShiftDate>
ttShifts.tmstart = <piShiftStart>
ttShifts.tmend = <piShiftEnd>
ttShifts.breaks = <peShiftBreak>
ttShifts.emsrtycd = <pcShiftServiceType>
ttShifts.cid = <piCompanyID>
ttShifts.bid = <piBranchID>
ttShifts.uid = <piShiftCustomerID>
ttShifts.sid = <piShiftSiteID>.
```

- There are two or more record in the dsShifts.ttShifts table. The operation will apply MULTIPLE SHIFTS AVAILABILITY logic. In this case input parameters <pdShiftDate>, <piShiftStart>, <piShiftEnd>, <peShiftBreak>, <pcShiftServiceType>, <piShiftCustomerID>, <piShiftSiteID> will be ignored - each shift will should contain

## Get Employee Availability

correct all correspondend field data.

### ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned dsShifts to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

EXAMPLE: See eTime Xpress Web Services API Examples - Get Employee Availability.

**User Security:** Employee Web Services --> READ  
Employee Availability Web Services --> READ

### Parameters:

Name	Description	Type
<b>INPUT*/OUTPUT* PARAMETERS</b>		
<b>Read Employee Data Parameters</b>	All Read Employee parameters EXCEPT pcStatus, pcSort, all *Batch* parameters. Also, pcWhere parameter is an optional. Based on parameters, the operation will select list of employees to be checked against specific availability rules. <b>Values:</b> See more details in the Read Employee data documentation part <b>Default:</b> N/A	
<b>INPUT PARAMETERS</b>		
<b>pdShiftDate</b>	Shift Scheduled Date. <b>Values:</b> <any valid date string that corresponds to the eTime Xpress Datatime data type>. For example, MM/DD/YYYY	<b>Required</b>
<b>piShiftStart</b>	Shift Scheduled Start. <b>Values:</b> HHMM - Etime's Time , where HH - 00..23, MM - 00..59	<b>Required</b>
<b>piShiftEnd</b>	Shift Scheduled End. <b>Values:</b> HHMM - Etime's Time , where HH - 00..23, MM - 00..59	<b>Required</b>
<b>piShiftCustomerID</b>	Shift Customer ID. <b>Values:</b> 1..999	<b>Required</b>
<b>piShiftSiteID</b>	Shift Site ID. <b>Values:</b> 1..999	<b>Required</b>
<b>peShiftBreak</b>	Shift Break, hours. <b>Values:</b> 0..99.999	<b>Required</b>
<b>pcShiftServiceType</b>	Shift Service Type. <b>Values:</b> <any string>	<b>Required</b>
<b>plExperienceAtCustomerSite</b>	Applies the "Experience at customer site" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plMeetsShiftQualifications</b>	Applies the "Meets shift qualifications" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plNoOvertimeCaused</b>	Applies the "No overtime caused" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>

## Get Employee Availability

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>plNotScheduledOnThisDay</b>	Applies the "Not scheduled on this day" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	Optional
<b>plListEmployeesWithDayOff</b>	Applies the "List employees with day off" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	Optional
<b>plServiceTypeInEmpPayRates</b>	Applies the "Service Type found in employee pay rates" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	Optional
<b>plDoesNotExceedSiteMaxPayRate</b>	Applies the "Does not exceed site max pay rate" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	Optional
<b>plMatchesDefServiceType</b>	Applies the "Matches default service type" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	Optional
<b>plUnionizedEmployee</b>	Applies the "Unionized employee" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	Optional
<b>piConsecutiveDayOff</b>	Applies the "Not during <piConsecutiveDayOff> consecutive days off" filter. <b>Values:</b> 1..99; not passed to the server - Does not apply the filter. <b>Default:</b> N/A	Optional
<b>pcDepartment</b>	Applies the "<pcDepartment>" filter. The filter can be a single department code or a comma-separated list of departments. <b>Values:</b> <any string> - Applies the filter; not passed to the server - Does not apply the filter. <b>Default:</b> N/A	Optional
<b>pcPosition</b>	Applies the "<pcPosition>" filter. The filter can be a single position code or a comma-separated list of positions. <b>Values:</b> <any string> - Applies the filter; not passed to the server - Does not apply the filter. <b>Default:</b> N/A	Optional
<b>pcEmployeeType</b>	Applies the "<pcEmployeeType>" filter. The filter can be a single employee type code or a comma-separated list of employee types. <b>Values:</b> <any string> - Applies the filter; not passed to the server - Does not apply the filter. <b>Default:</b> N/A	Optional
<b>pcVolunteerCode</b>	Applies the "<pcVolunteerCode>" filter. The filter can be a single volunteer code or a comma-separated list of volunteer codes.	Optional

# Get Employee Availability

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Values:</b> <any string> - Applies the filter; not passed to the server - Does not apply the filter. <b>Default:</b> N/A	
<b>pcTotalHoursPeriod</b>	Applies the "Total hrs less than <peTotalHoursMax> hrs for <pcTotalHoursPeriod>" filter. <b>Values:</b> DAY - One day period; WEEK - One week period; MONTH - One month period; not passed to the server - Does not apply the filter. <b>Default:</b> N/A	<b>Optional</b>
<b>peTotalHoursMax</b>	Applies the "Total hrs less than <peTotalHoursMax> hrs for <pcTotalHoursPeriod>" filter. <b>Values:</b> 0..999; not passed to the server - Does not apply the filter. <b>Default:</b> N/A	<b>Optional</b>
<b>peRestPeriodHours</b>	Applies the "Rest Period between shifts <peRest Period Hours> hrs" filter. <b>Values:</b> 0..504; not passed to the server - Does not apply the filter. <b>Default:</b> N/A	<b>Optional</b>
<b>plEmployeeIsDoubleBooked</b>	Applies a rule that corresponds to the Company/Branch Message Type "Employee is Double Booked to a Shift" = "Error". <b>Values:</b> YES - Applies the rule; NO - Does not apply the rule. <b>Default:</b> NO	<b>Optional</b>
<b>plLocationTravelTime</b>	Applies a rule that corresponds to the Company/Branch Message Type "Location Travel Time" = "Error". <b>Values:</b> YES - Applies the rule; NO - Does not apply the rule. <b>Default:</b> NO	<b>Optional</b>
<b>plEmployeeMinimumRestPeriod</b>	Applies a rule that corresponds to the Company/Branch Message Type "Employee Minimum Rest Period Between Shifts" = "Error". <b>Values:</b> YES - Applies the rule; NO - Does not apply the rule. <b>Default:</b> NO	<b>Optional</b>
<b>plShiftIsRestricted</b>	Applies a rule that corresponds to the Company/Branch Message Type "Shift is restricted" = "Error". <b>Values:</b> YES - Applies the rule; NO - Does not apply the rule. <b>Default:</b> NO	<b>Optional</b>
<b>plEmployeeHasTimeOff</b>	Applies a rule that corresponds to the Company/Branch Message Type "Employee has Time Off" = "Error". <b>Values:</b> YES - Applies the rule; NO - Does not apply the rule. <b>Default:</b> NO	<b>Optional</b>
<b>plEmployeeHasRequestedTimeOff</b>	Applies a rule that corresponds to the Company/Branch Message Type "Employee has Requested that Time Off" = "Error". <b>Values:</b> YES - Applies the rule; NO - Does not apply the rule. <b>Default:</b> NO	<b>Optional</b>
<b>plEmployeeBudgetIsExceeded</b>	Applies the "Employee Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter.	<b>Optional</b>



# Get Employee Availability

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Default:</b> NO	
<b>plSiteBudgetIsExceeded</b>	Applies the "Site Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plEmployeeDailyBudget</b>	Applies the "Employee Daily Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plEmployeeWeeklyBudget</b>	Applies the "Employee Weekly Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plEmployeeMonthlyBudget</b>	Applies the "Employee Monthly Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plCustomerDailyBudget</b>	Applies the "Customer Daily Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plCustomerWeeklyBudget</b>	Applies the "Customer Weekly Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plCustomerMonthlyBudget</b>	Applies the "Customer Monthly Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plSiteDailyBudget</b>	Applies the "Site Daily Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plSiteWeeklyBudget</b>	Applies the "Site Weekly Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plSiteMonthlyBudget</b>	Applies the "Site Monthly Budget Exceeded" filter. <b>Values:</b> YES - Applies the filter; <any other values> or not passed to the server - Does not apply the filter. <b>Default:</b> NO	<b>Optional</b>
<b>plAllowOverwriteGlobalRules</b>	Ignores rules that correspond to the following Company/Branch Message Types: plEmployeeIsDoubleBooked plLocationTravelTime plEmployeeMinimumRestPeriod plShiftIsRestricted plEmployeeHasTimeOff plEmployeeHasRequestedTimeOff <b>Values:</b> YES - Does not use the above rules;	<b>Optional</b>

# Get Employee Availability

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	NO - Uses the above rules (as in VS). <b>Default:</b> NO	
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

## Errors:

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1115	Period week start day is not defined. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1835,	<detail error string from shift' business-logic validation routine> <detail error string from shift' business-logic validation routine> Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980,	<query string> Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,2030,<parameter value> + <TAB> + <error description>	Parameter &1 error: &2 Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	

---

## Get Employee Availability

---

**Errors:**

**ID**

**Description**

---

User ID is not valid. Contact your system administrator for assistance.  
Level: OPERATION

# Read Employee Web Xpress Accounts

**Operation:** apSrvEmpWebAcGet  
 INPUT-OUTPUT dsContext, OUTPUT dsEmpWebAc

**Description:** This operation retrieves information about existing employee Web Xpress accounts from the system.

ENTITY TABLE: dsEmpWebAc.ttEmpWebAc  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsEmpWebAc Dataset  
 KEY FIELDS: cid,bid,eid

Fields returned by this operation can be of the following three types:  
 - DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
 - CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
 - SYSTEM FIELDS, required for system purposes.

NOTE: The value of the password field returned by this operation is always an empty string.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.

EXAMPLE: See eTime Xpress Web Services API Examples - Read Employee Web Xpress Accounts.

**User Security:** Employee Web Xpress Account --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: empwebac.loginid = "test" Values: <any string>	Required
pcWEBXPLoginID	Web Xpress System Administrator Login ID. Values: <any string>	Required
pcWEBXPPassword	Web Xpress System Administrator Password. Values: <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by empwebac.loginid Values: <any string> Default: <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will	Optional

# Read Employee Web Xpress Accounts

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined.</p> <p><b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.</p> <p><b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.</p> <p><b>Values:</b> &lt;any positive integer value&gt;</p> <p><b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttEmpWebAc.eid value that defines start record for a batch.</p> <p>NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.</p> <p><b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.</p> <p><b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.</p> <p>NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.</p>	<b>Optional</b>

# Read Employee Web Xpress Accounts

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Values:</b> <any string> <b>Default:</b> <empty string>	
pcBatchSearchKeyValue	<p>A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcBatchRepositionKeys	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2&gt; + ... + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <b>Values:</b> <any string> <b>Default:</b> N/A	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <b>Values:</b> See Errors part below	
pcIgnoredInServiceFieldsList	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <b>Values:</b> <any string that corresponds to the description>	
plBatchLast	<p>Last Record flag, which indicates whether the last</p>	

# Read Employee Web Xpress Accounts

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
	record was reached in the current batch operation. <b>Values:</b> YES - The last record was reached; NO - The last record was NOT reached.	
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,2035	Invalid Web Xpress System Administrator login or password. Level: OPERATION
RUN_ERROR,2050	[%Company%]/[%Branch%] access has not been set up for Web Xpress System Administrator. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	

---

## Read Employee Web Xpress Accounts

---

### Errors:

ID	Description
	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION



## Add/Update Employee Web Xpress Accounts

**Operation:** apSrvEmpWebAcSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsEmpWebAc

**Description:** This operation creates new employee Web Xpress accounts in the system.

ENTITY TABLE: dsEmpWebAc.ttEmpWebAc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsEmpWebAc Dataset  
KEY FIELDS: cid,bid,eid  
MANDATORY FIELDS: cid,bid,eid,loginid,password  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
- "A" or <empty string> - the ADD record mode

NOTE: There is no update for this operation.

To add new employee Web Xpress accounts (ADD record mode), the following condition should be met:  
- The loginid field of new records should be unique in the system  
The system will assign automatically generated values to the Employee Web Xpress Account ID (empwebid) fields of new records.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it. If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Employee Web Xpress Accounts.

**User Security:** Employee Web Xpress Account --> ADD

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcWEBXPLoginID	Web Xpress System Administrator Login ID. Values: <any string>	Required
pcWEBXPPassword	Web Xpress System Administrator Password. Values: <any string>	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both	Optional

# Add/Update Employee Web Xpress Accounts

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1710	[%Employee%] ID cannot be 0. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD

---

## Add/Update Employee Web Xpress Accounts

---

**Errors:**

<b>ID</b>	<b>Description</b>
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2035	Invalid Web Xpress System Administrator login or password. Level: OPERATION
RUN_ERROR,2040	Web Xpress Login ID already exists. Level: RECORD
RUN_ERROR,2045,<Fields Label> + <TAB> + <Min field's size>	&1 must be at least &2 characters in length. Level: RECORD
RUN_ERROR,2050	[%Company%]/[%Branch%] access has not been set up for Web Xpress System Administrator. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

# Delete Employee Web Xpress Accounts

**Operation:** apSrvEmpWebAcSave  
 INPUT-OUTPUT dsContext, INPUT-OUTPUT dsEmpWebAc

**Description:** This operation deletes existing employee Web Xpress accounts from the system.

ENTITY TABLE: dsEmpWebAc.ttEmpWebAc  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsEmpWebAc Dataset  
 KEY FIELDS: cid,bid,eid  
 MANDATORY FIELDS: cid,bid,eid

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete Employee Web Xpress Accounts.

**User Security:** Employee Web Xpress Account --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcWEBXPLoginID	Web Xpress System Administrator Login ID. Values: <any string>	Required
pcWEBXPPassword	Web Xpress System Administrator Password. Values: <any string>	Required

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. Values: See Errors part below
---------------	---

**Errors:**

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	

---

## Delete Employee Web Xpress Accounts

---

### Errors:

ID	Description
	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1785	<b>&lt;Error description from the Process Child records program when record is deleted&gt;</b> &1. Unable to delete. Level: RECORD
RUN_ERROR,2035	Invalid Web Xpress System Administrator login or password. Level: OPERATION
RUN_ERROR,2050	[%Company%]/[%Branch%] access has not been set up for Web Xpress System Administrator. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA	<b>&lt;Context Parameter Name&gt; + &lt;TAB&gt; + &lt;Comma-separated list of available values&gt;</b> The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Read Users

---

**Operation:** apSrvZUsersGet  
INPUT-OUTPUT dsContext, OUTPUT dsZUsers

**Description:** This operation retrieves information about existing users from the system.

ENTITY TABLE: dsZUsers.ttZUsers  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsers Dataset  
KEY FIELDS: usrid

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

NOTE: The value of the usrpwd field returned by this operation is always an empty string.

ERROR HANDLING:  
Use ExecuteStatus parameter to detect OPERATION level errors.

EXAMPLE: See eTime Xpress Web Services API Examples - Read Users.

**User Security:** User Security --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcStatus	User's Status Record Flag. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: zusers.usrid = "test" <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by zusers.usrtype <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults	Optional

---

## Read Users

---

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be read; <empty string> - All DATA fields will be read excluding pcReadExceptFieldsList fields <b>Default:</b> <empty string>	
pcReadExceptFieldsList	A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
plBatch	Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit<>0. <b>Values:</b> YES - Enables the Batch mode; NO - Does not enable the Batch mode. <b>Default:</b> NO	Optional
piBatchLimit	The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one. <b>Values:</b> <any positive integer value> <b>Default:</b> 0	Optional
pcBatchKey	A Batch Key is ttZUsers.usrid value that defines start record for a batch. NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description). <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcBatchDirection	Direction of the batch. <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST; PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST; FIRST - First batch; LAST - Last batch. <b>Default:</b> NEXT	Optional
pcBatchSearchKey	A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: <table>.<field>, where <field> - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter. NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch. <b>Values:</b> <any string>	Optional

---

## Read Users

---

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Default:</b> <empty string>	
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchRepositionKeys</b>	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2 &gt; + ... + &lt;TAB&gt; + &lt;TAB&gt;+ &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> N/A</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
<b>pcIgnoredInServiceFieldsList</b>	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
<b>plBatchLast</b>	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p>	



# Read Users

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
	<b>Values:</b> YES - The last record was reached; NO - The last record was NOT reached.	
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	

---

## Read Users

---

### Errors:

ID

Description

---

Invalid Session ID. Level: OPERATION

RUN\_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)

User ID is not valid. Contact your system administrator for assistance.  
Level: OPERATION

---

## Add/Update Users

---

**Operation:** apSrvZUsersSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZUsers

**Description:** This operation creates new and/or updates information about existing users in the system.

ENTITY TABLE: dsZUsers.ttZUsers  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsers Dataset  
KEY FIELDS: usrid  
MANDATORY FIELDS: usrid  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:

- "A" - the ADD record mode
- "U" - the UPDATE record mode
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new users (ADD record mode), the following condition should be met:

- The usrid field of new records should be unique in the system
- The usrpwd field is not empty

To update users password (UPDATE record mode), the following condition should be met:

- The usrpwd field is not empty.

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors. Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it. If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Users.

**User Security:** User Security --> ADD  
User Security --> UPDATE  
User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required

# Add/Update Users

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcUpdateRequestFieldsList</b>	<p>"" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined.</p> <p><b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be updated;            &lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcUpdateExceptFieldsList</b>	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plImportEnableUpdate</b>	<p>A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field).</p> <p><b>Values:</b> YES - Allows to update existing records;            NO - Does not allow to update existing records.</p> <p><b>Default:</b> YES</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	

## Errors:

ID	Description
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	List of context fields that are required for service and are not defined. Level: OPERATION
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION
<b>RUN_ERROR,1005</b>	Invalid entry. Level: RECORD
<b>RUN_ERROR,1015</b>	Required entry. Level: RECORD
<b>RUN_ERROR,1020</b>	Record already exists. Level: RECORD
<b>RUN_ERROR,1025</b>	Record is currently locked. Level: RECORD
<b>RUN_ERROR,1065</b>	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level: OPERATION, RECORD

---

## Add/Update Users

---

### Errors:

ID	Description
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,1865	Your User ID must be contain only numbers, letters, the hyphen symbol ( - ) or the underscore symbol ( _ ). Please rename your User ID. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2080	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Delete Users

---

**Operation:** apSrvZUsersSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZUsers

**Description:** This operation deletes existing users from the system.

ENTITY TABLE: dsZUsers.ttZUsers  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsers Dataset  
KEY FIELDS: usrid  
MANDATORY FIELDS: usrid

ERROR HANDLING:  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Users.

**User Security:** User Security --> DELETE  
User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

**Errors:**

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD

---

## Delete Users

---

### Errors:

ID	Description
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,2080	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
RUN_ERROR,2155	Unable to delete. You are currently logged in under this user account. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read User Security Company/Branch Access

**Operation:** apSrvZAccessGet  
INPUT-OUTPUT dsContext, OUTPUT dsZAccess

**Description:** This operation retrieves information about existing user security company/branch access records. Each record in the dataset contains information about access of one user to one company/branch combination.

ENTITY TABLE: dsZAccess.ttZAccess  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZAccess Dataset  
KEY FIELDS: cid,bid,usrid

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read User Security Company/Branch Access.

**User Security:** User Security Company/Branch Access --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcStatus	Status Record Flag. Reserved for future usage. <b>Values:</b> <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: zaccess.usrid = "test" <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by zaccess.usrid <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be	Optional



# Read User Security Company/Branch Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>read excluding pcReadExceptFieldsList fields;                      &lt;comma-separated list&gt; - List of DATA fields that will be read;                      &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields  <b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.  <b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.  <b>Values:</b> YES - Enables the Batch mode;                      NO - Does not enable the Batch mode.  <b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.  <b>Values:</b> &lt;any positive integer value&gt;  <b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttZAccess.zaccid value that defines start record for a batch.                      NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.  <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;                      PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;                      FIRST - First batch;                      LAST - Last batch.  <b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.                      NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key</p>	<b>Optional</b>

---

## Read User Security Company/Branch Access

---

### Parameters:

Name	Description	Type
------	-------------	------

---

#### INPUT PARAMETERS

search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:

1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: <TAB> + <TAB> + <Character to search> + <Current record key>, where <Current record key> is the business object ID value for the current record and has the same format as pcBatchKey.

2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: <String to search>. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.

NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.

**Values:** <any string>  
**Default:** <empty string>

---

#### pcBatchRepositionKeys

A double <TAB>-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: <pcBatchKey1> + <TAB> + <TAB> + <pcBatchKey2 > + ... + <TAB> + <TAB>+ <pcBatchKeyN>. If <pcBatchKey1> fails, then the operation will try using <pcBatchKey2>, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.

**Values:** <any string>  
**Default:** N/A

Optional

---

#### OUTPUT PARAMETERS

#### ExecuteStatus

Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.

**Values:** See Errors part below

---

#### pcIgnoredInServiceFieldsList

A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.

**Values:** <any string that corresponds to the description>

---

#### plBatchLast

Last Record flag, which indicates whether the last record was reached in the current batch operation.

**Values:** YES - The last record was reached;  
NO - The last record was NOT reached.

# Read User Security Company/Branch Access

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for	

---

## Read User Security Company/Branch Access

---

**Errors:**

ID	Description
assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Add/Update User Security Company/Branch Access

**Operation:** apSrvZAccessSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZAccess

**Description:** This operation creates new and/or updates information about existing user security company/branch access records in the system.

ENTITY TABLE: dsZAccess.ttZAccess  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZAccess Dataset  
 KEY FIELDS: cid,bid,usrid  
 MANDATORY FIELDS: cid,bid,usrid  
 IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
 - "A" - the ADD record mode  
 - "U" - the UPDATE record mode  
 - <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: The system will assign automatically generated values to the Company/Branch Access ID (zaccid) field of new records.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update User Security Company/Branch Access.

**User Security:** User Security Company/Branch Access --> ADD  
 User Security Company/Branch Access --> UPDATE  
 User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both	Optional

# Add/Update User Security Company/Branch Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
plImportEnableUpdate	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records. <b>Default:</b> YES	Optional

## OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	

---

## Add/Update User Security Company/Branch Access

---

### Errors:

ID	Description
	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2055	Security permissions cannot be updated for users with administrator rights. Level: RECORD
RUN_ERROR,2065	Unable to update. Allow Access to All [%Sites%] permission is inherited from a security group. Level: RECORD
RUN_ERROR,2080	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
RUN_ERROR,2085	Unable to update. Allow Access to All [%Employees%] permission is inherited from a security group. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete User Security Company/Branch Access

**Operation:** apSrvZAccessSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZAccess

**Description:** This operation deletes existing user security company/branch access records from the system.

ENTITY TABLE: dsZAccess.ttZAccess

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZAccess Dataset

KEY FIELDS: cid,bid,usrId

MANDATORY FIELDS: cid,bid,usrId

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.

Use cValmess and cObjectName fields in returned table to detect

RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete User Security Company/Branch Access.

**User Security:** User Security Company/Branch Access --> DELETE  
User should have an administration rights

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required

### OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. Values: See Errors part below
---------------	---

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD



---

## Delete User Security Company/Branch Access

---

### Errors:

ID	Description
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,2055	Security permissions cannot be updated for users with administrator rights. Level: RECORD
RUN_ERROR,2060,<Company ID> + <TAB> + <Branch ID>	Unable to delete. [%Company%] &1 / [%Branch%] &2 permission is inherited from a security group. Level: RECORD
RUN_ERROR,2080	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read User Security Customer/Site Access

**Operation:** apSrvZUsrSAccGet  
INPUT-OUTPUT dsContext, OUTPUT dsZUsrSAcc

**Description:** This operation retrieves information about existing user security customer/site access records from the system. Each record in the dataset contains information about access of one user to one customer/site combination or to one customer/all sites.

ENTITY TABLE: dsZUsrSAcc.ttZUsrSAcc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsrSAcc Dataset  
KEY FIELDS: cid,bid,usrid,uid,sid

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read User Security Customer/Site Access.

**User Security:** User Security Customer/Site Access --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcStatus	Status Record Flag. Reserved for future usage. <b>Values:</b> <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: zusrSacc.usrid = "test" <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by zusrSacc.usrid <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined.	Optional

## Read User Security Customer/Site Access

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p><b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.</p> <p><b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.</p> <p><b>Values:</b> &lt;any positive integer value&gt;</p> <p><b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttZUsrSAcc.zusrsaid value that defines start record for a batch.</p> <p>NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.</p> <p><b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.</p> <p><b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.</p> <p>NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>

# Read User Security Customer/Site Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchRepositionKeys</b>	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2 &gt; + ... + &lt;TAB&gt; + &lt;TAB&gt;+ &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> N/A</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
<b>pcIgnoredInServiceFieldsList</b>	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
<b>plBatchLast</b>	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p> <p><b>Values:</b> YES - The last record was reached;  NO - The last record was NOT reached.</p>	

# Read User Security Customer/Site Access

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &l. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION

---

## Read User Security Customer/Site Access

---

### Errors:

ID	Description
----	-------------

---

RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance.
--	---

Level: OPERATION
------------------

## Add/Update User Security Customer/Site Access

**Operation:** apSrvZUsrSAccSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZUsrSAcc

**Description:** This operation creates new user security customer/site access records in the system.

ENTITY TABLE: dsZUsrSAcc.ttZUsrSAcc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsrSAcc Dataset  
KEY FIELDS: cid,bid,usrid,uid,sid  
MANDATORY FIELDS: cid,bid,usrid,uid,sid  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
- "A" or <empty string> - the ADD record mode

NOTE: There is no update for this operation.  
The system will assign automatically generated values to the User Site Access ID (zusrsaid) fields of new records.  
In order to provide access to ALL customers and sites within a particular company/branch, set the corresponding acallsit field value to True by the Add/Update User Security Company/Branch Access operation.

ERROR HANDLING:  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update User Security Customer/Site Access.

**User Security:** User Security Customer/Site Access --> ADD  
User should have an administration rights

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields;	Optional

# Add/Update User Security Customer/Site Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>&lt;comma-separated list&gt; - List of DATA fields that will be updated;</p> <p>&lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
pcUpdateExceptFieldsList	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1170	[%Company%]/[%Branch%] access has not been set up for this user. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &l field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &l field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	



---

## Add/Update User Security Customer/Site Access

---

### Errors:

ID	Description
	&1. Unable to delete. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2055	Security permissions cannot be updated for users with administrator rights. Level: RECORD
RUN_ERROR,2075,<Company ID> + <TAB> + <Branch ID>	Unable to add. User already has permission to access all [%sites%] for [%Company%] &1/ [%Branch%] &2. Level: RECORD
RUN_ERROR,2080	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete User Security Customer/Site Access

**Operation:** apSrvZUsrSAccSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZUsrSAcc

**Description:** This operation deletes existing user security customer/site access records from the system.

ENTITY TABLE: dsZUsrSAcc.ttZUsrSAcc  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsrSAcc Dataset  
 KEY FIELDS: cid,bid,usrid,uid,sid  
 MANDATORY FIELDS: cid,bid,usrid,uid,sid

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete User Security Customer/Site Access.

**User Security:** User Security Customer/Site Access --> DELETE  
 User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. Values: See Errors part below
---------------	---

**Errors:**

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD

---

## Delete User Security Customer/Site Access

---

### Errors:

ID	Description
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1705	Invalid [%Customer%] ID. Level: RECORD
RUN_ERROR,1785	<Error description from the Process Child records program when record is deleted> &1. Unable to delete. Level: RECORD
RUN_ERROR,2055	Security permissions cannot be updated for users with administrator rights. Level: RECORD
RUN_ERROR,2070	<Customer ID> + <TAB> + <Site ID> Unable to delete. [%Customer%] &1/ [%Site%] &2 is inherited from a security group. Level: RECORD
RUN_ERROR,2080	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA	<Context Parameter Name> + <TAB> + <Comma-separated list of available values> The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Read User Security Employee Access

---

**Operation:** apSrvZUsrEAccGet  
INPUT-OUTPUT dsContext, OUTPUT dsZUsrEAcc

**Description:** This operation retrieves information about existing user security employee access records from the system. Each record in the dataset contains information about access of one user to one employee record.

ENTITY TABLE: dsZUsrEAcc.ttZUsrEAcc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsrEAcc Dataset  
KEY FIELDS: cid,bid,usrid,eid

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read User Security Employee Access.

**User Security:** User Security Employee Access --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcStatus	Status Record Flag. Reserved for future usage. <b>Values:</b> <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: zusreacc.usrid = "test" <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by zusreacc.usrid <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined.	Optional

# Read User Security Employee Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p><b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.</p> <p><b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.</p> <p><b>Values:</b> &lt;any positive integer value&gt;</p> <p><b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttZUsrEAcc.zusreadid value that defines start record for a batch.</p> <p>NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.</p> <p><b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.</p> <p><b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.</p> <p>NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>

# Read User Security Employee Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchRepositionKeys</b>	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2&gt; + ... + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> N/A</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
<b>pcIgnoredInServiceFieldsList</b>	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
<b>plBatchLast</b>	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p> <p><b>Values:</b> YES - The last record was reached;  NO - The last record was NOT reached.</p>	

# Read User Security Employee Access

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &l. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION

---

## Read User Security Employee Access

---

### Errors:

ID	Description
----	-------------

---

RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	
--	--

	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION
--	---



## Add/Update User Security Employee Access

**Operation:** apSrvZUsrEAccSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZUsrEAcc

**Description:** This operation creates new user security employee access records in the system.

ENTITY TABLE: dsZUsrEAcc.ttZUsrEAcc  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsrEAcc Dataset  
 KEY FIELDS: cid,bid,usrid,eid  
 MANDATORY FIELDS: cid,bid,usrid,eid  
 IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
 - "A" or <empty string> - the ADD record mode

NOTE: There is no update for this operation.  
 The system will assign automatically generated values to the User Employee Access ID (zusreaid) fields of new records.  
 In order to provide access to ALL employees within a particular company/branch, set the corresponding acallemp field value to True by the Add/Update User Security Company/Branch Access operation.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update User Security Employee Access.

**User Security:** User Security Employee Access --> ADD  
 User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. Values: * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that	Optional

# Add/Update User Security Employee Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1170	[%Company%]/[%Branch%] access has not been set up for this user. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD

---

## Add/Update User Security Employee Access

---

**Errors:**

<b>ID</b>	<b>Description</b>
<b>RUN_ERROR,1965</b>	There are wrong fields in the requested list: &1. Level: RECORD
<b>RUN_ERROR,2055</b>	Security permissions cannot be updated for users with administrator rights. Level: RECORD
<b>RUN_ERROR,2080</b>	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
<b>RUN_ERROR,2100,&lt;Company ID&gt; + &lt;TAB&gt; + &lt;Branch ID&gt;</b>	Unable to add. User already has permission to access all [%employees%] for [%Company%] &1/ [%Branch%] &2. Level: RECORD
<b>RUN_ERROR,INVALID_CONTEXT_DATA,&lt;Context Parameter Name&gt; + &lt;TAB&gt; + &lt;Comma-separated list of available values&gt;</b>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
<b>RUN_ERROR,LOCKED_CONTEXT_DATA</b>	Context record is currently locked. Level: OPERATION
<b>RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)</b>	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
<b>RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)</b>	Invalid Session ID. Level: OPERATION
<b>RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)</b>	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete User Security Employee Access

**Operation:** apSrvZUsrEAccSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZUsrEAcc

**Description:** This operation deletes existing user security employee access records from the system.

ENTITY TABLE: dsZUsrEAcc.ttZUsrEAcc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsrEAcc Dataset  
KEY FIELDS: cid,bid,usrid,eid  
MANDATORY FIELDS: cid,bid,usrid,eid

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete User Security Employee Access.

**User Security:** User Security Employee Access --> DELETE  
User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

**Errors:**

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD

---

## Delete User Security Employee Access

---

### Errors:

ID	Description
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1785	<Error description from the Process Child records program when record is deleted> &1. Unable to delete. Level: RECORD
RUN_ERROR,2055	Security permissions cannot be updated for users with administrator rights. Level: RECORD
RUN_ERROR,2080	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
RUN_ERROR,2095	<Employee ID> Unable to delete. [%Employee%] &1 is inherited from a security group. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA	<Context Parameter Name> + <TAB> + <Comma-separated list of available values> The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read User Security Groups

**Operation:** apSrvZUsrGrpxGet  
INPUT-OUTPUT dsContext, OUTPUT dsZUsrGrpx

**Description:** This operation retrieves information from the system about the membership of eTime Xpress users in security groups. Each record in the dataset contains information about the membership of one user in one security group.

ENTITY TABLE: dsZUsrGrpx.ttZUsrGrpx  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsrGrpx Dataset  
KEY FIELDS: usrid,grpId

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read User Security Groups.

**User Security:** User Security Groups --> READ  
User should have an administration rights

Parameters: Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcStatus	Status Record Flag. Reserved for future usage. <b>Values:</b> <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: zusrgpx.usrid = "test" <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by zusrgpx.usrid <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined.	Optional

# Read User Security Groups

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p><b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.</p> <p><b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.</p> <p><b>Values:</b> &lt;any positive integer value&gt;</p> <p><b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is (ttZusrGrpx.usrid + &lt;TAB&gt; + ttZusrGrpx.grpid) value that defines start record for a batch.</p> <p>NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.</p> <p><b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.</p> <p><b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.</p> <p>NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>

---

# Read User Security Groups

---

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchRepositionKeys</b>	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2&gt; + ... + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> N/A</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
<b>pcIgnoredInServiceFieldsList</b>	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
<b>plBatchLast</b>	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p> <p><b>Values:</b> YES - The last record was reached;</p>	



# Read User Security Groups

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
	NO - The last record was NOT reached.	
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION

---

## Read User Security Groups

---

**Errors:**

ID	Description
----	-------------

---

RUN_ERROR,	Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)
------------	--

User ID is not valid. Contact your system administrator for assistance. Level: OPERATION
---

## Add/Update User Security Groups

**Operation:** apSrvZUsrGrpxSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZUsrGrpx

**Description:** This operation creates new user/security group membership records in the system. Creating a new membership record is equivalent to adding a user to a security group.

ENTITY TABLE: dsZUsrGrpx.ttZUsrGrpx  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsrGrpx Dataset  
 KEY FIELDS: usrid,grpId  
 MANDATORY FIELDS: usrid,grpId  
 IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
 - "A" or <empty string> - the ADD record mode

NOTE: There is no update for this operation.

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update User Security Groups.

**User Security:** User Security Groups --> ADD  
 User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	Optional

# Add/Update User Security Groups

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,2055	Security permissions cannot be updated for users with administrator rights. Level: RECORD
RUN_ERROR,2080	

---

## Add/Update User Security Groups

---

### Errors:

ID	Description
	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete User Security Groups

**Operation:** apSrvZUsrGrpxSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZUsrGrpx

**Description:** This operation deletes existing user/security group membership records from the system. Deleting a membership record is equivalent to excluding a user from a security group.

ENTITY TABLE: dsZUsrGrpx.ttZUsrGrpx

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZUsrGrpx Dataset

KEY FIELDS: usrid,grpId

MANDATORY FIELDS: usrid,grpId

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.

Use cValmess and cObjectName fields in returned table to detect

RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete User Security Groups.

**User Security:** User Security Groups --> DELETE  
User should have an administration rights

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required

### OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. Values: See Errors part below
---------------	---

### Errors:

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	

---

## Delete User Security Groups

---

### Errors:

ID	Description
	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1785	<b>&lt;Error description from the Process Child records program when record is deleted&gt;</b> &1. Unable to delete. Level: RECORD
RUN_ERROR,2055	Security permissions cannot be updated for users with administrator rights. Level: RECORD
RUN_ERROR,2080	Security permissions cannot be added/updated/deleted for default admin user and system users. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA	<b>&lt;Context Parameter Name&gt; + &lt;TAB&gt; + &lt;Comma-separated list of available values&gt;</b> The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Security Groups

**Operation:** apSrvZGroupsGet  
INPUT-OUTPUT dsContext, OUTPUT dsZGroups

**Description:** This operation retrieves information about existing security groups from the system.

ENTITY TABLE: dsZGroups.ttZGroups  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGroups Dataset  
KEY FIELDS: grpid

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Security Groups.

**User Security:** Security Groups --> READ  
User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcStatus	Status Record Flag. Reserved for future usage. <b>Values:</b> <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: zgroups.grpid = "test" <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by zgroups.grpid <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be	Optional



# Read Security Groups

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields  <b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.  <b>Values:</b> &lt;any string that corresponds to the description&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.  <b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.  <b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.  <b>Values:</b> &lt;any positive integer value&gt;  <b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttZGroups.grpid value that defines start record for a batch.            NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.  <b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.  <b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.            NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.  <b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key</p>	<b>Optional</b>

---

## Read Security Groups

---

### Parameters:

Name	Description	Type
------	-------------	------

---

#### INPUT PARAMETERS

search to be performed, the `pcBatchSearchKey` parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:

1) **FIRST CHARACTER SEARCH:** The operation will try to find a first record (after the current record) whose `pcBatchSearchKey` field's value starts from a specified character. In this case the `pcBatchSearchKeyValue` parameter value format is: `<TAB> + <TAB> + <Character to search> + <Current record key>`, where `<Current record key>` is the business object ID value for the current record and has the same format as `pcBatchKey`.

2) **MULTIPLE CHARACTERS SEARCH:** The operation will try to find a first record whose `pcBatchSearchKey` field value matches as many symbols as possible in the specified string. In this case the `pcBatchSearchKeyValue` parameter value format is: `<String to search>`. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the `pcBatchSearchKey` field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.

**NOTE:** In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the `plBatchKeySearchOK` parameter, and the Batch Key value of the returned record will be returned in the `plBatchOneLetterSearchKey` parameter.

**Values:** `<any string>`  
**Default:** `<empty string>`

---

#### `pcBatchRepositionKeys`

A double `<TAB>`-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the `pcBatchKey` parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: `<pcBatchKey1> + <TAB> + <TAB> + <pcBatchKey2 > + ... + <TAB> + <TAB>+ <pcBatchKeyN>`. If `<pcBatchKey1>` fails, then the operation will try using `<pcBatchKey2>`, and so on, until the corresponding business object is found. **NOTE:** If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty `pcBatchKey` parameter value will be used by the present operation.

**Values:** `<any string>`  
**Default:** `N/A`

**Optional**

---

#### OUTPUT PARAMETERS

#### `ExecuteStatus`

Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output `dsContext` dataset.

**Values:** See Errors part below

---

#### `pcIgnoredInServiceFieldsList`

A comma-separated list of fields that are restricted for the current user (`pcUserID`). Corresponding fields in the table will have default values.

**Values:** `<any string that corresponds to the description>`

---

#### `plBatchLast`

Last Record flag, which indicates whether the last record was reached in the current batch operation.

**Values:** YES - The last record was reached;  
NO - The last record was NOT reached.

# Read Security Groups

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980,	<query string> Invalid QUERY string: &l. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,	<Context Parameter Name> + <TAB> + <Comma-separated list of available values> The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for	

---

## Read Security Groups

---

**Errors:**

ID	Description
assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Add/Update Security Groups

**Operation:** apSrvZGroupsSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZGroups

**Description:** This operation creates new and/or updates information about existing security groups in the system.

ENTITY TABLE: dsZGroups.ttZGroups  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGroups Dataset  
KEY FIELDS: grpId  
MANDATORY FIELDS: grpId  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:

- "A" - the ADD record mode
- "U" - the UPDATE record mode
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new security groups (ADD record mode), the following condition should be met:

- The grpId field of new records should be unique in the system

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Security Groups.

**User Security:** Security Groups --> ADD  
Security Groups --> UPDATE  
User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList	Optional

# Add/Update Security Groups

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
plImportEnableUpdate	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records. <b>Default:</b> YES	Optional

## OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	

---

## Add/Update Security Groups

---

### Errors:

ID	Description
	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1785	<Error description from the Process Child records program when record is deleted> &1. Unable to delete. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA	<Context Parameter Name> + <TAB> + <Comma-separated list of available values> The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

# Delete Security Groups

**Operation:** apSrvZGroupsSave  
 INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZGroups

**Description:** This operation deletes existing security groups from the system.

ENTITY TABLE: dsZGroups.ttZGroups  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGroups Dataset  
 KEY FIELDS: grpId  
 MANDATORY FIELDS: grpId

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Security Groups.

**User Security:** Security Groups --> DELETE  
 User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. Values: See Errors part below
---------------	---

**Errors:**

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD



---

## Delete Security Groups

---

### Errors:

ID	Description
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Security Group Company/Branch Access

**Operation:** apSrvZGrpBaccGet  
INPUT-OUTPUT dsContext, OUTPUT dsZGrpBacc

**Description:** This operation retrieves information about existing security group company/branch access records from the system. Each record in the dataset contains information about access of one group to one company/branch combination.

ENTITY TABLE: dsZGrpBacc.ttZGrpBacc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGrpBacc Dataset  
KEY FIELDS: cid,bid,grpId

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Security Group Company/Branch Access.

**User Security:** Security Group Company/Branch Access --> READ  
User should have an administration rights

Parameters: Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcStatus	Status Record Flag. Reserved for future usage. <b>Values:</b> <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: zgrpbacc.grpId = "test" <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by zusers.zgrpbacc <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined.	Optional

# Read Security Group Company/Branch Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p><b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.</p> <p><b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.</p> <p><b>Values:</b> &lt;any positive integer value&gt;</p> <p><b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttZGrpBAcc.zgrpbid value that defines start record for a batch.</p> <p>NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.</p> <p><b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.</p> <p><b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.</p> <p>NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>

---

# Read Security Group Company/Branch Access

---

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchRepositionKeys</b>	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2&gt; + ... + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> N/A</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
<b>pcIgnoredInServiceFieldsList</b>	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
<b>plBatchLast</b>	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p> <p><b>Values:</b> YES - The last record was reached; NO - The last record was NOT reached.</p>	

# Read Security Group Company/Branch Access

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &l. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION

---

## Read Security Group Company/Branch Access

---

**Errors:**

ID	Description
----	-------------

---

RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance.
--	---

Level: OPERATION
------------------

## Add/Update Security Group Company/Branch Access

**Operation:** apSrvZGrpBaccSave

INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZGrpBacc

**Description:** This operation creates new and/or updates information about existing security group company/branch access records in the system.

ENTITY TABLE: dsZGrpBacc.ttZGrpBacc

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGrpBacc Dataset

KEY FIELDS: cid,bid,grpId

MANDATORY FIELDS: cid,bid,grpId

IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:

- "A" - the ADD record mode

- "U" - the UPDATE record mode

- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: The system will assign automatically generated values to the Group Company/Branch Access ID (zgrpbid) field of new records.

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.

Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Security Group Company/Branch Access.

**User Security:** Security Group Company/Branch Access --> ADD  
Security Group Company/Branch Access --> UPDATE  
User should have an administration rights

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both	Optional

# Add/Update Security Group Company/Branch Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
plImportEnableUpdate	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records. <b>Default:</b> YES	Optional

## OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	



---

## Add/Update Security Group Company/Branch Access

---

### Errors:

ID	Description
	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete Security Group Company/Branch Access

**Operation:** apSrvZGrpBaccSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZGrpBacc

**Description:** This operation deletes existing security group company/branch access records from the system.

ENTITY TABLE: dsZGrpBacc.ttZGrpBacc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGrpBacc Dataset  
KEY FIELDS: cid,bid,grpId  
MANDATORY FIELDS: cid,bid,grpId

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete Security Group Company/Branch Access.

**User Security:** Security Group Company/Branch Access --> DELETE  
User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. Values: See Errors part below
---------------	---

**Errors:**

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD

---

## Delete Security Group Company/Branch Access

---

### Errors:

ID	Description
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Security Group Customer/Site Access

**Operation:** apSrvZGrpSaccGet  
INPUT-OUTPUT dsContext, OUTPUT dsZGrpSacc

**Description:** This operation retrieves information about existing security group customer/site access records from the system. Each record in the dataset contains information about access of one group to one customer/site combination or to one customer/all sites.

ENTITY TABLE: dsZGrpSacc.ttZGrpSacc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGrpSacc Dataset  
KEY FIELDS: cid,bid,grpId,uid,sid

Fields returned by this operation can be of the following three types:  
- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Security Group Customer/Site Access.

**User Security:** Security Group Customer/Site Access --> READ  
User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcStatus	Status Record Flag. Reserved for future usage. <b>Values:</b> <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: zgrpsacc.grpId = "test" <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by zgrpsacc.grpId <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined.	Optional

## Read Security Group Customer/Site Access

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p><b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.</p> <p><b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.</p> <p><b>Values:</b> &lt;any positive integer value&gt;</p> <p><b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttZGrpSAcc.zgrpsaid value that defines start record for a batch.</p> <p>NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.</p> <p><b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.</p> <p><b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.</p> <p>NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>

# Read Security Group Customer/Site Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchRepositionKeys</b>	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2 &gt; + ... + &lt;TAB&gt; + &lt;TAB&gt;+ &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt;  <b>Default:</b> N/A</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
<b>pcIgnoredInServiceFieldsList</b>	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
<b>plBatchLast</b>	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p> <p><b>Values:</b> YES - The last record was reached;  NO - The last record was NOT reached.</p>	

# Read Security Group Customer/Site Access

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &l. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION

---

## Read Security Group Customer/Site Access

---

**Errors:**

ID	Description
----	-------------

---

RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance.
--	---

Level: OPERATION
------------------



## Add/Update Security Group Customer/Site Access

**Operation:** apSrvZGrpSAccSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZGrpSAcc

**Description:** This operation creates new a security group customer/site access records in the system.

ENTITY TABLE: dsZGrpSAcc.ttZGrpSAcc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGrpSAcc Dataset  
KEY FIELDS: cid,bid,grpId,uid,sid  
MANDATORY FIELDS: cid,bid,grpId,uid,sid  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
- "A" or <empty string> - the ADD record mode

NOTE: There is no update for this operation.  
The system will assign automatically generated values to the Group Site Access ID (zgrpsaid) fields of new records.  
In order to provide access to ALL customers and sites within a particular company/branch, set the corresponding acallsit field value to True by the Add/Update Security Group Company/Branch Access operation.

ERROR HANDLING:  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Security Group Customer/Site Access.

**User Security:** Security Group Customer/Site Access --> ADD  
User should have an administration rights

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields;	Optional

## Add/Update Security Group Customer/Site Access

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p>&lt;comma-separated list&gt; - List of DATA fields that will be updated;</p> <p>&lt;empty string&gt; - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
pcUpdateExceptFieldsList	<p>A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	

### Errors:

ID	Description
NO_CONTEXT_DATA,	<p>&lt;comma separated list of missed context data&gt;</p> <p>List of context fields that are required for service and are not defined.</p> <p>Level: OPERATION</p>
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	<p>More than one service is set to be run by one call. Level: OPERATION</p>
RUN_ERROR,1005	<p>Invalid entry. Level: RECORD</p>
RUN_ERROR,1015	<p>Required entry. Level: RECORD</p>
RUN_ERROR,1020	<p>Record already exists. Level: RECORD</p>
RUN_ERROR,1025	<p>Record is currently locked. Level: RECORD</p>
RUN_ERROR,1065	<p>Record no longer exists. Refresh recommended. Level: OPERATION, RECORD</p>
RUN_ERROR,1240	<p>You do not have security privileges to access this function. Level: OPERATION, RECORD</p>
RUN_ERROR,1435	<p>This function is not licensed. Level: OPERATION</p>
RUN_ERROR,1755,	<p>&lt;Fields Label&gt; + &lt;TAB&gt; + &lt;Max field's size&gt;</p> <p>The &amp;1 field's size exceeds maximum size: &amp;2 character(s). Level: RECORD</p>
RUN_ERROR,1765,	<p>&lt;Fields Label&gt; + &lt;TAB&gt; + &lt;Comma-separated list of available values&gt;</p> <p>The &amp;1 field's value must be one of the following: &amp;2. Level: RECORD</p>
RUN_ERROR,1785,	<p>&lt;Error description from the Process Child records program when record is deleted&gt;</p> <p>&amp;1. Unable to delete. Level: RECORD</p>
RUN_ERROR,1965	<p>There are wrong fields in the requested list: &amp;1. Level: RECORD</p>

---

## Add/Update Security Group Customer/Site Access

---

### Errors:

ID	Description
----	-------------

---

RUN\_ERROR,2075,<Company ID> + <TAB> + <Branch ID>

Unable to add. User already has permission to access all [%sites%] for [%Company%] &1/ [%Branch%] &2. Level: RECORD

RUN\_ERROR,2090

[%Company%]/[%Branch%] access has not been set up for this security group. Level: RECORD

RUN\_ERROR,INVALID\_CONTEXT\_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>

The &1 parameter's value must be one of the following: &2. Level: OPERATION

RUN\_ERROR,LOCKED\_CONTEXT\_DATA

Context record is currently locked. Level: OPERATION

RUN\_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)

[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION

RUN\_ERROR,Service Session failure: Invalid Session ID. (#1670)

Invalid Session ID. Level: OPERATION

RUN\_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)

User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Delete Security Group Customer/Site Access

**Operation:** apSrvZGrpSAccSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZGrpSAcc

**Description:** This operation deletes existing security group customer/site access records from the system.

ENTITY TABLE: dsZGrpSAcc.ttZGrpSAcc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGrpSAcc Dataset  
KEY FIELDS: cid,bid,grpId,uid,sid  
MANDATORY FIELDS: cid,bid,grpId,uid,sid

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete Security Group Customer/Site Access.

**User Security:** Security Group Customer/Site Access --> DELETE  
User should have an administration rights

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required

### OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. Values: See Errors part below
---------------	---

### Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD

---

## Delete Security Group Customer/Site Access

---

**Errors:**

<b>ID</b>	<b>Description</b>
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1705	Invalid [%Customer%] ID. Level: RECORD
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Security Group Employee Access

**Operation:** apSrvZGrpEAccGet  
INPUT-OUTPUT dsContext, OUTPUT dsZGrpEAcc

**Description:** This operation retrieves information about existing security group employee access records from the system. Each record in the dataset contains information about access of one group to one employee record.

ENTITY TABLE: dsZGrpEAcc.ttZGrpEAcc

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGrpEAcc Dataset

KEY FIELDS: cid,bid,grpId,eid

Fields returned by this operation can be of the following three types:

- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and
- SYSTEM FIELDS, required for system purposes.

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.

EXAMPLE: See eTime Xpress Web Services API Examples - Read Security Group Employee Access.

**User Security:** Security Group Employee Access --> READ  
User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required
pcStatus	Status Record Flag. Reserved for future usage. Values: <empty string> - All records.	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: zgrpeacc.grpId = "test" Values: <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by zgrpeacc.grpId Values: <any string> Default: <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined.	Optional

# Read Security Group Employee Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<p><b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields;            &lt;comma-separated list&gt; - List of DATA fields that will be read;            &lt;empty string&gt; - All DATA fields will be read excluding pcReadExceptFieldsList fields</p> <p><b>Default:</b> &lt;empty string&gt;</p>	
<b>pcReadExceptFieldsList</b>	<p>A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "". Otherwise, the parameter will be ignored.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>plBatch</b>	<p>Enables the Batch mode. NOTE: All batch-related input/output parameters make sense only when the Batch mode is enabled: plBatch=YES and piBatchLimit&lt;&gt;0.</p> <p><b>Values:</b> YES - Enables the Batch mode;            NO - Does not enable the Batch mode.</p> <p><b>Default:</b> NO</p>	<b>Optional</b>
<b>piBatchLimit</b>	<p>The maximum number of records in the table which could be returned by the operation. If piBatchLimit=0 then the Batch mode is not enabled regardless of the value of the plBatch parameter. NOTE: If piBatchLimit=1 then the operation can return up to two records, including the current one.</p> <p><b>Values:</b> &lt;any positive integer value&gt;</p> <p><b>Default:</b> 0</p>	<b>Optional</b>
<b>pcBatchKey</b>	<p>A Batch Key is ttZGrpEAcc.zgrpeaid value that defines start record for a batch.</p> <p>NOTE: The value of this parameter is ignored if the Batch Key search is used (see gcBatchSearchKey parameter description).</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchDirection</b>	<p>Direction of the batch.</p> <p><b>Values:</b> NEXT - Next batch. If the pcBatchKey parameter has empty or zero value, it is equivalent to FIRST;            PREV - Previous batch. If the pcBatchKey parameter has empty value, it is equivalent to LAST;            FIRST - First batch;            LAST - Last batch.</p> <p><b>Default:</b> NEXT</p>	<b>Optional</b>
<b>pcBatchSearchKey</b>	<p>A field used in the Batch Key search. If the value of this parameter is not empty, the Batch Key search will be performed. A Batch Key, obtained as a result of this search, will be used in the pcBatchKey parameter, instead of its explicitly specified value. The format of the value of the present parameter is: &lt;table&gt;.&lt;field&gt;, where &lt;field&gt; - any DATA type field in the table (same as in the pcWhere parameter). The details of the Batch Key search algorithm are given in the description of the pcBatchSearchKeyValue parameter.</p> <p>NOTE: If pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record, the operation will still try returning piBatchLimit records. This will be done by adding to the batch a required number of records which go prior to the Batch Key record. In case of the success, the original Batch Key record will not be the first record in the batch.</p> <p><b>Values:</b> &lt;any string&gt;</p> <p><b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>

---

# Read Security Group Employee Access

---

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcBatchSearchKeyValue</b>	<p>A string used in the Batch Key search. For the Batch Key search to be performed, the pcBatchSearchKey parameter should not be empty. Depending on the value of the present parameter, two search algorithms can be used:</p> <p>1) FIRST CHARACTER SEARCH: The operation will try to find a first record (after the current record) whose pcBatchSearchKey field's value starts from a specified character. In this case the pcBatchSearchKeyValue parameter value format is: &lt;TAB&gt; + &lt;TAB&gt; + &lt;Character to search&gt; + &lt;Current record key&gt;, where &lt;Current record key&gt; is the business object ID value for the current record and has the same format as pcBatchKey.</p> <p>2) MULTIPLE CHARACTERS SEARCH: The operation will try to find a first record whose pcBatchSearchKey field value matches as many symbols as possible in the specified string. In this case the pcBatchSearchKeyValue parameter value format is: &lt;String to search&gt;. The comparison will be done on a character by character basis. At the first step, the operation is looking for a first record where the first character in the pcBatchSearchKey field value is equal to the first character in the specified string. Then, starting from just found record, it looks for a first record where the second character in the field value is equal to the second character in the string, and so on. The search is finished when it finds a record whose first characters of the field value are equal to the specified string, or there is a character in the specified string for which the above character by character comparison fails.</p> <p>NOTE: In both algorithms, if a record is not found during the first character comparison, the present operation will return an empty table. The success flag of the search operation is written to the plBatchKeySearchOK parameter, and the Batch Key value of the returned record will be returned in the plBatchOneLetterSearchKey parameter.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> &lt;empty string&gt;</p>	<b>Optional</b>
<b>pcBatchRepositionKeys</b>	<p>A double &lt;TAB&gt;-separated list of reposition (alternative) Batch Keys. If the present operation does not find a business object with the requested business object ID value, then, instead of the value of the pcBatchKey parameter, it will try to use one or more reposition Batch Keys values. The format of the list is: &lt;pcBatchKey1&gt; + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKey2&gt; + ... + &lt;TAB&gt; + &lt;TAB&gt; + &lt;pcBatchKeyN&gt;. If &lt;pcBatchKey1&gt; fails, then the operation will try using &lt;pcBatchKey2&gt;, and so on, until the corresponding business object is found. NOTE: If this parameter is empty, or no objects can be found for all specified reposition Batch Keys, an empty pcBatchKey parameter value will be used by the present operation.</p> <p><b>Values:</b> &lt;any string&gt; <b>Default:</b> N/A</p>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	<p>Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset.</p> <p><b>Values:</b> See Errors part below</p>	
<b>pcIgnoredInServiceFieldsList</b>	<p>A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.</p> <p><b>Values:</b> &lt;any string that corresponds to the description&gt;</p>	
<b>plBatchLast</b>	<p>Last Record flag, which indicates whether the last record was reached in the current batch operation.</p> <p><b>Values:</b> YES - The last record was reached; NO - The last record was NOT reached.</p>	



# Read Security Group Employee Access

## Parameters:

Name	Description	Type
<b>OUTPUT PARAMETERS</b>		
plBatchKeySearchOK	Batch Key search success flag. This parameter indicates whether the Batch Key search, performed using the pcBatchSearchKey and pcBatchSearchKeyValue parameters, was successful. <b>Values:</b> YES - The record was found; NO - The record was NOT found.	
plBatchReturnedExtraData	Indicates whether the present operation returned records starting not from the Batch Key value obtained as a result of the Batch Key search. This is possible if pcBatchDirection = "NEXT" and there are less than piBatchLimit-1 records in the batch after the Batch Key record. <b>Values:</b> YES - The operation returned extra data; NO - The operation did NOT return extra data.	
plBatchOneLetterSearchKey	A value returned by the Batch Key search. If the Batch Key search was successful, this parameter will return the corresponding Batch Key value. If the Batch Key search was not successful, this parameter will contain an empty string. <b>Values:</b> <any string>	

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1060	Record is not available. Level: RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &l. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION

---

## Read Security Group Employee Access

---

**Errors:**

ID	Description
----	-------------

---

RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance.
--	---

Level: OPERATION
------------------

## Add/Update Security Group Employee Access

**Operation:** apSrvZGrpEAccSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZGrpEAcc

**Description:** This operation creates new a security group employee access records in the system.

ENTITY TABLE: dsZGrpEAcc.ttZGrpEAcc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGrpEAcc Dataset  
KEY FIELDS: cid,bid,grpId,eid  
MANDATORY FIELDS: cid,bid,grpId,eid  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
- "A" or <empty string> - the ADD record mode

NOTE: There is no update for this operation.  
The system will assign automatically generated values to the Group Employee Access ID (zgrpeaid) fields of new records.  
In order to provide access to ALL employees within a particular company/branch, set the corresponding acallemp field value to True by the Add/Update Security Group Company/Branch Access operation.

ERROR HANDLING:  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Security Group Employee Access.

**User Security:** Security Group Employee Access --> ADD  
User should have an administration rights

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcUpdateRequestFieldsList	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that	Optional

# Add/Update Security Group Employee Access

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	
pcUpdateExceptFieldsList	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

## OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD

---

## Add/Update Security Group Employee Access

---

### Errors:

ID	Description
RUN_ERROR,2090	[%Company%]/[%Branch%] access has not been set up for this security group. Level: RECORD
RUN_ERROR,2100,<Company ID> + <TAB> + <Branch ID>	Unable to add. User already has permission to access all [%employees%] for [%Company%] &1/ [%Branch%] &2. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Delete Security Group Employee Access

---

**Operation:** apSrvZGrpEAccSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsZGrpEAcc

**Description:** This operation deletes existing security group employee access records from the system.

ENTITY TABLE: dsZGrpEAcc.ttZGrpEAcc  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsZGrpEAcc Dataset  
KEY FIELDS: cid,bid,grpId,eid  
MANDATORY FIELDS: cid,bid,grpId,eid

**ERROR HANDLING:**  
Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete Security Group Employee Access.

**User Security:** Security Group Employee Access --> DELETE  
User should have an administration rights

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

**Errors:**

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD

---

## Delete Security Group Employee Access

---

**Errors:**

<b>ID</b>	<b>Description</b>
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1785,<Error description from the Process Child records program when record is deleted>	&1. Unable to delete. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Pay Rates

**Operation:** apSrvPayRatesGet  
 INPUT-OUTPUT dsContext, OUTPUT dsPayRates

**Description:** This operation retrieves information about existing pay rates records from the system.

ENTITY TABLE: dsPayRates.ttPayRates  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsPayRates Dataset  
 KEY FIELDS: cid,bid,uid,sid,eid,srvcode,pyratety,effdate

Fields returned by this operation can be of the following three types:  
 - DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),  
 - CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and  
 - SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Pay Rates.

**User Security:** Pay Rates Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: payrates.uid = 1 <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by payrates.uid by payrates.sid <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcCustStat	Customer status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcSiteStat	Site status.	Optional



# Read Pay Rates

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	
pcEmpStat	Employee status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be read; <empty string> - All DATA fields will be read excluding pcReadExceptFieldsList fields <b>Default:</b> <empty string>	Optional
pcReadExceptFieldsList	A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

## OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
pcIgnoredInServiceFieldsList	A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values. <b>Values:</b> <any string that corresponds to the description>

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980,<query string>	Invalid QUERY string: &l. Level: OPERATION

---

## Read Pay Rates

---

### Errors:

ID	Description
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Add/Update Pay Rates

---

**Operation:** apSrvPayRatesSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsPayRates

**Description:** This operation creates new and/or updates information about existing pay rates records in the system.

ENTITY TABLE: dsPayRates.ttPayRates  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsPayRates Dataset  
KEY FIELDS: cid,bid,uid,sid,eid,srvcode,pyratety,effdate  
MANDATORY FIELDS: cid,bid,uid,sid,eid,srvcode,pyratety,effdate  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:  
- "A" - the ADD record mode  
- "U" - the UPDATE record mode  
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new pay rates record (ADD record mode), the following condition should be met:

- The values of the KEY FIELDS should be unique in the system.

NOTE: The result list of fields that will be updated in the present operation will be created based on the pcUpdateRequestFieldsList/pcUpdateExceptFieldsList input parameters and it should include all MANDATORY FIELDS mentioned above.

The valid Pay rate type (pyratety) should be set: "B" - Branch pay rates, "E" - Employee pay rates, "S" - Site pay rates, "X" - Combination pay rates.

NOTE: When Pay rate type field is set to "B", the Customer ID (uid), Site ID (sid), Employee ID (eid) fields values will be ignored.

When Pay rate type field is set to "E", the Customer ID (uid), Site ID (sid) fields values will be ignored.

When Pay rate type field is set to "S", the Employee ID (eid) field's value will be ignored.

When Use Custom Factors (usecustom) field is set to No, the factor-related fields will be ignored: dtfactortype,dtfactor,otfactor,otfactortype,stfactortype,stfactor,trfactortypetrfactor,ohfactortype,ohfactor.

### ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.

Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Pay Rates.

**User Security:** Pay Rates Web Services --> ADD  
Pay Rates Web Services --> UPDATE

**Parameters:**

# Add/Update Pay Rates

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcAction</b>	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	<b>Required</b>
<b>pcSessionID</b>	Unique Session ID string. <b>Values:</b> <any string>	<b>Required</b>
<b>pcUserID</b>	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	<b>Required</b>
<b>piCompanyID</b>	Unique Company ID. <b>Values:</b> 1..999	<b>Required</b>
<b>piBranchID</b>	Branch ID (unique within the Company). <b>Values:</b> 1..999	<b>Required</b>
<b>pcUpdateRequestFieldsList</b>	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcUpdateExceptFieldsList</b>	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcCustStat</b>	Customer status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcSiteStat</b>	Site status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcEmpStat</b>	Employee status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	<b>Optional</b>
<b>plRecalculateShifts</b>	Shifts Distribution Recalculation flag. <b>Values:</b> YES - Apply the shifts distribution recalculation; NO - Does not apply the shifts distribution recalculation. <b>Default:</b> NO	<b>Optional</b>
<b>plImportEnableUpdate</b>	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records. <b>Default:</b> YES	<b>Optional</b>
<b>plUseBeforeImage</b>	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the	<b>Optional</b>

# Add/Update Pay Rates

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	
pcConcurrencyFieldList	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

## OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
---------------	--

## Errors:

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1005	Invalid entry. Level: RECORD
RUN_ERROR,1015	Required entry. Level: RECORD
RUN_ERROR,1020	Record already exists. Level: RECORD
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1535	All required fields must be included in the import. Level: RECORD
RUN_ERROR,1665	Invalid [%Customer%] / [%Site%] combination. Level: RECORD
RUN_ERROR,1685	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD

---

## Add/Update Pay Rates

---

### Errors:

ID	Description
RUN_ERROR,1705	Invalid [%Customer%] ID. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1815	Invalid [%Service%]. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Delete Pay Rates

---

**Operation:** apSrvPayRatesSave

INPUT-OUTPUT dsContext, INPUT-OUTPUT dsPayRates

**Description:** This operation deletes existing pay rates records from the system.

ENTITY TABLE: dsPayRates.ttPayRates

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsPayRates Dataset

KEY FIELDS: cid,bid,uid,sid,eid,srvcode,pyratety,effdate

MANDATORY FIELDS: cid,bid,uid,sid,eid,srvcode,pyratety,effdate

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.

Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Pay Rates.

**User Security:** Pay Rates Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcCustStat	Customer status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcSiteStat	Site status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcEmpStat	Employee status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
plRecalculateShifts	Shifts Distribution Recalculation flag. <b>Values:</b> YES - Apply the shifts distribution recalculation; NO - Does not apply the shifts distribution recalculation. <b>Default:</b> NO	Optional

## Delete Pay Rates

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<code>plUseBeforeImage</code>	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: <code>plUseBeforeImage=YES</code> . Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
<code>pcConcurrencyFieldList</code>	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
<b>OUTPUT PARAMETERS</b>		
<code>ExecuteStatus</code>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

### Errors:

ID	Description
<code>NO_CONTEXT_DATA,</code>	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
<code>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</code>	More than one service is set to be run by one call. Level: OPERATION
<code>RUN_ERROR,1025</code>	Record is currently locked. Level: RECORD
<code>RUN_ERROR,1065</code>	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
<code>RUN_ERROR,1240</code>	You do not have security privileges to access this function. Level: OPERATION, RECORD
<code>RUN_ERROR,1435</code>	This function is not licensed. Level: OPERATION
<code>RUN_ERROR,1665</code>	Invalid [%Customer%] / [%Site%] combination. Level: RECORD
<code>RUN_ERROR,1685</code>	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
<code>RUN_ERROR,1700</code>	Invalid [%Employee%] ID. Level: RECORD
<code>RUN_ERROR,1705</code>	Invalid [%Customer%] ID. Level: RECORD
<code>RUN_ERROR,1980,</code>	<query string> Invalid QUERY string: &l. Level: OPERATION
<code>RUN_ERROR,2185</code>	Record has been changed by another user or system process. Refresh recommended. Level: RECORD



---

## Delete Pay Rates

---

### Errors:

ID	Description
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Read Bill Rates

---

**Operation:** apSrvBillRateGet  
INPUT-OUTPUT dsContext, OUTPUT dsBillRate

**Description:** This operation retrieves information about existing bill rates records from the system.

ENTITY TABLE: dsBillRate.ttBillRate

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsBillRate Dataset

KEY FIELDS: cid,bid,uid,sid,eid,bisrtycd,blratety,effdate

Fields returned by this operation can be of the following three types:

- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Bill Rates.

**User Security:** Bill Rates Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: billrate.uid = 1 <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by billrate.uid by billrate.sid <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
pcCustStat	Customer status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcSiteStat	Site status.	Optional

# Read Bill Rates

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	
pcEmpStat	Employee status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcReadRequestFieldsList	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be read; <empty string> - All DATA fields will be read excluding pcReadExceptFieldsList fields <b>Default:</b> <empty string>	Optional
pcReadExceptFieldsList	A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional

## OUTPUT PARAMETERS

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below
pcIgnoredInServiceFieldsList	A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values. <b>Values:</b> <any string that corresponds to the description>

## Errors:

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &l. Level: RECORD
RUN_ERROR,1980,<query string>	Invalid QUERY string: &l. Level: OPERATION

---

## Read Bill Rates

---

### Errors:

ID	Description
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &l parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Add/Update Bill Rates

---

**Operation:** apSrvBillRateSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsBillRate

**Description:** This operation creates new and/or updates information about existing bill rates records in the system.

ENTITY TABLE: dsBillRate.ttBillRate  
FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsBillRate Dataset  
KEY FIELDS: cid,bid,uid,sid,eid,bisrtycd,blratety,effdate  
MANDATORY FIELDS: cid,bid,uid,sid,eid,bisrtycd,blratety,effdate  
IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The cChkNew field is used as a record mode flag and can have the following values:

- "A" - the ADD record mode
- "U" - the UPDATE record mode
- <empty string> - the AUTO DETECT mode: if the value of the plImportEnableUpdate parameter is set to YES (default), and there is already a record in the system with the same KEY FIELDS as in the current record, then the record will be in the UPDATE record mode. Otherwise, the record will be in the ADD record mode.

NOTE: To add new bill rates record (ADD record mode), the following condition should be met:

- The values of the KEY FIELDS should be unique in the system.

NOTE: The result list of fields that will be updated in the present operation will be created based on the pcUpdateRequestFieldsList/pcUpdateExceptFieldsList input parameters and it should include all MANDATORY FIELDS mentioned above.

The valid Bill rate type (blratety) should be set: "B" - Branch bill rates, "E" - Employee bill rates, "S" - Site bill rates, "X" - Combination bill rates.

NOTE: When Bill rate type field is set to "B", the Customer ID (uid), Site ID (sid), Employee ID (eid) fields values will be ignored.

When Bill rate type field is set to "E", the Customer ID (uid), Site ID (sid) fields values will be ignored.

When Bill rate type field is set to "S", the Employee ID (eid) field's value will be ignored.

### ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.  
Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Bill Rates.

**User Security:** Bill Rates Web Services --> ADD  
Bill Rates Web Services --> UPDATE

### Parameters:

Name	Description	Type
------	-------------	------

---

### INPUT PARAMETERS

# Add/Update Bill Rates

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<b>pcAction</b>	Action for present operation. <b>Values:</b> ADD_UPDATE - Add/Update action is requested	<b>Required</b>
<b>pcSessionID</b>	Unique Session ID string. <b>Values:</b> <any string>	<b>Required</b>
<b>pcUserID</b>	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	<b>Required</b>
<b>piCompanyID</b>	Unique Company ID. <b>Values:</b> 1..999	<b>Required</b>
<b>piBranchID</b>	Branch ID (unique within the Company). <b>Values:</b> 1..999	<b>Required</b>
<b>pcUpdateRequestFieldsList</b>	"*" string or a comma-separated list of DATA fields that will be updated in the present operation. The system defaults its value to "*" when both pcUpdateRequestFieldsList and pcUpdateExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be updated; <empty string> - All DATA fields will be updated excluding pcUpdateExceptFieldsList fields <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcUpdateExceptFieldsList</b>	A comma-separated list of DATA fields that will be excluded from update by the present operation when pcUpdateRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcCustStat</b>	Customer status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcSiteStat</b>	Site status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcEmpStat</b>	Employee status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	<b>Optional</b>
<b>plRecalculateShifts</b>	Shifts Distriburion Recalculation flag. <b>Values:</b> YES - Apply the shifts distriburion recalculation; NO - Does not apply the shifts distriburion recalculation. <b>Default:</b> NO	<b>Optional</b>
<b>plImportEnableUpdate</b>	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the cChkNew field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records.	<b>Optional</b>

# Add/Update Bill Rates

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Default:</b> YES	
<b>plUseBeforeImage</b>	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	Optional
<b>pcConcurrencyFieldList</b>	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	Optional
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

## Errors:

ID	Description
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	List of context fields that are required for service and are not defined. Level: OPERATION
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION
<b>RUN_ERROR,1005</b>	Invalid entry. Level: RECORD
<b>RUN_ERROR,1015</b>	Required entry. Level: RECORD
<b>RUN_ERROR,1020</b>	Record already exists. Level: RECORD
<b>RUN_ERROR,1025</b>	Record is currently locked. Level: RECORD
<b>RUN_ERROR,1065</b>	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level: OPERATION, RECORD
<b>RUN_ERROR,1435</b>	This function is not licensed. Level: OPERATION
<b>RUN_ERROR,1535</b>	All required fields must be included in the import. Level: RECORD
<b>RUN_ERROR,1665</b>	Invalid [%Customer%] / [%Site%] combination. Level: RECORD
<b>RUN_ERROR,1685</b>	Invalid [%Company%] / [%Branch%] combination. Level: RECORD

---

## Add/Update Bill Rates

---

### Errors:

ID	Description
RUN_ERROR,1700	Invalid [%Employee%] ID. Level: RECORD
RUN_ERROR,1705	Invalid [%Customer%] ID. Level: RECORD
RUN_ERROR,1755,<Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
RUN_ERROR,1760,<Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
RUN_ERROR,1765,<Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
RUN_ERROR,1815	Invalid [%Service%]. Level: RECORD
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,2185	Record has been changed by another user or system process. Refresh recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION



## Delete Bill Rates

**Operation:** apSrvBillRateSave  
INPUT-OUTPUT dsContext, INPUT-OUTPUT dsBillRate

**Description:** This operation deletes existing bill rates records from the system.

ENTITY TABLE: dsBillRate.ttBillRate

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsBillRate Dataset

KEY FIELDS: cid,bid,uid,sid,eid,bisrtycd,blratety,effdate

MANDATORY FIELDS: cid,bid,uid,sid,eid,bisrtycd,blratety,effdate

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors.

Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Delete Bill Rates.

**User Security:** Bill Rates Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. <b>Values:</b> DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcCustStat	Customer status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcSiteStat	Site status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
pcEmpStat	Employee status. <b>Values:</b> A - Active records; I - Inactive records; V - Void records; <empty string> - All records. <b>Default:</b> <empty string>	Optional
plRecalculateShifts	Shifts Distriburion Recalculation flag. <b>Values:</b> YES - Apply the shifts distriburion recalculation; NO - Does not apply the shifts distriburion recalculation.	Optional

## Delete Bill Rates

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	<b>Default:</b> NO	
<b>plUseBeforeImage</b>	Enables the concurrency mode. NOTE: All concurrency-related input/output parameters make sense only when the concurrency mode is enabled: plUseBeforeImage=YES. Operation should receive Before Image for the INPUT-OUPTUT ENTITY dataset. <b>Values:</b> YES - Enables the concurrency mode; NO - Does not enable the concurrency mode. <b>Default:</b> NO	<b>Optional</b>
<b>pcConcurrencyFieldList</b>	A comma-separated list of DATA FIELDS that will be used for the concurrency check. If the list is empty string or it is not defined, the operation will use all DATA FIELDS. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>OUTPUT PARAMETERS</b>		
<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	

### Errors:

ID	Description
<b>NO_CONTEXT_DATA,</b> <comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION
<b>RUN_ERROR,1025</b>	Record is currently locked. Level: RECORD
<b>RUN_ERROR,1065</b>	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level: OPERATION, RECORD
<b>RUN_ERROR,1435</b>	This function is not licensed. Level: OPERATION
<b>RUN_ERROR,1665</b>	Invalid [%Customer%] / [%Site%] combination. Level: RECORD
<b>RUN_ERROR,1685</b>	Invalid [%Company%] / [%Branch%] combination. Level: RECORD
<b>RUN_ERROR,1700</b>	Invalid [%Employee%] ID. Level: RECORD
<b>RUN_ERROR,1705</b>	Invalid [%Customer%] ID. Level: RECORD
<b>RUN_ERROR,1980,</b> <query string>	Invalid QUERY string: &l. Level: OPERATION
<b>RUN_ERROR,2185</b>	Record has been changed by another user or system process. Refresh

---

## Delete Bill Rates

---

### Errors:

ID	Description
	recommended. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Shift Allowances

**Operation:** apSrvShAllownGet  
 INPUT-OUTPUT dsContext, INPUT-OUTPUT dsShAllown

**Description:** This operation retrieves information about existing Shift Allowances records from the system.

ENTITY TABLE: dsShAllown.ttShAllown  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShAllown Dataset  
 KEY FIELDS: shiftid, dbRow

Fields returned by this operation can be of the following three types:

- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**  
 Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Shift Allowances.

**User Security:** Allowance Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: shifts.uid = 1 <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system. NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. Parameters is not used by operation. <b>Values:</b> <any string> <b>Default:</b> <empty string>	Optional
<b>OUTPUT PARAMETERS</b>		
ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below	
pcIgnoredInServiceFieldsList	A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values.	

---

## Read Shift Allowances

---

### Parameters:

Name	Description	Type
------	-------------	------

---

### OUTPUT PARAMETERS

Values: <any string that corresponds to the description>

### Errors:

ID	Description
NO_CONTEXT_DATA,	<comma separated list of missed context data> List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,2225	This shift has not been worked. Unable to add/update allowances. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

---

## Add/Update Shift Allowances

---

**Operation:** apSrvShAllownSave

INPUT-OUTPUT dsContext, INPUT-OUTPUT dsShAllown

**Description:** This operation creates new and/or updates information about existing Shift Allowances records in the system.

ENTITY TABLE: dsShAllown.ttShAllown

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShAllown Dataset

KEY FIELDS: shiftid, dbRow

MANDATORY FIELDS: shiftid, dbRow

IGNORED FIELDS: all DATA-READ fields, all CALCULATED fields, all SYSTEM fields

The dbRow field is used as a record mode flag and can have the following values:

- <empty string> - the ADD record mode

- <non-empty string> - the UPDATE record mode. It will use dbRow as a key value.

NOTE: To add new Shift Allowances record (ADD record mode), the following condition should be met:

- The shiftid value should correspond to valid worked shift record.

- The dbRow field should be an empty string.

To update Shift Allowances record (UPDATE record mode), the following condition should be met:

- The shiftid value should correspond to valid worked shift record.

- The dbRow field should correspond to valid shift allowance record.

ERROR HANDLING:

Use ExecuteStatus parameter to detect OPERATION level errors.

Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.

If a record cannot be added/updated, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to add/update remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Add/Update Shift Allowances.

**User Security:** Allowance Web Services --> ADD  
Allowance Web Services --> UPDATE

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: ADD_UPDATE - Add/Update action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required

## Add/Update Shift Allowances

### Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
<code>plImportEnableUpdate</code>	A flag specifying whether the present operation can update existing records. This parameter is applicable only to the records in the table, which are in the AUTO DETECT mode (have empty value of the <code>cChkNew</code> field). <b>Values:</b> YES - Allows to update existing records; NO - Does not allow to update existing records. <b>Default:</b> YES	Optional

### OUTPUT PARAMETERS

<code>ExecuteStatus</code>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output <code>dsContext</code> dataset. <b>Values:</b> See Errors part below
----------------------------	---

### Errors:

ID	Description
<code>NO_CONTEXT_DATA</code> , <comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
<code>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</code>	More than one service is set to be run by one call. Level: OPERATION
<code>RUN_ERROR,1005</code>	Invalid entry. Level: RECORD
<code>RUN_ERROR,1015</code>	Required entry. Level: RECORD
<code>RUN_ERROR,1020</code>	Record already exists. Level: RECORD
<code>RUN_ERROR,1025</code>	Record is currently locked. Level: RECORD
<code>RUN_ERROR,1065</code>	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD
<code>RUN_ERROR,1240</code>	You do not have security privileges to access this function. Level: OPERATION, RECORD
<code>RUN_ERROR,1435</code>	This function is not licensed. Level: OPERATION
<code>RUN_ERROR,1755</code> , <Fields Label> + <TAB> + <Max field's size>	The &1 field's size exceeds maximum size: &2 character(s). Level: RECORD
<code>RUN_ERROR,1760</code> , <Fields Label> + <TAB> + <Min decimal value> + <TAB> + <Max decimal value>	The &1 field's value cannot be less than &2 and greater than &3. Level: RECORD
<code>RUN_ERROR,1765</code> , <Fields Label> + <TAB> + <Comma-separated list of available values>	The &1 field's value must be one of the following: &2. Level: RECORD
<code>RUN_ERROR,1965</code>	There are wrong fields in the requested list: &1. Level: RECORD
<code>RUN_ERROR,1980</code> , <query string>	Invalid QUERY string: &1. Level: OPERATION

---

## Add/Update Shift Allowances

---

### Errors:

ID	Description
----	-------------

---

RUN\_ERROR,2225

This shift has not been worked. Unable to add/update allowances. Level:  
RECORD

RUN\_ERROR,INVALID\_CONTEXT\_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of  
available values>

The &1 parameter's value must be one of the following: &2. Level:  
OPERATION

RUN\_ERROR,LOCKED\_CONTEXT\_DATA

Context record is currently locked. Level: OPERATION

RUN\_ERROR,Service Session failure: Company/Branch access has not been set up for this user.  
(#1170)

[%Company%]/[%Branch%] access has not been set up for this user. Level:  
OPERATION

RUN\_ERROR,Service Session failure: Invalid Session ID. (#1670)

Invalid Session ID. Level: OPERATION

RUN\_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for  
assistance. (#1460)

User ID is not valid. Contact your system administrator for assistance.  
Level: OPERATION



## Delete Shift Allowances

**Operation:** apSrvShAllownSave  
 INPUT-OUTPUT dsContext, INPUT-OUTPUT dsShAllown

**Description:** This operation deletes existing Shift Allowances records from the system.

ENTITY TABLE: dsShAllown.ttShAllown  
 FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShAllown Dataset  
 KEY FIELDS: dbRow  
 MANDATORY FIELDS: dbRow

ERROR HANDLING:  
 Use ExecuteStatus parameter to detect OPERATION level errors.  
 Use cValmess and cObjectName fields in returned table to detect RECORD level errors. If such an error occurred, these fields will contain error ID and name of the object, which caused it.  
 If a record cannot be deleted, the operation will not stop: only failed record transaction will be rolled back. The operation will proceed to delete remaining records.

EXAMPLE: See eTime Xpress Web Services API Examples - Delete Shift Allowances.

**User Security:** Allowance Web Services --> DELETE

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcAction	Action for present operation. Values: DELETE - Delete action is requested	Required
pcSessionID	Unique Session ID string. Values: <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. Values: <any string>	Required
piCompanyID	Unique Company ID. Values: 1..999	Required
piBranchID	Branch ID (unique within the Company). Values: 1..999	Required

**OUTPUT PARAMETERS**

ExecuteStatus	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. Values: See Errors part below
---------------	---

**Errors:**

ID	Description
NO_CONTEXT_DATA,<comma separated list of missed context data>	List of context fields that are required for service and are not defined. Level: OPERATION
RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE	More than one service is set to be run by one call. Level: OPERATION
RUN_ERROR,1025	Record is currently locked. Level: RECORD
RUN_ERROR,1065	Record no longer exists. Refresh recommended. Level: OPERATION, RECORD

---

## Delete Shift Allowances

---

### Errors:

ID	Description
RUN_ERROR,1240	You do not have security privileges to access this function. Level: OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,2225	This shift has not been worked. Unable to add/update allowances. Level: RECORD
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

## Read Shifts Tasks

**Operation:** apSrvShiftToDoGet INPUT-OUTPUT dsContext, OUTPUT dsShiftsWithTasks

**Description:** This operation retrieves information about existing shifts tasks records from the system.

ENTITY TABLE: dsShiftsWithTasks.ttShifts

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShiftsWithTasks Dataset

KEY FIELDS: shiftid

ENTITY TABLE: dsShiftsWithTasks.ttToDo

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShiftsWithTasks Dataset

KEY FIELDS: todoid

DATA RELATIONSHIP: dsShiftsWithTasks.ttShifts.shiftid - 1:N -> dsShiftsWithTasks.ttToDo.shiftid

Fields returned by this operation can be of the following three types:

- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Shifts Tasks.

**User Security:** Shifts Web Services --> READ  
Shift Tasks Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pdFromDate	Shift's Scheduled From Date. <b>Values:</b> <any valid date string that corresponds to the eTime Xpress Datatime data type>. For example, MM/DD/YYYY	Required
pdToDate	Shift's Scheduled To Date. <b>Values:</b> <any valid date string that corresponds to the eTime Xpress Datatime data type>. For example, MM/DD/YYYY	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: shifts.cid < 1 <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system.	Optional

# Read Shifts Tasks

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by shifts.shdate <b>Values:</b> <any string> <b>Default:</b> <empty string>	
<b>pcReadRequestFieldsList</b>	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be read; <empty string> - All DATA fields will be read excluding pcReadExceptFieldsList fields <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcReadExceptFieldsList</b>	A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>plGetMidnightDifference</b>	Performs additional calculations related with Midnight Cut Off. Additional Midnight cutoff calculations process the portion of the shift occurring after midnight separately filling-in the following fields: ttShifts.MidDiffScheduled, ttShifts.MidDiffWorked, ttShifts.MidDiffOverlap1, ttShifts.MidDiffOverlap2, ttShifts.MidDiffWorkedOverlap1, ttShifts.MidDiffWorkedOverlap2. See more fields description in eTime Xpress Web Services API Datasets - dsShifts Dataset. <b>Values:</b> YES - Performs additional calculations related with Midnight Cutoff; NO - Does not perform additional calculations related with Midnight Cutoff. <b>Default:</b> NO	<b>Optional</b>

## OUTPUT PARAMETERS

<b>pcIgnoredInServiceFieldsList</b>	A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values. <b>Values:</b> <any string that corresponds to the description>
<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below

## Errors:

ID	Description
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	List of context fields that are required for service and are not defined. Level: OPERATION
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level:

---

## Read Shifts Tasks

---

### Errors:

ID	Description
	OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION

# Read Shifts Events

**Operation:** apSrvShiftEventGet INPUT-OUTPUT dsContext, OUTPUT dsShiftWithEvent

**Description:** This operation retrieves information about existing shifts event records from the system.

ENTITY TABLE: dsShiftsWithEvent.ttShifts

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShiftsWithEvent Dataset

KEY FIELDS: shiftid

ENTITY TABLE: dsShiftsWithEvent.ttEventReadOnly

FIELDS DESCRIPTION: See eTime Xpress Web Services API Datasets - dsShiftsWithEvent Dataset

KEY FIELDS: eventUuid

DATA RELATIONSHIP: dsShiftsWithEvent.ttShifts.eventUuid - 1:1 -> dsShiftsWithEvent.ttEventReadOnly.eventUuid

Fields returned by this operation can be of the following three types:

- DATA FIELDS, which contain "raw" object information (some of these fields are marked as DATA-READ - see more information in the description of the Add/Update operation),
- CALCULATED FIELDS, whose values are calculated on the basis of the "raw" data, and
- SYSTEM FIELDS, required for system purposes.

**ERROR HANDLING:**

Use ExecuteStatus parameter to detect OPERATION level errors.

**EXAMPLE:** See eTime Xpress Web Services API Examples - Read Shifts Events.

**User Security:** Shifts Web Services --> READ  
Shifts Events Web Services --> READ

**Parameters:**

Name	Description	Type
<b>INPUT PARAMETERS</b>		
pcSessionID	Unique Session ID string. <b>Values:</b> <any string>	Required
pcUserID	Unique User ID, under which the operation is executed. <b>Values:</b> <any string>	Required
piCompanyID	Unique Company ID. <b>Values:</b> 1..999	Required
piBranchID	Branch ID (unique within the Company). <b>Values:</b> 1..999	Required
pdFromDate	Shift's Scheduled From Date. <b>Values:</b> <any valid date string that corresponds to the eTime Xpress Datatime data type>. For example, MM/DD/YYYY	Required
pdToDate	Shift's Scheduled To Date. <b>Values:</b> <any valid date string that corresponds to the eTime Xpress Datatime data type>. For example, MM/DD/YYYY	Required
pcWhere	An additional WHERE clause expression for the 4GL SELECT query that retrieves data from the system. Example: shifts.cid < 1 <b>Values:</b> <any string>	Required
pcSort	An additional SORT clause expression for the 4GL SELECT query that retrieves data from the system.	Optional

# Read Shifts Events

## Parameters:

Name	Description	Type
<b>INPUT PARAMETERS</b>		
	NOTE: Records in the returned table are sorted by default primary index and their order does not depend on the value of the present parameter. This parameter makes sense only when used in the Batch mode - sorting of records is performed before selecting them for a batch. Example: by shifts.shdate <b>Values:</b> <any string> <b>Default:</b> <empty string>	
<b>pcReadRequestFieldsList</b>	"*" string or a comma-separated list of fields that will be retrieved in the present operation. The system defaults its value to "*" when both pcReadRequestFieldsList and pcReadExceptFieldsList parameters are not defined. <b>Values:</b> * - All DATA fields will be read excluding pcReadExceptFieldsList fields; <comma-separated list> - List of DATA fields that will be read; <empty string> - All DATA fields will be read excluding pcReadExceptFieldsList fields <b>Default:</b> <empty string>	<b>Optional</b>
<b>pcReadExceptFieldsList</b>	A comma-separated list of fields that will be excluded from read by the present operation when pcReadRequestFieldsList is empty or "*". Otherwise, the parameter will be ignored. <b>Values:</b> <any string that corresponds to the description> <b>Default:</b> <empty string>	<b>Optional</b>
<b>plGetMidnightDifference</b>	Performs additional calculations related with Midnight Cut Off. Additional Midnight cutoff calculations process the portion of the shift occurring after midnight separately filling-in the following fields: ttShifts.MidDiffScheduled,ttShifts.MidDiffWorked,ttShifts.MidDiffOverlap1,ttShifts.MidDiffOverlap2,ttShifts.MidDiffWorkedOverlap1,ttShifts.MidDiffWorkedOverlap2. See more fields description in eTime Xpress Web Services API Datasets - dsShifts Dataset. <b>Values:</b> YES - Performs additional calculations related with Midnight Cutoff; NO - Does not perform additional calculations related with Midnight Cutoff. <b>Default:</b> NO	<b>Optional</b>

## OUTPUT PARAMETERS

<b>pcIgnoredInServiceFieldsList</b>	A comma-separated list of fields that are restricted for the current user (pcUserID). Corresponding fields in the table will have default values. <b>Values:</b> <any string that corresponds to the description>
<b>ExecuteStatus</b>	Operation status. If there are no OPERATION level errors, this parameter will be excluded from the output dsContext dataset. <b>Values:</b> See Errors part below

## Errors:

ID	Description
<b>NO_CONTEXT_DATA,&lt;comma separated list of missed context data&gt;</b>	List of context fields that are required for service and are not defined. Level: OPERATION
<b>RUN_ERROR, WRONG_CONTEXT_DATA_SERVICE</b>	More than one service is set to be run by one call. Level: OPERATION
<b>RUN_ERROR,1240</b>	You do not have security privileges to access this function. Level:

---

## Read Shifts Events

---

### Errors:

ID	Description
	OPERATION, RECORD
RUN_ERROR,1435	This function is not licensed. Level: OPERATION
RUN_ERROR,1965	There are wrong fields in the requested list: &1. Level: RECORD
RUN_ERROR,1980, <query string>	Invalid QUERY string: &1. Level: OPERATION
RUN_ERROR,INVALID_CONTEXT_DATA,<Context Parameter Name> + <TAB> + <Comma-separated list of available values>	The &1 parameter's value must be one of the following: &2. Level: OPERATION
RUN_ERROR,LOCKED_CONTEXT_DATA	Context record is currently locked. Level: OPERATION
RUN_ERROR,Service Session failure: Company/Branch access has not been set up for this user. (#1170)	[%Company%]/[%Branch%] access has not been set up for this user. Level: OPERATION
RUN_ERROR,Service Session failure: Invalid Session ID. (#1670)	Invalid Session ID. Level: OPERATION
RUN_ERROR,Service Session failure: User ID is not valid. Contact your system administrator for assistance. (#1460)	User ID is not valid. Contact your system administrator for assistance. Level: OPERATION







