

Automating User Import to the Inquisiq R3 LMS

Importing user information from back-end systems such as HR software has been greatly simplified in the Inquisiq R3 LMS. The process involves two basic steps – inserting the raw user account data into the database, and invoking the stored procedure that merges the raw data into the live user account data.

Insert the Raw User Data

The first step is to get your raw data into the database so that it can be processed. There are various ways to do this including manual or automated DTS or SSIS packages and bulk insert. The process you choose and the steps to implement are far beyond the scope of this document; however whatever process you choose should insert the data according to the details outlined here.

Your data should be inserted to the **tblUserBatch** table. This is the same table that the built-in User Batch process uses to store the raw data prior to merging it with the live user account data. This is the only table that you should insert your data into.

idSite Value

The idSite value is used to determine which LMS account on the server should receive the imported user data. It is important that this value be correct or your users will be imported into the wrong LMS account. You can find the idSite values by looking in the idSite column of the tblSite table.

Username Value

The username value is very important during the import process as that is the unique identifier that will be used to find and update existing users.

If the username value corresponds to an existing user account in the site specified by the idSite value, then that existing user account will be updated with the information contained in the user data record with the exception of the password (see below).

If the username value that you enter does not already exist as a user account in the site specified by the idSite value, then a new user account will be created with the information contained in the user data record including password.

Password Value

Passwords are required for all user accounts; null values are not allowed in either the user table or the user batch table. When updating existing user accounts you may want to retain the password that is contained within the LMS in case the user has changed it there. To do so, enter an empty string ("") in the password column of the user batch table and upon synchronization, the LMS will retain the existing password for that record rather than using the value from the user batch table.

Note that you may not do this for new user accounts; you must enter a non-null, non-empty value in the password column.

In either case, if you would like to force the user to change their password the next time they access the system, enter a “true” bit value (1) in the following column labeled mustChangePassword.

Token Value

In order to be able to distinguish the records associated with your user batch from others, you will need to generate and insert a token into the token column in each record. This token should not already exist in any record in the tblUserBatch table and should be the same for all records that you insert during each “run” of this process.

Note that if a previous insert process has failed, you may also use the token value to remove the records that were inserted during that process since the token value will be the same for all of them.

User Batch Table

Use the table below as a reference for inserting raw user data into the tblUserBatch table. Columns indicating “yes” in the Req. column are required for proper processing.

Note that each LMS account may be configured differently with regard to user fields that are required as well as formatting rules for those fields. At the moment, the import process does not enforce these rules, so your data should comply prior to being inserting. For a list of the rules and requirements as defined within your LMS account, go to Administrator Menu / Users / New User Batch.

Column	Datatype/Format	Req.	Notes
idSite	int	yes	The value entered in this column should correspond to the idSite value of the LMS account that your user data will be imported to. This value can be found in the idSite column of the tblSite table.
name.firstname	nvarchar(255)	yes	
name.middlename	nvarchar(255)		
name.lastname	nvarchar(255)	yes	
email	nvarchar(255)		
username	nvarchar(512)	yes	The value entered here will be used to lookup an existing user account. If no account is found, a new account will be created with the data found in this record.
password	nvarchar(512)	yes	
mustChangePassword	bit		A “true” value forces the user to change their password the next time they access the system. 1 = “true”, 0 = “false”
expires	datetime		Expiration date of the user account. After this date, the user will no longer be able to login.
company	nvarchar(255)		
address.street	nvarchar(512)		
address.city	nvarchar(255)		
address.province	nvarchar(255)		
address.postalcode	nvarchar(25)		

address.country	nvarchar(50)		
phone.primary	nvarchar(25)		
phone.work	nvarchar(25)		
phone.fax	nvarchar(25)		
phone.home	nvarchar(25)		
phone.mobile	nvarchar(25)		
phone.pager	nvarchar(25)		
phone.other	nvarchar(25)		
employeeid	nvarchar(255)		
jobtitle	nvarchar(255)		
jobclass	nvarchar(255)		
division	nvarchar(255)		
region	nvarchar(255)		
department	nvarchar(255)		
supervisor	nvarchar(255)		Username of the user's supervisor
hiredate	datetime		
termdate	datetime		
gender	nvarchar(1)		"m" or "f"
race	nvarchar(64)		
ssn	nvarchar(9)		
dob	datetime		
field00	nvarchar(255)		
field01	nvarchar(255)		
field02	nvarchar(255)		
field03	nvarchar(255)		
field04	nvarchar(255)		
field05	nvarchar(255)		
field06	nvarchar(255)		
field07	nvarchar(255)		
field08	nvarchar(255)		
field09	nvarchar(255)		
token	nvarchar(20)	Yes	Insert the same token for all records within your batch so that they can be differentiated from other batches that may be running at the same time. It is recommended that you insert a full, 20 character token. This token will be passed to the stored procedure in the next step in so that it only processes records containing this token.
timestamp	datetime	Yes	Insert the current GMT timestamp using the SQL command: getutcdate()

Merge the Raw User Data

To merge your raw user data with the existing live user account data, simply execute the stored procedure [system.synchronizeOnBatchUserTable].

This procedure requires 3 parameters as follows:

- idSite (integer) - the idSite value of the LMS account that you are merging to. This value should correspond to the idSite value that was inserted with your raw user data. Only records containing this idSite and the token noted below will be merged.
- token (nvarchar[20]) – corresponds to the token value that was inserted into the token column of your raw user data records. Only records containing this token and the idSite noted above will be merged.
- appendMode (bit) – indicates how the raw user data should be merged.
 - 1 appends the raw user data to the existing user base.
 - 0 replaces the existing user base with the raw user data (existing users that are not found in the raw user data will be disabled).

When completed successfully, this procedure will clear the raw user data that was just merged.

Validation

Note that because this process inserts data directly to the database on a potentially large scale, validation of server-configured rules (as defined in the /_lib/_global.asp file) and site-configured rules (as defined on the User Account Data interface) are not currently performed. Therefore, data that does not comply with these rules will be allowed, however any subsequent attempt to update or save a user account containing such invalid data within the LMS interface will be rejected unless the invalid data is corrected within that interface.

Also note that while invalid data will prevent a user account from being updated or saved within the LMS interface, it will not prevent the user from logging into the system with that account.