

IccMySQLConsole Manual

Contents

Description	1
Installation	2
Pre-requisites	2
Steps.....	2
Logic File	2
Purpose	2
Syntax.....	2
Keys.....	2
Creating Encrypted Key Values.....	8
Running The Program	8
Using the default 'lccMySQLConsole-logic.txt' path name	8
Defining a custom Logic File path.....	8
Testing Connection Values For ODBC Driver	9
Example Logic Files.....	9
Read two tables and output to two files.....	9
Read a table and only output records who fall between two dates.	9
Definitions.....	10
Modifications	10

Description

This document describes the lccMySQLConsole program.

The lccMySQLConsole is a Command Line program that performs actions against a MySQL server/database.

By default, the program will log out to a log:

[lccMySQLConsoleLog-YYYYMMDD.log](#)

This can be overridden with the Logic File key 'lcc:logPath' (see the Logic File section below).

Installation

Pre-requisites

- install MySQL ODBC Driver (**32 bit version**), ref:
- <https://dev.mysql.com/downloads/connector/odbc>

Steps

- copy lccMySQLConsole.exe to a folder
- create a Logic File
- (run)

Logic File

Purpose

The Logic File instructs the program on how to send the Request and process the Response.

Syntax

The Logic File expects settings to be supplied in the following manner. However, any lines that do not match will be ignore, i.e. can be used to place comments/etc.

All keys can have encrypted values. To specify that a key value is encrypted, prepend the value with "[lcc:encrypted]":

If you supply encrypted keys, you must also provide these keys:

- lcc:sourceEncryptedPasswordHash
- lcc:sourceEncryptedSaltKey
- lcc:sourceEncryptedVIKey

Syntax

[key] [tab] [value]

Example

lcc:exampleKey exampleValue

Keys

lcc:debugLevel (*optional*)(*none to many per Logic File*)

Provide each Debug Level # to instruct the program on what to output.

The Debug Level #s Are:

- Level 1 Program start/finish
- Level 2 Logic File Records
- Level 3 Logic File Key/Values
- Level 4 Logic File Valid Key/Values
- Level 5 Logic File Skips
- Level 6 Output File - Filename

Level 7	Running Command
Level 8	Start/End Date Checks
Level 9	Valid records

Syntax

lcc:debugLevel [tab] [value]

Example

lcc:debugLevel 2

Example #2 (multiple)

lcc:debugLevel 2

lcc:debugLevel 5

lcc:logPath *(optional)(zero to many per Logic File)*

The path/root name of the Log File. The program will place the '.log' extension automatically and will also append a Year/Month date.

Example: lcc:logPath lccMySQLConsoleLog

lcc:connectionDriver *(optional/mandatory)(one per Logic File)*

The connection driver value.

If providing the 'lcc:connectionString', this key is not used.

Syntax

lcc:connectionDriver [tab] [value]

Example

lcc:connectionDriver {MySQL ODBC 5.3 ANSI Driver}

lcc:connectionServer *(optional/mandatory)(one per Logic File)*

The server used for a connection.

If providing the 'lcc:connectionString', this key is not used.

Syntax

lcc:connectionServer [tab] [value]

Example

lcc:connectionServer ourServer

lcc:connectionServerPort *(optional/mandatory)(one per Logic File)*

The server port used for a connection.

If providing the 'lcc:connectionString', this key is not used.

Syntax

lcc:connectionServerPort [tab] [value]

Example

`lcc:connectionServerPort` `3306`

lcc:connectionDatabase *(optional/mandatory)(one per Logic File)*

The database used for a connection.

If providing the '`lcc:connectionString`', this key is not used.

Syntax

`lcc:connectionDatabase` [tab] [value]

Example

`lcc:connectionDatabase` `ourDatabase`

lcc:connectionUser *(optional/mandatory)(one per Logic File)*

The user used for a connection to the server/database.

We highly recommend using the '`lcc:encrypted`' option for this key/value. See the 'Creating Encrypted Key Values' section.

If providing the '`lcc:connectionString`', this key is not used.

Syntax

`lcc:connectionUser` [tab] [value]

Example

`lcc:connectionUser` `ourUser`

lcc:connectionPassword *(optional/mandatory)(one per Logic File)*

The password used for a connection to the server/database.

We highly recommend using the '`lcc:encrypted`' option for this key/value. See the 'Creating Encrypted Key Values' section.

If providing the '`lcc:connectionString`', this key is not used.

Syntax

`lcc:connectionPassword` [tab] [value]

Example

`lcc:connectionPassword` `ourServer`

lcc:connectionString *(optional/mandatory)(one per Logic File)*

A full connection string to connect to the server/database.

If provided will ignore all of the other `lcc:connection...` keys.

Syntax

`lcc:connectionString` [tab] [value]

Example

```
lcc:connectionString Driver={MySQL ODBC 5.3 ANSI  
Driver};Server=ourServer;Database=ourDatabase;User=ourUser;Password=ourPass  
word
```

lcc:outputFileDelimiter *(one per Logic File)*

Sets the column delimiter in the output files.

By default, a TAB character is used.

The following white-space characters can defined with value:

- [space]
- [tab]

Syntax

```
lcc:outputFileDelimiter [tab] [..delimiter...]
```

Example

```
lcc:outputFileDelimiter      ,
```

Example #2

```
lcc:outputFileDelimiter      [tab]
```

lcc:command *(mandatory)(one to many per Logic File)*

A mySQL command to execute against the server/database.

Provide this key for each line of the command(s).

Use the paired keys 'lcc:outputFile' to export the results to a file. Each set of 'lcc:command' keys should have a single 'lcc:outputFile' key directly following the commands.

Syntax

```
lcc:command [tab] [value]
```

Example

```
lcc:command SHOW TABLES;
```

Example #2 (multiple lines)

```
lcc:command SELECT col1, col2 FROM Table1  
lcc:commandAppend      LEFT JOIN  
lcc:commandAppend      SELECT col3, FROM Table2  
lcc:commandAppend      PK ON Table1.col1 = Table2.col1  
lcc:commandAppend      ORDER BY Table1.col2
```

lcc:commandSkip *(optional)(one to many per Logic File, one per command)*

If provided with 'YES', will skip this command.

Syntax

`lcc:commandSkip`[tab] [YES]

Example

`lcc:commandSkip` YES

lcc:commandHeaderRow (optional)(one to many per Logic File, one per command)

Write a header row to the output file. The header columns should be separated by a comma, but will end up with the defined output file delimiter.

Syntax

`lcc:commandHeaderRow` [tab] [header,header,...]

Example

`lcc:commandHeaderRowp` col1,col2,col3

lcc:commandFlag (optional)(one to many per Logic File, per command)

Flags that affect how the command is processed.

Valid Flags

- header row from table [...table name...]
- looks up the columns in the table and inserts as a header row in the output file

Syntax

`lcc:commandFlag` [tab] [...flag...] [tab] [...value...]

Example

`lcc:commandFlag` header row from table ourTable

lcc:outputFile (mandatory)(one to many per Logic File)

A path/file where the results of the previous set of 'lcc:command' commands are recorded.

Use the paired keys 'lcc:command' to configure what commands will produce the output that will be recorded to this file. This key should directly follow each set of commands.

Syntax

`lcc:outputFile` [tab] [value]

Example

`lcc:outputFile` ourOutputFile.txt

lcc:startDate (optional)(one to many per Logic File, one per Command/Output set)

Instruct the program to only return records who have a start date column equal or greater than the DATE value provide. This key also supplies which column to look at.

Syntax

`lcc:startDate` [tab] [column #] [tab] [DATE]

Example

lcc:startDate 7 2/13/2014 1:19:45 PM

lcc:endtDate *(optional)(one to many per Logic File, one per Command/Output set)*

Instruct the program to only return records who have an end date column equal or less than the DATE value provide. This key also supplies which column to look at.

Syntax

lcc:endtDate [tab] [column #] [tab] [DATE]

Example

lcc:endtDate 6 5/53/2014 5:01:00 PM

lcc:sourceEncryptedPasswordHash *(optional)(one per Logic File)*

Hash value supplied for encryption. Must be used with keys 'lcc:sourceEncryptedSaltKey' and 'lcc:sourceEncryptedVIKey'.

Any set of characters can be used.

Syntax: lcc: sourceEncryptedPasswordHash [tab] [...]

Example: lcc: sourceEncryptedPasswordHash s0m3C00lPhr@\$e

lcc:sourceEncryptedSaltKey *(optional)(one per Logic File)*

Salt Key value supplied for encryption. Must be used with keys 'lcc:sourceEncryptedPasswordHash' and 'lcc:sourceEncryptedVIKey'.

Any set of characters can be used.

Syntax: lcc: sourceEncryptedSaltKey [tab] [...]

Example: lcc: sourceEncryptedSaltKey s0m3C00lPhr@\$e

lcc:sourceEncryptedVIKey *(optional)(one per Logic File)*

VI Key value supplied for encryption. Must be used with keys 'lcc:sourceEncryptedPasswordHash' and 'lcc:sourceEncryptedSaltKey'.

Any set of characters can be used, but, must be exactly 16 characters long.

Syntax: lcc:sourceEncryptedVIKey [tab] [...]

Example: lcc:sourceEncryptedVIKey s0m3C00lPhr@\$e12

lcc:skipUntilLabel *(optional)(none to many per Logic File)*

Where to skip to when you want some Logic to be ignored (not processed). The program will ignore all Logic until it finds the key 'lcc:skipLabel' and the value matching this key's value.

This key is paired with key 'lcc:skipLabel'.

Syntax

lcc:skipUntilLabel [tab] [value]

Example

`lcc:skipUntilLabel` `some skip label`

lcc:skipLabel *(optional)(none to many per Logic File)*

A place to skip to when you want some Logic to be ignored (not processed). The same label can be used multiple times, as the 'lcc:skipUntilLabel' only looks for the 'next' 'lcc:skipLabel'.

This key is paired with key 'lcc:skipUntilLabel'.

Syntax

`lcc:skipLabel` [tab] [value]

Example

`lcc:skipLabel` `some skip label`

Creating Encrypted Key Values

The encrypting option uses the same Logic File used to run the program, which looks for the three encryption keys:

- `lcc:sourceEncryptedPasswordHash`
- `lcc:sourceEncryptedSaltKey`
- `lcc:sourceEncryptedVIKey`

To encrypt a value, run the following from a Command Line:

`lccMySQLConsole.exe lcc:encryptValue "..."`

example:

`lccMySQLConsole.exe lcc:encryptValue "our secret information"`

The 'encrypted' value will then be copied to your clipboard. Edit your Logic File and 'paste' into the desired value.

Running The Program

Run the lccMySQLConsole.exe with one of the following Syntaxes:

Using the default 'lccMySQLConsole-logic.txt' path name

- `lccMySQLConsole.exe`

Defining a custom Logic File path

- `lccMySQLConsole.exe lcc:logicPath` [logic file path]
- ex: `lccMySQLConsole.exe lcc:logicPath lccMySQLConsole-logic.txt`

Testing Connection Values For ODBC Driver

1. Create a text file on desktop and rename it to .udl.
2. Double click it to open the Data Link Properties dialog box.
3. Select provider and connection to connect to the ODBC data source. Click Test Connection on the Connection tab to test.

To View The Connection String

4. Click OK button and open the .udl file in the Notepad to see the connection string.
5. Use this connection string to open the OdbcConnection object.

Example Logic Files

Read two tables and output to two files.

```
lcc:debugLevel      1      Program start/finish
lcc:debugLevel      6      Output File - Filename
lcc:debugLevel      7      Running Command
```

```
lcc:sourceEncryptedPasswordHash s0m3C00lPhr@$e
lcc:sourceEncryptedSaltKey s0m3C00lPhr@$e
lcc:sourceEncryptedVIKey  s0m3C00lPhr@$e12
```

```
lcc:connectionDriver {MySQL ODBC 5.3 ANSI Driver}
lcc:connectionServersrv00883
lcc:connectionServerPort  3306
lcc:connectionDatabase   ourDatabase
lcc:connectionUser  [lcc:encrypted]EMCLDEDBHGPEGFGBGHEEEDNDN
lcc:connectionPassword  [lcc:encrypted]DCDMGBDECEDHGEBGLGEOFHNDN
```

```
lcc:command SHOW TABLES;
lcc:outputFile ourData-tables.txt
```

```
lcc:command SHOW FULL TABLES IN ourDatabase WHERE TABLE_TYPE LIKE 'VIEW';
lcc:outputFile ourData-views.txt
```

Read a table and only output records who fall between two dates.

```
lcc:debugLevel      1      Program start/finish
lcc:debugLevel      6      Output File - Filename
lcc:debugLevel      7      Running Command
```

```
lcc:sourceEncryptedPasswordHash s0m3C00lPhr@$e
lcc:sourceEncryptedSaltKey s0m3C00lPhr@$e
lcc:sourceEncryptedVIKey  s0m3C00lPhr@$e12
```

```
lcc:connectionDriver {MySQL ODBC 5.3 ANSI Driver}
lcc:connectionServersrv00883
lcc:connectionServerPort 3306
lcc:connectionDatabase ourDatabase
lcc:connectionUser [lcc:encrypted]EMCLDEDBHGPEGFGBGHEEEDNDN
lcc:connectionPassword [lcc:encrypted]DCDMGBDECEDHGEBGLGEOFHNDN

lcc:command SELECT * FROM ourTable;
lcc:startDate 7 2/13/2014 1:19:45 PM
lcc:endDate 6 3/14/2015 2:05:00 PM
lcc:outputFile ourData-specificRecordsFilteredByDate.txt
```

Definitions

none

Modifications

NAME	DATE	MODIFICATION
David Mielcarek	9/1/2015	Created
David Mielcarek	7/20/2016	Added note about installing MySQL ODBC Driver.
David Mielcarek	11/29/2022	Added keys "lcc:commandFlag", "lcc:outputFileDelimiter"
David Mielcarek	11/30/2022	Added keys "lcc:commandSkip", "lcc:commandHeaderRow"

End of document