
Load a big CSV file into a MySQL DB

By rac

Published: 02.01.2008 - 13:16

Used Libraries

mysql-connector-java-5.1.5-bin.jar

CSV2SQLImporter.java

```
1.
import java.sql.Connection;

2.
import java.sql.DriverManager;

3.
import java.sql.Statement;

4.
import java.sql.ResultSet;

5.
import java.sql.SQLException;

6.

7.
class CSV2SQLImporter
```

```
8.
{

9.
    Connection conn = null;

10.
    public CSV2SQLImporter()

11.
    {

12.
    }

13.

14.
    public boolean connect( String db_connect_str,

15.
    String db_userid, String db_password)

16.
    {

17.
        Connection conntmp;

18.
        try

19.
        {

20.
```

Class.forName(

21.

"com.mysql.jdbc.Driver").newInstance());

22.

23.

conntmp = [DriverManager](#).getConnection(db_connect_str,

24.

db_userid, db_password);

25.

26.

}

27.

catch([SQLException](#) es)

28.

{

29.

[System](#).out.println("SQL Error Code: "+es.getErrorCode()+"\nSQL state: "+es.
getSQLState()+"\nException Message: "+es.getMessage());

30.

[System](#).out.println("\n");

31.

es.printStackTrace();

32.

```
        conntmp = null;

33.     return false;

34. }

35. catch( Exception e)

36. {

37.     e.printStackTrace();

38.     conntmp = null;

39.     return false;

40. }

41.

42. this.conn = conntmp;

43. return true;

44. }
```

45.

46.

```
public void importData(String filename, String table)
```

47.

```
{
```

48.

```
    Statement stmt;
```

49.

```
    String query;
```

50.

51.

```
    try
```

52.

```
    {
```

53.

```
        stmt = this.conn.createStatement(
```

54.

```
        ResultSet.TYPE_SCROLL_SENSITIVE,
```

55.

```
        ResultSet.CONCUR_UPDATABLE);
```

56.

57.

```
query = "LOAD DATA LOCAL " +
```

```
58.
```

```
    "INFILE '" + filename + "'" +
```

```
59.
```

```
    "    REPLACE" +
```

```
60.
```

```
    "    INTO TABLE " + table +
```

```
61.
```

```
    "    FIELDS" +
```

```
62.
```

```
    "          TERMINATED BY ';';"
```

```
63.
```

```
    stmt.executeUpdate(query);
```

```
64.
```

```
65.
```

```
    }
```

```
66.
```

```
    catch( Exception e)
```

```
67.
```

```
    {
```

```
68.
```

```
        e.printStackTrace();
```

```
69.
```

```
    stmt = null;
```

```
70.     }  
  
71.     }  
  
72.  
};
```

sampleGUImport.java

```
1.  
import javax.swing.*;  
  
2.  
  
3.  
public class sampleGUImport extends JFrame  
  
4.  
{  
  
5.  
    JTextArea response;  
  
6.  
    /**
```

7.
* @param importfile.csv table

8.
*/

9.
public static void main([String](#) [] args)

10.
{

11.
sampleGUIimport gui = new sampleGUIimport();

12.
gui.setDefaultCloseOperation([JFrame](#) .DO_NOTHING_ON_CLOSE);

13.
[String](#) server = "remote.server.net";

14.
[String](#) port = "3306";

15.
[String](#) mysqldb = "remote_db";

16.
[String](#) user = "dbuser";

17.
[String](#) password = "dbpassword";

18.

19.

```
//String csvfile = "C:/temp/sample.txt";
```

20.

```
String csvfile1 = "//source.server.net/windowsshare/sample1.txt";
```

21.

```
String table1 = "tbl_sample1";
```

22.

```
String csvfile2 = "//source.server.net/windowsshare/sample2.txt";
```

23.

```
String table2 = "tbl_sample2";
```

24.

25.

```
CSV2SQLImporter db = new CSV2SQLImporter();
```

26.

```
boolean connect = db.connect("jdbc:mysql://" + server + ":" + port + "/" + mysqlDb, user, password)
```

```
;
```

27.

```
if(connect){
```

28.

```
    gui.addResponse("Importing sample1 Data...");
```

29.

```
    db.importData(csvfile1, table1);
```

30.

```
    gui.addResponse("Importing sample2 Data...");
```

31.

```
        db.importData(csvfile2, table2);

32.        gui.addResponse("Data import done");

33.    } else {

34.        gui.addResponse("Couldn't connect to mysql db");

35.    }

36.    gui.setDefaultCloseOperation( JFrame .EXIT_ON_CLOSE);

37.    }

38.

39.    public sampleGUILimport() {

40.        getContentPane().setLayout(null);

41.

42.        response = new JTextArea ();

43.        response.setLocation(,);
```

```
44. response.setSize(546,528);

45. response.setRows(5);

46. response.setColumns(5);

47. response.setToolTipText("Here you see what I'm doing");

48. response.setEditable(false);

49. getContentPane().add(response);

50.

51. setTitle("multi sample file Import Utility");

52. setSize(546,528);

53. setVisible(true);

54. setResizable(false);

55. }

56.
```

57.

```
public void addResponse( String newresponse) {
```

58.

```
    System .out.println(newresponse);
```

59.

```
    this.response.setText(this.response.getText()+newresponse+"n");
```

60.

```
}
```

61.

62.

```
}
```

Sample ANT script to build JAR

1.

```
<?xml version="1.0" encoding="UTF-8"?>
```

2.

```
<project default="jar">
```

3.

<!-- Ordnerstruktur -->

4.

```
<property name="project.root" location="../../../../" />
```

5.

```
<property name="src.dir" location="${project.root}/src/" />
```

6.

```
<property name="srctmp.dir" location="${project.root}/srctmp/" />
```

7.

```
<property name="bin.dir" location="${project.root}/bin/" />
```

8.

```
<property name="lib.dir" location="${project.root}/lib/" />
```

9.

```
<property name="tmp.dir" location="${project.root}/tmp/" />
```

10.

11.

```
<!-- wichtige Files -->
```

12.

```
<property name="mysql-connector-java-5.1.5-bin.jar" location=
"${lib.dir}/mysql-connector-java-5.1.5-bin.jar" />
```

13.

```
<property name="SAMPLEImporter.jar" location="${bin.dir}/SAMPLEImporter.jar" />
```

14.

```
<property name="SAMPLEImporterWOLIB.jar" location="${bin.dir}/SAMPLEImporterWOLIB.jar" />
```

15.

`<property name="main.class" value="org.test.sample.csv2sql.SAMPLEimport" />`

16.

17.

`<!-- Alle libs die verwendet werden -->`

18.

`<fileset id="lib.jars" dir="{lib.dir}">`

19.

`<include name="mysql-connector-java-5.1.5-bin.jar"/>`

20.

`</fileset>`

21.

22.

`<path id="libs">`

23.

`<fileset refid="lib.jars"/>`

24.

`</path>`

25.

26.

`<!-- Compiliert das Ganze Projekt -->`

27.

`<target name="compile">`

28.

```
<mkdir dir="${bin.dir}" />
```

29.

```
<javac srcdir="${src.dir}" destdir="${bin.dir}" classpathref="libs" />
```

30.

```
</target>
```

31.

32.

```
<!-- Erstellt ein Projekt jar File -->
```

33.

```
<target name="jar" depends="compile">
```

34.

```
<jar destfile="${SAMPLEImporterWOLIB.jar}" basedir="${bin.dir}" includes=
"org/test/sample/csv2sql/*.class">
```

35.

```
<manifest>
```

36.

```
<attribute name="Main-Class" value="${main.class}" />
```

37.

```
</manifest>
```

38.

```
</jar>
```

39.

```
</target>
```

40.

41.

```
<!-- Erstellt ein Jar welches alle Librarys enthaellt -->
```

42.

```
<target name="onejar" depends="jar">
```

43.

```
<mkdir dir="${tmp.dir}" />
```

44.

```
<unjar dest="${tmp.dir}" src="${SAMPLEimporterWOLIB.jar}" />
```

45.

```
<unjar dest="${tmp.dir}">
```

46.

```
<fileset refid="lib.jars" />
```

47.

```
</unjar>
```

48.

```
<delete dir="${tmp.dir}/META-INF"/>
```

49.

```
<jar destfile="${SAMPLEimporter.jar}" basedir="${tmp.dir}" includes="**/*">
```

50.

```
<manifest>
```

51.

```
<attribute name="Main-Class" value="${main.class}" />
```

52.

</manifest>

53.

</jar>

54.

<delete dir="{tmp.dir}" />

55.

</target>

56.

57.

<!-- Fuehrt SAMPLEimporter aus -->

58.

<target name="run" depends="oneJar">

59.

<java classname="{main.class}" classpath="{SAMPLEimporter.jar}" fork="true" />

60.

</target>

61.

62.

63.

<!-- Loescht alle erstellten Files -->

64.

<target name="clean">

65.

```
<delete dir="{bin.dir}" />
```

66.

67.

```
</target>
```

68.

```
</project>
```

-->

Trackback URL for this post:

<http://www.2030.tk/trackback/23>