

Homework 3

Remember HW1 question where you had to create a number of CSV files and populate it based on a University Database.

In this homework assignment, you need to write a Java program where it connects to MySQL server, reads your CSV data files and populates the MySQL database.

Instructions:

Step 1: You will need to install MySQL installer on your system. Please find the instructions here (<https://dev.mysql.com/downloads/windows/installer/5.7.html>).

Select the OS version of your system and download the MySQL installer.

Generally Available (GA) Releases

MySQL Installer 5.7.23

Select Version:
5.7.23

Select Operating System:
Microsoft Windows

Looking for the latest GA version?

Click this

Windows (x86, 32-bit), MSI Installer (mysql-installer-web-community-5.7.23.0.msi)	5.7.23	15.9M	Download
Windows (x86, 32-bit), MSI Installer (mysql-installer-community-5.7.23.0.msi)	5.7.23	371.0M	Download

MD5: 348be559b268b9ec56bdf0b46ba19c61 | Signature

MD5: 95e92084366994464ab377ec196d9ecc | Signature


Step 2: Create the tables in the MySQL database using the SQL DDL statements.

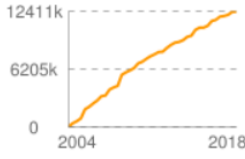
For example: Student table can be created as below:

```
create database CS480HW3; // naming the database
use CS480HW3;
create table student(
studid integer,
sname varchar(20),
phonenumber varchar(20),
gender char, dob date, address varchar(15));
```

Step 3: You will also need to download and install any of the available IDE(Integrated development environment), if you do not already have. Check out some of the IDE here: [Eclipse](#), [NetBeans](#), [IntelliJ](#)

Step 4: You need to download the library MySQL-JDBC connector JAR file (<http://mvnrepository.com/artifact/mysql/mysql-connector-java/5.1.23>). Add this JAR file into the library of your java application.




Indexed Artifacts (12.4M)


Popular Categories

- Aspect Oriented
- Actor Frameworks
- Application Metrics
- Build Tools
- Bytecode Libraries
- Command Line Parsers
- Cache Implementations

Home » mysql » mysql-connector-java » 5.1.23


MySQL Connector/J » 5.1.23
JDBC Type 4 driver for MySQL

License	GPL 2.0
Categories	MySQL Drivers Download
HomePage	http://dev.mysql.com/usingmysql/java/
Date	(May 08, 2015)
Files	pom (1 KB) jar (823 KB) View All
Repositories	Central Aspose
Used By	2,976 artifacts

Note: There is a new version for this artifact

Step 5: Finally, you need to write a Java program which does the following:

1. Establish connection to MySQL server. You will need to read through the following page here to start coding (<https://dev.mysql.com/doc/connector-j/5.1/en/connector-j-usagenotes-connect-drivermanager.html>)

Below is shown the code snippet for establishing connection with MySQL:

```
// JDBC driver name and database URL

static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
static final String DB_URL = "jdbc:mysql://localhost/DatabaseName";

// Database credentials
static final String USER = "root";
static final String PASS = "*****";

public static void main(String[] args) {
    Connection conn = null;
    Statement stmt = null;

    //Register JDBC driver
    Class.forName("com.mysql.jdbc.Driver");

    //Open a connection
    System.out.println("Connecting to a selected database...");
    conn = DriverManager.getConnection(DB_URL, USER, PASS);
    System.out.println("Connected database successfully...");
}
```

2. Finally, you need to insert data from CSV files into the N tables you already created in MySQL. Please go through the page <https://dev.mysql.com/doc/connector-j/5.1/en/connector-j-usagenotes-statements.html> to understand how it works.

Code for inserting values into one of the tables is given below:

```
// inserting into table Student (6 arguments since I have 6 attributes for Student table)
String sql1 = " INSERT INTO student VALUES(?,?,?, ?,?,?) ";
BufferedReader bReader1 = new BufferedReader(new FileReader("Location of your CSV file"));
String line1 = "";
while ((line1 = bReader1.readLine()) != null) {
    String[] str = line1.split(",");
    try (
        //Create preparedStatement here, set them and execute them
        PreparedStatement ps = conn.prepareStatement(sql1)) {
        ps.setString(1, str[0]);
        ps.setString(2, str[1]);
        ps.setString(3, str[2]);
        ps.setString(4, str[3]);
        ps.setString(5, str[4]);
        ps.setString(6, str[5]);
        ps.executeUpdate();
    }
}
```

Please submit your solution electronically to Blackboard (<http://uic.blackboard.com/>) under the assignment "HW3". Zip your SQL file and java file together and upload to HW3 assignment in Blackboard. Make sure your name and other information appears at the top of your SQL document and java document, the best approach is to use comments like this:

```
--
--
HW3 MySQL-JDBC
--
--NAME and NetID
--U. of Illinois, Chicago
--CS 480, Fall 2018
--Homework 3
--
```