TUTORIAL 6 – CONNECT JAVA TO MYSQL

❖ Content:

- Setup MySQL connection and create a database in MySQL Workbench
- Create a new Java project and establish a connection to MySQL in IntelliJ

❖ Instructions:

1) Open MySQL Workbench then create a new connection to MySQL

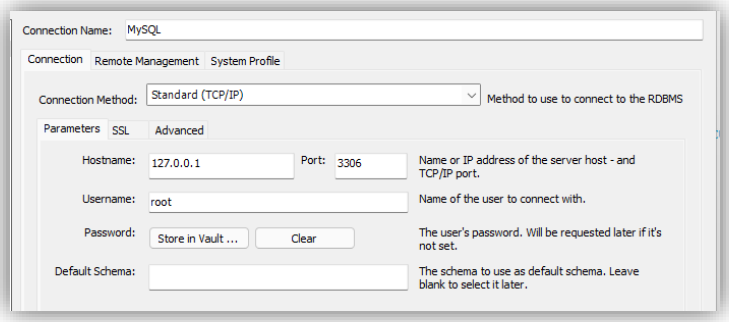

Figure 1 - Create connection to MySQL

2) Create a new database (schema) and at least one table then insert some sample records (rows) to that table

```
/* Create new database */
CREATE DATABASE bookdb;
/* Use this database */
USE bookdb;
/* Create new table */
CREATE TABLE book (
id INT PRIMARY KEY AUTO_INCREMENT,
title VARCHAR(50) NOT NULL,
author VARCHAR(30) NOT NULL,
price FLOAT NOT NULL);
/* Insert data to this table */
INSERT INTO book (title, author, price)
VALUES ("Java Web","John",100),("Spring Boot","David",120);
```

Figure 2 - Sample SQL Script

3) Open IntelliJ then create a new Java project

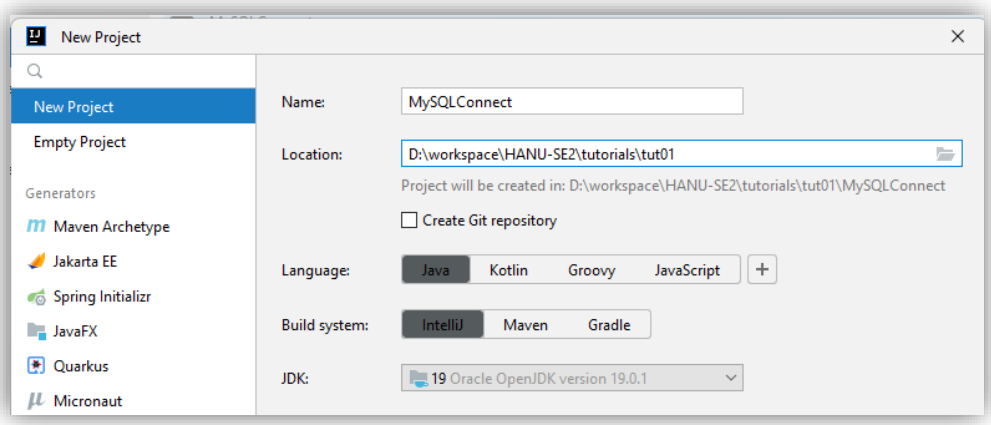

Figure 3 - Create new Java project

4) Import MySQL connector library

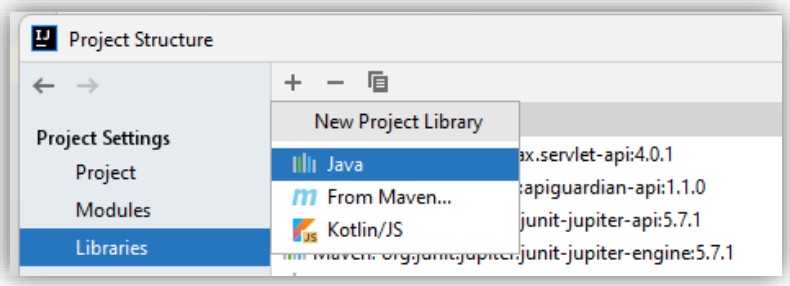

Figure 4 - Import MySQL library (1)

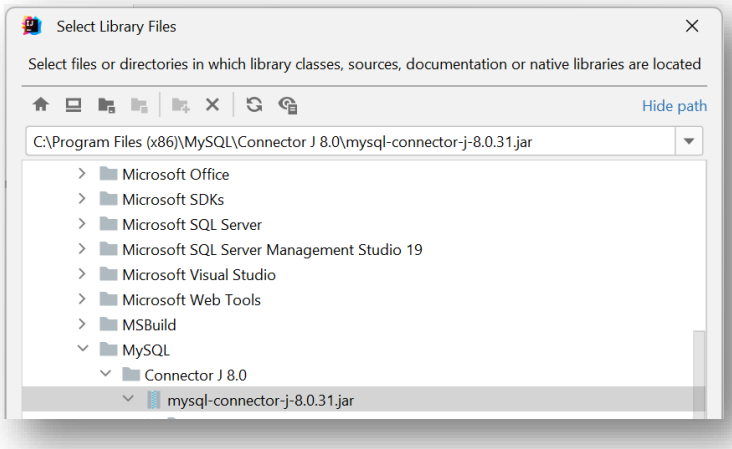

Figure 5 - Import MySQL library (2)

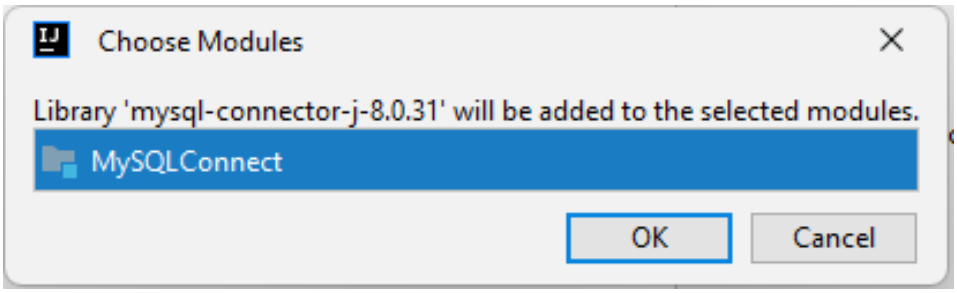

Figure 6 - Import MySQL library (3)

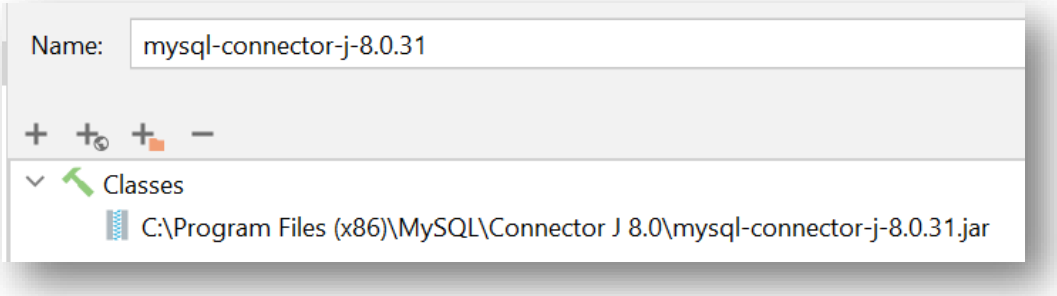

Figure 7 - Import MySQL library (4)

5) Create a new Java package and Java class for MySQL database connection

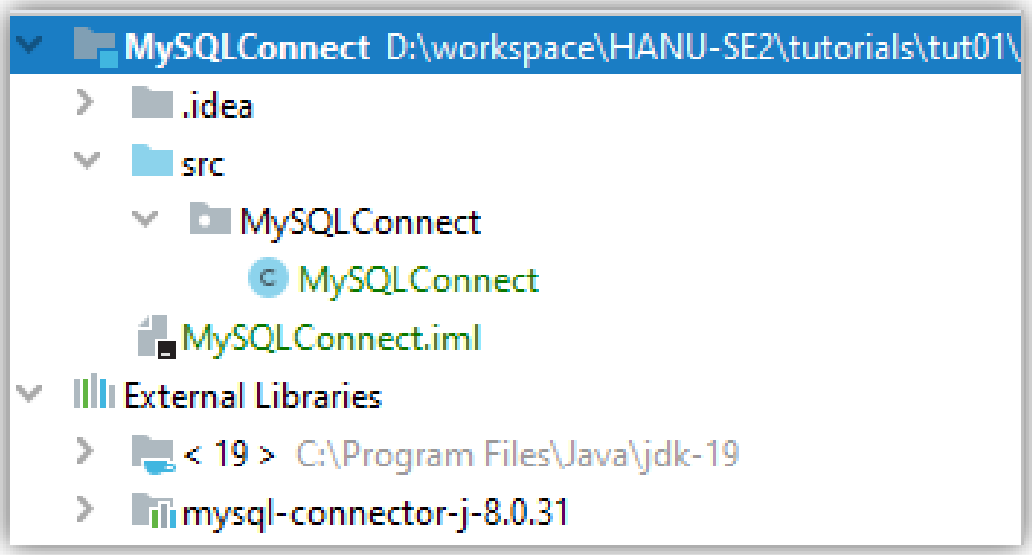

Figure 8 - Project structure

```java
public class MySQLConnect {
    //set MySQL connection parameters
    private static final String DB_URL = "jdbc:mysql://127.0.0.1:3306/bookdb";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "root";

    //main method for testing
    public static void main(String args[]) {
        try {
            // connect to database
            Connection con = getConnection(DB_URL, DB_USERNAME, DB_PASSWORD);
            /* System.out.println("Connection to DB succeed !"); */
            // create statement
            Statement stm = con.createStatement();
            // create SQL query
            String query = "SELECT * FROM book";
            // execute SQL query
            ResultSet rs = stm.executeQuery(query);
            // display data to console
            while (rs.next()) {
                System.out.println("Book ID: " + rs.getInt( columnIndex: 1));
                System.out.println("Book Title:  " + rs.getString( columnIndex: 2));
                System.out.println("Book Author: " + rs.getString( columnIndex: 3));
                System.out.println("Book Price: " + rs.getString( columnIndex: 4));
                System.out.println("------------------------");
            }
            // close JDBC connection
            con.close();
        } catch (Exception ex) {
            ex.printStackTrace();
            /* System.err.println("Connection to DB failed !"); */
        }
    }
}
```

*Figure 9 – **MySQLConnect.java***

```
Book ID: 1
Book Title:  Java Web
Book Author: John
Book Price: 100.0
------------------------
Book ID: 2
Book Title:  Spring Boot
Book Author: David
Book Price: 120.0
------------------------
```

Figure 10 - Console output

❖ TASKS:

- Create the connection to MySQL then write SQL script to create a different database and at least 2 tables (ex: *StudentDB, EmployeeDB, HospitalDB*,…) in MySQL Workbench
- Create new Java project and write code to connect to that database and show the table data to console in IntelliJ IDEA
- Submit the evidences (screenshots) that you already setup MySQL & establish a connection from Java to MySQL successfully.
    - Screenshot 1: MySQL Workbench interface which you have created a database (schema)
    - Screenshot 2: SQL script to create database & tables
    - Screenshot 3: IntelliJ interface which you have created Java project with a Java class to connect to MySQL
    - Screenshot 4: Console output which shows the table data from MySQL
- Put every above screenshots into a Word document then submit to FIT LMS with file name syntax: *FullName_StudentID_SE1_Tut6.docx*