### **CSD420-H326 Advanced Java Programming (2251-DD)**

**Date**: 12/09/2024

**Name**: Arun Sharma

**Assignment no.**: 9 of Module 9: JDBC Part I

\*(Code examples used provided in instruction)

### **Objective**

The objective of this assignment was to test the JDBC connection between a Java application and a MySQL database using a provided code example. The code examples include operations to create a table, insert data, and select data from the database.

GitHub Repository: [CSD-420 Repository](https://github.com/SharmaArun017/CSD-420)

### **Code Examples**

#### **1. CreateTable.java**

/\*  
 \* Date: 12/09/2024  
 \* Course: CSD420-H326 Advanced Java Programming  
 \* GitHub Repository: <https://github.com/SharmaArun017/CSD-420> \*   
 \* Program: CreateTable  
 \* Description: This program drops the "address33" table if it exists  
 \* and creates a new "address33" table in the "databasedb" database.  
 \*/  
  
import java.sql.\*;  
  
public class CreateTable {  
  
 private Connection con;  
 private Statement stmt;  
  
 public CreateTable() {  
 try {  
 // Load the MySQL JDBC driver  
 Class.forName("com.mysql.cj.jdbc.Driver");  
  
 // Database connection URL  
 String url = "jdbc:mysql://localhost:3306/databasedb?";  
 con = DriverManager.getConnection(url + "user=student1&password=pass");  
  
 // Create a statement object  
 stmt = con.createStatement();  
 } catch (Exception e) {  
 System.out.println("Error connecting to the database: " + e.getMessage());  
 System.exit(0);  
 }  
  
 // Drop the table if it exists  
 try {  
 stmt.executeUpdate("DROP TABLE address33");  
 System.out.println("Table 'address33' dropped successfully.");  
 } catch (SQLException e) {  
 System.out.println("Table 'address33' does not exist. Skipping drop operation.");  
 }  
  
 // Create the table  
 try {  
 String createTableSQL = "CREATE TABLE address33 (" +  
 "ID INT PRIMARY KEY, " +  
 "LASTNAME VARCHAR(40), " +  
 "FIRSTNAME VARCHAR(40), " +  
 "STREET VARCHAR(40), " +  
 "CITY VARCHAR(40), " +  
 "STATE VARCHAR(40), " +  
 "ZIP VARCHAR(40))";  
 stmt.executeUpdate(createTableSQL);  
 System.out.println("Table 'address33' created successfully.");  
 } catch (SQLException e) {  
 System.out.println("Failed to create table 'address33': " + e.getMessage());  
 }  
  
 // Close the database connection  
 try {  
 stmt.close();  
 con.close();  
 System.out.println("Database connections closed.");  
 } catch (SQLException e) {  
 System.out.println("Failed to close database connections: " + e.getMessage());  
 }  
 }  
  
 public static void main(String[] args) {  
 // Instantiate CreateTable class to execute the operations  
 new CreateTable();  
 }  
}

#### **2. InsertData.java**

/\*  
 \* Date: 12/09/2024  
 \* Course: CSD420-H326 Advanced Java Programming  
 \* GitHub Repository: <https://github.com/SharmaArun017/CSD-420> \*   
 \* Program: InsertData  
 \* Description: This program inserts multiple rows into the "address33" table  
 \* in the "databasedb" database.  
 \*/  
  
import java.sql.\*;  
  
public class InsertData {  
  
 private Connection con;  
 private Statement stmt;  
  
 public InsertData() {  
 try {  
 // Load the MySQL JDBC driver  
 Class.forName("com.mysql.cj.jdbc.Driver");  
  
 // Database connection URL  
 String url = "jdbc:mysql://localhost:3306/databasedb?";  
 con = DriverManager.getConnection(url + "user=student1&password=pass");  
  
 // Create a statement object  
 stmt = con.createStatement();  
 } catch (Exception e) {  
 System.out.println("Error connecting to the database: " + e.getMessage());  
 System.exit(0);  
 }  
  
 // Insert data into the "address33" table  
 try {  
 System.out.println(  
 stmt.executeUpdate("INSERT INTO address33 VALUES (55, 'Larry', 'Rich', '1111 Redwing Circle888', 'Bellevue', 'NE', '68123')")   
 + " row(s) updated.");  
  
 System.out.println(  
 stmt.executeUpdate("INSERT INTO address33 VALUES (1, 'Fine', 'Ruth', '1111 Redwing Circle', 'Bellevue', 'NE', '68123')")   
 + " row(s) updated.");  
 System.out.println(  
 stmt.executeUpdate("INSERT INTO address33 VALUES (2, 'Howard', 'Curly', '1000 Galvin Road South', 'Bellevue', 'NE', '68005')")   
 + " row(s) updated.");  
 System.out.println(  
 stmt.executeUpdate("INSERT INTO address33 VALUES (3, 'Howard', 'Will', '2919 Redwing Circle', 'Bellevue', 'NE', '68123')")   
 + " row(s) updated.");  
 System.out.println(  
 stmt.executeUpdate("INSERT INTO address33 VALUES (4, 'Wilson', 'Larry', '1121 Redwing Circle', 'Bellevue', 'NE', '68124')")   
 + " row(s) updated.");  
 System.out.println(  
 stmt.executeUpdate("INSERT INTO address33 VALUES (5, 'Johnson', 'George', '1300 Galvin Road South', 'Bellevue', 'NE', '68006')")   
 + " row(s) updated.");  
 System.out.println(  
 stmt.executeUpdate("INSERT INTO address33 VALUES (6, 'Long', 'Matthew', '2419 Redwing Circle', 'Bellevue', 'NE', '68127')")   
 + " row(s) updated.");  
 System.out.println(  
 stmt.executeUpdate("INSERT INTO address33 VALUES (44, 'Tom', 'Matthew', '1999 Redwing Circle', 'Bellevue', 'NE', '68123')")   
 + " row(s) updated.");  
  
 // Commit the transaction  
 stmt.executeUpdate("COMMIT");  
 System.out.println("Data inserted successfully.");  
 } catch (SQLException e) {  
 System.out.println("Error inserting data: " + e.getMessage());  
 }  
  
 // Close the database connection  
 try {  
 stmt.close();  
 con.close();  
 System.out.println("Database connections closed.");  
 } catch (SQLException e) {  
 System.out.println("Failed to close database connections: " + e.getMessage());  
 }  
 }  
  
 public static void main(String[] args) {  
 // Instantiate InsertData class to execute the operations  
 new InsertData();  
 }  
}

#### **3. Select5.java**

/\*  
 \* Date: 12/09/2024  
 \* Course: CSD420-H326 Advanced Java Programming  
 \* GitHub Repository: <https://github.com/SharmaArun017/CSD-420> \*   
 \* Program: Select5  
 \* Description: This program retrieves and displays all rows from the "address33" table  
 \* in the "databasedb" database.  
 \*/  
  
import java.sql.\*;  
  
public class Select5 {  
  
 public static void main(String[] args) {  
 Connection con = null;  
 Statement stmt = null;  
  
 try {  
 // Load the MySQL JDBC driver  
 Class.forName("com.mysql.cj.jdbc.Driver");  
  
 // Database connection URL  
 String url = "jdbc:mysql://localhost:3306/databasedb?";  
 con = DriverManager.getConnection(url + "user=student1&password=pass");  
  
 System.out.println("Connection established - now executing a SELECT query");  
  
 // Create a statement object  
 stmt = con.createStatement();  
  
 // Execute the SELECT query  
 ResultSet rs = stmt.executeQuery("SELECT \* FROM address33");  
  
 System.out.println("Received Results:");  
  
 // Retrieve metadata to get column count  
 int columnCount = rs.getMetaData().getColumnCount();  
  
 // Iterate through the result set and display the results  
 while (rs.next()) {  
 for (int i = 1; i <= columnCount; i++) {  
 System.out.print(rs.getString(i) + "\t");  
 }  
 System.out.println();  
 }  
  
 } catch (ClassNotFoundException e) {  
 System.out.println("MySQL JDBC Driver not found: " + e.getMessage());  
 } catch (SQLException e) {  
 System.out.println("Error executing SQL query: " + e.getMessage());  
 } finally {  
 // Close resources  
 try {  
 if (stmt != null) {  
 stmt.close();  
 }  
 if (con != null) {  
 con.close();  
 }  
 System.out.println("Database connections closed.");  
 } catch (SQLException e) {  
 System.out.println("Failed to close database connections: " + e.getMessage());  
 }  
 }  
 }  
}

### **Screenshots**



