Java MySQL connection CURD

- **Retrieve data from MySQL using Java:**

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.ResultSet;
import java.sql.Statement;

class java_test {
   public static void main(String[] args) {
      // JDBC URL, username, and password of MySQL server
      String url = "jdbc:mysql://localhost:3306/employeedata";
      String user = "root";
      String password = "Prdc@123";

      try {
         // Register the JDBC driver
         Class.forName("com.mysql.cj.jdbc.Driver");

         // Open a connection
         System.out.println("Connecting to database...");
         Connection connection = DriverManager.getConnection(url, user, password);

         // Do something with the connection...
         // SELECT statement
         String selectQuery = "select * from employee";

         // Creating a Statement
         Statement statement = connection.createStatement();

         // Executing the SELECT query
         ResultSet resultSet = statement.executeQuery(selectQuery);

         // Processing the results
         while (resultSet.next()) {
           String id = resultSet.getString("id");
           String name = resultSet.getString("name");
           String address = resultSet.getString("address");
           String email = resultSet.getString("email");
           String mobile = resultSet.getString("mobile");
           String job_title = resultSet.getString("job_title");
           System.out.println("ID= " +id);
           System.out.println("Name= " +name);
           System.out.println("Email= " +email);
           System.out.println("Mobile= " +mobile);
           System.out.println("job_title= " +job_title);
           System.out.println();
         }

         // Close the connection
         connection.close();
         System.out.println("Connection closed.");

      } catch (ClassNotFoundException | SQLException e) {
         e.printStackTrace();
      }
   }
}
```

Download jar file and keep in db_test folder or same java file folder.
Compile and run:
javac -cp .;db_test\mysql-connector-j-8.3.0.jar java_test.java
java -cp .;db_test\mysql-connector-j-8.3.0.jar java_test

- **Insert Data in MySQL Using Java:**

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Scanner;

class InsertDataFromUser {
   public static void main(String[] args) {
      // JDBC URL, username, and password of MySQL server
```

Java MySQL connection CURD

```
        String url = "jdbc:mysql://localhost:3306/employeedata";
        String user = "root";
        String password = "Prdc@123";

        try {
            // Register the JDBC driver
            Class.forName("com.mysql.cj.jdbc.Driver");

            // Open a connection
            System.out.println("Connecting to database...");
            Connection connection = DriverManager.getConnection(url, user, password);

            // Get data from the user
            Scanner scanner = new Scanner(System.in);
            System.out.print("Enter employee ID: ");
            String id = scanner.nextLine();
            System.out.print("Enter employee name: ");
            String name = scanner.nextLine();
            System.out.print("Enter employee address: ");
            String address = scanner.nextLine();
            System.out.print("Enter employee email: ");
            String email = scanner.nextLine();
            System.out.print("Enter employee mobile: ");
            String mobile = scanner.nextLine();
            System.out.print("Enter employee job title: ");
            String jobTitle = scanner.nextLine();

            // INSERT statement
            String insertQuery = "INSERT INTO employee (id, name, address, email, mobile, job_title) VALUES (?, ?, ?, ?, ?, ?)";

            // Creating a PreparedStatement
            try (PreparedStatement preparedStatement = connection.prepareStatement(insertQuery)) {
                // Set values for the parameters
                preparedStatement.setString(1, id);
                preparedStatement.setString(2, name);
                preparedStatement.setString(3, address);
                preparedStatement.setString(4, email);
                preparedStatement.setString(5, mobile);
                preparedStatement.setString(6, jobTitle);

                // Execute the INSERT query
                int rowsAffected = preparedStatement.executeUpdate();

                if (rowsAffected > 0) {
                    System.out.println("Data inserted successfully.");
                } else {
                    System.out.println("Failed to insert data.");
                }
            }

            // Close the connection
            connection.close();
            System.out.println("Connection closed.");

        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
        }
    }
}
```

Note: Downloaded jar file should be same project folder.
To Compile: javac -cp .;mysql-connector-j-8.3.0.jar InsertDataFromUser.java
To Run: java -cp .;mysql-connector-j-8.3.0.jar InsertDataFromUser

- **Delete Data in MySQL Using Java:**

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Scanner;

class DeleteDataFromMySQL {
    public static void main(String[] args) {
```

```
        // JDBC URL, username, and password of MySQL server
        String url = "jdbc:mysql://localhost:3306/employeedata";
        String user = "root";
        String password = "Prdc@123";

        try {
            // Register the JDBC driver
            Class.forName("com.mysql.cj.jdbc.Driver");

            // Open a connection
            System.out.println("Connecting to database...");
            Connection connection = DriverManager.getConnection(url, user, password);

            // Prompt the user for the ID to delete
            Scanner scanner = new Scanner(System.in);
            System.out.print("Enter the ID to delete: ");
            String idToDelete = scanner.nextLine();

            // DELETE statement
            String deleteQuery = "DELETE FROM employee WHERE id = ?";

            // Creating a PreparedStatement
            try (PreparedStatement preparedStatement = connection.prepareStatement(deleteQuery)) {
                // Set the value for the parameter (ID to delete)
                preparedStatement.setString(1, idToDelete);

                // Execute the DELETE query
                int rowsAffected = preparedStatement.executeUpdate();

                if (rowsAffected > 0) {
                    System.out.println("Data deleted successfully.");
                } else {
                    System.out.println("No rows deleted. ID not found.");
                }
            }

            // Close the connection
            connection.close();
            System.out.println("Connection closed.");

        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
        }
    }
}
```

To compile and Run:

```
javac -cp .;mysql-connector-j-8.3.0.jar DeleteDataFromMySQL.java
java -cp .;mysql-connector-j-8.3.0.jar DeleteDataFromMySQL
```

- **Update Data in MySQL Using Java:**

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Scanner;

class UpdateDataInMySQL {
    public static void main(String[] args) {
        // JDBC URL, username, and password of MySQL server
        String url = "jdbc:mysql://localhost:3306/employeedata";
        String user = "root";
        String password = "Prdc@123";

        try {
            // Register the JDBC driver
            Class.forName("com.mysql.cj.jdbc.Driver");

            // Open a connection
            System.out.println("Connecting to database...");
            Connection connection = DriverManager.getConnection(url, user, password);

            // Prompt the user for the ID to update
```

```
            Scanner scanner = new Scanner(System.in);
            System.out.print("Enter the ID to update: ");
            String idToUpdate = scanner.nextLine();

            // Prompt the user for the new data
            System.out.print("Enter the new name (or press Enter to skip): ");
            String newName = scanner.nextLine();

            System.out.print("Enter the new address (or press Enter to skip): ");
            String newAddress = scanner.nextLine();

            System.out.print("Enter the new email (or press Enter to skip): ");
            String newEmail = scanner.nextLine();

            System.out.print("Enter the new mobile (or press Enter to skip): ");
            String newMobile = scanner.nextLine();

            System.out.print("Enter the new job title (or press Enter to skip): ");
            String newJobTitle = scanner.nextLine();

            // UPDATE statement
            String updateQuery = "UPDATE employee SET name = COALESCE(?, name), " +
                                 "address = COALESCE(?, address), " +
                                 "email = COALESCE(?, email), " +
                                 "mobile = COALESCE(?, mobile), " +
                                 "job_title = COALESCE(?, job_title) " +
                         "WHERE id = ?";

            // Creating a PreparedStatement
            try (PreparedStatement preparedStatement = connection.prepareStatement(updateQuery)) {
               // Set values for the parameters
               preparedStatement.setString(1, newName.isEmpty() ? null : newName);
               preparedStatement.setString(2, newAddress.isEmpty() ? null : newAddress);
               preparedStatement.setString(3, newEmail.isEmpty() ? null : newEmail);
               preparedStatement.setString(4, newMobile.isEmpty() ? null : newMobile);
               preparedStatement.setString(5, newJobTitle.isEmpty() ? null : newJobTitle);
               preparedStatement.setString(6, idToUpdate);

               // Execute the UPDATE query
               int rowsAffected = preparedStatement.executeUpdate();

               if (rowsAffected > 0) {
                  System.out.println("Data updated successfully.");
               } else {
                  System.out.println("No rows updated. ID not found.");
               }
            }

            // Close the connection
            connection.close();
            System.out.println("Connection closed.");

         } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
         }
      }
   }
```

To compile and Run:

```
javac -cp .;mysql-connector-j-8.3.0.jar UpdateDataInMySQL.java
java -cp .;mysql-connector-j-8.3.0.jar UpdateDataInMySQL
```

- **Search Data in MySQL Using Java:**

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Scanner;

class SearchDataInMySQL {
   public static void main(String[] args) {
      // JDBC URL, username, and password of MySQL server
```

Java MySQL connection CURD

```
        String url = "jdbc:mysql://localhost:3306/employeedata";
        String user = "root";
        String password = "Prdc@123";

        try {
            // Register the JDBC driver
            Class.forName("com.mysql.cj.jdbc.Driver");

            // Open a connection
            System.out.println("Connecting to database...");
            Connection connection = DriverManager.getConnection(url, user, password);

            // Prompt the user for the ID to search
            Scanner scanner = new Scanner(System.in);
            System.out.print("Enter the employee ID to search: ");
            String searchId = scanner.nextLine();

            // SELECT statement with a parameterized query
            String selectQuery = "SELECT * FROM employee WHERE id = ?";

            // Creating a PreparedStatement
            try (PreparedStatement preparedStatement = connection.prepareStatement(selectQuery)) {
                // Set the value for the parameter (ID to search)
                preparedStatement.setString(1, searchId);

                // Executing the SELECT query
                ResultSet resultSet = preparedStatement.executeQuery();

                // Processing the results
                if (resultSet.next()) {
                    String id = resultSet.getString("id");
                    String name = resultSet.getString("name");
                    String address = resultSet.getString("address");
                    String email = resultSet.getString("email");
                    String mobile = resultSet.getString("mobile");
                    String job_title = resultSet.getString("job_title");

                    System.out.println("ID= " + id);
                    System.out.println("Name= " + name);
                    System.out.println("Email= " + email);
                    System.out.println("Mobile= " + mobile);
                    System.out.println("Job Title= " + job_title);
                } else {
                    System.out.println("No data found for the given ID.");
                }
            }

            // Close the connection
            connection.close();
            System.out.println("Connection closed.");

        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
        }
    }
}
```

To compile and Run:

```
javac -cp .;mysql-connector-j-8.3.0.jar SearchDataInMySQL.java
java -cp .;mysql-connector-j-8.3.0.jar SearchDataInMySQL
```

ThankYou