1. Write a Java program to connect to a MySQL database using JDBC.

```
 package day11_Assessment;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class JdbcConnection {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/mydb";
        String user = "root";
        String password = "admin";

        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection(url, user, password);
            System.out.println("Connection established successfully.");
            con.close();
        } catch (ClassNotFoundException e) {
            System.out.println("MySQL JDBC Driver not found.");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Failed to connect to the database.");
            e.printStackTrace();
        }
    }
}
```

Output:

Connecton established successfully.

2. Create a Java class to insert student records into a database table.

```
package day11_Assessment;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class StudentTable {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/mydb";
```

```
        String user = "root";
        String password = "admin";

        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection(url, user, password);

            String query = "INSERT INTO student (id, name, age) VALUES (?, ?,
?)";
            PreparedStatement pstmt = con.prepareStatement(query);
            pstmt.setInt(1, 1);
            pstmt.setString(2, "John");
            pstmt.setInt(3, 20);

            int rowsInserted = pstmt.executeUpdate();
            if (rowsInserted > 0) {
                System.out.println("Student record inserted successfully.");
            }

            pstmt.close();
            con.close();
        } catch (ClassNotFoundException e) {
            System.out.println("MySQL JDBC Driver not found.");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Database error occurred.");
            e.printStackTrace();
        }
    }
}
```

3. Write a JDBC program to fetch and display all student records from the database.

```
package day11_Assessment;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class FetchStudents {
```

```
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/mydb";
        String user = "root";
        String password = "admin";

        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection(url, user, password);

            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT * FROM student");

            System.out.println("ID\tName\tAge");
            while (rs.next()) {
                int id = rs.getInt("id");
                String name = rs.getString("name");
                int age = rs.getInt("age");
                System.out.println(id + "\t" + name + "\t" + age);
            }

            rs.close();
            stmt.close();
            con.close();
        } catch (ClassNotFoundException e) {
            System.out.println("MySQL JDBC Driver not found.");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Database error occurred.");
            e.printStackTrace();
        }
    }
}
```

4. Develop a program to search a student by ID using JDBC.

```
package day11_Assessment;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
```

```java
import java.util.Scanner;

public class SearchStudentByID {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/mydb";
        String user = "root";
        String password = "admin";

        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection(url, user, password);

            Scanner sc = new Scanner(System.in);
            System.out.print("Enter Student ID: ");
            int id = sc.nextInt();

            String query = "SELECT * FROM student WHERE id = ?";
            PreparedStatement pstmt = con.prepareStatement(query);
            pstmt.setInt(1, id);

            ResultSet rs = pstmt.executeQuery();

            if (rs.next()) {
                System.out.println("ID: " + rs.getInt("id"));
                System.out.println("Name: " + rs.getString("name"));
                System.out.println("Age: " + rs.getInt("age"));
            } else {
                System.out.println("No student found with ID " + id);
            }

            rs.close();
            pstmt.close();
            con.close();
            sc.close();
        } catch (ClassNotFoundException e) {
            System.out.println("MySQL JDBC Driver not found.");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Database error occurred.");
            e.printStackTrace();
        }
    }
}
```

5. Implement an update operation to modify student details in the database using JDBC.

```
package day11_Assessment;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Scanner;

public class UpdateStudent {
   public static void main(String[] args) {
      String url = "jdbc:mysql://localhost:3306/mydb";
      String user = "root";
      String password = "admin";

      try {
         Class.forName("com.mysql.cj.jdbc.Driver");
         Connection con = DriverManager.getConnection(url, user, password);

         Scanner sc = new Scanner(System.in);
         System.out.print("Enter Student ID to update: ");
         int id = sc.nextInt();
         sc.nextLine();

         System.out.print("Enter new name: ");
         String name = sc.nextLine();

         System.out.print("Enter new age: ");
         int age = sc.nextInt();

         String query = "UPDATE student SET name = ?, age = ? WHERE id =
?";
         PreparedStatement pstmt = con.prepareStatement(query);
         pstmt.setString(1, name);
         pstmt.setInt(2, age);
         pstmt.setInt(3, id);

         int rowsUpdated = pstmt.executeUpdate();

         if (rowsUpdated > 0) {
```

```
            System.out.println("Student record updated successfully.");
        } else {
            System.out.println("No student found with the given ID.");
        }

        pstmt.close();
        con.close();
        sc.close();
    } catch (ClassNotFoundException e) {
        System.out.println("MySQL JDBC Driver not found.");
        e.printStackTrace();
    } catch (SQLException e) {
        System.out.println("Database error occurred.");
        e.printStackTrace();
    }
  }
}
```

6. Write a Java program to delete a student record from the database using JDBC.

```
package day11_Assessment;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Scanner;

public class DeleteStudent {
  public static void main(String[] args) {
    String url = "jdbc:mysql://localhost:3306/mydb";
    String user = "root";
    String password = "admin";

    try {
      Class.forName("com.mysql.cj.jdbc.Driver");
      Connection con = DriverManager.getConnection(url, user, password);

      Scanner sc = new Scanner(System.in);
      System.out.print("Enter Student ID to delete: ");
      int id = sc.nextInt();
```

```
            String query = "DELETE FROM student WHERE id = ?";
            PreparedStatement pstmt = con.prepareStatement(query);
            pstmt.setInt(1, id);

            int rowsDeleted = pstmt.executeUpdate();

            if (rowsDeleted > 0) {
                System.out.println("Student record deleted successfully.");
            } else {
                System.out.println("No student found with the given ID.");
            }

            pstmt.close();
            con.close();
            sc.close();
        } catch (ClassNotFoundException e) {
            System.out.println("MySQL JDBC Driver not found.");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Database error occurred.");
            e.printStackTrace();
        }
    }
}
```

---

7. Design a Java application to perform all CRUD (Create, Read, Update, Delete) operations on an **Employee** table using JDBC.

```
 package day11_Assessment;
import java.sql.*;
import java.util.Scanner;

public class EmployeeCRUD {
    static final String URL = "jdbc:mysql://localhost:3306/mydb";
    static final String USER = "root";
    static final String PASSWORD = "admin";

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

```
        try (Connection con = DriverManager.getConnection(URL, USER,
PASSWORD);
             Scanner sc = new Scanner(System.in)) {

            Class.forName("com.mysql.cj.jdbc.Driver");
            boolean exit = false;

            while (!exit) {
                System.out.println("\nEmployee Management System");
                System.out.println("1. Insert Employee");
                System.out.println("2. Display All Employees");
                System.out.println("3. Update Employee");
                System.out.println("4. Delete Employee");
                System.out.println("5. Search Employee by ID");
                System.out.println("6. Exit");
                System.out.print("Choose an option: ");
                int choice = sc.nextInt();
                sc.nextLine(); // consume newline

                switch (choice) {
                    case 1 -> insertEmployee(con, sc);
                    case 2 -> displayAllEmployees(con);
                    case 3 -> updateEmployee(con, sc);
                    case 4 -> deleteEmployee(con, sc);
                    case 5 -> searchEmployeeByID(con, sc);
                    case 6 -> exit = true;
                    default -> System.out.println("Invalid choice. Try again.");
                }
            }

        } catch (ClassNotFoundException e) {
            System.out.println("MySQL JDBC Driver not found.");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Database error.");
            e.printStackTrace();
        }
    }

    private static void insertEmployee(Connection con, Scanner sc) throws
SQLException {
        System.out.print("Enter Employee ID: ");
        int id = sc.nextInt();
```

```
        sc.nextLine();
        System.out.print("Enter Employee Name: ");
        String name = sc.nextLine();
        System.out.print("Enter Salary: ");
        int salary = sc.nextInt();
        System.out.print("Enter Phone Number: ");
        float phone = sc.nextFloat();

        String query = "INSERT INTO employee (emp_id, emp_name, salary,
phonenum) VALUES (?, ?, ?, ?)";
        try (PreparedStatement pstmt = con.prepareStatement(query)) {
            pstmt.setInt(1, id);
            pstmt.setString(2, name);
            pstmt.setInt(3, salary);
            pstmt.setFloat(4, phone);

            int rows = pstmt.executeUpdate();
            if (rows > 0) System.out.println("Employee inserted successfully.");
            else System.out.println("Insertion failed.");
        }
    }

    private static void displayAllEmployees(Connection con) throws
SQLException {
        String query = "SELECT * FROM employee";
        try (Statement stmt = con.createStatement();
             ResultSet rs = stmt.executeQuery(query)) {
            System.out.println("ID\tName\tSalary\tPhone Number");
            while (rs.next()) {
                System.out.printf("%d\t%s\t%d\t%.0f%n",
                        rs.getInt("emp_id"),
                        rs.getString("emp_name"),
                        rs.getInt("salary"),
                        rs.getFloat("phonenum"));
            }
        }
    }

    private static void updateEmployee(Connection con, Scanner sc) throws
SQLException {
        System.out.print("Enter Employee ID to update: ");
        int id = sc.nextInt();
        sc.nextLine();
```

```
        System.out.print("Enter new Employee Name: ");
        String name = sc.nextLine();
        System.out.print("Enter new Salary: ");
        int salary = sc.nextInt();
        System.out.print("Enter new Phone Number: ");
        float phone = sc.nextFloat();

        String query = "UPDATE employee SET emp_name = ?, salary = ?,
phonenum = ? WHERE emp_id = ?";
        try (PreparedStatement pstmt = con.prepareStatement(query)) {
            pstmt.setString(1, name);
            pstmt.setInt(2, salary);
            pstmt.setFloat(3, phone);
            pstmt.setInt(4, id);

            int rows = pstmt.executeUpdate();
            if (rows > 0) System.out.println("Employee updated successfully.");
            else System.out.println("Employee ID not found.");
        }
    }

    private static void deleteEmployee(Connection con, Scanner sc) throws
SQLException {
        System.out.print("Enter Employee ID to delete: ");
        int id = sc.nextInt();

        String query = "DELETE FROM employee WHERE emp_id = ?";
        try (PreparedStatement pstmt = con.prepareStatement(query)) {
            pstmt.setInt(1, id);

            int rows = pstmt.executeUpdate();
            if (rows > 0) System.out.println("Employee deleted successfully.");
            else System.out.println("Employee ID not found.");
        }
    }

    private static void searchEmployeeByID(Connection con, Scanner sc)
throws SQLException {
        System.out.print("Enter Employee ID to search: ");
        int id = sc.nextInt();

        String query = "SELECT * FROM employee WHERE emp_id = ?";
        try (PreparedStatement pstmt = con.prepareStatement(query)) {
```

```
            pstmt.setInt(1, id);

            try (ResultSet rs = pstmt.executeQuery()) {
                if (rs.next()) {
                    System.out.println("ID: " + rs.getInt("emp_id"));
                    System.out.println("Name: " + rs.getString("emp_name"));
                    System.out.println("Salary: " + rs.getInt("salary"));
                    System.out.println("Phone Number: " + rs.getFloat("phonenum"));
                } else {
                    System.out.println("Employee not found.");
                }
            }
        }
    }
}
```

8. Create a JDBC-based program to count the total number of rows in a table.

```
package day11_Assessment;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class CountRows {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/mydb";
        String user = "root";
        String password = "admin";

        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection(url, user, password);

            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT COUNT(*) AS total
FROM employee");

            if (rs.next()) {
                int count = rs.getInt("total");
```

```
            System.out.println("Total number of rows in employee table: " +
count);
        }

        rs.close();
        stmt.close();
        con.close();
      } catch (ClassNotFoundException e) {
        System.out.println("MySQL JDBC Driver not found.");
        e.printStackTrace();
      } catch (SQLException e) {
        System.out.println("Database error occurred.");
        e.printStackTrace();
      }
   }
}
```

9. Develop a program to sort student data in ascending order by name using SQL in JDBC.

```
package day11_Assessment;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class SortStudentByName {
   public static void main(String[] args) {
      String url = "jdbc:mysql://localhost:3306/mydb";
      String user = "root";
      String password = "admin";

      try {
        Class.forName("com.mysql.cj.jdbc.Driver");
        Connection con = DriverManager.getConnection(url, user, password);

        Statement stmt = con.createStatement();
        String query = "SELECT * FROM student ORDER BY name ASC";
        ResultSet rs = stmt.executeQuery(query);

        System.out.println("ID\tName\tAge");
```

```
            while (rs.next()) {
                int id = rs.getInt("id");
                String name = rs.getString("name");
                int age = rs.getInt("age");
                System.out.println(id + "\t" + name + "\t" + age);
            }

            rs.close();
            stmt.close();
            con.close();
        } catch (ClassNotFoundException e) {
            System.out.println("MySQL JDBC Driver not found.");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Database error occurred.");
            e.printStackTrace();
        }
    }
}
```

10. Write a program to display all students whose percentage is greater than 75 using JDBC and SQL WHERE clause.

```
package day11_Assessment;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class StudentsAbove75Percent {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/mydb";
        String user = "root";
        String password = "admin";

        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection(url, user, password);

            Statement stmt = con.createStatement();
            String query = "SELECT * FROM student WHERE percentage > 75";
```

```
            ResultSet rs = stmt.executeQuery(query);

            System.out.println("ID\tName\tPercentage");
            while (rs.next()) {
                int id = rs.getInt("id");
                String name = rs.getString("name");
                float percentage = rs.getFloat("percentage");
                System.out.println(id + "\t" + name + "\t" + percentage);
            }

            rs.close();
            stmt.close();
            con.close();
        } catch (ClassNotFoundException e) {
            System.out.println("MySQL JDBC Driver not found.");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Database error occurred.");
            e.printStackTrace();
        }
    }
}
```

---

11. Use **PreparedStatement** to insert multiple student records into the database.

```
package day11_Assessment;
import java.sql.*;
import java.util.ArrayList;
        import java.util.List;
        import java.sql.*;
        import java.util.*;

        // 1. Student class (Java Bean)
        class Student {
            int rollno;
            String name;
            int age;
```

```
    Student(int rollno, String name, int age) {
        this.rollno = rollno;
        this.name = name;
        this.age = age;
    }
}

public class PreparedStatement {
    public static void main(String[] args) {
        // 2. Database credentials
        String url = "jdbc:mysql://localhost:3306/mydb"; // Replace with
your DB name
        String user = "root"; // Replace with your DB user
        String password = "root"; // Replace with your DB password

        // 3. List of students to insert
        List<Student> students1 = new ArrayList<>();
        students1.add(new Student(101, "Neeva Sharma", 20));
        students1.add(new Student(102, "Reeva Sharma", 20));
        students1.add(new Student(103, "Shiva Upadhyay", 20));
        students1.add(new Student(104, "Amit Verma", 22));
        students1.add(new Student(105, "Sonal Mehta", 19));

        // 4. JDBC code
        try (Connection con = DriverManager.getConnection(url, user,
password)) {
            // Insert records
            String insertQuery = "INSERT INTO students1 (rollno, name,
age) VALUES (?, ?, ?)";
            PreparedStatement insertPst = (PreparedStatement)
con.prepareStatement(insertQuery);

            for (Student s : students1) {
                ((java.sql.PreparedStatement) insertPst).setInt(1, s.rollno);
                ((java.sql.PreparedStatement) insertPst).setString(2, s.name);
                ((java.sql.PreparedStatement) insertPst).setInt(3, s.age);
                ((java.sql.PreparedStatement) insertPst).addBatch(); // Adds
this set of data to the batch
            }

            int[] result = ((Statement) insertPst).executeBatch(); // Executes
all insertions
```

```
                System.out.println(result.length + " records inserted
successfully.\n");

                // Fetch and print all records
                String selectQuery = "SELECT * FROM students1";
                PreparedStatement selectPst = (PreparedStatement)
con.prepareStatement(selectQuery);
                ResultSet rs = ((java.sql.PreparedStatement)
selectPst).executeQuery();

                System.out.println("ID\tName\t\t\tMarks");
                System.out.println("-----------------------------------------");
                while (rs.next()) {
                    int rollno = rs.getInt("rollno");
                    String name = rs.getString("name");
                    int age = rs.getInt("age");

                    System.out.printf("%d\t%-20s\t%d\n", rollno, name, age);
                }

            } catch (SQLException e) {
                e.printStackTrace();
            }
        }

            }
```

---

12. Create a **Hospital Management System** database. Using JDBC, implement:

- Register new patient
- Assign doctor
- Generate billing

```
import java.sql.*;
public class HospitalCallable {
   public static void main(String[] args) {
      String url = "jdbc:mysql://localhost:3306/mydb";
```

```
        String user = "root";
        String password = "admin";
        try (Connection con = DriverManager.getConnection(url, user, password))
{
            System.out.println("Database Connected Successfully\n");
            // patient count
            System.out.println("Average Patient Count Per Day:");
            CallableStatement cst1 = con.prepareCall("{CALL
PrintAvgPatientCountDaily()}");
            ResultSet rs1 = cst1.executeQuery();
            if (rs1.next()) {
               double avgCount = rs1.getDouble("Avg_Patient_Count_Per_Day");
               System.out.println("Average Patient Count Per Day = " + avgCount +
"\n");
            }
            // number of patients in same ward
            System.out.println("Patients in Ward No 2:");
            CallableStatement cst2 = con.prepareCall("{CALL
PrintPatientsSameWard(?)}");
            cst2.setInt(1, 2);
            ResultSet rs2 = cst2.executeQuery();

            System.out.println("ID\tName\t\tWard\tAdmission Date");
            while (rs2.next()) {
               int id = rs2.getInt("patient_id");
               String name = rs2.getString("patient_name");
               int ward = rs2.getInt("ward_no");
               Date date = rs2.getDate("admission_date");
               System.out.printf("%d\t%-15s\t%d\t%s\n", id, name, ward, date);
            }
            System.out.println();
            System.out.println("Patients Admited on  same Date:");
            CallableStatement cst3 = con.prepareCall("{CALL
PrintPatientsByAdmissionDate()}");
            ResultSet rs3 = cst3.executeQuery();

            System.out.println("ID\tName\t\tWard\tAdmission Date");
            while (rs3.next()) {
               int id = rs3.getInt("patient_id");
               String name = rs3.getString("patient_name");
               int ward = rs3.getInt("ward_no");
               Date date = rs3.getDate("admission_date");
               System.out.printf("%d\t%-15s\t%d\t%s\n", id, name, ward, date);
```

```
            }

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