Lab 9

1. Perform the following tasks:

a. Create Student table with following attributes (STUDENT_ID , FIRST_NAME, LAST_NAME, PHONE_NUMBER, MARKS, COURSE_ID).

Query :

```
CREATE TABLE Student (

  STUDENT_ID INT PRIMARY KEY,
  FIRST_NAME VARCHAR(50),
  LAST_NAME VARCHAR(50),
  PHONE_NUMBER VARCHAR(15),
  MARKS INT,
  COURSE_ID INT
);
```

Output:

```
    -> );
Query OK, 0 rows affected (0.04 sec)
```

b. Create Course table with following attributes (COURSE_ID, COURSE_NAME).

Query :

```
CREATE TABLE Course (

  COURSE_ID INT PRIMARY KEY,
  COURSE_NAME VARCHAR(100)
);

INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (1, 'Mathematics');
INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (2, 'Physics');
INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (3, 'Chemistry');
INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (4, 'Computer Science');
```

Output:

```
Query OK, 0 rows affected (0.10 sec)

mysql>
mysql> INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (1, 'Mathematics');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (2, 'Physics');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (3, 'Chemistry');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (4, 'Computer Science');
```

c. Write a SQL statement to insert 8 records with your own value into the tables.

Query :

```
INSERT INTO Student VALUES (101, 'Ashish', 'Kashyap', '1234567890', 85, 1);
INSERT INTO Student  VALUES (102, 'Sudeep', 'Prajapti', '2345678901', 92, 1);
INSERT INTO Student  VALUES (103, 'Akash ', 'Paswan', '3456789012', 76, 2);
INSERT INTO Student  VALUES (104, 'Sneha', 'Dhodi', '4567890123', 88, 3);
INSERT INTO Student  VALUES (105, 'Priyanka', 'Dhule', '5678901234', 91, 4);
INSERT INTO Student  VALUES (106, 'Archna', 'Prajapati', '6789012345', 79, 2);
INSERT INTO Student VALUES (107, 'Aditi', 'Bhoir', '7890123456', 85, 3);
INSERT INTO Student VALUES (108, 'Akash', 'Prajapati', '8901234567', 95, 1);
```

Output:

```
mysql> INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (3, 'Chemistry');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Course (COURSE_ID, COURSE_NAME) VALUES (4, 'Computer Science');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Student VALUES (101, 'Ashish', 'Kashyap', '1234567890', 85, 1);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Student  VALUES (102, 'Sudeep', 'Prajapti', '2345678901', 92, 1);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Student  VALUES (103, 'Akash ', 'Paswan', '3456789012', 76, 2);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Student  VALUES (104, 'Sneha', 'Dhodi', '4567890123', 88, 3);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Student  VALUES (105, 'Priyanka', 'Dhule', '5678901234', 91, 4);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Student  VALUES (106, 'Archna', 'Prajapati', '6789012345', 79, 2);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Student VALUES (107, 'Aditi', 'Bhoir', '7890123456', 85, 3);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Student VALUES (108, 'Akash', 'Prajapati', '8901234567', 95, 1);
```

d. Write a query to get the number of students with the same course.

Query :

```
SELECT COURSE_ID, COUNT(*) AS NumberOfStudents FROM Student GROUP BY COURSE_ID;
```

Output:

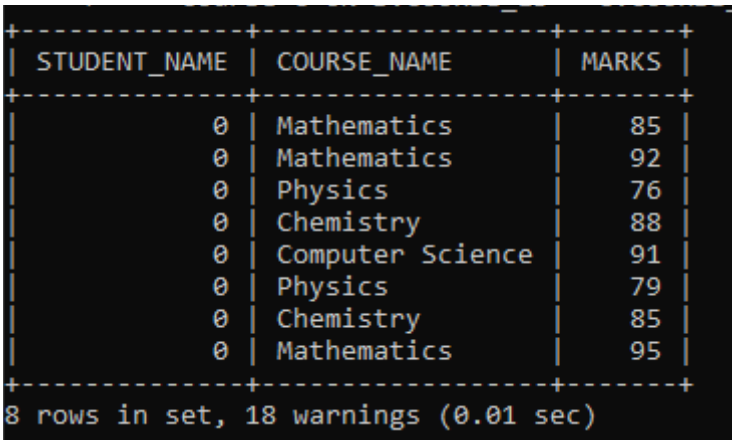
```
+-----------+------------------+
| COURSE_ID | NumberOfStudents |
+-----------+------------------+
|         1 |                3 |
|         2 |                2 |
|         3 |                2 |
|         4 |                1 |
+-----------+------------------+
4 rows in set (0.02 sec)
```

f. Write a query to get the student name, course name and marks of the students.

Query :

```
SELECT
  S.FIRST_NAME || ' ' || S.LAST_NAME AS STUDENT_NAME,
  C.COURSE_NAME,
  S.MARKS
FROM
  Student S
JOIN
  Course C ON S.COURSE_ID = C.COURSE_ID;
```

Output:



STUDENT_NAME	COURSE_NAME	MARKS
0	Mathematics	85
0	Mathematics	92
0	Physics	76
0	Chemistry	88
0	Computer Science	91
0	Physics	79
0	Chemistry	85
0	Mathematics	95

8 rows in set, 18 warnings (0.01 sec)

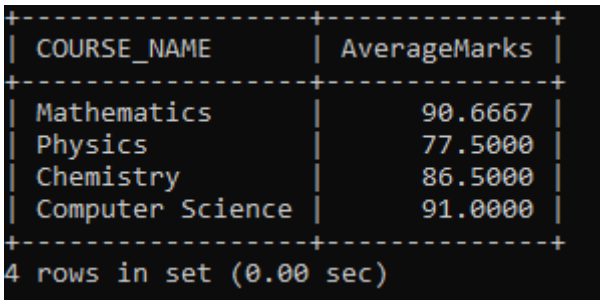
g. Write a query to get the Average marks of students course wise.

Query :

```
SELECT
  C.COURSE_NAME,
  AVG(S.MARKS) AS AverageMarks
FROM
  Student S
JOIN
  Course C ON S.COURSE_ID = C.COURSE_ID
GROUP BY
  C.COURSE_NAME;
```

Output:

.



COURSE_NAME	AverageMarks
Mathematics	90.6667
Physics	77.5000
Chemistry	86.5000
Computer Science	91.0000

4 rows in set (0.00 sec)

2. Create database for hospital management system & Perform the following tasks:

a. Create HEALTH CARE WORKERS table with following attributes (EMPLOYEE_ID , FIRST_NAME, LAST_NAME,EMAIL, PHONE_NUMBER, HIRE_DATE, SALARY, DESIGNATION)..

Query :

```
CREATE DATABASE HospitalManagement1;


USE HospitalManagement1;


CREATE TABLE HEALTH_CARE_WORKERS (
  EMPLOYEE_ID INT PRIMARY KEY,
  FIRST_NAME VARCHAR(50),
  LAST_NAME VARCHAR(50),
  EMAIL VARCHAR(100),
  PHONE_NUMBER VARCHAR(15),
  HIRE_DATE DATE,
  SALARY DECIMAL(10, 2),
  DESIGNATION VARCHAR(50)
);
```

Output:

```
Query OK, 0 rows affected (0.03 sec)
```

b. Create PATIENT table with following attributes (PATIENT_ID,NAME, PHONE_NUMBER).

Query :

```
CREATE TABLE PATIENT (

  PATIENT_ID INT PRIMARY KEY,
  NAME VARCHAR(100),
  PHONE_NUMBER VARCHAR(15)
);
```

output:

```
Query OK, 0 rows affected (0.03 sec)
```

c. Write a SQL statement to insert 10 records with your own value into the tables.

Query :

```
INSERT INTO PATIENT VALUES (1, 'SACHIN PRASAD', '7505354678');

INSERT INTO PATIENT VALUES (2, 'ARCHI PRAJAPATI', '653546767');
INSERT INTO PATIENT VALUES (3, 'POONAM', '56705354678');
INSERT INTO PATIENT VALUES (4, 'ANUJ SADA', '653546767');
INSERT INTO PATIENT VALUES (5, 'DON KKR', '7705354676');
INSERT INTO PATIENT VALUES (6, 'KAJU SOD', '6535676667');
INSERT INTO PATIENT VALUES (7, 'MASSIMO', '6995354678');
INSERT INTO PATIENT VALUES (8, 'DIVID', '653546767');
INSERT INTO PATIENT VALUES (9, 'RIYA PAL', '6995354678');
INSERT INTO PATIENT VALUES (10, 'DIVID', '653546767');
```

output:

```
mysql> USE HospitalManagement1;
Database changed
mysql> INSERT INTO PATIENT VALUES (1, 'SACHIN PRASAD', '7505354678');
Query OK, 1 row affected (0.58 sec)

mysql> INSERT INTO PATIENT VALUES (2, 'ARCHI PRAJAPATI', '653546767');
Query OK, 1 row affected (1.49 sec)

mysql> INSERT INTO PATIENT VALUES (3, 'POONAM', '56705354678');
Query OK, 1 row affected (0.30 sec)

mysql> INSERT INTO PATIENT VALUES (4, 'ANUJ SADA', '653546767');
Query OK, 1 row affected (0.27 sec)

mysql> INSERT INTO PATIENT VALUES (5, 'DON KKR', '7705354676');
Query OK, 1 row affected (0.30 sec)

mysql> INSERT INTO PATIENT VALUES (6, 'KAJU SOD', '6535676667');
Query OK, 1 row affected (0.33 sec)

mysql> INSERT INTO PATIENT VALUES (7, 'MASSIMO', '6995354678');
Query OK, 1 row affected (0.23 sec)
```

d. Write a query to get the names (first_name, last_name),Designation, salary.

Query :

```
SELECT FIRST_NAME, LAST_NAME, DESIGNATION, SALARY
FROM EMPLOYEE;
```

e. Write a query to get the number of employees with the same Designation

Query :

```
SEELCT FIRST_NAME,LAST_NAME,DESIGNATION,SALARY
FROM HEALTH_CARE_WORKER;
```

output:

```
+------------+-----------+--------------------+--------+
| FIRST_NAME | LAST_NAME | DESIGNATION        | SALARY |
+------------+-----------+--------------------+--------+
| John       | Doe       | Health Care Worker | 30000  |
| Jane       | Smith     | Health Care Worker | 32000  |
| Alice      | Johnson   | Health Care Worker | 28000  |
+------------+-----------+--------------------+--------+
3 rows in set (0.00 sec)
```

f. Write a query to get employee name who are getting salary more than 25000.

Query :

```
SELECT designation, COUNT(*) AS NO_OF_EMP

FROM employees
GROUP BY designation;
```

Output:

```
mysql> SELECT designation, COUNT(*) AS NO_OF_EMP FROM employees GROUP BY designation;
+---------------------+-----------+
| designation         | NO_OF_EMP |
+---------------------+-----------+
| Health Care Worker  | 5         |
| Nurse               | 3         |
| Doctor              | 7         |
+---------------------+-----------+
3 rows in set (0.01 sec)
```

g. Fetch HEALTH CARE WORKERS name using their employee id.

Query :

```
SELECT NAME FROM EMP WHERE DESIGNATION = 'HEALTH CARE WORKER' AND EMP_ID
IN(10,20,30,40);
```

output:

```
mysql> SELECT NAME FROM EMP WHERE DESIGNATION = 'HEALTH CARE WORKER' AN
+--------------+
| NAME         |
+--------------+
| John Doe     |
| Jane Smith   |
| Alice Johnson|
+--------------+
3 rows in set (0.00 sec)
```

3.Consider two tables, customers and orders, with the following structures:

Customers Table: customer_id (Primary Key) first_name Last_name

Orders Table: order_id (Primary Key) customer_id (Foreign Key) order_date Total_amount

Write an SQL query to retrieve the first and last names of customers along with the order date and total amount of their orders.

Query :

```
SELECT
```

```
  Customers.first_name,
  Customers.last_name,
  Orders.order_date,
  Orders.total_amount
FROM
  Customers
JOIN
  Orders ON Customers.customer_id = Orders.customer_id;
```

output:

```
+------------+-----------+------------+--------------+
| first_name | last_name | order_date | total_amount |
+------------+-----------+------------+--------------+
| Rajesh     | Sharma    | 2024-09-01 | 5000.00      |
| Priya      | Kumar     | 2024-09-02 | 7500.50      |
| Anjali     | Verma     | 2024-09-03 | 12000.75     |
| Sunil      | Patil     | 2024-09-04 | 3000.00      |
| Meena      | Gupta     | 2024-09-05 | 4500.25      |
+------------+-----------+------------+--------------+
5 rows in set (0.00 sec)
```

4.Consider two tables, departments and employees, with the following structures:

Departments Table: department_id (Primary Key) department_name

Employees Table: employee_id (Primary Key) first_name last_name department_id (Foreign Key)

Write an SQL query to retrieve a list of all departments and the names of employees who belong to each department. Use a LEFT JOIN to include departments that have no employees.

Query:

```
SELECT
  DEPARTMENTS.DEPARTMENT_NAME,
  EMPLOYEES.FIRST_NAME,
  EMPLOYEES.LAST_NAME
FROM   DEPARTMENTS
LEFT JOIN
EMPLOYEES ON DEPARTMENTS.DEPARTMENT_ID = EMPLOYEES.DEPARTMENT_ID;
```

Output:

```
+-----------------+------------+-----------+
| DEPARTMENT_NAME | FIRST_NAME | LAST_NAME |
+-----------------+------------+-----------+
| HR              | Sunita     | Rao       |
| Engineering     | Rakesh     | Mehta     |
| Marketing       | Priya      | Deshmukh  |
| Finance         | NULL       | NULL      |
+-----------------+------------+-----------+
4 rows in set (0.00 sec)
```

4. Write a program to show JDBC connection with MYSQL and perform the following operations:

Create table Customer with following fields:

Custno, Custame,Custaddress,Phoneno, City, Pincode, Country

Insert 5 records in Customer table.

a. Insert values

b. Delete values

c. update city name Shimla to Shilong.

d. Show table in the console

code:

Note : i was created two file in my idle but hare i have marged both files code.

```
package project2;


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

public class JdbcMySQLExample {


   private static final String URL = "jdbc:mysql://localhost:3306/demo";
    private static final String USER = "roots";
    private static final String PASSWORD = "1234";

  private static Connection connection;

  public static void main(String[] args) {
    try {
      // Establish a connection to the database
      connection = DriverManager.getConnection(URL, USER, PASSWORD);

      //Create the Customer table
      createCustomerTable();

      //Insert 5 records into the Customer table
      insertCustomers();

      //Delete a record from the Customer table
      deleteCustomer(3);  // Example: Delete customer with Custno = 3

      //Update the city name from Shimla to Shilong
      updateCustomerCity("Shimla", "Shilong");
```

```
      //Display the Customer table in the console
      displayCustomers();

    } catch (SQLException e) {
      e.printStackTrace();
    } finally {
      // 7. Close the connection
      try {
        if (connection != null) {
          connection.close();
        }
      } catch (SQLException e) {
        e.printStackTrace();
      }
    }
  }

  private static void createCustomerTable() throws SQLException {
    String createTableSQL = "CREATE TABLE IF NOT EXISTS Customer ("
        + "Custno INT PRIMARY KEY AUTO_INCREMENT, "
        + "Custname VARCHAR(50), "
        + "Custaddress VARCHAR(100), "
        + "Phoneno VARCHAR(15), "
        + "City VARCHAR(50), "
        + "Pincode VARCHAR(10), "
        + "Country VARCHAR(50))";

    try (Statement statement = connection.createStatement()) {
      statement.execute(createTableSQL);
      System.out.println("Table 'Customer' is created!");
    }
  }

  private static void insertCustomers() throws SQLException {
    String insertSQL = "INSERT INTO Customer (Custname, Custaddress, Phoneno, City,
Pincode, Country) VALUES (?, ?, ?, ?, ?, ?)";

    try (PreparedStatement preparedStatement =
connection.prepareStatement(insertSQL)) {
      connection.setAutoCommit(false);

      preparedStatement.setString(1, "Ashish Kashyap");
      preparedStatement.setString(2, "Boisar");
      preparedStatement.setString(3, "1234567890");
      preparedStatement.setString(4, "Shimla");
      preparedStatement.setString(5, "171001");
      preparedStatement.setString(6, "India");
      preparedStatement.addBatch();

      preparedStatement.setString(1, "mohan thakur");
```

```
        preparedStatement.setString(2, "Boisar");
        preparedStatement.setString(3, "0987654321");
        preparedStatement.setString(4, "maharastra");
        preparedStatement.setString(5, "110001");
        preparedStatement.setString(6, "India");
        preparedStatement.addBatch();

        preparedStatement.setString(1, "sanika jadhav");
        preparedStatement.setString(2, "palghar");
        preparedStatement.setString(3, "2345678901");
        preparedStatement.setString(4, "maharastra");
        preparedStatement.setString(5, "400001");
        preparedStatement.setString(6, "India");
        preparedStatement.addBatch();

        preparedStatement.setString(1, "neha gupta");
        preparedStatement.setString(2, "gujrat");
        preparedStatement.setString(3, "3456789012");
        preparedStatement.setString(4, "gujrat");
        preparedStatement.setString(5, "700001");
        preparedStatement.setString(6, "India");
        preparedStatement.addBatch();

        preparedStatement.setString(1, "mohan singh");
        preparedStatement.setString(2, "boisar");
        preparedStatement.setString(3, "4567890123");
        preparedStatement.setString(4, "Chennai");
        preparedStatement.setString(5, "600001");
        preparedStatement.setString(6, "India");
        preparedStatement.addBatch();

        preparedStatement.executeBatch();
        connection.commit();
        System.out.println("Records inserted successfully!");
      }
    }

    private static void deleteCustomer(int custNo) throws SQLException {
      String deleteSQL = "DELETE FROM Customer WHERE Custno = ?";

      try (PreparedStatement preparedStatement =
connection.prepareStatement(deleteSQL)) {
        preparedStatement.setInt(1, custNo);
        int rowsAffected = preparedStatement.executeUpdate();
        System.out.println("Deleted " + rowsAffected + " record(s) where Custno = " +
custNo);
      }
    }

    private static void updateCustomerCity(String oldCity, String newCity) throws
SQLException {
```

```
    String updateSQL = "UPDATE Customer SET City = ? WHERE City = ?";

    try (PreparedStatement preparedStatement =
connection.prepareStatement(updateSQL)) {
      preparedStatement.setString(1, newCity);
      preparedStatement.setString(2, oldCity);
      int rowsAffected = preparedStatement.executeUpdate();
      System.out.println("Updated " + rowsAffected + " record(s) from " + oldCity +
" to " + newCity);
    }
  }

  private static void displayCustomers() throws SQLException {
    String selectSQL = "SELECT * FROM Customer";

    try (Statement statement = connection.createStatement();
       ResultSet resultSet = statement.executeQuery(selectSQL)) {


      while (resultSet.next()) {
        int custNo = resultSet.getInt("Custno");
        String custName = resultSet.getString("Custname");
        String custAddress = resultSet.getString("Custaddress");
        String phoneNo = resultSet.getString("Phoneno");
        String city = resultSet.getString("City");
        String pincode = resultSet.getString("Pincode");
        String country = resultSet.getString("Country");

        System.out.printf("%-6d | %-12s | %-15s | %-10s | %-8s | %-7s | %-7s%n",
                  custNo, custName, custAddress, phoneNo, city, pincode, country);
      }
    }
  }
}
```

output: