**Program 1:**

Write a program with Java database connectivity to insert data in MySQL database using prepared statement only.

You can take  random data attributes of a student or employee to insert into table.

**Code:**

import java.sql.\*;

import java.util.Scanner;

public class DataInsertion

{

public static void main(String[] args) throws Exception

{

Scanner sc=new Scanner(System.in);

//taking input from user

System.out.println("Enter Department Id:");

int id=Integer.parseInt(sc.nextInt());

System.out.println("Enter Department name:");

String name=sc.nextLine();

System.out.println("Enter location:");

String location=sc.nextLine();

//registering driver class

Class.forName("oracle.jdbc.driver.OracleDriver");

//Establishing conneection

Connection connection=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","scott","tiger");

//creatinh sql query

String sql="insert into department values(?,?,?)";

PreparedStatement statement=connection.prepareStatement(sql);

//set the values

statement.setInt(1,id);

statement.setString(2,name);

statement.setString(3,location);

//executing the statement

statement.executeUpdate();

System.out.println("New Record inserted);

//closing the connection

connection.close();

}

}

**Program 2:**

Write a program with Java database connectivity to insert data in MySQL database using callable statement only.

You can take  random data attributes of a student or employee to insert into table.

**Code:**

/\*Store procedure to insert employee Details in database:

Sql Statement:

DELIMITER //

CREATE PROCEDURE InsertEmployeeDetail(IN eid INT,IN name VARCHAR(50), IN address VARCHAR(50), IN salary INT)

BEGIN

    INSERT INTO employee (eid,name,address,salary) VALUES (eid,name, address, salary);

END;

//

DELIMITER ;

\*/

package jdbcConnectivity;

import java.sql.Connection;

import java.sql.CallableStatement;

import java.sql.Types;

public class CallableStatementExercise {

    public static void main(String[] args) {

        try (Connection con = ConnectDB.dbConnect();) {

              // 3. Prepare the CallableStatement to call the stored procedure.

             // 3. Prepare the CallableStatement to call the stored procedure.

            String callStatement = "{CALL InsertEmployeeDetail(?,?, ?, ?)}";

            CallableStatement callableStatement = con.prepareCall(callStatement);

            // 4. Set the input parameters (employee details).

            callableStatement.setInt(1, 7);

            callableStatement.setString(2, "Peter Doe");

            callableStatement.setString(3, "Software Engineer");

            callableStatement.setInt(4, 75000);

            // 5. Execute the stored procedure to insert a new employee.

            callableStatement.execute();

            // 6. Close the resources.

            callableStatement.close();

            System.out.println("Employee inserted successfully");

        }catch (Exception e) {

            System.out.println(e);

        }

    }

}