

Practcial 07

Simple Statement

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

public class CRUDSS {
    public static Connection con;
    static int ch;
    static String selectQ = "SELECT * FROM
employee";
    static String eid, ename, esalary;

    public static void main(String[] args) throws
SQLException, ClassNotFoundException {

        Class.forName("com.mysql.cj.jdbc.Driver");
        CRUDSS.con =
DriverManager.getConnection("jdbc:mysql://localhost:3
306/javajdbc", "root", "password");
        Scanner sc = new Scanner(System.in);

        while (true) {
            System.out.println(
                "1. create
employee \n 2. update employee info \n 3. delete
employee \n 0. Exit \n enter your choice: ");

            ch = sc.nextInt();
            switch (ch) {
                case 1:
                    createEmp();
                    break;
                case 2:
                    updateEmp();
                    break;
                case 3:
                    deleteEmp();
                    break;
                case 0:
                    System.exit(0);
                default:

                    System.out.println("Invalid choice! Please make a
valid choice. \n\n");
            }
        }

        private static void createEmp() throws
SQLException, ClassNotFoundException {
            ResultSet rs;
            String insertQ = "INSERT INTO employee
VALUES ('104','Rupesh','90000')";

            // obtain a statement
            Statement st = con.createStatement();
            rs = st.executeQuery(selectQ);
            System.out.println("Employee table
before Insertion");
            System.out.println("EmpId \t Empname \
t EmpSalary");
            while (rs.next()) {
                eid = rs.getString(1);
```

```
                ename = rs.getString(2);
                esalary = rs.getString(3);
                System.out.println(eid + "\t" +
ename + "\t" + esalary);
            }
            int exeUpdate =
st.executeUpdate(insertQ);
            if (exeUpdate == 1) {
                System.out.println("1 row
inserted");
            }
            rs = st.executeQuery(selectQ);
            System.out.println("Employee table after
Insertion");
            System.out.println("EmpId \t Empname \
t EmpSalary");
            while (rs.next()) {
                eid = rs.getString(1);
                ename = rs.getString(2);
                esalary = rs.getString(3);
                System.out.println(eid + "\t" +
ename + "\t" + esalary);
            }

            private static void updateEmp() throws
SQLException, ClassNotFoundException {
                ResultSet rs;
                String updateQ = "UPDATE employee SET
emp_sal= '160000' WHERE emp_id = '102'";

                // obtain a statement
                Statement st = con.createStatement();
                rs = st.executeQuery(selectQ);
                System.out.println("Employee table
before Updation");
                System.out.println("EmpId \t Empname \
t EmpSalary");
                while (rs.next()) {
                    eid = rs.getString(1);
                    ename = rs.getString(2);
                    esalary = rs.getString(3);
                    System.out.println(eid + "\t" +
ename + "\t" + esalary);
                }
                int exeUpdate1 =
st.executeUpdate(updateQ);
                if (exeUpdate1 == 1) {
                    System.out.println("1 row
updated");
                }
                rs = st.executeQuery(selectQ);
                System.out.println("Employee table after
Updation");
                System.out.println("EmpId \t Empname \
t EmpSalary");
                while (rs.next()) {
                    eid = rs.getString(1);
                    ename = rs.getString(2);
                    esalary = rs.getString(3);
                    System.out.println(eid + "\t" +
ename + "\t" + esalary);
                }
            }
        }
    }
}
```

```

    }

    private static void deleteEmp() throws
SQLException, ClassNotFoundException {
        ResultSet rs;
        String deleteQ = "      DELETE FROM
employee WHERE emp_id = '102'";

        // obtain a statement
        Statement st = con.createStatement();
        rs = st.executeQuery(selectQ);
        System.out.println("Employee table
before deletion");
        System.out.println("EmpId \t Empname \
t EmpSalary");
        while (rs.next()) {
            eid = rs.getString(1);
            ename = rs.getString(2);
            esalary = rs.getString(3);
            System.out.println(eid + "\t" +
ename + "\t" + esalary);
        }
        int exeUpdate1 =
st.executeUpdate(deleteQ);
        if (exeUpdate1 == 1) {
            System.out.println("Employee is
deleted");
        }
        rs = st.executeQuery(selectQ);
        System.out.println("Employee table after
deletion");
        System.out.println("EmpId \t Empname \
t EmpSalary");
        while (rs.next()) {
            eid = rs.getString(1);
            ename = rs.getString(2);
            esalary = rs.getString(3);
            System.out.println(eid + "\t" +
ename + "\t" + esalary);
        }
    }
}

```

Output

```

1. create employee
2. update employee info
3. delete employee
0. Exit
enter your choice:
1

```

```

Employee table before Insertion
EmpId  Empname    EmpSalary
101    Akshay      160000
102    Tushar      90000
103    Sourabh     90000
1 row inserted

```

```

Employee table after Insertion
EmpId  Empname    EmpSalary
101    Akshay      160000
102    Tushar      90000

```

```

103    Sourabh     90000
104    Rupesh      90000

```

```

1. create employee
2. update employee info
3. delete employee
0. Exit
enter your choice:
2

```

```

Employee table before Updation
EmpId  Empname    EmpSalary
101    Akshay      160000
102    Tushar      90000
103    Sourabh     90000
104    Rupesh      90000
1 row updated

```

```

Employee table after Updation
EmpId  Empname    EmpSalary
101    Akshay      160000
102    Tushar      160000
103    Sourabh     90000
104    Rupesh      90000

```

```

1. create employee
2. update employee info
3. delete employee
0. Exit
enter your choice:
3

```

```

Employee table before deletion
EmpId  Empname    EmpSalary
101    Akshay      160000
102    Tushar      160000
103    Sourabh     90000
104    Rupesh      90000
Employee is deleted

```

```

Employee table after deletion
EmpId  Empname    EmpSalary
101    Akshay      160000
103    Sourabh     90000
104    Rupesh      90000

```

```

1. create employee
2. update employee info
3. delete employee
0. Exit
enter your choice:
0

```

PreparedStatement

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

public class PreparedDemo {

    PreparedStatement pstmt;
    public static Connection con;

    void printTable() {
        PreparedStatement printTbl;

        String query = "SELECT * FROM student";
        try {
            System.out.println("\n\nPrinting
Table : \n");
            printTbl =
PreparedDemo.con.prepareStatement(query);

            ResultSet rs =
printTbl.executeQuery();

            System.out.println("RollNo\
tName\tMarks");
            while (rs.next()) {
                int roll = rs.getInt(1);
                String name =
rs.getString(2);
                int mark = rs.getInt(3);
                System.out.println(roll +
"\t" + name + "\t" + mark);
            }
            System.out.println("\n");
            printTbl.close();
        } catch (SQLException e) {
            System.out.println(e);
        }
    }

    void insertInTable(int roll, String name, int
marks) {
        PreparedStatement insertTbl;

        String query = "INSERT INTO student
VALUES(?,?,?)";
        try {
            insertTbl =
PreparedDemo.con.prepareStatement(query);

            insertTbl.setInt(1, roll);
            insertTbl.setString(2, name);
            insertTbl.setInt(3, marks);

            int i = insertTbl.executeUpdate();

            insertTbl.close();
        } catch (SQLException e) {
            System.out.println(e);
        }
    }

    void updateTable(int roll, int marks) {
```

```
        PreparedStatement updateTbl;

        String query = "update student set
marks=? where roll_no=? ";
        try {
            updateTbl =
PreparedDemo.con.prepareStatement(query);

            updateTbl.setInt(1, marks);

            updateTbl.setInt(2, roll);

            int i =
updateTbl.executeUpdate();
            updateTbl.close();
        } catch (SQLException e) {
            System.out.println(e);
        }
    }

    void deleteData(int roll) {
        PreparedStatement delete;

        String query = "delete from student
where roll_no = ?";
        try {
            delete =
PreparedDemo.con.prepareStatement(query);

            delete.setInt(1, roll);

            int i = delete.executeUpdate();

            delete.close();
        } catch (SQLException e) {
            System.out.println(e);
        }
    }

    public static void main(String args[]) {
        PreparedDemo pd;

        try {
            Scanner sc = new
Scanner(System.in);

            Class.forName("com.mysql.cj.jdbc.Driver");
            PreparedDemo.con =
DriverManager.getConnection("jdbc:mysql://localhost:3
306/javajdbc", "root", "password");
            pd = new PreparedDemo();

            while (true) {
                System.out.println("1.
Print table\n2. Insert table\n3. Update Data(marks)\n4.
Delete Data\n5. Exit\n");

                int i = sc.nextInt();
                int roll_no, marks;
                String name;
                switch (i) {
                    case 1:

                        pd.printTable();

                        break;
```

```

        case 2:
            System.out.println("Enter roll no");
            roll_no =
sc.nextInt();
            System.out.println("\nEnter name");
            name =
sc.next();
            System.out.println("\nEnter marks");
            marks =
sc.nextInt();
            pd.insertInTable(roll_no, name, marks);
            pd.printTable();
            break;
        case 3:
            System.out.println("Enter roll no");
            roll_no =
sc.nextInt();
            System.out.println("\nEnter marks");
            marks =
sc.nextInt();
            pd.updateTable(roll_no, marks);
            pd.printTable();
            break;
        case 4:
            System.out.println("Enter roll no");
            roll_no =
sc.nextInt();
            pd.deleteData(roll_no);
            pd.printTable();
            break;
        case 5:
            break;
        default:
            break;
    }
} catch (Exception e) {
    System.out.println(" main" + e);
}
}

```

Output

```

1. Print table
2. Insert table
3. Update Data(marks)
4. Delete Data
5. Exit

```

1

Printing Table :

RollNo	Name	Marks
2	Vinay	90
3	Vijay	95
25	Akshay	95

```

1. Print table
2. Insert table
3. Update Data(marks)
4. Delete Data
5. Exit

```

2

Enter roll no
30

Enter name
Pranavraj

Enter marks
95

Printing Table :

RollNo	Name	Marks
2	Vinay	90
3	Vijay	95
25	Akshay	95
30	Pranavraj	95

```

1. Print table
2. Insert table
3. Update Data(marks)
4. Delete Data
5. Exit

```

3

Enter roll no
2

Enter marks
95

Printing Table :

RollNo	Name	Marks
2	Vinay	95

3	Vijay	95
25	Akshay	95
30	Pranavraj	95

1. Print table
2. Insert table
3. Update Data(marks)
4. Delete Data
5. Exit

4

Enter roll no
25

Printing Table :

RollNo	Name	Marks
2	Vinay	95
3	Vijay	95
30	Pranavraj	95

1. Print table
2. Insert table
3. Update Data(marks)
4. Delete Data
5. Exit

5

CallableStatement

```
CREATE DEFINER=`root`@`localhost` PROCEDURE
`max_salary`()
BEGIN
    SELECT * FROM employee WHERE emp_sal >
    100000;
END
```

```
CREATE DEFINER=`root`@`localhost` PROCEDURE
`total_emp`(OUT x INT)
BEGIN
    SELECT COUNT(emp_id) INTO x FROM employee;
END
```

```
CREATE DEFINER=`root`@`localhost` PROCEDURE
`salary_emp`(IN id INT,OUT sal INT)
BEGIN
    SELECT emp_sal INTO sal FROM employee WHERE
    emp_id = id;
END
```

Program

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

public class CollableDemo {
    public static Connection con;
    static String query = "{call max_salary()}";

    public static void main(String agrs[]) throws
Exception{
    Scanner sc = new Scanner(System.in);

    Class.forName("com.mysql.cj.jdbc.Driver");
    con =
    DriverManager.getConnection("jdbc:mysql://localhost:3
306/javajdbc", "root", "password");

    CallableStatement stmt =
con.prepareCall(query);
    System.out.println("Executing...");
    stmt.execute();

    ResultSet rs =
stmt.getResultSet();

    System.out.println("EMP_ID\
tName\tSALARY");
    while (rs.next()) {
        int emp_id = rs.getInt(1);
        String emp_name =
rs.getString(2);
        int emp_salary =
rs.getInt(3);

        System.out.println(emp_id + "\t" + emp_name +
"\t" + emp_salary);
    }
    System.out.println("\n");

    query = "{call total_emp(?)}";
    stmt = con.prepareCall(query);

    stmt.registerOutParameter(1,Types.INTEGER);
    System.out.println("Executing...");
    stmt.execute();

    System.out.println("Total count
of employess in company is " + stmt.getInt(1));
    System.out.println("\n");

    query = "{call salary_emp(?,?)}";
    stmt = con.prepareCall(query);
    System.out.println("Enter
Employee id");

    int ch = sc.nextInt();

    stmt.setInt(1,ch);

    stmt.registerOutParameter(2,Types.INTEGER);
    System.out.println("Executing...");
```

```
        stmt.execute();

        System.out.println("Salay of
Employee "+ch+" : " + stmt.getInt(2));
        System.out.println("\n");

    }
}
```

Output

```
cd
"/home/akshay/Akshay/Codes/Java/java2_prgms/Java
jdbc" ; /usr/bin/env /opt/java/jdk1.8.0_321/bin/java -
cp /tmp/cp_4ms2pnrnh8tf408d7ld2q6og1.jar
CollableDemo
```

Executing...

EMP_ID	Name	SALARY
101	Akshay	160000
104	Rupesh	120000

Executing...

Total count of employess in company is 3

Enter Employee id

101

Executing...

Salay of Employee 101 : 160000