

Name : Kalyani Kangle
Branch: CSE 3rd Year

Task 1: Database Connection Code

Description: In this task , I have performed the database connectivity with java application. This task includes connection code for connecting java application to database here the database used is MySQL.

For Connecting Java Application to Database , we have to follow two most important and mandatory steps:

1.Register Database Driver:

For Registering database driver , In JDBC the `forName()` method of `Class` type is used.

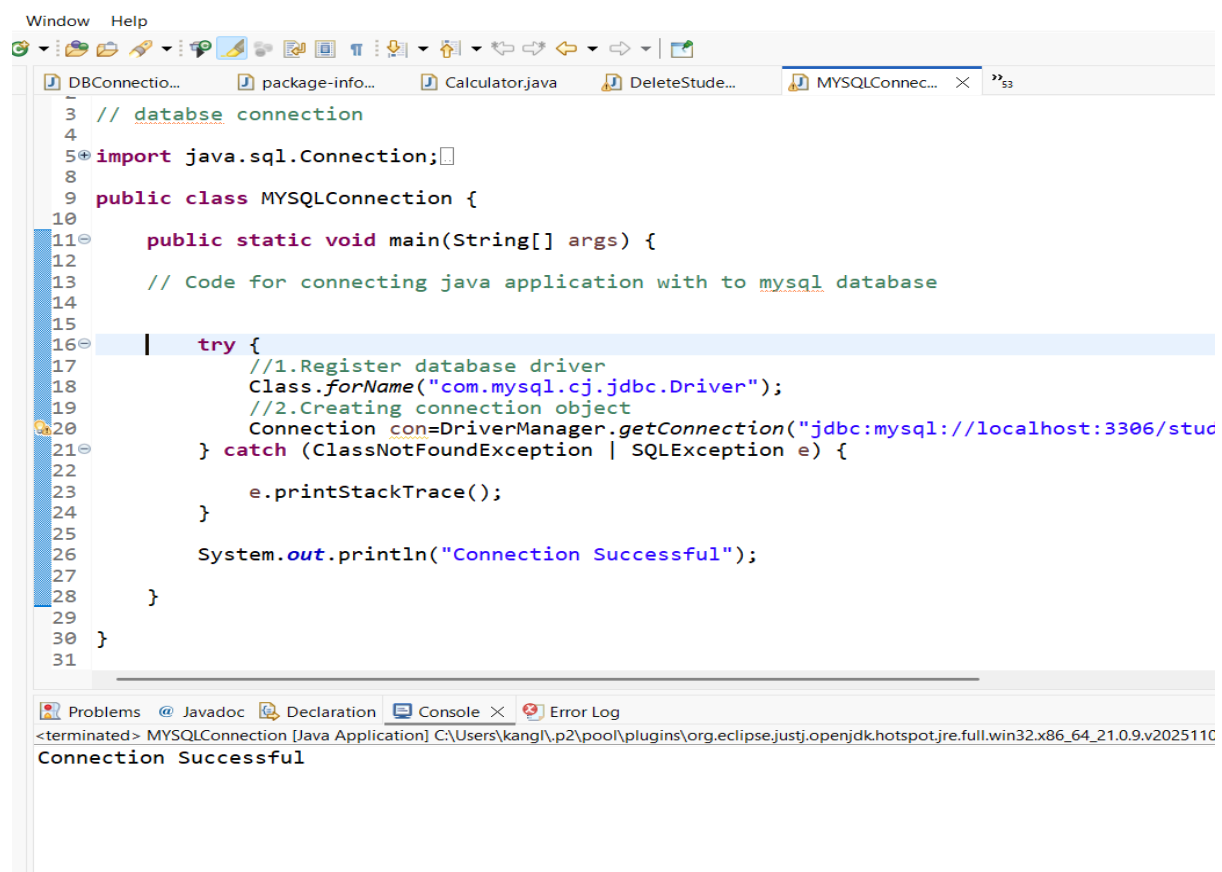
Syntax: `Class.forName("DriverName");`

2.Create Connection Object:

In this step , Connection reference is created , where `Connection` is an Interface.

Syntax: `Connection con=DriverManager.getConnection("URL","USERNAME","PASSWORD");`

Here , the `getConnection()` method of `DriverManager` class takes url , username and password as arguments to create reference on `Connection` , which is we need to use further steps.



```
Window  Help
DBConnec...  package-info...  Calculator.java  DeleteStude...  MySQLConnec...  »53
3  // databse connection
4
5+ import java.sql.Connection;
8
9  public class MySQLConnection {
10
11  public static void main(String[] args) {
12
13      // Code for connecting java application with to mysql database
14
15
16  |  try {
17      //1.Register database driver
18      Class.forName("com.mysql.cj.jdbc.Driver");
19      //2.Creating connection object
20      Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/stuc
21  } catch (ClassNotFoundException | SQLException e) {
22
23      e.printStackTrace();
24  }
25
26      System.out.println("Connection Successful");
27
28  }
29
30 }
31

Problems  @ Javadoc  Declaration  Console  Error Log
<terminated> MySQLConnection [Java Application] C:\Users\kangl\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.9.v2025110
Connection Successful
```

Name : Kalyani Kangle
Branch: CSE 3rd Year

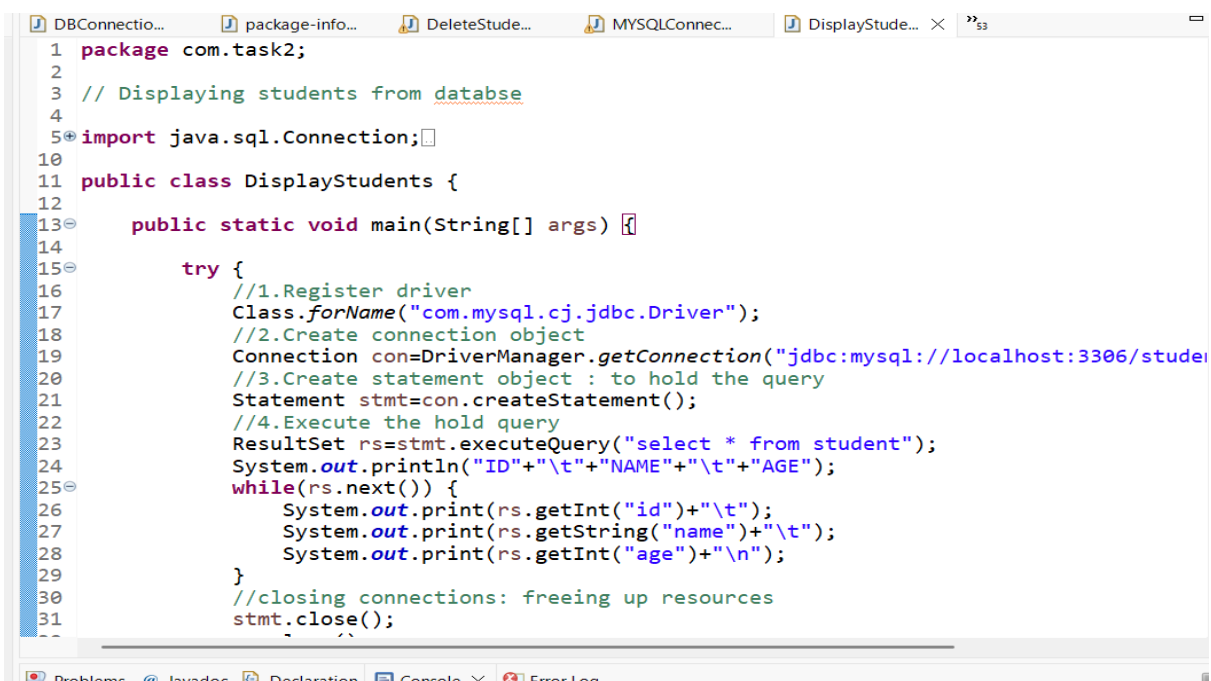
Task 2: Display All Students

Description: This task focuses on displaying the data from database to the console through java application.

In this task , First I have created a database name as students where student table is present in which student data id , name , marks are stored . This task mainly performed to display all the records from student table. For displaying data from database using java application , the java application should be connected with database.

I have completed this task by performing following 5 steps:

- 1.Registering database driver using Class.forName("DriverName")
- 2.Creating Connection reference that stores url , username and password through getConnection() method of DriverManager class.
- 3.Statement reference is created using Connection reference and use of createStatement() here createStatement() do not hold any query.
- 4.In this step , select query is hold and execute using executeQuery() , the result return by executeQuery() is stored in ResultSet rs , now next step is to extract data from rs for that rs.next() function is used to check whether there is next element is present in rs or not , if element or data is present rs.next() returns true otherwise returns false.
- 5.At last getter functions from ResultSet are used to get value from database (table) and display it.
- 6.Closing the connection using close() method.



```
1 package com.task2;
2
3 // Displaying students from database
4
5 import java.sql.Connection;
10
11 public class DisplayStudents {
12
13     public static void main(String[] args) {
14
15         try {
16             //1.Register driver
17             Class.forName("com.mysql.cj.jdbc.Driver");
18             //2.Create connection object
19             Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/studen
20             //3.Create statement object : to hold the query
21             Statement stmt=con.createStatement();
22             //4.Execute the hold query
23             ResultSet rs=stmt.executeQuery("select * from student");
24             System.out.println("ID"+"\\t"+"NAME"+"\\t"+"AGE");
25             while(rs.next()) {
26                 System.out.print(rs.getInt("id")+"\\t");
27                 System.out.print(rs.getString("name")+"\\t");
28                 System.out.print(rs.getInt("age")+"\\n");
29             }
30             //closing connections: freeing up resources
31             stmt.close();
32         }
33     }
34 }
```

Name : Kalyani Kangle
Branch: CSE 3rd Year

```
17 Class.forName("com.mysql.cj.jdbc.Driver");
18 //2.Create connection object
19 Connection con=DriverManager.getConnection("jdbc:mysql://localhost:
20 //3.Create statement object : to hold the query
21 Statement stmt=con.createStatement();
22 //4.Execute the hold query
23 ResultSet rs=stmt.executeQuery("select * from student");
24 System.out.println("ID"+"\\t"+"NAME"+"\\t"+"AGE");
25 while(rs.next()) {
26     System.out.print(rs.getInt("id")+"\\t");
27     System.out.print(rs.getString("name")+"\\t");
28     System.out.print(rs.getInt("age")+"\\n");
29 }
30 //closing connections: freeing up resources
31 stmt.close();
32 rs.close();
33 con.close();
34
35 } catch (ClassNotFoundException | SQLException e) {
36     // TODO Auto-generated catch block
37     e.printStackTrace();
38 }
39
40 }
41
42 }
43
```

Problems Javadoc Declaration Console Error Log

<terminated> DisplayStudents [Java Application] C:\Users\kangl\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21

ID	NAME	AGE
1	Kalyani	20
2	Madhavi	20
3	Vivek	21
4	Aakash	21
5	Aarya	23

Name : Kalyani Kangle
Branch: CSE 3rd Year

Task 3: Inserting Student Records

Description: In this task students records are inserted into table student , using PreparedStatement and Parameterized query. The parameterised query is used to take values from users.

This task is performed by implementing following steps:

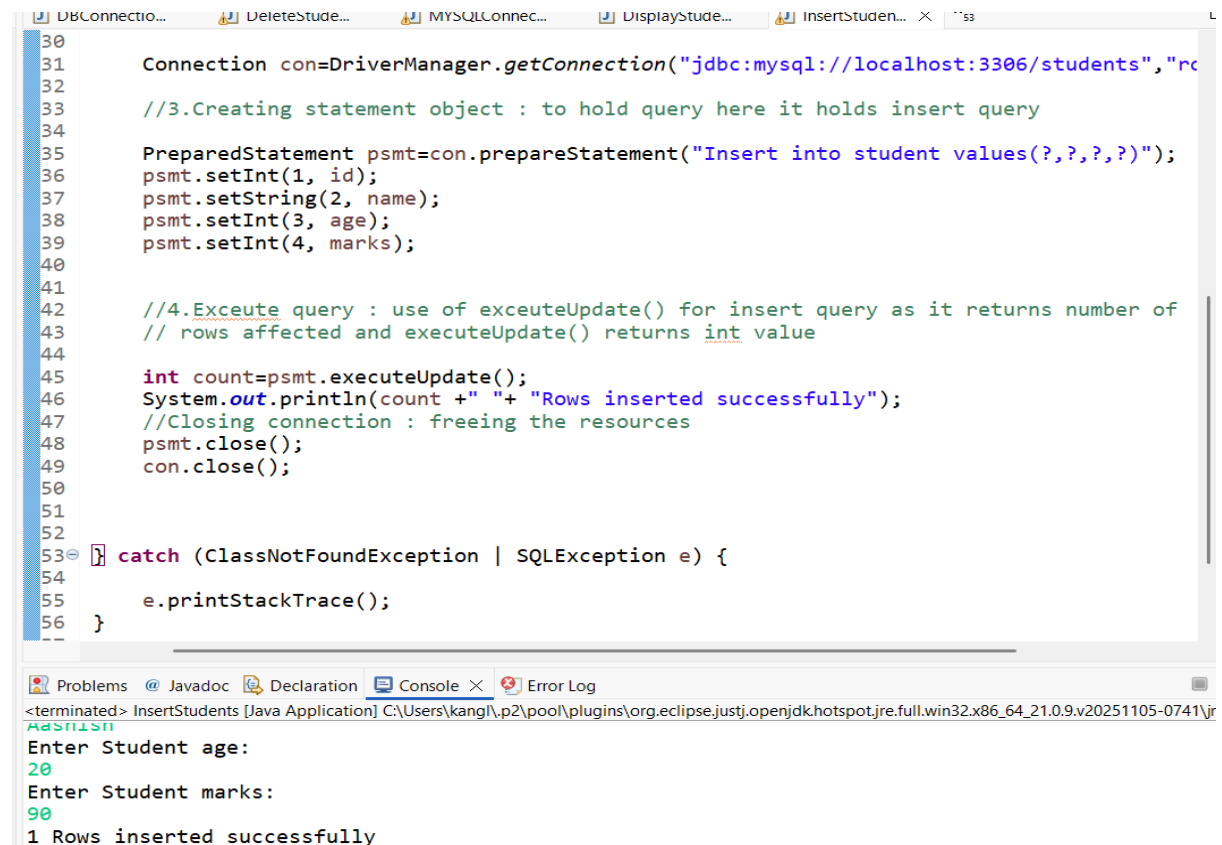
- 1.The most basic and primary steps is to register database and establishing connection with database.
- 2.Scanner class object is used to take input from users
- 3.Insert query with parameters is hold by prepareStatement() , where the parameters values are passed by user and store into that parameters using setter methods , that setter methods are used or call using PreparedStatement reference.
- 4.As Insert query is a DML statement , executeUpdate() is used for exceuting it , it returns int value that is number of rows affected.
- 5.At last Connection is closed to free resources

```
1 package com.task3;
2
3 import java.sql.Connection;
4
5 //Inserting students details in student table by taking input from user through console
6 //Inserting student record
7
8 public class InsertStudents {
9
10     public static void main(String[] args) {
11         Scanner sc=new Scanner(System.in);
12         System.out.println("Enter Student id: ");
13         int id=sc.nextInt();
14         System.out.println("Enter Student name: ");
15         String name=sc.next();
16         System.out.println("Enter Student age: ");
17         int age=sc.nextInt();
18         System.out.println("Enter Student marks: ");
19         int marks=sc.nextInt();
20
21         try {
22             //1. Register database driver
23             Class.forName("com.mysql.cj.jdbc.Driver");
24
25             //2.Creating connection object
26             Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/stude
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Problems @ Javadoc Declaration Console X Error Log
<terminated> InsertStudents [Java Application] C:\Users\kangl\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.9.v20251105-074
Enter Student id:
11
Enter Student name:
Aashish
Enter Student age:
20

Name : Kalyani Kangle

Branch: CSE 3rd Year



```
30
31     Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/students","root","root");
32
33     //3.Creating statement object : to hold query here it holds insert query
34
35     PreparedStatement psmt=con.prepareStatement("Insert into student values(?,?,?,?)");
36     psmt.setInt(1, id);
37     psmt.setString(2, name);
38     psmt.setInt(3, age);
39     psmt.setInt(4, marks);
40
41
42     //4.Excute query : use of exceuteUpdate() for insert query as it returns number of
43     // rows affected and executeUpdate() returns int value
44
45     int count=psmt.executeUpdate();
46     System.out.println(count + " " + "Rows inserted successfully");
47     //Closing connection : freeing the resources
48     psmt.close();
49     con.close();
50
51
52
53 catch (ClassNotFoundException | SQLException e) {
54
55     e.printStackTrace();
56 }
--
```

Problems @ Javadoc Declaration Console × Error Log

<terminated> InsertStudents [Java Application] C:\Users\kangl\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.9.v20251105-0741\jre\bin\java.exe

Enter Student age:
20
Enter Student marks:
90
1 Rows inserted successfully

Name : Kalyani Kangle
Branch: CSE 3rd Year

Task 4: Update Student Records

Description: This task is to update students marks using student id which is to be inserted by user and also takes new marks from the user.

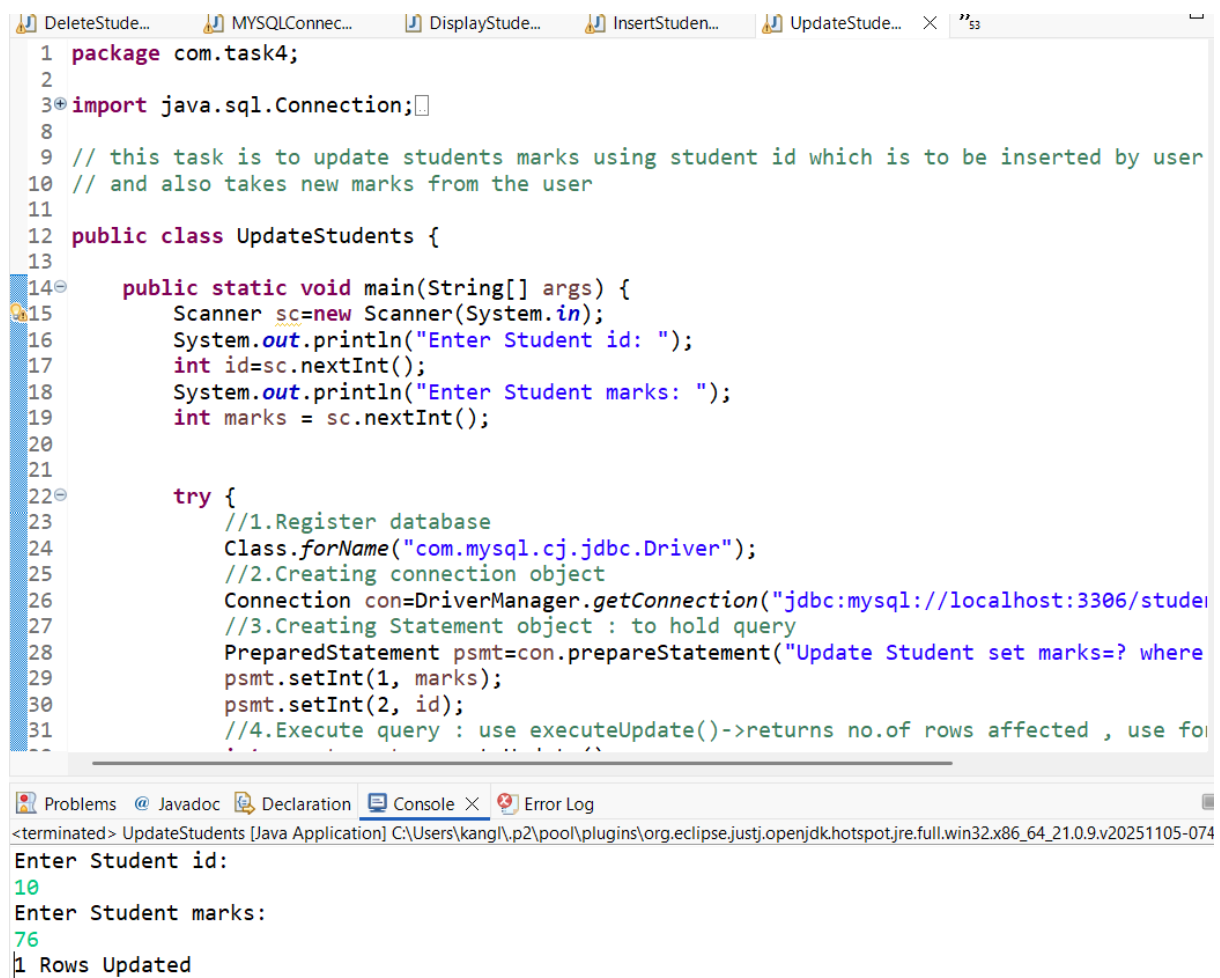
In this task , student marks are updated using Update query which is a parameterised query here that takes input student id at place of parameter from user and also takes new marks from the user.

1.Here , Firstly database driver is registered and Connection is established between java application and Database (MySQL).

2.Input id and marks are taken from user using Scanner Class object , that inputs are passed at the place of parameters to the query using setter methods called on PreparedStatement reference.

3.executeUpdate() method is used to execute query and returns number of rows affected (updated here) .

4.At last , Connections are closed for freeing up the resources by calling close() method.



```
1 package com.task4;
2
3 import java.sql.Connection;
4
5 // this task is to update students marks using student id which is to be inserted by user
6 // and also takes new marks from the user
7
8 public class UpdateStudents {
9
10     public static void main(String[] args) {
11         Scanner sc=new Scanner(System.in);
12         System.out.println("Enter Student id: ");
13         int id=sc.nextInt();
14         System.out.println("Enter Student marks: ");
15         int marks = sc.nextInt();
16
17         try {
18             //1.Register database
19             Class.forName("com.mysql.cj.jdbc.Driver");
20             //2.Creating connection object
21             Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/student");
22             //3.Creating Statement object : to hold query
23             PreparedStatement psmt=con.prepareStatement("Update Student set marks=? where id=?");
24             psmt.setInt(1, marks);
25             psmt.setInt(2, id);
26             //4.Execute query : use executeUpdate()->returns no.of rows affected , use for
27
28         } catch (Exception e) {
29             e.printStackTrace();
30         }
31     }
32 }
```

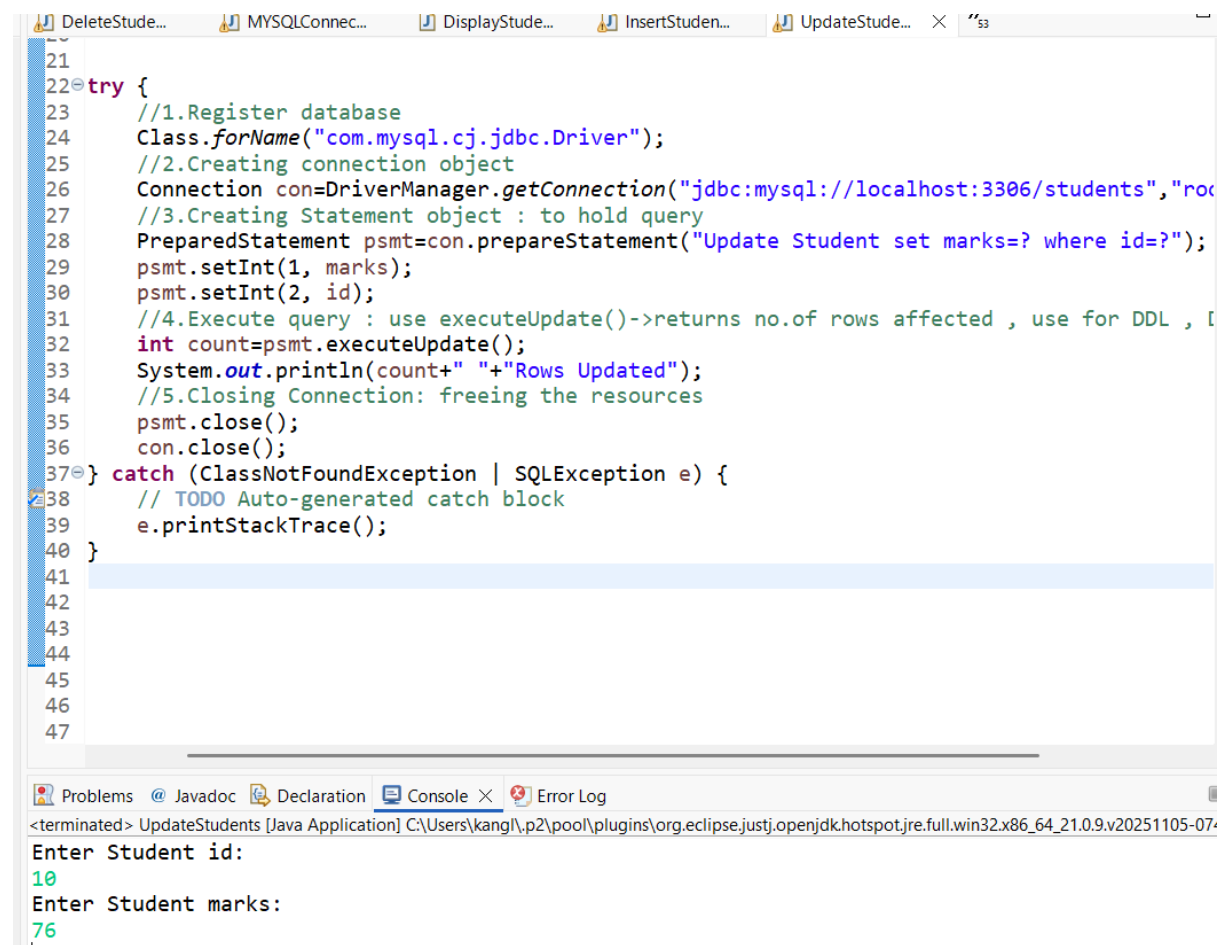
Problems @ Javadoc Declaration Console Error Log

<terminated> UpdateStudents [Java Application] C:\Users\kangl\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.9.v20251105-074

Enter Student id:
10
Enter Student marks:
76
1 Rows Updated

Name : Kalyani Kangle

Branch: CSE 3rd Year



```
21
22 try {
23     //1.Register database
24     Class.forName("com.mysql.cj.jdbc.Driver");
25     //2.Creating connection object
26     Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/students","root","root");
27     //3.Creating Statement object : to hold query
28     PreparedStatement psmt=con.prepareStatement("Update Student set marks=? where id=?");
29     psmt.setInt(1, marks);
30     psmt.setInt(2, id);
31     //4.Execute query : use executeUpdate()->returns no.of rows affected , use for DDL , [
32     int count=psmt.executeUpdate();
33     System.out.println(count+" "+"Rows Updated");
34     //5.Closing Connection: freeing the resources
35     psmt.close();
36     con.close();
37 } catch (ClassNotFoundException | SQLException e) {
38     // TODO Auto-generated catch block
39     e.printStackTrace();
40 }
41
42
43
44
45
46
47
```

Problems @ Javadoc Declaration Console X Error Log

<terminated> UpdateStudents [Java Application] C:\Users\kangl\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.9.v20251105-07-

Enter Student id:

10

Enter Student marks:

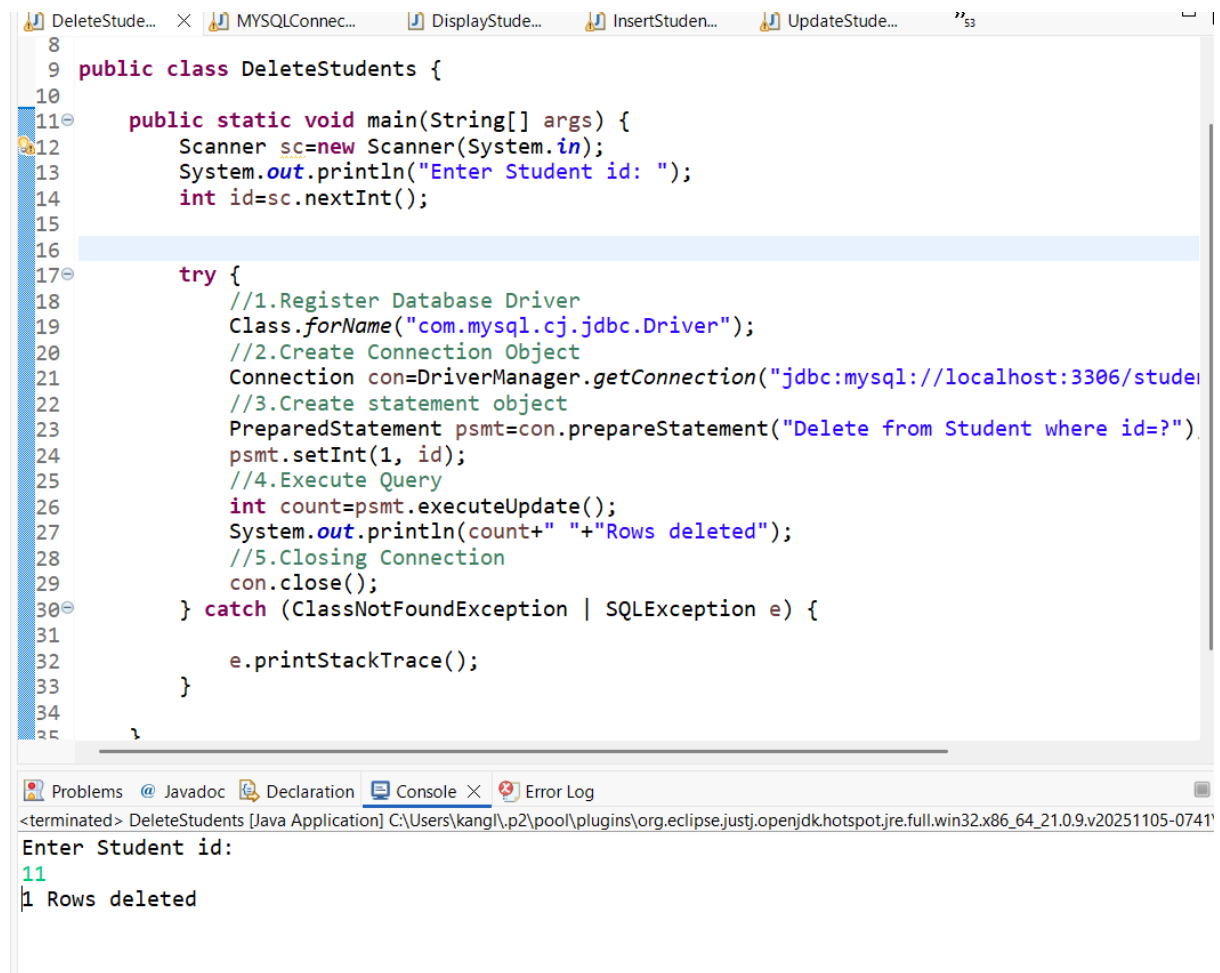
76

Name : Kalyani Kangle
Branch: CSE 3rd Year

Task 5: Delete Student Record

Description: In this task , id is taken from user as input , and the record corresponding to that input id is deleted.

- 1.Database Driver is registered and Connection is established between java application and database.
2. id is taken as input from user using Scanner class object and passed it to the Parameterized query using setter method called on Statement reference.
- 3.prepareStatement() is used to hold Delete query.
- 4.executeUpdate() is used to execute query and it returns number of rows deleted.
- 5.At last Connections are closed for freeing up the resources.



```
8
9 public class DeleteStudents {
10
11     public static void main(String[] args) {
12         Scanner sc=new Scanner(System.in);
13         System.out.println("Enter Student id: ");
14         int id=sc.nextInt();
15
16
17         try {
18             //1.Register Database Driver
19             Class.forName("com.mysql.cj.jdbc.Driver");
20             //2.Create Connection Object
21             Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/studei
22             //3.Create statement object
23             PreparedStatement psmt=con.prepareStatement("Delete from Student where id=?").
24             psmt.setInt(1, id);
25             //4.Execute Query
26             int count=psmt.executeUpdate();
27             System.out.println(count+" "+"Rows deleted");
28             //5.Closing Connection
29             con.close();
30         } catch (ClassNotFoundException | SQLException e) {
31
32             e.printStackTrace();
33         }
34
35     }
```

Problems @ Javadoc Declaration Console Error Log

<terminated> DeleteStudents [Java Application] C:\Users\kangl\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.9.v20251105-0741'

Enter Student id:

11

1 Rows deleted