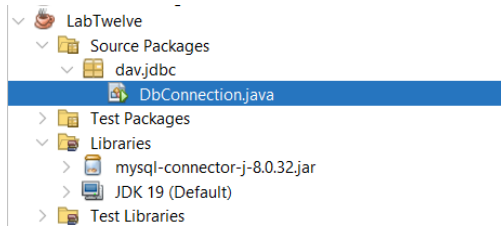


## Appendix:



```
1 package dav.jdbc;
2 import java.sql.*;
3 public class DbConnection {
4     public static final String DB_NAME = "lab_twelve";
5     public static final String DB_HOST = "localhost";
6     public static final String DB_USER = "root";
7     public static final String DB_PASSWORD = "";
8     public static final int PORT = 3306;
9     public static final String URL = "jdbc:mysql://" + DB_HOST + ":" + PORT + "/" + DB_NAME;
10    public static Connection con;
11    public static Statement stmt;
12
13    public static void doConnection() {
14        try {
15            Class.forName(className:"com.mysql.cj.jdbc.Driver");
16            con = DriverManager.getConnection(url: URL, user:DB_USER, password: DB_PASSWORD);
17            if (con != null) {
18                System.out.println(x: "Connection Success");
19            } else {
20                System.out.println(x: "Connection failed");
21            }
22        } catch (SQLException | ClassNotFoundException e) {
23            System.out.println("Error: " + e.getMessage());
24        }
25    }
26
27    public static void insertData(String first_name, String last_name, String username, String password) {
28        try {
29            String sql = "INSERT INTO lab_12_user(first_name,last_name,username,password) VALUES ('"+ first_name + "','"+
30                + last_name + "','"+ username + "','"+ password + "')";
31            stmt = con.createStatement();
32            int status = stmt.executeUpdate(string:sql);
33            if(status == 1){
34                System.out.println(x: "data inserted");
35            }else{
36                System.out.println(x: "Data not inserted");
37            }
38        } catch (SQLException e) {
39            System.out.println("Error: " + e.getMessage());
40        }
41    }
42
43    public static void updateData(String first_name, String last_name, String username, String password, int id) {
44        try {
45            String sql = "UPDATE lab_12_user SET first_name='"+first_name+"', last_name='"+last_name+"',
46                + " username='"+username+"', password='"+password+"' WHERE id='"+id+"'";
47            stmt = con.createStatement();
48            int status = stmt.executeUpdate(string:sql);
49            if(status == 1){
50                System.out.println(x: "data inserted");
51            }else{
52                System.out.println(x: "Data not inserted");
53            }
54        } catch (SQLException e) {
55            System.out.println("Error: " + e.getMessage());
56        }
57    }
58
59    public static void getUserList() {
60        try{
61            String sql="SELECT * FROM lab_12_user";
62            stmt=con.createStatement();
63            ResultSet rs = stmt.executeQuery(string:sql);
64            while(rs.next()){
65                System.out.println("First Name: "+rs.getString(string:"first_name"));
66                System.out.println("Last Name: "+rs.getString(string:"last_name"));
67                System.out.println("Username: "+rs.getString(string:"username"));
68                System.out.println("Password: "+rs.getString(string:"password"));
69            }
70        } catch (SQLException e) {
71            System.out.println("Error:"+e.getMessage());
72        }
73    }
74 }
```

```

72 public static void main(String[] args) {
73     doConnection();
74     insertData(first_name: "md", last_name: "khan", username: "noor", password: "noor123");
75     updateData(first_name: "Noorullah", last_name: "Khan", username: "meow", password: "meow123", id: 1);
76     getUserList();
77 }
78 }

```

Output:

doConnection():

Output - LabTwelve (run) x

```

run:
Connection Success
BUILD SUCCESSFUL (total time: 0 seconds)

```

insertData():

```

public static void main(String[] args) {
    doConnection();
    insertData(first_name: "md", last_name: "khan", username: "noor", password: "noor123");
    // updateData("Noorullah", "Khan", "meow", "meow123", 1);
    // getUserList();
}

```

Output - LabTwelve (run) x

```

run:
Connection Success
data inserted
BUILD SUCCESSFUL (total time: 0 seconds)

```

	id	first_name	last_name	username	password
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	md	khan	noor	noor123

updateData():



```

public static void main(String[] args) {
    doConnection();
    // insertData("md", "khan", "noor", "noor123");
    updateData(first_name: "Noorullah", last_name: "Khan", username: "meow", password: "meow123", id: 1);
    // getUserList();
}

```

Output - LabTwelve (run) x

```
run:
Connection Success
Data update successfully
BUILD SUCCESSFUL (total time: 0 seconds)
||
```

	id	first_name	last_name	username	password
<input type="checkbox"/> Edit  Copy  Delete	1	Noorullah	Khan	meow	meow123

getUserList():

```
public static void main(String[] args) {
    doConnection();
    // insertData("md", "khan", "noor", "noor123");
    // updateData("Noorullah", "Khan", "meow", "meow123", 1);
    getUserList();
}
```

Output - LabTwelve (run) x

```
run:
Connection Success
First Name: Noorullah
Last Name: Khan
Username: meow
Password: meow123
BUILD SUCCESSFUL (total time: 1 second)
||
```

## Appendix:

```
LabCaseStudyOne D:\BCA_3rdsem\java lab\LabCaseStudyOne
> .idea
> out
src
  casestudy.one.bmi
    BMICalculator
    User
  LabCaseStudyOne.iml
> External Libraries
> Scratches and Consoles
```

```
1 package casestudy.one.bmi;
2 import java.util.Scanner;
  2 usages
3 public class BMICalculator {
  9 usages
4     User user;
  1 usage
5     public BMICalculator() { this.user= new User(); }
  1 usage
8     public double calculateBMI(){
9         Scanner sc = new Scanner(System.in);
10        System.out.println("Enter your Weight in KG: ");
11        try{
12            this.user.setWeight(sc.nextDouble());
13        }catch (Exception e){
14            System.out.println("Please enter valid input");
15        }
16        System.out.println("Enter your height in Meter: ");
17        try{
18            this.user.setHeight(sc.nextDouble());
19        }
20        catch (Exception e){
21            System.out.println("Please enter valid input");
22        }
23        double bmi= this.user.getWeight()/Math.pow(this.user.getHeight(),2);
24        return bmi;
25    }
  2 usages
26    public String showIndex(double bmi){
27        String index=null;
28        if(bmi<0&&bmi>100){
29            index="please insert proper data: Height in meter and weight in kg";
30        }else if(bmi>0 && bmi<=16){
31            index = "You're severly underweight";
32        } else if (bmi>16 && bmi <=18.4){
33            index = "You are underweight";
34        } else if (bmi>18.4 && bmi<=24.9) {
35            index = "You are Normal";
36        }else if (bmi>24.9 && bmi<= 29.9){
37            index ="You are overweight";
38        }else if (bmi>29.9 && bmi<= 34.9){
39            index ="You are moderately obese";
40        }else if (bmi>34.9 && bmi<= 39.9){
41            index ="You are severely obese";
42        }else if (bmi>39.9){
43            index ="You are Morbidly obese";
44        }
45        return index;
46    }
  1 usage
47    @ public void calculate(String flag){
48        switch (flag.toLowerCase()){
49            case "us":
50                System.out.println("Your index: " + this.showIndex(this.calculateUsBmi()));
51                break;
52            case "metric":
53                System.out.println("Your index: " + this.showIndex(this.calculateBMI()));
54                break;
```

```

55         default:
56             System.out.println("Enter either 'us' or 'metric'");}}
57     1 usage
58     public double calculateUsBmi(){
59         Scanner sc = new Scanner(System.in);
60         System.out.println("Enter your Weight in LBS: ");
61         try{
62             this.user.setWeight(sc.nextDouble());
63         }catch (Exception e){
64             System.out.println("Please enter valid input");}
65         System.out.println("Enter your height in Inches: ");
66         try{
67             this.user.setHeight(sc.nextDouble());}
68         catch (Exception e){
69             System.out.println("Please enter valid input");}
70         double bmi= this.user.getWeight()/(Math.pow(this.user.getHeight(),2)*703);
71         return bmi;
72     }
73     no usages
74     public static void main(String[] args) {
75         BMICalculator s = new BMICalculator();
76         Scanner sc = new Scanner(System.in);
77         System.out.println("Enter the unit 'us' or 'metric' ");
78         s.calculate(sc.next());
79     }

```

#### User.java:

```

1  package casestudy.one.bmi;
2  import java.util.ArrayList;
3  2 usages
4  public class User{
5      2 usages
6      private String name;
7      2 usages
8      private String dob;
9      2 usages
10     private String blood_group;
11     2 usages
12     private String email;
13     2 usages
14     private int age;
15     2 usages
16     private double weight;
17     2 usages
18     private double height;
19     no usages
20     public String getName(){return this.name;}
21     no usages
22     public void setName(String name){this.name=name;}
23     no usages
24     public String getDob(){return this.dob;}
25     no usages
26     public void setDob(String dob){this.dob=dob;}
27     no usages
28     public String getBlood_group() {return this.blood_group;}
29     no usages
30     public void setBlood_group(String blood_group){this.blood_group=blood_group;}
31     no usages
32     public String getEmail() {return this.email;}
33     no usages
34     public void setEmail(String email){this.email=email;}
35     no usages
36     public int getAge(){return this.age;}
37     no usages
38     public void setAge(int age) {this.age = age;}
39     2 usages
40     public double getWeight(){return this.weight;}

```

```

22     2 usages
    public void setWeight(double weight) {this.weight = weight;}
23     2 usages
    public double getHeight() {return height;}
24     2 usages
    public void setHeight(double height) {this.height = height;}
25 }

```

Output:

Run: BMI Calculator ×

```

"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent
Enter the unit 'us' or 'metric'
metric
Enter your Weight in KG:
70
Enter your height in Meter:
1.79
Your index: You are Normal

```

Run: BMI Calculator ×

```

"C:\Program Files\Java\jdk-19\bin\java.exe" "-java.
Enter the unit 'us' or 'metric'
metric
Enter your Weight in KG:
80
Enter your height in Meter:
1.79
Your index: You are overweight

```