

CS23333-Object Oriented Programming Using Java-2023

Dashboard / My courses / CS23333-OOPJ-2023 / Lab-04-Classes and Objects / Lab-04-Logic Building

Quiz navigation



Show one page at a time

Finish review

Question **1**

Correct

Marked out of 5.00

Flag question

| | |
|-----------|------------------------------------|
| Status | Finished |
| Started | Saturday, 5 October 2024, 9:22 PM |
| Completed | Saturday, 5 October 2024, 10:12 PM |
| Duration | 49 mins 24 secs |

Create a class Student with two private attributes, name and roll number. Create three objects by invoking different constructors available in the class Student.

Student()

Student(String name)

Student(String name, int rollno)

Input:

No input

Output:

No-arg constructor is invoked

1 arg constructor is invoked

2 arg constructor is invoked

Name =null , Roll no = 0

Name =Rajalakshmi , Roll no = 0

Name =Lakshmi , Roll no = 101

For example:

| Test | Result |
|------|---|
| 1 | No-arg constructor is invoked 1 arg constructor is invoked 2 arg constructor is invoked Name =null , Roll no = 0 Name =Rajalakshmi , Roll no = 0 Name =Lakshmi , Roll no = 101 |

Answer: (penalty regime: 0 %)

```
1 public class Student
2 {
3     private String name;
4     private int rollno;
5     public Student()
6     {
7         this.name=null;
8         this.rollno=0;
9         System.out.println("No-arg constructor is invoked");
10    }
11    public Student(String name)
12    {
13        this.name=name;
14        this.rollno=0;
15        System.out.println("1 arg constructor is invoked");
16    }
17    public Student(String name,int rollno)
18    {
19        this.name=name;
20        this.rollno=rollno;
21        System.out.println("2 arg constructor is invoked");
22    }
23    public void display()
24    {
25        System.out.println("Name =" +this.name + " , Roll no = " +this.rollno);
26    }
27    public static void main(String args[])
28    {
29        Student s1=new Student();
30        Student s2=new Student("Rajalakshmi");
31        Student s3=new Student("Lakshmi",101);
32        s1.display();
33        s2.display();
34        s3.display();
35    }
36 }
```

| Test | Expected | Got |
|------|---|---|
| 1 | No-arg constructor is invoked 1 arg constructor is invoked 2 arg constructor is invoked Name =null , Roll no = 0 Name =Rajalakshmi , Roll no = 0 Name =Lakshmi , Roll no = 101 | No-arg constructor is invoked 1 arg constructor is invoked 2 arg constructor is invoked Name =null , Roll no = 0 Name =Rajalakshmi , Roll no = 0 Name =Lakshmi , Roll no = 101 |

Passed all tests!

Question **2**

Correct

Marked out of 5.00

Flag question

Create a Class Mobile with the attributes listed below,

```
private String manufacturer;
private String operating_system;
public String color;
private int cost;
```

Define a Parameterized constructor to initialize the above instance variables.

Define getter and setter methods for the attributes above.

for example : setter method for manufacturer is

```
void setManufacturer(String manufacturer){
    this.manufacturer= manufacturer;
}
```

```
String getManufacturer(){
    return manufacturer;
}
```

Display the object details by overriding the toString() method.

For example:

| Test | Result |
|------|--|
| 1 | <pre> manufacturer = Redmi operating_system = Andriod color = Blue cost = 34000 </pre> |

Answer: (penalty regime: 0 %)

```

1 public class Mobile
2 {
3     private String manufacturer;
4     private String operating_system;
5     public String color;
6     private int cost;
7
8     public Mobile(String manufacturer,String operating_system,String color,int cost)
9     {
10         this.manufacturer=manufacturer;
11         this.operating_system=operating_system;
12         this.color=color;
13         this.cost=cost;
14     }
15     public void setManufacturer(String manufacturer)
16     {
17         this.manufacturer=manufacturer;
18     }
19     public String getManufacturer()
20     {
21         return manufacturer;
22     }
23     public void setOperating_system(String operating_system)
24     {
25         this.operating_system=operating_system;
26     }
27     public String getOperating_system()
28     {
29         return operating_system;
30     }
31     public void setColor(String color)
32     {
33         this.color=color;
34     }
35     public String getColor()
36     {
37         return color;
38     }
39     public void setCost(int cost)
40     {
41         this.cost=cost;
42     }
43     public int getCost()
44     {
45         return cost;
46     }
47     @Override
48     public String toString()
49     {
50         return "manufacturer = "+manufacturer+"\noperating_system = "+operating_system+"\ncolor = "+color+"\ncost = "+cost;
51     }
52     public static void main(String args[])
53     {
54         Mobile m = new Mobile("Redmi","Andriod","Blue",34000);
55         System.out.println(m.toString());
56     }
57
58 }

```

| Test | Expected | Got |
|------|--|--|
| 1 | <pre> manufacturer = Redmi operating_system = Andriod color = Blue cost = 34000 </pre> | <pre> manufacturer = Redmi operating_system = Andriod color = Blue cost = 34000 </pre> |

Passed all tests!

Question 3

Correct

Marked out of 5.00

[Flag question](#)

Create a class called "Circle" with a radius attribute. You can access and modify this attribute using getter and setter methods. Calculate the area and circumference of the circle.

Area of Circle = πr^2

Circumference = $2\pi r$

Input:

2

Output:

Area = 12.57

Circumference = 12.57

For example:

| Test | Input | Result |
|------|-------|---|
| 1 | 4 | <pre> Area = 50.27 Circumference = 25.13 </pre> |

Answer: (penalty regime: 0 %)

Reset answer

```

1 import java.io.*;
2 import java.util.*;
3 class Circle
4 {
5     private double radius;
6     public Circle(double radius){
7         // set the instance variable radius
8         this.radius=radius;
9
10    }
11
12    public void setRadius(double radius){
13        // set the radius
14        this.radius=radius;
15    }
16
17    public double getRadius() {
18        // return the radius
19        return radius;
20    }
21
22    public double calculateArea() { // complete the below statement
23        return (Math.PI)*radius*radius;

```

```

24     }
25 }
26 public double calculateCircumference() {
27     // complete the statement
28     return 2*(Math.PI)*radius;
29 }
30 }
31 class prog{
32     public static void main(String[] args) {
33         int r;
34         Scanner sc= new Scanner(System.in);
35         r=sc.nextInt();
36         Circle c= new Circle(r);
37         System.out.println("Area = "+String.format("%.2f", c.calculateArea()));
38         // invoke the calculateCircumference method
39         System.out.println("Circumference = "+String.format("%.2f", c.calculateCircumference()));
40         sc.close();
41     }
42 }
43 }
44

```

| Test | Input | Expected | Got |
|------|-------|--|--|
| 1 | 4 | Area = 50.27 Circumference = 25.13 | Area = 50.27 Circumference = 25.13 |
| 2 | 6 | Area = 113.10 Circumference = 37.70 | Area = 113.10 Circumference = 37.70 |
| 3 | 2 | Area = 12.57 Circumference = 12.57 | Area = 12.57 Circumference = 12.57 |

Passed all tests!

[Finish review](#)

[◀ Lab-04-MCQ](#)

Jump to...

[Number of Primes in a specified range ▶](#)