Status	Finished
Started	Thursday, 3 October 2024, 10:59 PM
Completed	Thursday, 3 October 2024, 11:00 PM
Duration	1 min 21 secs

Question **1**

Correct

Marked out of 5.00

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 = πr²

Circumference = 2πr

Input:

2

Output:

Area = 12.57
Circumference = 12.57

For example:

Test	Input	Result
1	4	Area = 50.27 Circumference = 25.13

Answer: (penalty regime: 0 %)

Reset answer

```java
import java.util.Scanner;
class Circle
{
    private double radius;
    public Circle(double radius){
        this.radius=radius;


    }
    public void setRadius(double radius){
        this.radius=radius;


    }
    public double getRadius()    {
        return radius;


    }
    public double calculateArea()  { // complete the below statement
       return Math.PI*radius*radius;

    }
    public double calculateCircumference()    {
       return 2*Math.PI*radius;
    }
}
public class Main{
    public static void main(String[] args)  {
        Scanner sc= new Scanner(System.in);
        double r=sc.nextDouble();
        Circle circle= new Circle(r);
        System.out.printf("Area = %.2f\n" ,circle.calculateArea());
        System.out.printf("Circumference = %.2f" ,circle.calculateCircumference());
        sc.close();

    }
}
```

	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! ✓

	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	✓

Question **2**

Correct

Marked out of 5.00

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	manufacturer = Redmi operating_system = Andriod color = Blue cost = 34000

Answer: (penalty regime: 0 %)

```
 1  public class Mobile{
 2      private String manufacturer;
 3      private String operatingSystem;
 4      private int cost;
 5      private String color;
 6      public Mobile(String manufacturer,String operatingSystem,int cost,String color){
 7          this.manufacturer=manufacturer;
 8          this.operatingSystem=operatingSystem;
 9          this.cost=cost;
10          this.color=color;
11      }
12      @Override
13      public String toString(){
14          return "manufacturer = " + manufacturer+ "\noperating_system = " + operatingSystem + "\ncolor = " + color
15
16      }
17      public static void main(String[] args){
18          Mobile mobile=new Mobile("Redmi","Andriod",34000,"Blue");
19          System.out.println(mobile.toString());
20      }
21
22
23
```

	Test	Expected	Got	
✓	1	manufacturer = Redmi operating_system = Andriod color = Blue cost = 34000	manufacturer = Redmi operating_system = Andriod color = Blue cost = 34000	✓

Passed all tests! ✓

Question **3**

Correct

Marked out of 5.00

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 %)

```java
1  public class Student{
2      private String name;
3      private int rollNo;
4      public Student(){
5          this.name=null;
6          this.rollNo=0;
7          System.out.println("No-arg constructor is invoked");
8      }
9      public Student(String name){
10         this.name=name;
11         this.rollNo=0;
12         System.out.println("1 arg constructor is invoked");
13     }
14     public Student(String name,int rollNo){
15         this.name=name;
16         this.rollNo=rollNo;
17         System.out.println("2 arg constructor is invoked");
18     }
19     public void displayInfo(){
20         System.out.println("Name ="+ this.name + " , Roll no = " +this.rollNo);
21
22     }
23     public static void main(String[] args){
24         Student student1=new Student();
25         Student student2=new Student("Rajalakshmi");
26         Student student3=new Student("Lakshmi",101);
27         student1.displayInfo();
28         student2.displayInfo();
29         student3.displayInfo();
30     }
31  }
```

	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! ✓

◄ Lab-04-MCQ

Jump to...

Number of Primes in a specified range ►