CS23333-Object Oriented Programming Using Java-2023

Quiz navigation

1 2 3

Show one page at a time

Finish review

Status	Finished
Started	Saturday, 5 October 2024, 1:02 PM
Completed	Saturday, 5 October 2024, 1:23 PM
Duration	20 mins 36 secs

Question **1**

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 = 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;

    // Constructor to set the instance variable radius
    public Circle(double radius) {
        this.radius = radius;
    }

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

    // Getter for radius
    public double getRadius() {
        return radius;
    }

    // Method to calculate the area of the circle
    public double calculateArea() {
        return Math.PI * radius * radius; // Area = πr²
    }

    // Method to calculate the circumference of the circle
    public double calculateCircumference() {
        return 2 * Math.PI * radius; // Circumference = 2πr
    }
}

class prog {
    public static void main(String[] args) {
        int r;
        Scanner sc = new Scanner(System.in);
        r = sc.nextInt(); // Read the radius input
        Circle c = new Circle(r); // Create a Circle object with the radius

        // Calculate and display the area
        System.out.println("Area = " + String.format("%.2f", c.calculateArea()));

        // Invoke the calculateCircumference method and display the result
        System.out.println("Circumference = " + String.format("%.2f", c.calculateCircumference()));

        sc.close(); // Close the scanner
    }
}

```

	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!

Question **2**

Correct

Marked out of 5.00

Flag question

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
public class Student {
    private String name;
    private int rollNumber;

    // No-argument constructor
    public Student() {
        System.out.println("No-arg constructor is invoked");
        this.name = null;
        this.rollNumber = 0;
    }

    // One-argument constructor
    public Student(String name) {
        System.out.println("1 arg constructor is invoked");
        this.name = name;
        this.rollNumber = 0;
    }

    // Two-argument constructor
    public Student(String name, int rollNumber) {
        System.out.println("2 arg constructor is invoked");
        this.name = name;
        this.rollNumber = rollNumber;
    }

    // Method to display student details
    public void display() {
        System.out.println("Name =" + name + " , Roll no = " + rollNumber);
    }

    public static void main(String[] args) {
        // Creating objects using different constructors
        Student student1 = new Student(); // No-arg constructor
        Student student2 = new Student("Rajalakshmi"); // One-arg constructor
        Student student3 = new Student("Lakshmi", 101); // Two-arg constructor

        // Displaying student details
        student1.display();
        student2.display();
        student3.display();
    }
}
```

	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 **3**

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

Answer: (penalty regime: 0 %)

```java
public class Mobile {
    private String manufacturer;
    private String operating_system;
    public String color;
    private int cost;

    // Parameterized constructor
    public Mobile(String manufacturer, String operating_system, String color, int cost) {
        this.manufacturer = manufacturer;
        this.operating_system = operating_system;
        this.color = color;
        this.cost = cost;
    }

    // Setter for manufacturer
    public void setManufacturer(String manufacturer) {
        this.manufacturer = manufacturer;
    }

    // Getter for manufacturer
    public String getManufacturer() {
        return manufacturer;
    }

    // Setter for operating_system
    public void setOperatingSystem(String operating_system) {
        this.operating_system = operating_system;
    }

    // Getter for operating_system
    public String getOperatingSystem() {
        return operating_system;
    }

    // Setter for color
    public void setColor(String color) {
        this.color = color;
    }

    // Getter for color
    public String getColor() {
        return color;
    }

    // Setter for cost
    public void setCost(int cost) {
        this.cost = cost;
    }

    // Getter for cost
    public int getCost() {
        return cost;
```

	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!

Finish review

◄ Lab-04-MCQ

Jump to...

Number of Primes in a specified range ►