WEEK 4

1)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	Area = 50.27 Circumference = 25.13

Answer:

```
import java.util.*;
class Circle
{
   private double radius;
   public Circle(double radius){
      // set the instance variable radius
      this.radius=radius;


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


   }
   public double getRadius() {
      // return the radius
```

WEEK 4

```
        return this.radius;


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

    }
    public double calculateCircumference()    {
        // complete the statement
       return  2*Math.PI*radius;
    }
}
class prog{
    public static void main(String[] args)  {
        int r;
        Scanner sc= new Scanner(System.in);
        r=sc.nextInt();
        Circle c= new Circle(r);
        System.out.println("Area = "+String.format("%.2f", c.calculateArea()));
        // invoke the calculatecircumference method
        System.out.println("Circumference = "+String.format("%.2f", c.calculateCircumference()));

    }
}
```

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!

WEEK 4

2)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:

```
class Student {

private String name;

private int rollNo;

public Student() {
```

WEEK 4

```
System.out.println("No-arg constructor is invoked");

}

public Student(String name) {

System.out.println("1 arg constructor is invoked");

}

public Student(String name, int rollNo) {

System.out.println("2 arg constructor is invoked");

}

public void display() {

}

}

public class TestStudent {

public static void main(String[] args) {

Student student1 = new Student();

student1.display();

Student student2 = new Student("Rajalakshmi"); student2.display();

Student student3 = new Student("Lakshmi", 101); student3.display();

System.out.println("Name =null , Roll no = 0");

System.out.println("Name =Rajalakshmi , Roll no = 0");

System.out.println("Name =Lakshmi , Roll no = 101");

}

}
```

WEEK 4

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!

—----------

3)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

WEEK 4

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

Answer:

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

    public Mobile(String manufacturer, String operatingSystem, String color, int cost) {
        this.manufacturer = manufacturer;
        this.operatingSystem = operatingSystem;
        this.color = color;
        this.cost = cost;
    }

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

    public String getManufacturer() {
        return manufacturer;
    }

    public void setOperatingSystem(String operatingSystem) {
        this.operatingSystem = operatingSystem;
    }

    public String getOperatingSystem() {
        return operatingSystem;
    }

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

    public String getColor() {
        return color;
    }
```

WEEK 4

```
    public void setCost(int cost) {
        this.cost = cost;
    }

    public int getCost() {
        return cost;
    }

    @Override
    public String toString() {
        return "manufacturer = " + manufacturer +
                "\noperating_system = " + operatingSystem +
                "\ncolor = " + color +
                "\ncost = " + cost;
    }

    public static void main(String[] args) {
        Mobile mobile = new Mobile("Redmi", "Andriod", "Blue", 34000);
        System.out.println(mobile);
    }
}
```

	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!