Question 3:

**package** guviTask2;

**import** java.util.Scanner;

**public** **class** Circle {

**private** **double** radius;

// No-argument constructor that sets the radius to a default value, e.g., 1.0

**public** Circle() {

**this**.radius = 1.0;

}

// Constructor with one argument to set the radius

**public** Circle(**double** radius) {

**this**.radius = radius;

}

// Constructor with two arguments (we'll use the average of the two as the radius)

**public** Circle(**double** radius1, **double** radius2) {

**this**.radius = (radius1 + radius2) / 2; // Average of two radii

}

// Method to calculate the circumference

**public** **double** calculateCircumference() {

**return** 2 \* Math.***PI*** \* radius;

}

// Getter for radius

**public** **double** getRadius() {

**return** radius;

}

**public** **static** **void** main(String[] args) {

Scanner scanner = **new** Scanner(System.***in***);

// Prompt the user to choose a constructor

System.***out***.println("Choose a constructor:");

System.***out***.println("1. No-argument constructor (default radius)");

System.***out***.println("2. One-argument constructor (enter radius)");

System.***out***.println("3. Two-argument constructor (enter two radii to average)");

**int** choice = scanner.nextInt();

Circle circle;

**switch** (choice) {

**case** 1:

// Use no-argument constructor

circle = **new** Circle();

**break**;

**case** 2:

// Use single-argument constructor

System.***out***.print("Enter the radius: ");

**double** radius = scanner.nextDouble();

circle = **new** Circle(radius);

**break**;

**case** 3:

// Use two-argument constructor

System.***out***.print("Enter the first radius: ");

**double** radius1 = scanner.nextDouble();

System.***out***.print("Enter the second radius: ");

**double** radius2 = scanner.nextDouble();

circle = **new** Circle(radius1, radius2);

**break**;

**default**:

System.***out***.println("Invalid choice! Using default constructor.");

circle = **new** Circle();

}

// Display the radius and circumference

System.***out***.println("Radius: " + circle.getRadius());

System.***out***.println("Circumference: " + circle.calculateCircumference());

scanner.close();

}}

Output:

|  |
| --- |
| Choose a constructor:  1. No-argument constructor (default radius)  2. One-argument constructor (enter radius)  3. Two-argument constructor (enter two radii to average)  2  Enter the radius: 5.0  Radius: 5.0  Circumference: 31.41592653589793 |