Lab worksheet 1: Introduction

1. Write a program that accepts an odd-length word and prints out the middle character. For example, if the input is magnificent, which has 11 characters, you output the sixth character f.

```java
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        // Create a Scanner object to read input from the console.
        Scanner scanner = new Scanner(System.in);
        // Prompt the user to enter a word.
        System.out.print("Enter a word: ");
        String word = scanner.nextLine();
        // Get the length of the word.
        int length = word.length();
        // Get the middle character of the word.
        char middleCharacter = word.charAt(length / 2);
        // Print the middle character.
        System.out.println("The middle character is " + middleCharacter);
    }
}
```

2. Write a program that asks the user for her or his full name in the format **first middle last** and replies with the name in the format **last, first middle-initial**. where the last name is followed by a comma and the middle initial is followed by a period. For example, if the input is **Antony Edward Stark** then the output is **Stark, Antony E**.

```java
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        // Create a Scanner object to read input from the user.
        Scanner scanner = new Scanner(System.in);
        // Prompt the user to enter their full name.
        System.out.print("Enter your full name (first middle last): ");
        // Read the user's full name into a string.
        String fullName = scanner.nextLine();
        // Find the first and last spaces in the full name.
        int firstSpaceIndex = fullName.indexOf(' ');
        int lastSpaceIndex = fullName.lastIndexOf(' ');
        // Extract the first, middle, and last name from the full name.
        String firstName = fullName.substring(0, firstSpaceIndex);
```

```
        String middleName = fullName.substring(firstSpaceIndex + 1,
        lastSpaceIndex);
        String lastName = fullName.substring(lastSpaceIndex + 1);
        // Extract the middle initial from the middle name.
        String middleInitial = middleName.charAt(0) + ".";
        // Format the name into last, first, middle initial format.
        String formattedName = lastName + ", " + firstName + " " +
        middleInitial;
        // Print the formatted name.
        System.out.println("Formatted name: " + formattedName);
    }
}
```

3. Write a Java program to convert centimetres (input) to feet and inches (output). (**1 inch 2.54 cm**)

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        // Create a Scanner object to read input from the user.
        Scanner scanner = new Scanner(System.in);
        // Print a prompt to the user.
        System.out.print("Enter the length in centimetres: ");
        // Read the length in centimetres from the user.
        int centimetres = scanner.nextInt();
        // Convert the centimetres to inches.
        double inches = centimetres / 2.54;
        // Get the number of feet.
        int feet = (int) inches / 12;
        // Get the number of inches that are left over after dividing by 12.
        inches %= 12;
        // Print the result.
        System.out.println("Length = " + feet + " feet and " + inches + "
    inches.");
    }
}
```

Lab worksheet 1: Introduction

4. Write a Java program that displays a frame window 300 pixels wide and 200 pixels high with the title **My First Frame**. Place the frame so that its top left corner is at a position 50 pixels from the top of the screen and 100 pixels from the left of the screen.

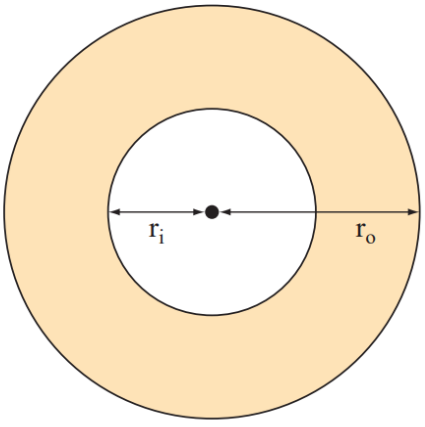
    - To position a window at a specified location, you can use the **setLocation** method like this, `frame.setLocation( 50, 50 );`
    - Through experimentation, determine how the two arguments in the setLocation method affect the positioning of the window.

```
import javax.swing.*;

class Main {

   public static void main(String[] args) {
       // Create a JFrame object
       JFrame myWindow = new JFrame();
       // Set the frame's size
       myWindow.setSize(300, 200);
       // Set the frame's location
       myWindow.setLocation(100, 50);
       // Add a label to the frame
       myWindow.setTitle("My First Frame");
       // Display the frame
       myWindow.setVisible(true);
   }
}
```

5. Write a Java program that computes the area of a circular region (the shaded area in the diagram), given the radii of the inner and the outer circles, ri and ro, respectively. We compute the area of the circular region by subtracting the area of the inner circle from the area of the outer circle. Define a **Circle** class that has methods **computeArea** and **computeCircumference** to compute the area and circumference. You set the circle's radius with the **setRadius** method or via a constructor.



Lab worksheet 1: Introduction

File: Circle.java

```
public class Circle {
   private double radius;

   public Circle() {
       this.radius = 0;
   }
   public Circle(double radius) {
       this.radius = radius;
   }
   public double getRadius() {
       return radius;
   }
   public void setRadius(double radius) {
       this.radius = radius;
   }
   public double computeArea() {
       return Math.PI * radius * radius;
   }
   public double computeCircumference() {
       return 2 * Math.PI * radius;
   }
}
```

File: Main.java

```
import java.util.Scanner;

public class Main {

   public static void main(String[] args) {
       Scanner scanner = new Scanner(System.in);
       System.out.print("Enter the radius of the inner circle: ");
       double ri = scanner.nextDouble();
       System.out.print("Enter the radius of the outer circle: ");
       double ro = scanner.nextDouble();
       Circle innerCircle = new Circle(ri);
       Circle outerCircle = new Circle(ro);
       double area =  outerCircle.computeArea() -
innerCircle.computeArea();
       System.out.println("The area of the circular region is " +
area);
   }
}
```