

DEEBAK KUMAR K
192324064
ASSIGNMENT - 7

Question 1:

Write a program called SumProductMinMax3 that prompts user for three integers. The program shall read the inputs as int; compute the sum, product, minimum and maximum of the three integers; and print the results. For examples,

Enter 1st integer: 8
Enter 2nd integer: 2
Enter 3rd integer: 9
The sum is: 19
The product is: 144
The min is: 2
The max is: 9

Code :

```
import java.util.Scanner;

public class SumProductMinMax3 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Prompt user for three integers
        System.out.print("Enter 1st integer: ");
        int num1 = scanner.nextInt();

        System.out.print("Enter 2nd integer: ");
        int num2 = scanner.nextInt();

        System.out.print("Enter 3rd integer: ");
        int num3 = scanner.nextInt();

        // Compute the sum, product, minimum and maximum
        int sum = num1 + num2 + num3;
        int product = num1 * num2 * num3;
        int min = Math.min(num1, Math.min(num2, num3));
        int max = Math.max(num1, Math.max(num2, num3));
```

```

        // Print the results
        System.out.println("The sum is: " + sum);
        System.out.println("The product is: " + product);
        System.out.println("The min is: " + min);
        System.out.println("The max is: " + max);

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

```

The screenshot shows a Java IDE with a file named 'Main.java'. The code defines a class 'SumProductMinMax3' with a 'main' method. It uses a 'Scanner' to take three integers as input (2, 4, 6). It then calculates the sum (12), product (48), minimum (2), and maximum (6). The output window on the right shows the execution results, confirming the calculations and stating 'Code Execution Successful'.

```

Main.java
1- import java.util.Scanner;
2
3- public class SumProductMinMax3 {
4-     public static void main(String[] args) {
5-         Scanner scanner = new Scanner(System.in);
6
7-         // Prompt user for three integers
8-         System.out.print("Enter 1st integer: ");
9-         int num1 = scanner.nextInt();
10
11        System.out.print("Enter 2nd integer: ");
12        int num2 = scanner.nextInt();
13
14        System.out.print("Enter 3rd integer: ");
15        int num3 = scanner.nextInt();
16
17        // Compute the sum, product, minimum and maximum
18        int sum = num1 + num2 + num3;
19        int product = num1 * num2 * num3;
20        int min = Math.min(num1, Math.min(num2, num3));
21        int max = Math.max(num1, Math.max(num2, num3));
22
23        // Print the results
24        System.out.println("The sum is: " + sum);
25        System.out.println("The product is: " + product);
26        System.out.println("The min is: " + min);
27        System.out.println("The max is: " + max);
28
Output
java -cp /tmp/CYZzbVF1ca/SumProductMinMax3
Enter 1st integer: 2
Enter 2nd integer: 4
Enter 3rd integer: 6
The sum is: 12
The product is: 48
The min is: 2
The max is: 6

=== Code Execution Successful ===

```

Question 2 :

Calculate BMI Using Java

The user enters his height (in inches) and weight (in pounds). The variables passed by the user are assigned to the float type. After calculating the BMI value, the value will be assigned to the appropriate range and the correct message will appear on the console. You can use the if-else-if ladder for printing the message on the console.

Intervals of BMI index:

- 16.00 or less = starvation
- 16.00-16.99 = emaciation
- 17.00-18.49 = underweight
- 18.50-22.99 = normal, low range
- 23.00-24.99 = normal high range

25.00-27.49 = overweight low range
27.50-29.99 = overweight high range
30.00-34.99 = 1st degree obesity
35.00-39.99 = 2nd degree obesity
40.00 or above = 3rd degree obesity

Code:

```
import java.util.Scanner;

public class BMICalculator {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Prompt user for height in inches
        System.out.print("Enter your height (in inches): ");
        float heightInches = scanner.nextFloat();

        // Prompt user for weight in pounds
        System.out.print("Enter your weight (in pounds): ");
        float weightPounds = scanner.nextFloat();

        // Convert height to meters and weight to kilograms
        float heightMeters = heightInches * 0.0254f;
        float weightKilograms = weightPounds * 0.453592f;

        // Calculate BMI
        float bmi = weightKilograms / (heightMeters * heightMeters);

        // Determine BMI category
        String bmiCategory;
        if (bmi <= 16.00) {
            bmiCategory = "starvation";
        } else if (bmi <= 16.99) {
            bmiCategory = "emaciation";
        } else if (bmi <= 18.49) {
            bmiCategory = "underweight";
        } else if (bmi <= 22.99) {
            bmiCategory = "normal, low range";
        } else if (bmi <= 24.99) {
            bmiCategory = "normal, high range";
        } else if (bmi <= 27.49) {
            bmiCategory = "overweight, low range";
        }
    }
}
```

```

    } else if (bmi <= 29.99) {
        bmiCategory = "overweight, high range";
    } else if (bmi <= 34.99) {
        bmiCategory = "1st degree obesity";
    } else if (bmi <= 39.99) {
        bmiCategory = "2nd degree obesity";
    } else {
        bmiCategory = "3rd degree obesity";
    }

    // Print the BMI value and category
    System.out.printf("Your BMI is: %.2f\n", bmi);
    System.out.println("You are classified as: " + bmiCategory);

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

```

The screenshot shows a Java IDE with a file named 'Main.java'. The code in the file is a BMI calculator that takes height in inches and weight in pounds as input, converts them to meters and kilograms, calculates the BMI, and then determines the BMI category based on a series of if-else statements. The output window on the right shows the program's execution with the following text:

```

java -cp /tmp/IziNwsipqG/BMICALculator
Enter your height (in inches): 70
Enter your weight (in pounds): 150
Your BMI is: 21.52
You are classified as: normal, low range
=== Code Execution Successful ===

```

Question 3:

Write a program that will use the while loop to find the largest and smallest number from the set of 10 randomly drawn integers from 1 to 100. In this task, do not use arrays or other collections.

Code:

```
import java.util.Random;
```

```

public class FindLargestAndSmallest {
    public static void main(String[] args) {
        Random random = new Random();

        // Initialize variables to store the largest and smallest numbers
        int largest = Integer.MIN_VALUE;
        int smallest = Integer.MAX_VALUE;

        int count = 0; // Counter for the number of random integers

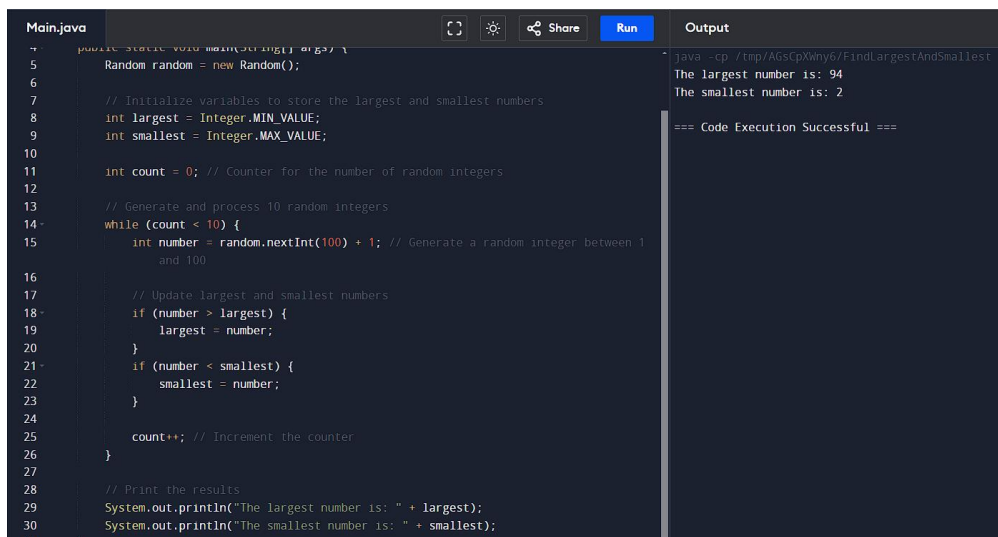
        // Generate and process 10 random integers
        while (count < 10) {
            int number = random.nextInt(100) + 1; // Generate a
            random integer between 1 and 100

            // Update largest and smallest numbers
            if (number > largest) {
                largest = number;
            }
            if (number < smallest) {
                smallest = number;
            }

            count++; // Increment the counter
        }

        // Print the results
        System.out.println("The largest number is: " + largest);
        System.out.println("The smallest number is: " + smallest);
    }
}

```



The screenshot shows a Java IDE with a dark theme. On the left, the code for `Main.java` is displayed, matching the code in the previous block. On the right, the `Output` panel shows the command `java -cp ./tmp/AGsCpXWny6/FindLargestAndSmallest` and the output: `The largest number is: 94` and `The smallest number is: 2`. Below the output, it says `=== Code Execution Successful ===`.