NATIONAL UNIVERSITY
551 M.F. JHOCSON ST., SAMPALOC, MANILA

COMPUTER ENGINEERING
CPOOPG2L: Object Oriented Programming

NATIONAL UNIVERSITY
COLLEGE OF ENGINEERING

Name/Section	Martin, Seth Marcus S. COE21	Rating
Date performed	DECEMBER 12, 2022	
Date submitted		
Professor	ENG. MARLON BAGARA	

ACTIVITY NO. 1
INPUT/OUTPUT STATEMENT

I. OBJECTIVES:

The activity aims to:

1. Implement the System.out and the System.in objects to generate output and input.
2. Implement the statement of using dialog box

LEARNING OUTCOMES (LOs) *At the end of the activity, the students should be able to:*	COURSE LEARNING OUTCOMES (CLOs)				
	1	2	3	4	5
1. Write a program to accept and display output data to the console.	•				
2. Write a program to accept and display output data using dialog box		•			

COURSE LEARNING OUTCOMES (CLOs)

1. Understand the fundamental concept of OOP through Java programming
2. Write programs using console and dialog box.
3. Apply the concept of iterative, control, and array structure programming..
4. Construct classes, objects, methods and constructor.
5. Write programs in GUI environment

II. SOFTWARE/HARDWARE/EQUIPMENT NEEDED:

1. Eclipse
2. Computer unit

III. SAFETY GUIDELINES:

1. Make sure you have both an adjustable table and chair so that ergonomic accommodations can be made for each person using the computer.
2. The computer screen should be front and centre so neck turning is unnecessary.
3. Keep your lab space clean and organized.
4. Clean your lab bench and equipment, and lock the door before you leave the laboratory.

5. Never eat, drink, or smoke while working in the laboratory.
6. DO NOT TOUCH ANYTHING WITH WHICH YOU ARE NOT COMPLETELY FAMILIAR!!! It is always better to ask questions to laboratory technicians or to your instructors than to risk harm to yourself or damage to the equipment.

IV. THEORY

Java provides a static object called **System.out** which performs output to a standard output device. Most OS allows users to redirect standard output to files or as an input to other files, but the default output is the Java console window. To display data on the output device, the given syntax must be followed:

System.in.print(identifier);

Example:

1. Sytem.out.print(area);
2. System.out.print("the area is: " + area);

The **System,in** is an object intended for performing input from the Java console window. The input is actually coming from the standard input device. The **System.in** object is associated with the standard input device. A simple way of reading input with this object is to create a Scanner object using the expression: **new Scanner(System.in).** For an instance, to input data into the variable x, the given statement must be considered:

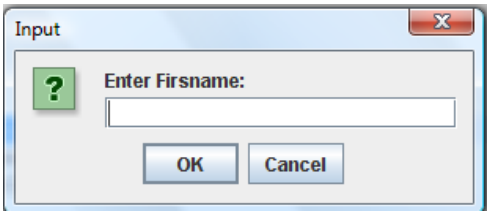
```
Scanner input=new Scanner(System.in);
int x=input.nextInt();
```

Another way of inputting and displaying data is through dialog box. A dialog box is a simple GUI based for input and output. The dialog box can be implemented by using the JOptionPane class

Example:

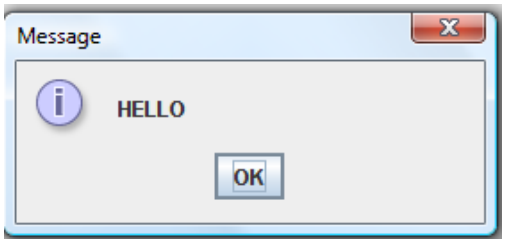
Statement for inputting data:

```
Fname=JOptionPane.showInputDialog("Enter Firsname:");
```



Statement for displaying data

```
JOptionPane.showMessageDialog(null,"HELLO");
```



V. PROCEDURE:

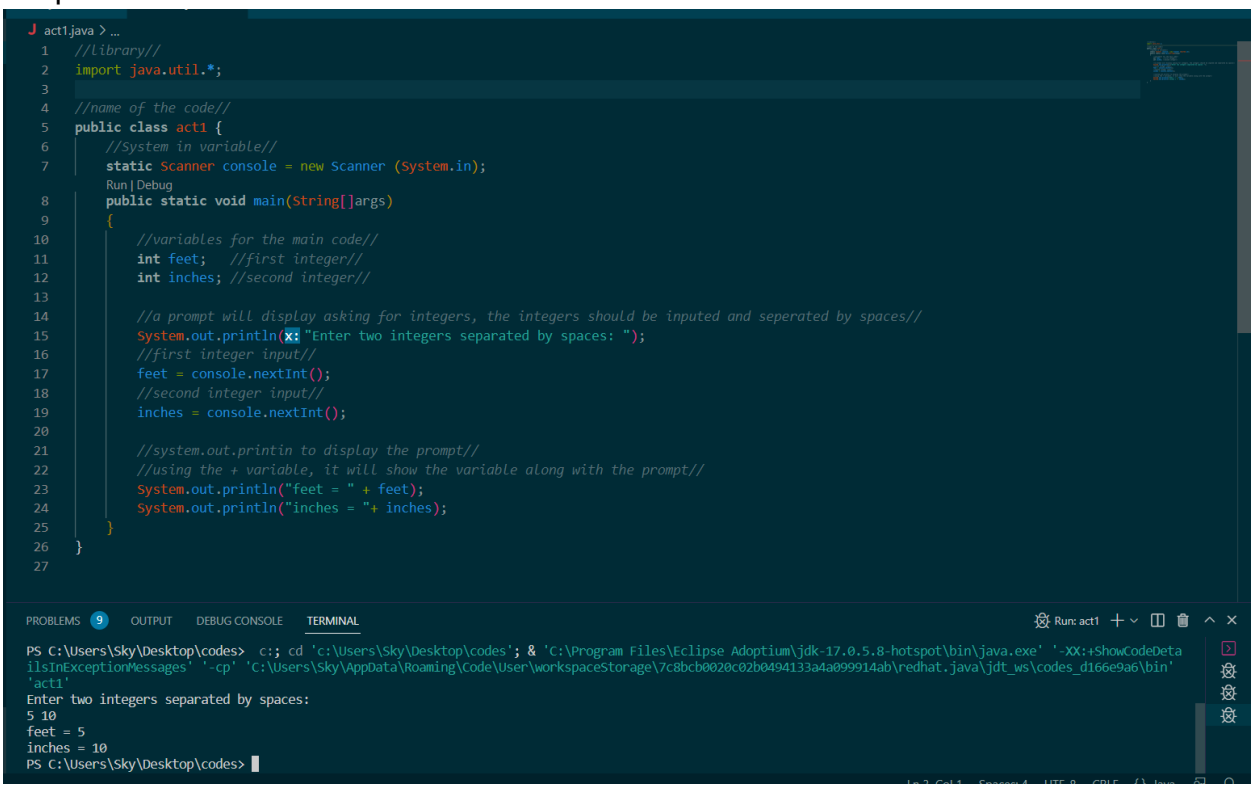
A.

1. Write for the given program using the eclipse editor screen.
2. Write a comment after each statement by to explain of what each line performs.
3. Execute the program.
4. Illustrate the output of the program.

```
1. import java.util.*;
2. public class Sample1_Input {
      static Scanner console = new Scanner (System.in);

3. public static void main(String[]args)
      {
4. int feet;
5. int inches;

6. System.out.println("Enter two integers separated by spaces: ");
7. feet = console.nextInt();
8. inches = console.nextInt();
```

```
9.  System.out.println("feet = " + feet);
10.     System.out.println("inches = "+ inches);
}
}
```

Output:



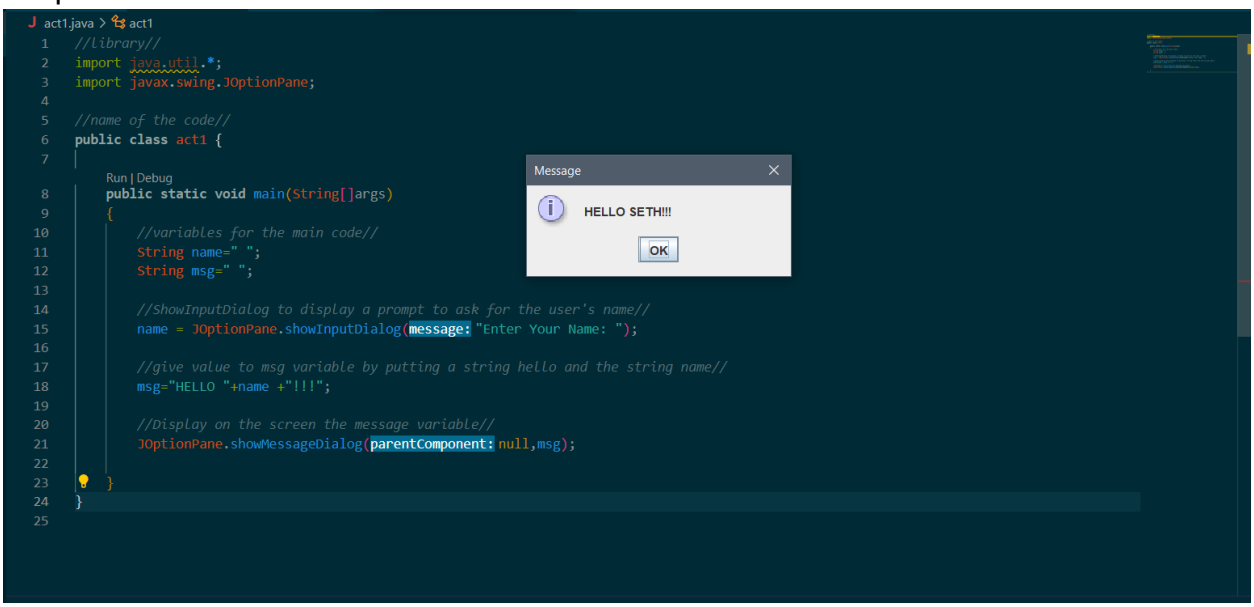
B.

1. Write for the given program using the eclipse editor screen.
2. Write a comment after each statement by to explain of what each line performs.
3. Execute the program.
4. Illustrate the output of the program.

```
1.  import javax.swing.JOptionPane;
2.  public class GetInput1 {
5.      public static void main(String[]args)
        {
              String name=" ";
6.            String msg=" ";
```

```
7.          name = JOptionPane.showInputDialog("Enter Your Name: ");

8.          msg="HELLO  "+name +" !!!";

9.          JOptionPane.showMessageDialog(null,msg);

        }

    }
```

Output:



VI. PROBLEMS/QUESTIONS:

1. Write a program that allows the user to input a name and age in year. Convert the age in year to each equivalent age in month and age in days. Put a comments after each line. Use the following output display:
    a. The java output console
    b. Dialog box.

2. Write a Java program that accepts two input numbers. Compute the sum and average of the inputted numbers

VII. INTERPRETATION/ANALYSIS OF DATA:

act1no1.java > act1no1 > main(String[])

```java
import java.util.*;
import javax.swing.*;


//name of the code//
public class act1no1 {

    Run | Debug
    public static void main(String[]args)
    {
        //variables for the main code//
        int bday;
        String msg=" ";
        String name=" ";
        int month;
        int days;
        //ShowInputDialog to display a prompt to ask for the user's name and birthyear//
        name = JOptionPane.showInputDialog(message: "Enter your Name: ");
        bday = Integer.parseInt(JOptionPane.showInputDialog(message: "Enter your Birthyear: "));
        bday=2022-bday;
        month=bday*12;
        days=bday*365;
        //give value to msg variable by putting a string hello and the string name//
        msg="Hello "+ name +"!\nYour age in year is: "+ bday +
        "\nYour age in months is: "+ month +"\nYour age in days is: "+ days;


        //Display on the screen the message variable//
        JOptionPane.showMessageDialog(parentComponent: null,msg);
```

PROBLEMS 11 OUTPUT DEBUG CONSOLE TERMINAL

The data that was inputted through the variables of int bday and String name.

The variable month and days were used for the calculation/formula

The variable msg was used for the last output, together with the inputted for each variables, it was collected to msg.

The code “JOptionPane” can only be used as string, converting the code using “Integer.parseInt”

Output is straight forward and used the previous given code as reference

act1no2.java > act1no2

```java
//library//
import java.util.*;
import javax.swing.*;


/name of the code//
public class act1no2 {

    Run | Debug
    public static void main(String[]args)
    {
        //variables for the main code//
        int first;
        int second;
        int sum;
        String msg = " ";
        //ShowInputDialog to display a prompt to ask for the user's first and second number//
        first = Integer.parseInt(JOptionPane.showInputDialog(message: "First number: "));
        second = Integer.parseInt(JOptionPane.showInputDialog(message: "Second number: "));

        //give value to msg variable//
        sum = first + second;
        msg= first +" + "+ second +" = "+sum;

        //Display on the screen the message variable//
        JOptionPane.showMessageDialog(parentComponent: null,msg);

    }
```

The data that was inputted through the variables of int first and int second.

The variable sum was used for the calculation/formula with the inputted variables first and second

The variable msg was used for the last output, together with the inputted for each variables, it was collected to msg.

The code “JOptionPane” can only be used as string, converting the code using “Integer.parseInt”

Output is straight forward and used the previous given code as reference

VIII. CONCLUSION/RECOMMENDATION:

Familiar with C# / C++ making this experiment quick and easy. Some unfamiliar command but can be easily picked-up. Being familiar with the software and critical thinking is needed to finish each step. Recommend having a more thorough and student friendly

explanation for each step. Some question / problem have typo and too broad of a statement making it confusing and adapting is a must in that situation.

NAME	**Seth Marcus Martin**	**DATE PERFORMED**	**December 13, 2022**
Activity No. / Title	**Activity 1: Input/Output Statement**	**SECTION**	**COE211**

Course: CPOOPG2L - Object Oriented Programming

Performance Indicators	Fair 40%	Good 60%	Excellent 70%	Score
1. Programming and logic coding (70%)	The program code and its logic formulation are indicated and runnable but displayed not the required output.	The program code and its logic formulation are correct and runnable, but the output require is insufficient	The program code and its logic formulation are indicated, and the output require is completely displayed	
2. Coding style (20%)	**Fair 10%**	**Good 15%**	**Excellent 20%**	
	Variables are properly declared but not follows proper indention	The proper use and declaration of variable and indention are followed	The proper use and declaration of variable and indention are followed with the use of comments for readability is applied.	
3. Output format (10)	**Fair 5%**	**Good 8%**	**Excellent 10%**	
	The output is displayed but not orderly arranged	The required output is displayed properly	The output is complete and creatively displayed	
			Grade	

Assessed by:

ENGR. MARLON G. BAGARA
Printed Name and Signature of Faculty Member

Date