JAVA ELEMENTARY PROGRAMMING & SELECTIONS

Machine Problem No. 2 Worksheet

Course:	**NCP2103 : Object-Oriented Programming**		
Student Name:	**VILLAROSA**	**JEDH CARLO**	**B.**
	Surname (UPPERCASE)	Given Name (UPPERCASE)	M.I.
Student No.:	**20190140068**	Section:	**CPE1**
Date Performed:	**AUGUST 29, 2023**	Date Submitted:	**AUGUST 2023**

Machine Problems

Instructions: Solve the following problem using Java programming. Use JOption Dialog Boxes for user I/O. Provide a screenshot of your code and sample outputs for each item.

1. Create a Java program that enables the cashier staff of XYZ Restaurants to enter a customer's subtotal bill and tip rate (in percentage). Then, the program shall compute the tip amount and the total bill. For example, if customer ABC's bill is P1000.00 and gives a 15% tip, the program shall display P150.00 as a tip and 1150 as the total bill.

   Sample Test Data:
   See the supplementary [test data](#) *sheet as needed.*

Source Code

```java
import javax.swing.JOptionPane;
public class jOption2 {
    public static void main(String[] args)
    {
        double bill = 0;
        double tip = 0.0;

        bill = Double.parseDouble(JOptionPane.showInputDialog("Insert your bill: "));
        tip = Double.parseDouble(JOptionPane.showInputDialog("Insert your tip in percentage: "));

        double subtotal = bill*(tip/100);
        double total = subtotal+bill;

        JOptionPane.showMessageDialog( parentComponent: null,  message: "Your tip is: " + String.format("%.2f",subtotal) + " Your total is: " + String.format
    }
}
```
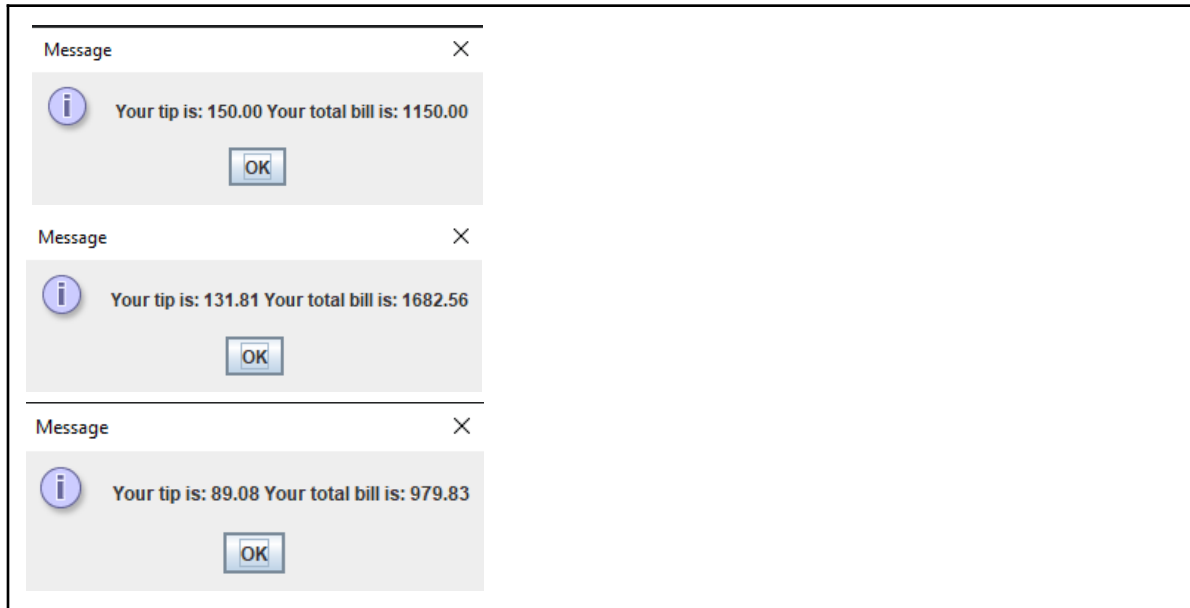
Sample Outputs

```
Message                          ×
 (i)   Your tip is: 150.00 Your total bill is: 1150.00
              [ OK ]

Message                          ×
 (i)   Your tip is: 131.81 Your total bill is: 1682.56
              [ OK ]

Message                          ×
 (i)   Your tip is: 89.08 Your total bill is: 979.83
              [ OK ]
```

2. Based on the lengths of its sides, a triangle can be classified as equilateral, isosceles, or scalene. The sizes of all three sides of an **equilateral** triangle are identical. Two sides of an **isosceles** triangle have the same length, whereas the third side has a variable length. If each side of a triangle has a different length, the triangle is **scalene**.

   Also, a triangle can be determined based on its angle – **Acute Triangle**: When all the angles of a triangle are acute, that is, they measure less than 90°, it is called an acute-angled triangle or acute triangle. **Right Triangle:** When one of the angles of a triangle is 90°, it is called a right-angled triangle or right triangle. **Obtuse Triangle:** When one of the angles of a triangle is an obtuse angle, that is, it **measures greater than 90°, it is called an obtuse-angled triangle or obtuse triangle.**

   Note:
   - The sum of two sides of a triangle must be greater than the third side. If this is true for all three combinations, then you will have a valid triangle.
   - The sum of all three angles of a triangle must be equal to 180°. Otherwise, the triangle is invalid.

   Create a Java program that asks for in which they want to determine the type of triangle **[1] "By Sides"** or **[2] "By Angle"**. Then, ask the user the three parameters needed and determine the type of triangle.

Sample Test Data:
See the supplementary [test data](#) sheet as needed.

Source Code

```java
import javax.swing.JOptionPane;
public class jOption2_2 {

    public static void main(String[] args)
    {
        //Choice if user want to determine the triangle by side or by angle
        int choice = Integer.parseInt(JOptionPane.showInputDialog("Please enter [1] if you want to determine triangle by sides or [2] by angles"));

        if(choice == 1)
        {
            //user input for 3 sides
            int side1 = Integer.parseInt(JOptionPane.showInputDialog("Please enter value of side 1: "));
            int side2 = Integer.parseInt(JOptionPane.showInputDialog("Please enter value of side 2: "));
            int side3 = Integer.parseInt(JOptionPane.showInputDialog("Please enter value of side 3: "));

            //conditions for triangle by sides
            if(side1 + side2 > side3)
            {
                if(side1 == side2 && side2 == side3)
                {
                    JOptionPane.showMessageDialog( parentComponent: null,  message: "Side 1: "+ side1 +"\n Side 2: "+ side2 +"\n You've entered an equilateral triangle");
                }
                else if(side1 == side2 || side1 == side3 || side2 == side3)
                {
                    JOptionPane.showMessageDialog( parentComponent: null,  message: "Side 1: "+ side1 +"\n Side 2: "+ side2 +"\n Side 3: "+ side3 + "\n You've entered an isosceles triangle");
                }
                else
                {
                    JOptionPane.showMessageDialog( parentComponent: null,  message: "Side 1: "+ side1 +"\n Side 2: "+ side2 +"\n Side 3: "+ side3 + "\n You've entered a scalene triangle");
                }
            }
            else
            {
                JOptionPane.showMessageDialog( parentComponent: null,  message: "Side 1: "+ side1 +"\n Side 2: "+ side2 +"\n Side 3: "+ side3 + "\n You've entered an invalid triangle");
            }
        }
        else if(choice == 2)
        {
            //user input for 3 angles
            int angle1 = Integer.parseInt(JOptionPane.showInputDialog("Please enter value of angle 1: "));
            int angle2 = Integer.parseInt(JOptionPane.showInputDialog("Please enter value of angle 2: "));
            int angle3 = Integer.parseInt(JOptionPane.showInputDialog("Please enter value of angle 3: "));

            //conditions
            if(angle1 + angle2 + angle3 == 180)
            {
                if(angle1 < 90 && angle2 < 90 && angle3 < 90)
                {
                    JOptionPane.showMessageDialog( parentComponent: null,  message: "Angle 1: "+ angle1 + "\n Angle 2: "+ angle2 +"\n Angle 3: "+ angle3 +"\n You've entered an acute triangle");
                }
                else if(angle1 == 90 || angle2 == 90 || angle3 == 90)
                {
                    JOptionPane.showMessageDialog( parentComponent: null,  message: "Angle 1: "+ angle1 + "\n Angle 2: "+ angle2 +"\n Angle 3: "+ angle3 +"\n You've entered a right triangle");
                }
                else if(angle1 > 90 || angle2 < 90 || angle3 < 90)
                {
                    JOptionPane.showMessageDialog( parentComponent: null,  message: "Angle 1: "+ angle1 + "\n Angle 2: "+ angle2 +"\n Angle 3: "+ angle3 +"\n You've entered an obtuse triangle");
                }
            }
            else
            {
                JOptionPane.showMessageDialog( parentComponent: null,  message: "Angle 1: "+ angle1 + "\n Angle 2: "+ angle2 +"\n Angle 3: "+ angle3 +"\n You've entered an invalid triangle");
            }

        }

    }

}
```
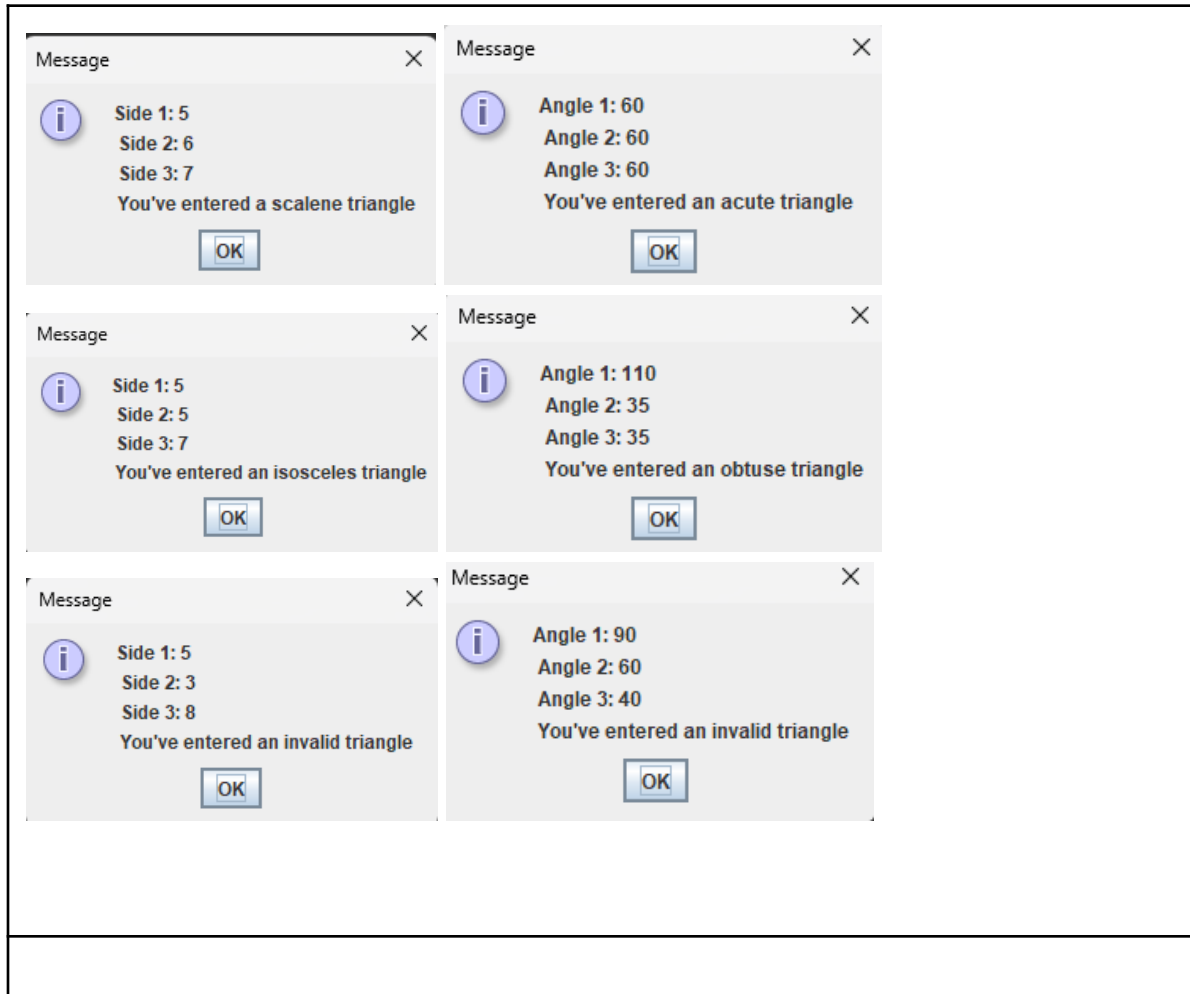
Sample Outputs

Message ✕

ℹ Side 1: 5
Side 2: 6
Side 3: 7
You've entered a scalene triangle

OK

Message ✕

ℹ Angle 1: 60
Angle 2: 60
Angle 3: 60
You've entered an acute triangle

OK

Message ✕

ℹ Side 1: 5
Side 2: 5
Side 3: 7
You've entered an isosceles triangle

OK

Message ✕

ℹ Angle 1: 110
Angle 2: 35
Angle 3: 35
You've entered an obtuse triangle

OK

Message ✕

ℹ Side 1: 5
Side 2: 3
Side 3: 8
You've entered an invalid triangle

OK

Message ✕

ℹ Angle 1: 90
Angle 2: 60
Angle 3: 40
You've entered an invalid triangle

OK

3. Mr. XYZ is a car enthusiast who loves to drive sports cars. He wants to determine the cost of fuel efficiency of his rides. Help him by writing a Java program that prompts the user to enter the distance to drive in kilometers, the fuel efficiency of the car in kilometers per gallon, and the price per gallon, and displays the cost of the trip.

Sample Test Data:
See the supplementary test data sheet as needed.
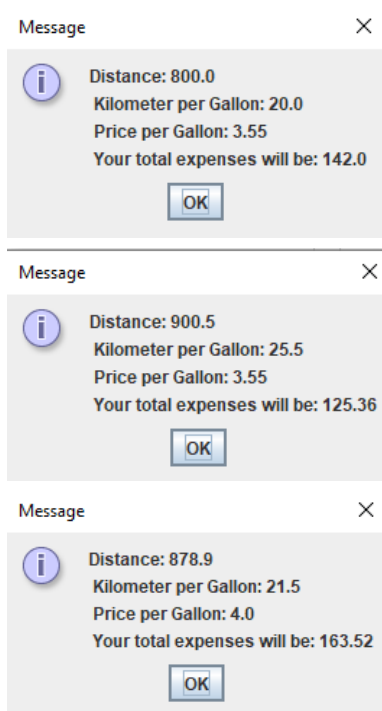
Source Code

```
import javax.swing.JOptionPane;
public class jOption2_3 {
    public static void main(String[] args)
    {
        //d/km/p
        double distance = Double.parseDouble(JOptionPane.showInputDialog("Enter the distance you want to travel: "));
        double kmperG = Double.parseDouble(JOptionPane.showInputDialog("Enter the Kilometer per Gallon: "));
        double pperg = Double.parseDouble(JOptionPane.showInputDialog("Enter the price per gallon: "));

        double computation = distance/kmperG;
        double computation1 = computation*pperg;

        JOptionPane.showMessageDialog( parentComponent: null,  message: "Distance: "+ distance + "\n Kilometer per Gallon: "+ kmperG +"\n Price per Gallon:
    }
}
```

Sample Outputs

Message ✕

(i) Distance: 800.0
Kilometer per Gallon: 20.0
Price per Gallon: 3.55
Your total expenses will be: 142.0

OK

Message ✕

(i) Distance: 900.5
Kilometer per Gallon: 25.5
Price per Gallon: 3.55
Your total expenses will be: 125.36

OK

Message ✕

(i) Distance: 878.9
Kilometer per Gallon: 21.5
Price per Gallon: 4.0
Your total expenses will be: 163.52

OK

4. Write a program to help Basya, a cashier at XYZ supermarket, to give exact change to their customers. The program shall compute the number of bills and coins needed to make changes for a customer. For instance, if you need to give 380.36 in the change you would need one two-hundred peso bill, one one-hundred peso bill, one fifty, one twenty, one ten coins, three ten-centavo coin, one five-centavo coin, and one- centavo coin. Use 500s, 200s, 100s, 50s, 20s, 10s, 5s, 1s, 10cs, 5cs, 1cs as monetary guide. (Hint: You can solve this problem by doing division, multiplication, subtraction, and converting floats to ints. If a

currency does not have any count, replace it with "--"

Sample Test Data:
See the supplementary test data sheet as needed.

Source Code
Sample Outputs

5. Write a Java program that can identify the mobile network of a given mobile number using a **switch** statement. The program shall accept only 11-digit numbers, otherwise display an error message. The network is identified by the first four or five digits of the mobile number. Visit this **link** for the complete list of prefixes.

   Note: Due to the number of prefixes, you are only to consider **five** prefixes for each network group (Globe/TM, Smart/TNT, Others).

   Sample Test Data:
   See the supplementary test data sheet as needed.

Source Code

```java
import javax.swing.JOptionPane;

public class NetworkIdentifier {
    public static void main(String [] args)
    {

        long digits = Long.parseLong(JOptionPane.showInputDialog( parentComponent: null, message: "Please enter a Mobile Number"));
        long number =  digits/10000000;

        if (number == 817 || number == 905 || number == 906 || number <= 917 && number >= 915 || number == 926 || number == 927 || number <= 937 && number >= 935 || number == 945
            || number <= 956 && number >= 953 || number <= 967 && number >= 965 || number <= 979 && number >= 975 || number <= 995 && number >= 997)
        {
            long postpaid = digits/1000000;
            if(postpaid == 9173 || postpaid <= 9178 && postpaid >= 9175 || postpaid == 9253 || postpaid <= 9258 && postpaid >= 9255)
            {
                JOptionPane.showMessageDialog( parentComponent: null, message: "The number you have entered 0" + digits + " " + "is Globe Postpaid.");
            }
            else
            {
                JOptionPane.showMessageDialog( parentComponent: null, message: "The number you have entered 0" + digits + " " + "is Globe/TM.");
            }
        }
        else if (number == 907 || number == 909 || number == 910 || number == 912 || number == 930 || number == 938 || number == 946 || number == 948 || number == 950)
        {
            JOptionPane.showMessageDialog( parentComponent: null, message: "The number you have entered 0" + digits + " " + "is Talk N' Text.");
        }
        else if (number == 908 ||  number <= 921 && number >= 918 || number == 928 || number == 929  || number == 939 || number == 947 || number == 949
            || number == 951 || number == 961 || number == 998 || number == 999)
        {
            JOptionPane.showMessageDialog( parentComponent: null, message: "The number you have entered 0" + digits + " " + "is Smart.");
        }
        else if (number <= 925 && number >= 922 || number <= 934 && number >= 931 || number <= 943 && number >= 940 || number == 973 || number == 974)
        {
            JOptionPane.showMessageDialog( parentComponent: null, message: "The number you have entered 0" + digits + " " + "is Sun.");
        }
        else
        {
            JOptionPane.showMessageDialog( parentComponent: null, message: "The number you have entered 0" + digits + " " + "is an invalid number");
        }
```

Sample Outputs

Input
Enter mobile Number
098212344
OK Cancel

Message
Invalid Number
OK

Input
Please enter a Mobile Number
09178091152
OK Cancel

```
Message                                    ×

(i)   The number you have entered 09178091152 is Globe Postpaid.

              OK

Input                                      ×

[?]   Please enter a Mobile Number
      09218091152

         OK          Cancel

Message                                    ×

(i)   The number you have entered 09218091152 is Smart.

              OK
```

6. Write a Java program that simulates the "HEP-HEP HOORAY" program.

> Say "Hep-Hep" if the number is divisible by 3.
>
> Say "Hooray" if the number is divisible by 5.
>
> Say "Hep-Hep Hooray" if the number is divisible by both 3 and 5.
>
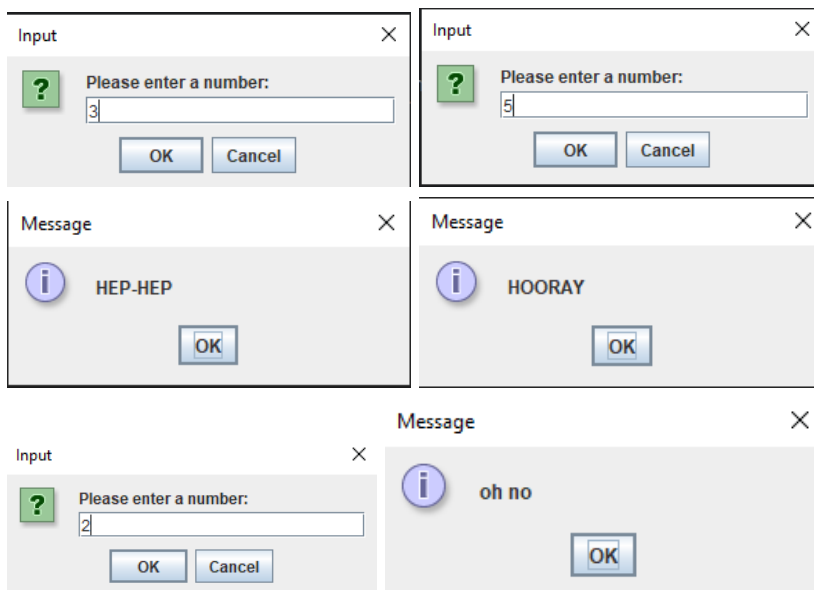> Return "oh no!" if the number is not divisible by 3 and 5.

Sample Test Data:
See the supplementary [test data](#) sheet as needed.

Source Code

```java
import javax.swing.JOptionPane;
public class jOption2_6 {
    public static void main(String[] args)
    {
        int number = Integer.parseInt(JOptionPane.showInputDialog("Please enter a number: "));

        if(number % 3 == 0)
        {
            JOptionPane.showMessageDialog( parentComponent: null,  message: "HEP-HEP");
        }
        if(number % 3 == 0 && number % 5 == 0)
        {
            JOptionPane.showMessageDialog( parentComponent: null,  message: "HEP_HEP HOORAY");
        }
        if(number % 5 == 0)
        {
            JOptionPane.showMessageDialog( parentComponent: null,  message: "HOORAY");
        }

        else
        {
            JOptionPane.showMessageDialog( parentComponent: null,  message: "oh no");
        }
```

Sample Outputs

Input — Please enter a number: 3 — OK / Cancel

Input — Please enter a number: 5 — OK / Cancel

Message — (i) HEP-HEP — OK

Message — (i) HOORAY — OK

Input — Please enter a number: 2 — OK / Cancel

Message — (i) oh no — OK

Advanced Problems

7 - 8. The new income tax table below is applied for the year 2020 following the new BIR TRAIN, meaning, the Tax Reform for Acceleration and Inclusion Act implementing the

Income Tax Provisions of the Republic Act 10963. Below is the tax "annual" tax matrix. Write a program that will accept the monthly salary of an employee and display the monthly income tax. Annual taxable income is monthly income multiplied by 12.

Amount of Taxable Income		Rate
Over	But Not Over	
-	250,000	0%
250,000	400,000	20% of the excess over P250,000
400,000	800,000	P30,000 + 25% of the excess over P400,000
800,000	2,000,000	P130,000 + 30% of the excess over P800,000
2,000,000	8,000,000	P490,000 +32% of the excess over P2 Million
8,000,000		P2,410,000 + 35% of the excess over P8 Million

Source Code

Sample Outputs

9 - 10. Write a Java program that will simulate an in-between game. The game draws two random cards and will allow the user to choose whether the player will "DEAL" or "NO DEAL". The user will win the game whenever the third random card (which will be shown after the user makes his deal) is in between exclusively the first two random cards. When the user bets NO DEAL, automatically he/she loses the game.

Note: In case the two random cards were the same, the user has a choice to choose whether "HIGHER" or "LOWER".

Below are the sample outputs with the assumption of the "Deal".

First Card	Second Card	Third Card	Status
8	13	10	Wins
8	13	3	Loses
2	2 *HIGHER*	10	Wins
8	8 *LOWER*	10	Loses
5	10	10	Loses
5	10	5	Loses

Source Code

```java
import javax.swing.JOptionPane;
import java.util.Random;

public class jOption2_9 {
    public static void main(String[] args){
        Random random = new Random();

        int card1 = random.nextInt( bound: 13)+1;
        int card2 = random.nextInt( bound: 13)+1;

        //Draws 2 card
        JOptionPane.showMessageDialog( parentComponent: null, message: "Your first card is: "+ card1 + "\nYour second card is: "+ card2);

        //choose deal or no deal
        int choice = Integer.parseInt(JOptionPane.showInputDialog("Please enter [1] if Deal or enter [2] if No Deal"));

        if (choice == 1)
        {
            //draw third card
            int card3 = random.nextInt( bound: 13)+1;

            //check
            if(card1<card2)
            {
                if (card3 > card1 && card3 < card2)
                {
                    JOptionPane.showMessageDialog( parentComponent: null, message: "You win! The third card is: "+ card3);
                }else {
                    JOptionPane.showMessageDialog( parentComponent: null, message: "You lose. The third card is: "+ card3);
                }
            }else{
                if (card3 < card1 && card3 > card2)
                {
                    JOptionPane.showMessageDialog( parentComponent: null, message: "You win! The third card is: "+ card3);
                }else{
                    JOptionPane.showMessageDialog( parentComponent: null, message: "You lose. The third card is: "+ card3);
                }
            }
        }else {
            JOptionPane.showMessageDialog( parentComponent: null, message: "You lose.");
        }
```

Sample Outputs

Message ✕

ⓘ Your first card is: 2
Your second card is: 6

[OK]

Message ✕

ⓘ Your first card is: 6
Your second card is: 13

[OK]

Input ✕

❓ Please enter [1] if Deal or enter [2] if No Deal
`1`

[OK] [Cancel]

Input ✕

❓ Please enter [1] if Deal or enter [2] if No Deal
`2`

[OK] [Cancel]

Message ✕

ⓘ You win! The third card is: 4

[OK]

Message ✕

ⓘ You lose.

[OK]

Grading Rubric

See the attached grading rubric.