PROGRAMMING ASSIGNMENT

Task 1:

Ans.

Explanation of the program:

The program is started by giving a class (‘public class Flag{’)defining the methods and variables and the main method (‘public static void main(String[] args’))which is to perform the actions written under it.

Printing the flag:

The print() method has been used to perform the action printing the ‘-’ as specified in the program or called by the method and also in the flag the lines ‘|’ are looped using ‘for’ loop to repeat in each layer of the flag. Next only print() method is used to print the stick and stairs part.

Pseudocode:

```
Start
        DECLARE
            int i;
           int j;
           int k;
           int l;
  //first layer of flag//
    DISPLAY(“-“)
    for loop(i) ‘|’
    DISPLAY(i)
    DISPLAY(“-“)
```

```
    for loop (j) '|'
    DISPLAY(j)
    DISPLAY("-")
      for loop (k) '|'
    DISPLAY(k)
    DISPLAY("-")
    //Stick part//
      for loop(l) '|'
   //Stairs part//
    DISPLAY("-")
    DISPLAY("|")
    END
EXIT
```

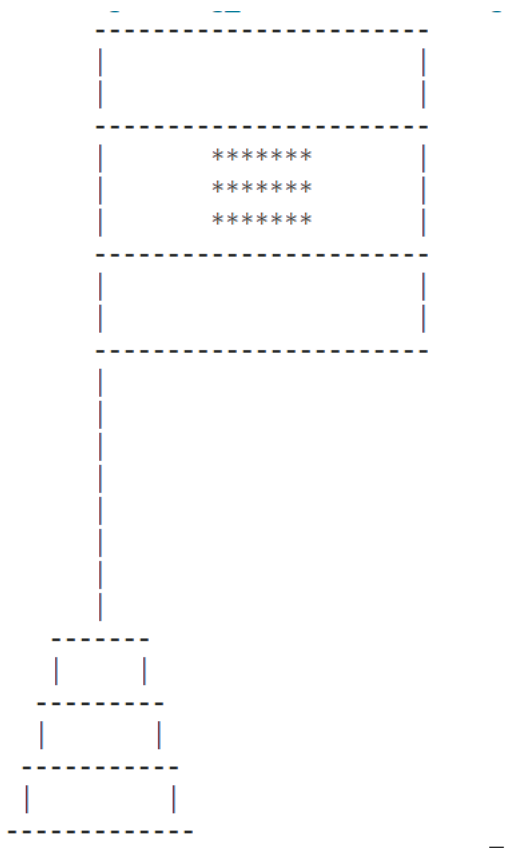
Source Code:

```
//Name : Thaspeeha Vahithu, Instructor Name: Ms.Shibili Said, Student
Id:32146925//
//This program prints the Indian Flag using for loops//
public class Flag //Defining the class//
{
    public static void main(String[] args)  //Defining the main method//
    {
        //Prints the flag part//
        System.out.println("      ------------------------");
        for (int i = 0; i < 2; i++){
        System.out.println("      |                      |");
        }
        System.out.println("      ------------------------");
        for (int j = 0; j < 3; j++){
        System.out.println("      |       *******        |");
        }
        System.out.println("      ------------------------");
        for (int k = 0; k < 2; k++){
        System.out.println("      |                      |");
        }
        System.out.println("      ------------------------");

        //Prints the stick part//
        for (int l = 0; l < 8; l++){
        System.out.println("      |                       ");
        }

        //Prints the last part with stairs//
        System.out.println("   -------");
        System.out.println("   |     |");
        System.out.println("  ---------");
        System.out.println("  |       |");
        System.out.println(" -----------");
        System.out.println(" |         |");
        System.out.println("-------------");
         }

}
```

Program code:

```
1   //Name : Thaspeeha Vahithu, Instructor Name: Ms.Shibili Said, Student Id:32146925//
2   //This program prints the Indian Flag using for loops//
3
4   public class Flag  //Defining the class//
5   {
        Run | Debug
6       public static void main(String[] args) //Defining the main method//
7       {
8           //Prints the flag part//
9           System.out.println(x:"      ------------------------");
10          for (int i = 0; i < 2; i++){
11          System.out.println(x:"      |                      |");
12          }
13          System.out.println(x:"      ------------------------");
14          for (int j = 0; j < 3; j++){
15          System.out.println(x:"      |       *******       |");
16          }
17          System.out.println(x:"      ------------------------");
18          for (int k = 0; k < 2; k++){
19          System.out.println(x:"      |                      |");
20          }
21          System.out.println(x:"      ------------------------");
22
23          //Prints the stick part//
24          for (int l = 0; l < 8; l++){
25          System.out.println(x:"      |                       ");
26          }
27
28          //Prints the last part with stairs//
29          System.out.println(x:"   -------");
30          System.out.println(x:"   |     |");
31          System.out.println(x:"  ---------");
32          System.out.println(x:"  |        |");
33          System.out.println(x:" -----------");
34          System.out.println(x:" |          |");
35          System.out.println(x:"-------------");
36      }
37  }
```

Console:



Task 2:

Explanation:
The program is about to calculate Body Mass Index and find overweight, normal, obesity, underweight accordingly.

The program first starts with (import java.util.Scanner;) since the program will be asking the user to input values and next is public class which is to define methods and variables, the main method is used to perform action written under it or calling the method.

Printing the title:
Uses the print()method to print main title for the user to easily use the program.
Scanner class:
Declaring the scanner class or an instance using various variables to ask the user to input.
Creating an user input to ask the user enter their weight and height and once its entered the BMI calculation starts using the height and weight entered.

If-elseif:
After the calculation it checks the if-else statements and result the type of weight accordingly i.e. for example : If BMI is 25.7 then it checks each statement which is one is appropriate, so here its appropriate for if (BMI<=34.9) and prints the required statement.

Pseudocode:

```
Start

    PRINT “BMI(Body Mass Index) Calculator”

DECLARE
        Scanner class()
        double weight;
        double height;
        double BMI;

PRINT “Enter your weight(kg)”
     INPUT weight
PRINT “Enter your height(m)”
      INPUT height
BMI = (weight/height*height));
PRINT BMI
     IF BMI <= 18.4
PRINT "Underweight"
     ELSE IF BMI <=24.9
   PRINT "Normal"
         ELSE IF BMI >= 34.9
   PRINT "Overweight"
         ELSE
   PRINT "Obesity"

 END IF-ELSE

EXIT
```

Source code:

```
//Name: Thaspeeha Vahithu, Instructor Name: Ms.Shibili, Student Id: 32146925//
//The below program calculates your Body Mass Index(BMI) and displays your type
of weight//
import java.util.Scanner; //Import for Scanner class//
 public class BodyMassIndex //Defining the class//
 {
    public static void main(String[] args) //Defining main method//
    {
        //Prints the title//
        System.out.println("BMI(Body Mass Index) Calculator\n");
        //Declaring an user input with scanner class, asking for data values//
           Scanner input = new Scanner(System.in);

        System.out.println("Enter your weight(kg): ");
           double weight = input.nextDouble();
        System.out.println("Enter your height(m): ");
           double height = input.nextDouble();

        //Calculation for BMI//
        System.out.println("Your BMI is: " );
           double BMI = (weight/(height*height));
        System.out.println(BMI);

        //Prints whether the calculated BMI result corresponds to the following
weights respectively using if-elseif condition//
        if (BMI <=18.4)
        {
         System.out.println("Underweight");
        }
        else if (BMI <=24.9)
        {
        System.out.println("Normal Weight");
        }
        else if (BMI <=34.9)
        {
         System.out.println("Overweight");
        }
        else
        {
         System.out.println("Obesity");
        }
}
}
```

Program code:

```java
//Name: Thaspeeha Vahithu, Instructor Name: Ms.Shibili, Student Id: 32146925//
//The below program calculates your Body Mass Index(BMI) and displays your type of weight//

import java.util.Scanner; //Import for Scanner class//
public class BodyMassIndex //Defining the class//
{
    Run | Debug
    public static void main(String[] args) //Defining main method//
    {
        //Prints the title//
        System.out.println("BMI(Body Mass Index) Calculator\n");

        //Declaring an user input with scanner class, asking for data values//
            Scanner input = new Scanner(System.in);

        System.out.println("Enter your weight(kg): ");
            double weight = input.nextDouble();

        System.out.println("Enter your height(m): ");
            double height = input.nextDouble();

        //Calculation for BMI//
        System.out.println("Your BMI is: " );
            double BMI = (weight/(height*height));

        //Prints whether the calculated BMI result corresponds to the following weights respectively using if-elseif condition//
        if (BMI <=18.4)
        {
         System.out.println("Underweight");
        }
        else if (BMI <=24.9)
        {
        System.out.println("Normal Weight");
        }
        else if (BMI <=34.9)
        {
         System.out.println("Overweight");
        }
        else
        {
         System.out.println("Obesity");
        }

         }
         }
```

Console 1:

```
BMI(Body Mass Index) Calculator

Enter your weight(kg):
45
Enter your height(m):
1.6
Your BMI is:
17.578124999999996
Underweight
```

Console 2:

```
BMI(Body Mass Index) Calculator

Enter your weight(kg):
65
Enter your height(m):
1.8
Your BMI is:
20.061728395061728
Normal Weight
```

Console 3:

```
BMI(Body Mass Index) Calculator

Enter your weight(kg):
75
Enter your height(m):
1.6
Your BMI is:
29.296874999999993
Overweight
```

Console 4:

```
BMI(Body Mass Index) Calculator

Enter your weight(kg):
70
Enter your height(m):
1.4
Your BMI is:
35.71428571428572
Obesity
```

Task 3:

Explanation:
The program is based on choosing the products and calculating the total amount of the products.

Printing the title:
Print() method is used here to print the required details for the user to easily input values accordingly that is products with options to choose easily and their prices.

Scanner class:
This is used to input or choose the options from the list displayed in the program.

While Loop:
The loop is declared here to sum up the product prices after the products are looped with their quantities or chosen.

Switch statements (case):
The prices and their quantites get calculated according to what product has been chosen so switch statements are declared here so the program can easily recognize the correct case or product to be calculated.

Summing up the total amount :
A for loop has been declared to check each products and their quantities and using sum variable the prices are multiplied by their quanitities.

Pseudocode:

```
Start
PRINT "ONLINE STORE"
    "\n"
    "Choose your required products and quantities from the given list of items given below:"
    "    PRODUCT              PRICE     "
    "1.Matcha Powder        32 per 1 bottle powder"
    "2.Lipstain             12 per bottle"
    "3.Rice                70 per Kilogram"
    "4.Bread               10 per bread"
      "Exit by typing any number(from 5) or q or exit"

DECLARE
      Scanner input = new Scanner(System.in)
      ArrayList <Integer> quantity = new ArrayList <Integer> ()
      int condition = 1;

WHILE LOOP
      while (condition==1)

         PRINT "\nEnter your option:"
         INPUT option

SWITCH STATEMENT
         switch(option)

 DECLARE case 1:
  PRINT "Enter the number of bottles of Match powder you want to buy:"
         INPUT number1

    DECLARE int  total1
         total1 = number1 * 32;
         PRINT "total1"
         break;

 DECLARE case 2:
  PRINT "Enter the number of bottles of Lipstain you want to buy:"
         INPUT number2

    DECLARE int total2
         total2 =number2 * 12;
         PRINT "total2"
         break;
```

```
 DECLARE case 3:
 PRINT “Enter the number of kilograms you want to buy:”
        INPUT number3

    DECLARE int total3
        total3 = number3 * 70;
        PRINT “total3”
        break;
  DECLARE case 4:
   PRINT “Enter the number of breads you want to buy:”
          INPUT number4

   DECLARE int total4
          total4 = number4 * 10;
          PRINT “total4”
          break;

   default:
   PRINT “Products are over and you are done buying.”
          condition = 0;
          break;

   DECLARE int list_length
          list_length = quantity.size()
   DECLARE int sum
          sum = 0;
   FOR LOOP
          for(int i = 0; i<list_length; i++)
          sum = sum + quantity.get(i)

   PRINT (“\nThe products you have chosen costs "+ sum + "."”)

   END WHILE LOOP
END SWITCH

END FOR LOOP

EXIT
```

Source code:

```
//Name: Thaspeeha Vahithu, Instructor Name: Ms.Shibili Said, Student Id:
32146925//
//The below program computes the tally in for buying your products and calculates
the total amount//

import java.util.Scanner;  //Import for Scanner class//
import java.util.ArrayList; //Import for arraylist//
public class OnlineStore //Defining class//
{
    public static void main(String[] args) //Defining a method that contains all
the products with their prices//
    {
         //Printing the required details//
         System.out.println("ONLINE STORE");
         System.out.println("\n");
         System.out.println("Choose your required products and quantities from
the given list of items given below: ");
         System.out.println("    PRODUCT                  PRICE     ");
         System.out.println("1.Matcha Powder          32 per 1 bottle powder");
         System.out.println("2.Lipstain               12 per bottle");
         System.out.println("3.Rice                   70 per Kilogram");
         System.out.println("4.Bread                  10 per bread");
         System.out.println("Exit by typing any number(from 5) or q or exit");

         //Scanner class for user to input values//
         Scanner input = new Scanner(System.in);

        //Array list for integers to input values and for calculation//
        ArrayList <Integer> quantity = new ArrayList <Integer> ();

        int condition = 1;
        //Using while condition for checking the switch statements(case)//
        while (condition==1)
        {
            System.out.println("\nEnter your option:");
            int option = input.nextInt();

       //Switch statements to declare the cases and check the correct cases once
the user input values//
            switch(option)
```

```
 {


case 1 :
System.out.println("Enter the number of bottles of Match powder you want to
buy:");
                int number1 = input.nextInt();
                //calculate the total price//
                int total1 = number1 * 32;
                //add the the total to the list//
                quantity.add(total1);
                break;


case 2:
  System.out.println("Enter the number of bottles of Lipstain you want to buy:");
                int number2 = input.nextInt();
                //calculate the total price//
                int total2 =number2 * 12;
                //add the total to the list//
                quantity.add(total2);
                break;

  case 3:
  System.out.println("Enter the number of kilograms you want to buy:");
                int number3 = input.nextInt();
                //calculate the total price//
                int total3 = number3 * 70;
                //add the total to the list//
                quantity.add(total3);
                break;

  case 4:
  System.out.println("Enter the number of breads you want to buy:");
                int number4 = input.nextInt();
                //calculate the total price//
                int total4 = number4 * 10;
                //add the total to the list//
                quantity.add(total4);
                break;

  default:
  //This is for exiting the products if it exceeds 4 products//
  System.out.println("Products are over and you are done buying.");
                 condition = 0;
```

```
                    break;
                }
                }
                  //Now let's get the length of the list//
                  int list_length = quantity.size();
                  //declare total sum here//
                  int sum = 0;
                for(int i = 0; i<list_length; i++)
                {
                  sum = sum + quantity.get(i);
                }
                  //Now prints the sum//
                  System.out.println("\nThe products you have chosen costs "+ sum +
"."); 
                }

                }
```

Program code:

```java
//Name: Thaspeeha Vahithu, Instructor Name: Ms.Shibili Said, Student Id: 32146925//
//The below program computes the tally in for buying your products and calculates the total amount//

import java.util.Scanner; //Import for Scanner class//
import java.util.ArrayList; //Import for arraylist//
public class OnlineStore //Defining class//
{
    Run | Debug
    public static void main(String[] args) //Defining a method that contains all the products with their prices//
    {
        //Printing the required details//
        System.out.println(x:"ONLINE STORE");
        System.out.println(x:"\n");
        System.out.println(x:"Choose your required products and quantities from the given list of items given below: ");
        System.out.println(x:"    PRODUCT                PRICE     ");
        System.out.println(x:"1.Matcha Powder          32 per 1 bottle powder");
        System.out.println(x:"2.Lipstain               12 per bottle");
        System.out.println(x:"3.Rice                   70 per Kilogram");
        System.out.println(x:"4.Bread                  10 per bread");
        System.out.println(x:"Exit by typing any number(from 5) or q or exit");

        //Scanner class for user to input values//
        Scanner input = new Scanner(System.in);

        //Array list for integers to input values and for calculation//
        ArrayList <Integer> quantity = new ArrayList <Integer> ();

        int condition = 1;
        //Using while condition for checking the switch statements(case)//
        while (condition==1)
        {
            System.out.println(x:"\nEnter your option:");
            int option = input.nextInt();

            //Switch statements to declare the cases and check the correct cases once the user input values//
            switch(option)
            {
        {
            System.out.println(x:"\nEnter your option:");
            int option = input.nextInt();

            //Switch statements to declare the cases and check the correct cases once the user input values//
            switch(option)
            {
              case 1:
                System.out.println(x:"Enter the number of bottles of Match powder you want to buy:");
                int number1 = input.nextInt();
                //calculate the total price//
                int total1 = number1 * 32;
                //add the the total to the list//
                quantity.add(total1);
                break;

              case 2:
                System.out.println(x:"Enter the number of bottles of Lipstain you want to buy:");
                int number2 = input.nextInt();
                //calculate the total price//
                int total2 =number2 * 12;
                //add the total to the list//
                quantity.add(total2);
                break;

```

```
55                  case 3:
56                    System.out.println(x:"Enter the number of kilograms you want to buy:");
57                    int number3 = input.nextInt();
58                    //calculate the total price//
59                    int total3 = number3 * 70;
60                    //add the total to the list//
61                    quantity.add(total3);
62                    break;
63
64                  case 4:
65                    System.out.println(x:"Enter the number of breads you want to buy:");
66                    int number4 = input.nextInt();
67                    //calculate the total price//
68                    int total4 = number4 * 10;
69                    //add the total to the list//
70                    quantity.add(total4);
71                    break;
72
73                  default:
74                    //This is for exiting the products if it exceeds 4 products//
75                    System.out.println(x:"Products are over and you are done buying and thank you for choosing the products!");
76                    //String number5 = input.nextLine();
77                     condition = 0;
78                     break;
79                  }
80                  }
81                      //Now lets get the length of the list//
82                      int list_length = quantity.size();
83                      //declare total sum here//
84                      int sum = 0;
85                    for(int i = 0; i<list_length; i++)
86                    {
87                      sum = sum + quantity.get(i);
88                    }
89                      //Now prints the sum//
90                      System.out.println("\nThe products you have chosen costs "+ sum + ".");
91                    }
92
93                    }
```

Console 1:

```
ONLINE STORE


Choose your required products and quantities from the given list of items given below:
    PRODUCT                  PRICE
1.Matcha Powder          32 per 1 bottle powder
2.Lipstain               12 per bottle
3.Rice                   70 per Kilogram
4.Bread                  10 per bread
Exit by typing any number(from 5) or q or exit

Enter your option:
2
Enter the number of bottles of Lipstain you want to buy:
15

Enter your option:
4
Enter the number of breads you want to buy:
8

Enter your option:
5
Products are over and you are done buying and thank you for choosing the products!

The products you have chosen costs 260.
```

Console 2:

```
ONLINE STORE


Choose your required products and quantities from the given list of items given below:
    PRODUCT                  PRICE
1.Matcha Powder          32 per 1 bottle powder
2.Lipstain               12 per bottle
3.Rice                   70 per Kilogram
4.Bread                  10 per bread
Exit by typing any number(from 5) or q or exit

Enter your option:
4
Enter the number of breads you want to buy:
100

Enter your option:
4
Enter the number of breads you want to buy:
2

Enter your option:
3
Enter the number of kilograms you want to buy:
12

Enter your option:
5
Products are over and you are done buying and thank you for choosing the products!

The products you have chosen costs 1860.
```

Task 4 :

Explanation:
The program tally in a write in election and announces the winner.
Declaring two arraylists for writing the candidate names and their votes or count as mentioned in the program and a scanner class for the user to input names

Printing title:
Print() method has been used to print required details for the user to know.

Declaring Variables:
Person is the variable for the user to input in the place of arraylist 'names' and int i,j,k,l has been given for the votes to add in and to find the maximum votes for person.

While Loop:
Once the user starts voting since there is no pre-determined set of candidates and they can end it by typing exit that is while ((person.equals("exit")== false) .

For Loop and If:
After the votes add in it will start setting the names to be equal to person by adding 1 and returning them to for loop and so on the for loops and if else is repeated at the end the most votes are calculated and the winner is announced.

Pseudo code:

```
Start
    DECLARE
            ArrayList <String> names = new ArrayList <String> ()
            ArrayList <Integer> count = new ArrayList <Integer> ()
            Scanner input = new Scanner (System.in)


   PRINT "Election Votes";
            "\n";
            "########################################";
            "#Enter the votes, one vote per line. #";
            "#End with -1.                       #";
            "########################################";
            "\n";
   DECLARE
     String person;
     int m;
     int n;
     int k =0;
     int l= 0;

WHILE LOOP
while (person.equals("-1") == false)
   INPUT person
    m =0;

FOR LOOP
for (int j=0; j<names.size(); j++)

IF LOOP
if(names.get(j).equals(person))
 GET
    n = count.get(j)+1;
SET
   count.set(j,n);
   m=1;

if(m==0 && person.equals("-1")==false)
  ADD
   names.add(person);
```

```
  ADD
   count.add(1);

FOR LOOP
 for(int j=0; j < names.size(); j++){
   PRINT "\n"
   PRINT (names.get(j) + " received " + count.get(j) + " votes ")

         if (count.get(j) > l)
         l = count.get(j);
         k = j;
  PRINT( "----------------------------------------")
  PRINT ("The Winner is " + names.get(k) + " with " + count.get(k) + " votes ")

       END WHILE LOOP
    END FOR

  END IF

 END FOR

EXIT
```

Source Code:

```
//Name: Thaspeeha Vahithu, Instructor Name: Ms.Shibili Said, Student Id:
32146925//
//The below program computes the tally in a write-in election and announces the
winner//

import java.util.Scanner; //Import for Scanner class//
import java.util.ArrayList; //Import for arraylist//
public class Election //Defining class//
{
    public static void main (String[] args)//Main method//
    {
//Declaring two arraylists for String and Integer//
        ArrayList <String> names = new ArrayList <String> ();
        ArrayList <Integer> count = new ArrayList <Integer> ();

        //Scanner class for the user to input values//
        Scanner input = new Scanner (System.in);

        //Prints required details for the user to know//
        System.out.println("Election Votes");
        System.out.println("\n");
        System.out.println("########################################");
        System.out.println("#Enter the votes, one vote per line. #");
        System.out.println("#End with -1.                        #");
        System.out.println("########################################");
        System.out.println("\n");

        //Declaring the variables//
        String person = "";
        int m;
        int n;
        int k = 0;
        int l = 0;

 //While condition for writing votes per line//
 while (person.equals("-1")== false){
    person = input.nextLine();
    m =0;

//Starts the loop when user input votes each line//
for (int j=0; j<names.size(); j++){
    if(names.get(j).equals(person)){  //Names equal to user's values for person//
    n = count.get(j)+1;  //return the value to n and increment of votes//
```

```
    count.set(j,n);  //set the value to j//
    m=1;
    }
    }
    if(m==0 && person.equals("-1")==false){
    names.add(person); //Adds the user input values to names//
    count.add(1); //the program adds 1 to the value of the element in count.//
        }
        }
  //Prints how many votes have been recieved//
for(int j=0; j < names.size(); j++){
     System.out.println("\n");
     System.out.println(names.get(j) + " received " + count.get(j) + " votes ");

   //To find who got the most votes and announce the winner//
   if (count.get(j) > l){
     l = count.get(j);
     k = j; // Setting the votes in this variable//
        }
        }

  //Prints the winner//
System.out.println("----------------------------------------");
System.out.println("The Winner is " + names.get(k) + " with " + count.get(k) + "
votes ");
        }
    }
```

Program Code:

```
1  //Name: Thaspeeha Vahithu, Instructor Name: Ms.Shibili Said, Student Id: 32146925//
2  //The below program computes the tally in a write-in election and announces the winner//
3
4  import java.util.Scanner; //Import for Scanner class//
5  import java.util.ArrayList; //Import for arraylist//
6  public class Election //Defining class//
7  {
       Run | Debug
8      public static void main (String[] args)//Main method//
9      {
10         //Declaring two arraylists for String and Integer//
11         ArrayList <String> names = new ArrayList <String> ();
12         ArrayList <Integer> count = new ArrayList <Integer> ();
13
14         //Scanner class for the user to input values//
15         Scanner input = new Scanner (System.in);
16
17         //Prints required details for the user to know//
18         System.out.println(x:"Election Votes");
19         System.out.println(x:"\n");
20         System.out.println(x:"########################################");
21         System.out.println(x:"#Enter the votes, one vote per line. #");
22         System.out.println(x:"#End with exit.                       #");
23         System.out.println(x:"########################################");
24         System.out.println(x:"\n");
25
26         //Declaring the variables//
27         String person = "";
28         int m;
29         int n;
30         int k = 0;
31         int l = 0;
32
33         //While condition for writing votes per line//
34         while (person.equals(anObject:"exit")== false){
35             person = input.nextLine();
36             m =0;
37
38         //Starts the loop when user input votes each line//
39         for (int j=0; j<names.size(); j++){
40             if(names.get(j).equals(person)){ //Names equal to user's values for person//
41             n = count.get(j)+1; //return the value to variable n and increment when votes add in//
42             count.set(j,n); //set the value of n to j//
43             m=1;
44             break;
45         }
46         }
47            if(m==0 && person.equals(anObject:"exit")==false){
48            names.add(person); //Adds the user input values to names//
49            count.add(e:1); //the program adds 1 to the value of the element in count.//
50         }
51         }
```

```
52          //Prints how many votes have been recieved//
53          for(int j=0; j < names.size(); j++){
54              //System.out.println("\n");
55              System.out.println(names.get(j) + " received " + count.get(j) + " vote ");
56
57          //To find who got the most votes and announce the winner//
58              if (count.get(j) > l){
59              l = count.get(j);
60              k = j; // Setting the votes in this variable//
61          }
62
63          }
64          //Prints the winner//
65              System.out.println(x:"-----------------------------------------");
66              System.out.println("The Winner is " + names.get(k) + " with " + count.get(k) + " votes "); //Prints the winner//
67          }
68      }
```

Console:

```
Election Votes


########################################
#Enter the votes, one vote per line. #
#End with exit.                       #
########################################


jess
sam
carl
carl
carl
jack
jack
sam
exit
jess received 1 vote
sam received 2 vote
carl received 3 vote
jack received 2 vote
-----------------------------------------
The Winner is carl with 3 votes
```

Task 5:

Explanation:

The program is based on Object Oriented Programming to calculate interest, balance and the date when the account was created.

Declaring the variables:

Its is to print the id, balance, annual interest rate and to import date when the account is created with the access modifier private.

A new object for the date to get created when the account is created.

Parameterised constructor:

- Account(int id, double balance)
- dateCreated = new java.util.Date();

This constructor is parameterised to easily recognize the attributes.

Set attibutes:

- setId(int id)
- setBalance(double balance)
- setAnnualInterestRate(double newannualInterestRate)

It's to set the attributes since they are using access modifier private.

Get attributes:

- getId()
- getBalance()
- getAnnualInterestRate()

Its to return the values of the attributes.

- getMonthlyInterestRate()

It returns the monthly interest using the annual interest rate.

- getDateCreated()

Returns the date.

- Withdraw(double amount)
- balance -= amount;

When withdrawn from the balance it subtracts the withdrawing amount from the initial balance.

- Deposit(double amount)
- balance += amount;

When deposited from the balance it adds the deposited amount to the initial balance.

There is also another program to implement the class Account

->Account myaccount = new Account(id, balance)

It first sets the values and then use get to return the values of id, balance and monthly interest rate.

Pseudocode:

Start

DECLARE

int id;

double balance;

double annualInterestRate;

import java.util.Date datecreated;

Account()

NEW OBJECT

dateCreated = new java.util.Date();

CONSTRUCTOR

Account(int id, double balance)
dateCreated = new java.util.Date();

SET
setId
setBalance
setAnnualInterestRate

GET
getId
getBalance
getAnnualInterestRate

RETURN (Id, Balance, AnnualInterestRate)

getMonthlyInterestRate
RETURN (annualInterestRate/12)*(balance/100)

getDateCreated
RETURN dateCreated;

Withdraw(double amount)
balance -= amount;

```
Deposit(double amount)
   balance += amount;
      END GET
    END SET
   END WITHDRAW
  END DEPOSIT
EXIT
```

Source code:

```
//Name: Thaspeeha Vahithu, Instructor Name: Ms.Shibili Said, Student Id:
32146925//
//The below program is based on Object Oriented Programming to calculate
interest, balance and the date when the account was created//

//Listing the properties of class Account using access modifier private to use
set and get methods//
class Account {
private int id;
private double balance;
private static double annualInterestRate;
private java.util.Date dateCreated;

//New object stores the date when the ac
count was created//
public Account() {
dateCreated = new java.util.Date();
}

//A parameterised constructor displays Id and balance for the current account//
public Account(int id, double balance) {
this.id = id;
this.balance = balance;
dateCreated = new java.util.Date();
}

//Mutators used since the attributes are private which are used to change the
current values//
public void setId(int id) {
this.id = id;
}
public void setBalance(double balance) {
this.balance = balance;
}
public static void setAnnualInterestRate(double newannualInterestRate) {
annualInterestRate = newannualInterestRate;
}

//Accessors used since the attributes are private which are used to return the
values//
public int getId() {
return this.id;
}
public double getBalance() {
```

```
return this.balance;
}
public double getAnnualInterestRate() {
return annualInterestRate;
}

//To calculate and return monthly interest rate//
public double getMonthlyInterestRate() {
return (annualInterestRate/12)*(balance/100);
}

//To return the date when the account is created//
public java.util.Date getDateCreated() {
return dateCreated;
}
//To calculate the balance or amount when withdrawn//
public void withdraw(double amount) {
balance -= amount;
}
//To calculate the balance or amount when deposited//
public void deposit(double amount) {
balance += amount;
}
}
```

Program code:

```
1   //Name: Thaspeeha Vahithu, Instructor Name: Ms.Shibili Said, Student Id: 32146925//
2   //The below program is based on Object Oriented Programming to calculate interest,balance and the date when the account was created//
3
4   //Listing the properties of class Account using access modifier private to use set and get methods//
5   class Account {
6   private int id;
7   private double balance;
8   private static double annualInterestRate;
9   private java.util.Date dateCreated;
10
11  //New object stores the date when the account was created//
12  public Account() {
13  dateCreated = new java.util.Date();
14  }
15
16  //A parameterised constructor displays Id and balance for the current account//
17  public Account(int id, double balance) {
18  this.id = id;
19  this.balance = balance;
20  dateCreated = new java.util.Date();
21  }
22
23  //Mutators used since the attributes are private which are used to change the current values//
24  public void setId(int id) {
25  this.id = id;
26  }
27  public void setBalance(double balance) {
28  this.balance = balance;
29  }
30  public static void setAnnualInterestRate(double newannualInterestRate) {
31  annualInterestRate = newannualInterestRate;
32  }
33
34  //Accessors used since the attributes are private which are used to return the values//
35  public int getId() {
36  return this.id;
37  }
38  public double getBalance() {
39  return this.balance;
40  }
41  public double getAnnualInterestRate() {
42  return annualInterestRate;
43  }
44
44
45  //To calculate and return monthly interest rate//
46  public double getMonthlyInterestRate() {
47  return (annualInterestRate/12)*(balance/100);
48  }
49
50  //To return the date when the account is created//
51  public java.util.Date getDateCreated() {
52  return dateCreated;
53  }
54  //To calculate the balance or amount when withdrawn//
55  public void withdraw(double amount) {
56  balance -= amount;
57  }
58  //To calculate the balance or amount when deposited//
59  public void deposit(double amount) {
60  balance += amount;
61  }
62  }
```

UML Class Diagram:

Account
private int id private double balance private double annualInterestRate import java.util.Date dateCreated
public Account() public Account(int id, double balance) public void set(int id, double balance, double double newannualInterestRate) public void setId (set id) public void setBalance (set balance) public static void setAnnualInterestRate (set newannualInterestRate) public int getId (get id) public double getBalance (get balance) public doube getAnnualInterestRate (get annualInterestRate) public double getMonthlyInterestRate (get monthlyInterestRate) public java.util.Date getDateCreated (getdateCreated) public void withdraw (double amount) public void deposit(double amount)

Source code for the implementation of class Account:

```
//This program is written to implement the class Account//
public class AccountTest //Defining the class//
{
    public static void main (String[] args) //Defining the main method//
    {
        //A new object for significant testing the class Account//
        Account myaccount = new Account(1122, 20000);
        System.out.println("Your Account.");
        System.out.println("Account ID: 1122");
        System.out.println("Balance: $20,000");
        System.out.println("Annual Interest Rate: 4.5%\n");
        Account.setAnnualInterestRate(4.5);
        myaccount.withdraw(2500);
        myaccount.deposit(3000);
        System.out.println("Withdrawing $2500 and depositing $3000");
        System.out.println("Balance is $" + myaccount.getBalance());
        System.out.println("Monthly interest is " +
myaccount.getMonthlyInterestRate() + "%");
```

```
        System.out.println("This account was created at " +
myaccount.getDateCreated());
}
}
```

Implementation of the class Account:

```
1   //This program is written to implement the class Account//
2   public class AccountTest //Defining the class//
3   {
        Run | Debug
4       public static void main (String[] args) //Defining the main method//
5       {
6           //A new object for significant testing the class Account//
7           Account myaccount = new Account(id:1122, balance:20000);
8           System.out.println(x:"Your Account.");
9           System.out.println(x:"Account ID: 1122");
10          System.out.println(x:"Balance: $20,000");
11          System.out.println(x:"Annual Interest Rate: 4.5%\n");
12          Account.setAnnualInterestRate(newannualInterestRate:4.5);
13          myaccount.withdraw(amount:2500);
14          myaccount.deposit(amount:3000);
15          System.out.println(x:"Withdrawing $2500 and depositing $3000");
16          System.out.println("Balance is $" + myaccount.getBalance());
17          System.out.println("Monthly interest is " + myaccount.getMonthlyInterestRate() + "%");
18          System.out.println("This account was created at " + myaccount.getDateCreated());
19  }
20  }
```

Console:

```
Your Account.
Account ID: 1122
Balance: $20,000
Annual Interest Rate: 4.5%

Withdrawing $2500 and depositing $3000
Balance is $20500.0
Monthly interest is 76.875%
This account was created at Tue Jan 02 14:46:17 GST 2024
```