**Assignment 1**

1

Accept i/ps from User , till user enters "quit" or any other option.

I/P : operation(add|sub|mult|div) , number1(double) , number2(double)

Display the result of the operation.

It should be done in a loop , till user enters "quit"

import java.util.Scanner;

class Arithmatic

{

 public static void main(String[] args)

{

        int choice=0;

         double no1;

         double no2;

         boolean flag=true;

         Scanner sc=new Scanner(System.in);

            System.out.println("Which operation would you like to perform:");

            System.out.println("1:Addition +");

            System.out.println("2:Subtraction -");

            System.out.println("3:Multiplication \*");

            System.out.println("4:Division /");

            System.out.println("5:Quit");

            while(flag!=false)

            {

                System.out.println("Enter choice");

                choice=sc.nextInt();

                System.out.println("Enter two double no");

                  no1=sc.nextDouble();

                 no2=sc.nextDouble();

            switch(choice)

            {

            case 1:

            double addition=no1 + no2;

            System.out.println("Addition" +addition);

            break;

            case 2:

            double subtraction=no1 - no2;

            System.out.println("Subtraction" +subtraction);

            break;

            case 3:

            double multiplication=no1 \* no2;

            System.out.println("Multiplication" +multiplication);

            break;

            case 4:

            double division=no1 / no2;

            System.out.println("Division" +division);

            break;

            case 5:

            System.out.println("Quit");

            break;

            default :

            System.out.println("Enter valid option:");

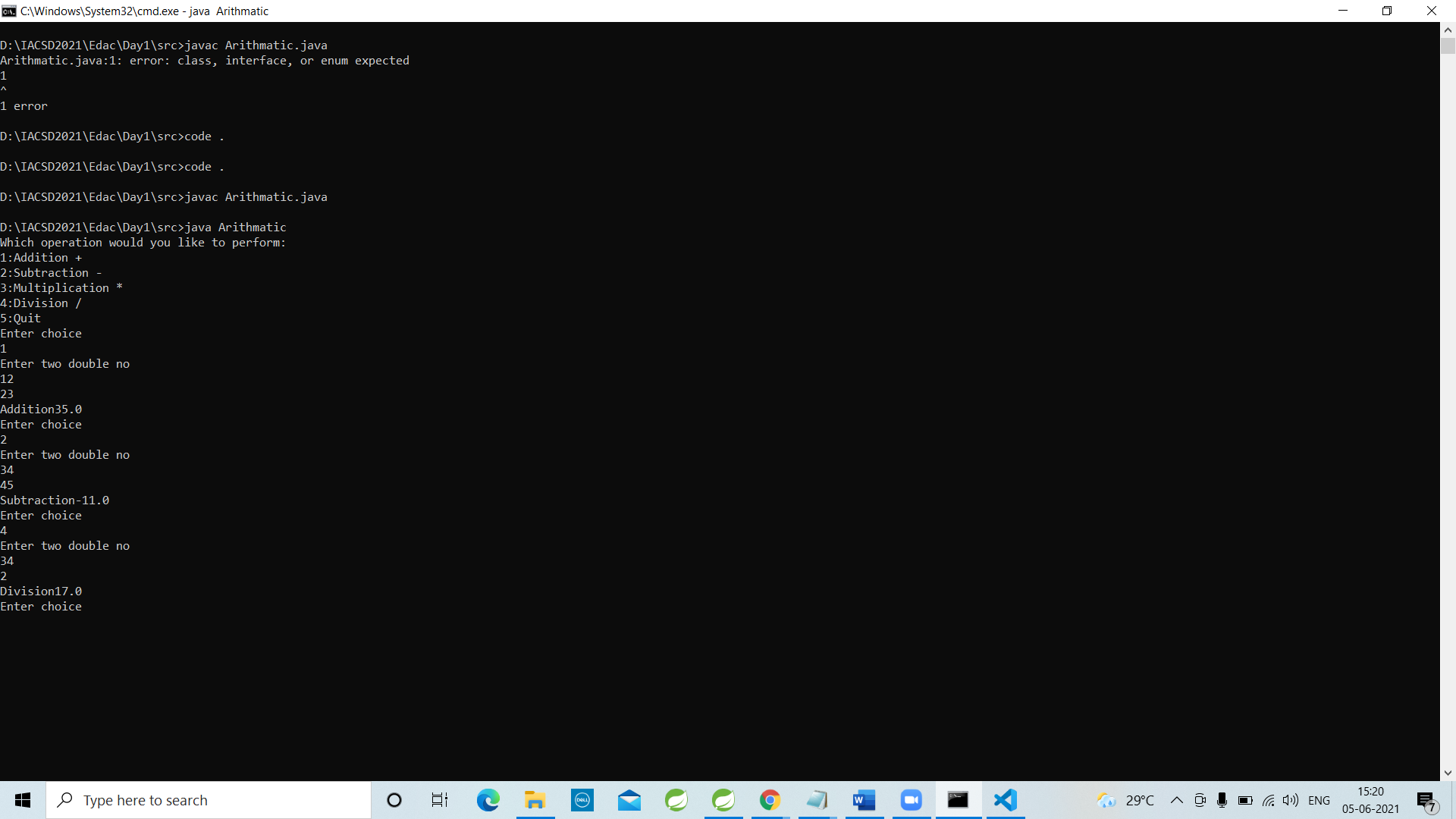
            }

            }

}

}

**Output:**



2. Accept 2 double values from User (using Scanner). Check data type.

. If arguments are not doubles , supply suitable error message & terminate.

If numbers are double values , print its average.

eg :

sop("Enter 2 double nos");

boolean flag=sc.hasNextDouble();

if(flag)

{

//read off the 1st token

double d1=sc.nextDouble();

if(....)

{

double d2=sc.nextDouble();

sop("Avg+....);

}

else {...}

}

else {...}

import java.util.Scanner;

public class Double {

    public static void main(String args[]){

        double  num1,num2,avg;

        //Initialize the scanner

        Scanner scan = new Scanner(System.in);

        System.out.println("Enter First no");

        //Iterate through the tokens

        if(scan.hasNextDouble())

        {

                   num1=scan.nextDouble();

                   System.out.println("Enter Second no");

                   if(scan.hasNextDouble())

                   {

                    num2=scan.nextDouble();

                    avg=(num1+num2)/2;

                    System.out.println("Average of two number is " +avg);

                    }

                     else

                    {

                      System.out.println("error");

                    }

        }

        else

        {

            System.out.println("error");

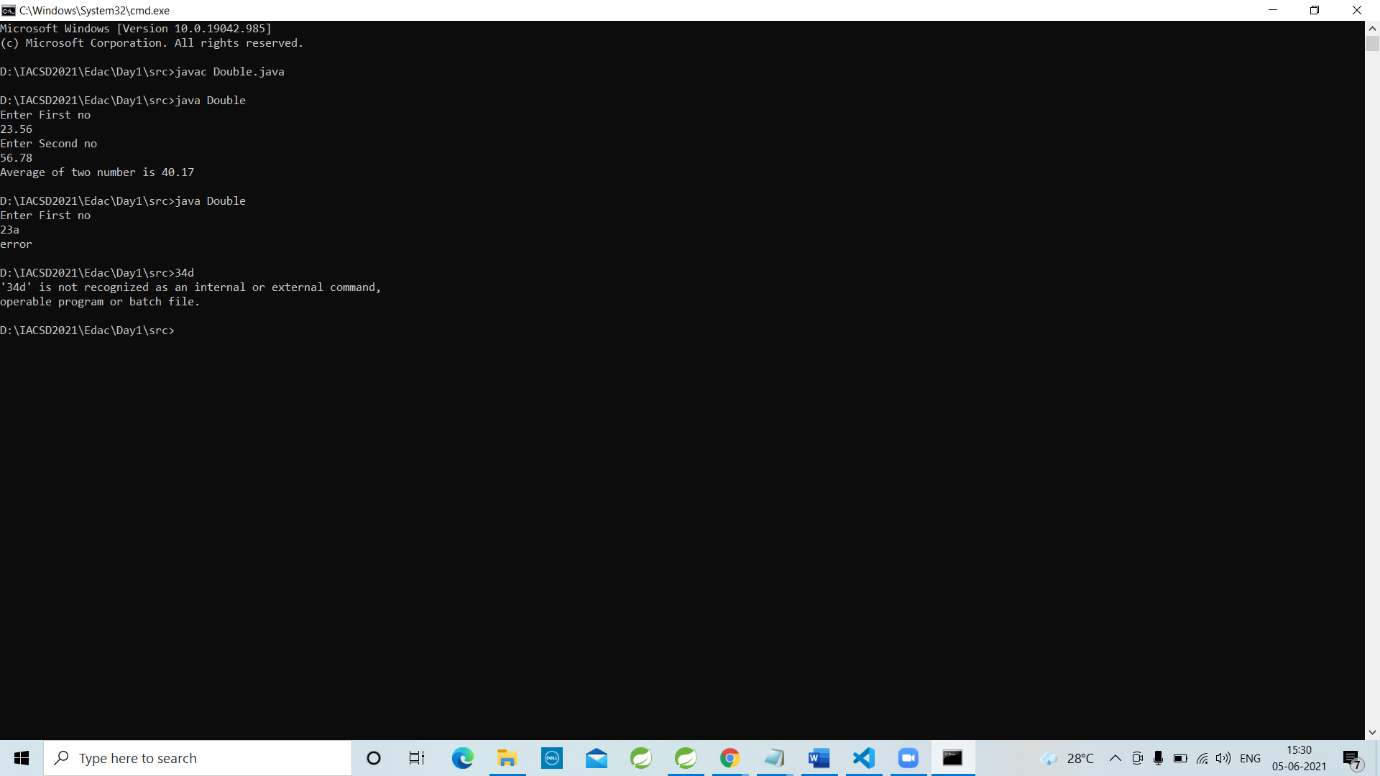
        }

        scan.close();

    }

}

**Output:**



3. Display food menu to user. User will select items from menu along with the quantity. (eg 1. Dosa 2. Samosa .......10 . Generate Bill ) Assign fixed prices to food items

When user enters 'Generate Bill' option(10) , display total bill & exit.

import java.util.Scanner;

class Food

{

 public static void main(String[] args)

{

        int choice=0;

         int total=0;

         boolean flag=true;

         Scanner sc=new Scanner(System.in);

            System.out.println("What you would you like to order:");

            System.out.println("1:Dosa       50/-");

            System.out.println("2:Samosa     20/-");

            System.out.println("3:Idali      40/-");

            System.out.println("4:Vada       15/-");

            System.out.println("5:Poha       20/-");

            System.out.println("6:Uttapa     40/-");

            System.out.println("7:Tea        30/-");

            System.out.println("8:Juice      50/-");

            System.out.println("9:Parathe     30/-");

            System.out.println("10:Generate Bill");

while (flag!=false)

    {

            System.out.println("Enter choice:");

            choice=sc.nextInt();

    switch(choice)

            {

            case 1:

            System.out.println("Enter quantity of Dosa:");

            int Dosa =sc.nextInt();

            total=total+(Dosa\*50);

            break;

                case 2:

            System.out.println("Enter quantity of Samosa:");

            int Samosa=sc.nextInt();

            total=total+(Samosa\*20);

            break;

            case 3:

            System.out.println("Enter quantity of Idali:");

            int Idali=sc.nextInt();

            total=total+(Idali\*40);

            break;

            case 4:

            System.out.println("Enter quantity of Vada:");

            int Vada =sc.nextInt();

            total=total+(Vada\*15);

            break;

            case 5:

            System.out.println("Enter quantity of Poha:");

            int Poha =sc.nextInt();

            total=total+(Poha\*20);

            break;

            case 6:

            System.out.println("Enter quantity of Uttapa:");

            int Uttapa=sc.nextInt();

            total=total+(Uttapa\*40);

            break;

            case 7:

            System.out.println("Enter quantity of Tea:");

            int Tea =sc.nextInt();

            total=total+(Tea\*30);

            break;

            case 8:

            System.out.println("Enter quantity of Juice:");

            int Juice =sc.nextInt();

            total=total+(Juice\*50);

            break;

            case 9:

            System.out.println("Enter quantity of Parathe:");

            int Parathe=sc.nextInt();

            total=total+(Parathe\*30);

            break;

            case 10:

            flag =false;

            break;

            default :

            System.out.println("Enter valid option:");

   }

}

if(total!=0)

{

        System.out.println("Total cost : Rs. " +total);

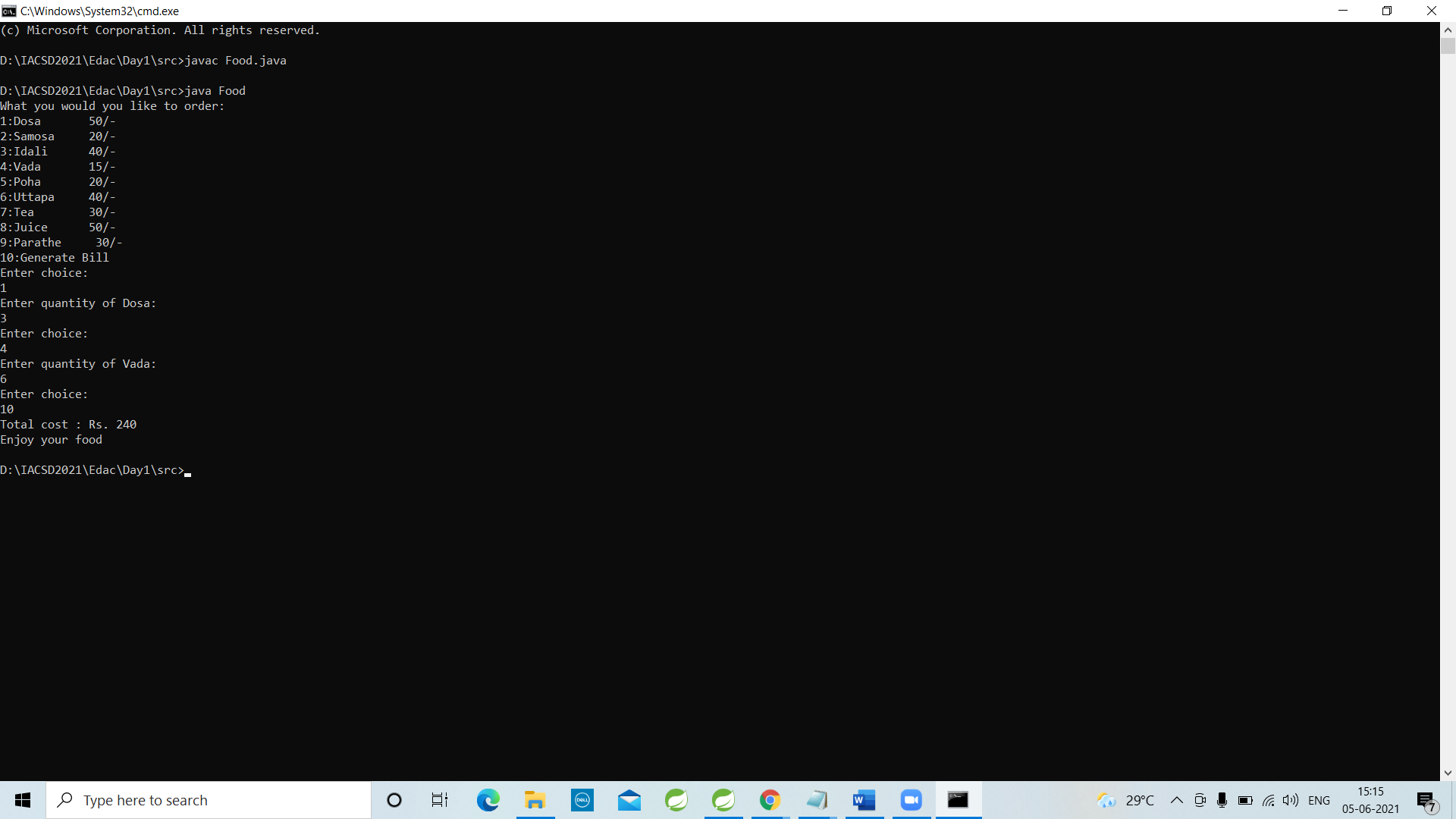
        System.out.println("Enjoy your food");

}

        }

        }

**Output:**



**Assignment 2**

Day2

1. Do complete revision of class work & then solve today's assignemnts

2.

Solve Tank assignment along with memory picture.

What will be the output ?

class Tank {

private int level;

Tank(int l)

{

level=l;

}

public void setLevel(int level1)

{

level=level1;

}

public int getLevel()

{

return level;

}

}

public class Assignment {

public static void main(String[] args) {

Tank t1 = new Tank(10);

Tank t2 = new Tank(20);

s.o.p("1: t1.level: " + t1.getLevel() +

", t2.level: " + t2.getLevel());

t1 = t2;//1st tank obj is marked GC

s.o.p("2: t1.level: " + t1.getLevel() +

", t2.level: " + t2.getLevel());

t1.setLevel(27);

s.o.p("3: t1.level: " + t1.getLevel() +

", t2.level: " + t2.getLevel());

t2.setLevel(t1.getLevel()+10);

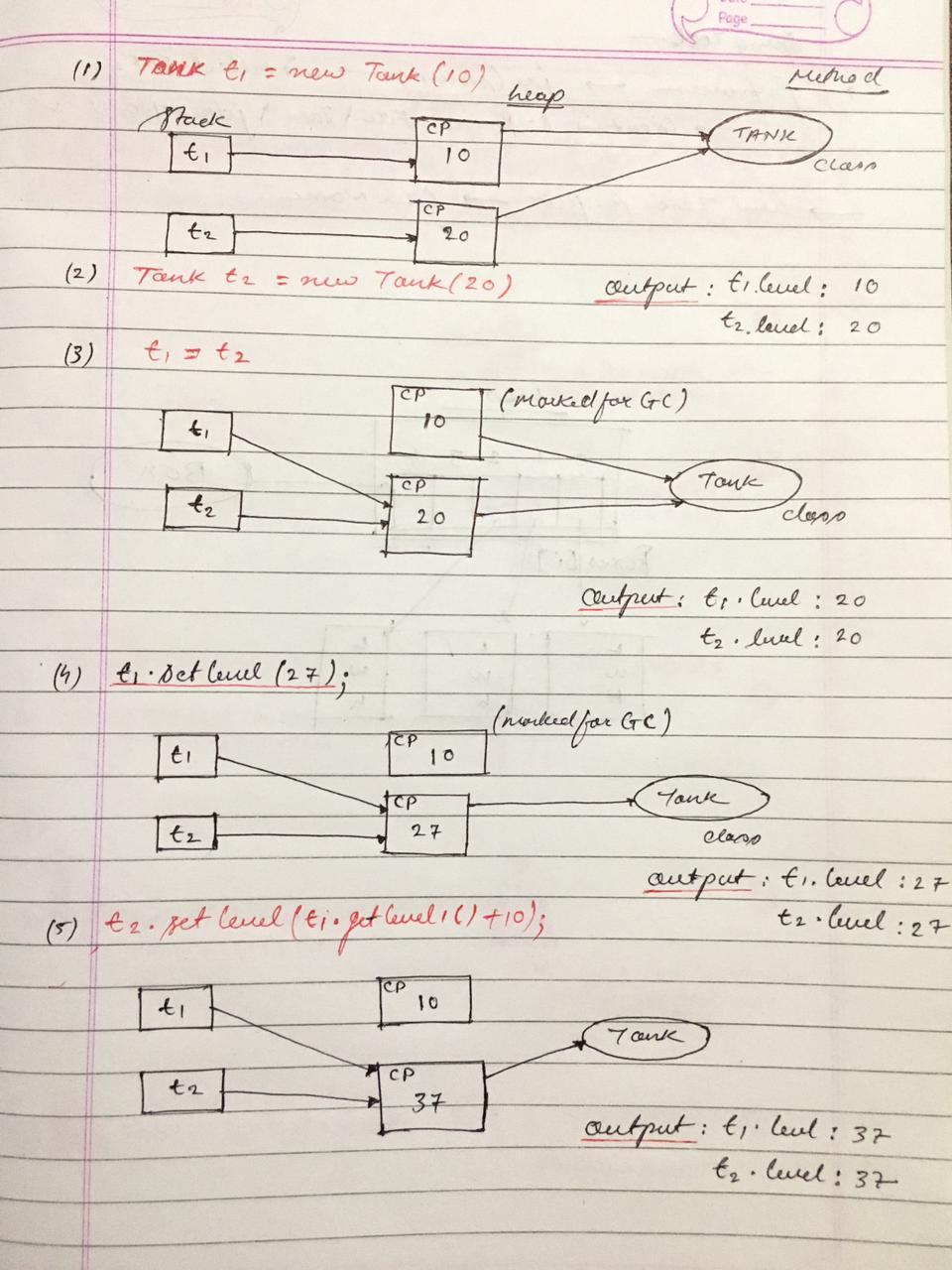
s.o.p("4: t1.level: " + t1.getLevel() +

", t2.level: " + t2.getLevel());

}

}

**Output:**



3. Solve

Create Point class Point2D : for representing a point in x-y co-ordinate system.

3.1 Create a parameterized constructor to accept x & y co-ords.

3.2 Add public String getDetails()) --to return string form point's x & y co-ords

eg : public String getDetails()

{

return "Point ("+x+","+y+")";

}

3.3 Add isEqual method to Point2D class : boolean returning method : must return true if both points are having same x,y co-ords or false otherwise.

eg : p1 , p2

p1.isEqual(p2) --> true/false

Method in Point2D class

boolean isEqual(Point2D anotherPoint)

{...}

3.4 Add a method to Point2D class -- to create and return new point having given x & y offset.

eg : Point2D p1=new Point2D(10,20);

Point2d p3=p1.createNewPoint(5,-2);//p3 : 15,18

If user supplies offset of (5, -2) : your method should return a new point object placed at (15,18)

eg : Point2D createNewPoint(int xOff,iny yOff){

Point2D newPoint=new Point2D(this.x+xOff,this.y+yOff);

return newPoint;

}

3.5 Add calculateDistance method to calculate distance between current point & specified point & return the distance to the caller.

(eg double calcDistance(Point2D anotherPoint))

Hint : Use distance formula . Use java.lang.Math class methods --sqrt, pow etc.

3.6 Write TestPoint class , with a main method

Accept co ordinates of 2 points from user (Scanner) --p1 & p2

3.7 Use getDetails method to display point details.(p1's details & p2's details)

3.8 Invoke isEqual & display if points are same or different (i.e p1 & p2 are located at the same position)

3.9 Create new point p3 , with the dimensions offset from p1.

input --x offset & y offset

eg : p1 : 50,100

Point2D p1=new Point2D(sc.nextInt(),sc.nextInt());//50,100

sop("Enter x off n y offset);

Point2D p3=p1.createNewPoint(sc.nextInt(),sc.nextInt());// 5 -3

//p3 : 55 97

3.10 Display distance between 2 points.(between p1 & p2

import java.lang.Math;

class Point2D

{

    private int x,y;

    //parameterized constructor

    Point2D(int x,int y )

    {

        this.x=x;

        this.y=y;

    }

    //return co-ordinate details in String form

    public String getdetails()

    {

        return "Co-ordinates:" + this.x + " " + this.y ;

    }

    //check equality of 2 point co-ordinate

    boolean isEqual(Point2D  anotherPoint)

    {

        return this.x == anotherPoint.x && this.y == anotherPoint.y;

    }

    //offset values method

    Point2D createNewPoint(int xOff,int yOff)

    {

        Point2D NewPoint=new Point2D(this.x + xOff ,this.y + yOff);

        return NewPoint;

    }

    double calculateDistance(Point2D p2)

    {

        double diff1 = p2.x - this.x;

        double diff2 = p2.y - this.y;

        double pow1 = Math.pow(diff1,2);

        double pow2 =Math.pow(diff2,2);

        double sum= pow1 +pow2;

        double squareroot = Math.sqrt(sum);

        return squareroot;

    }

}

import java.util.Scanner;

class TestPoint2D

{

    public static void main(String[] args)

    {

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter the first point coordinate(x.y):" );

        Point2D p1=new Point2D(sc.nextInt(),sc.nextInt());

          System.out.println("Enter the second point coordinate(x.y):" );

          Point2D p2=new Point2D(sc.nextInt(),sc.nextInt());

           System.out.println(p1.isEqual(p2) ? "Same Co-ordinate point" : "Different Co-ordinate point " );

            System.out.println("Enter offset of 1st C0-ordinate points" );

             Point2D p3 =p1.createNewPoint(sc.nextInt(),sc.nextInt());//b1 value is add in b3 new offset value

            System.out.println(p1.getdetails() );

              System.out.println(p3.getdetails() );

        //Distance between two points p1 and p2

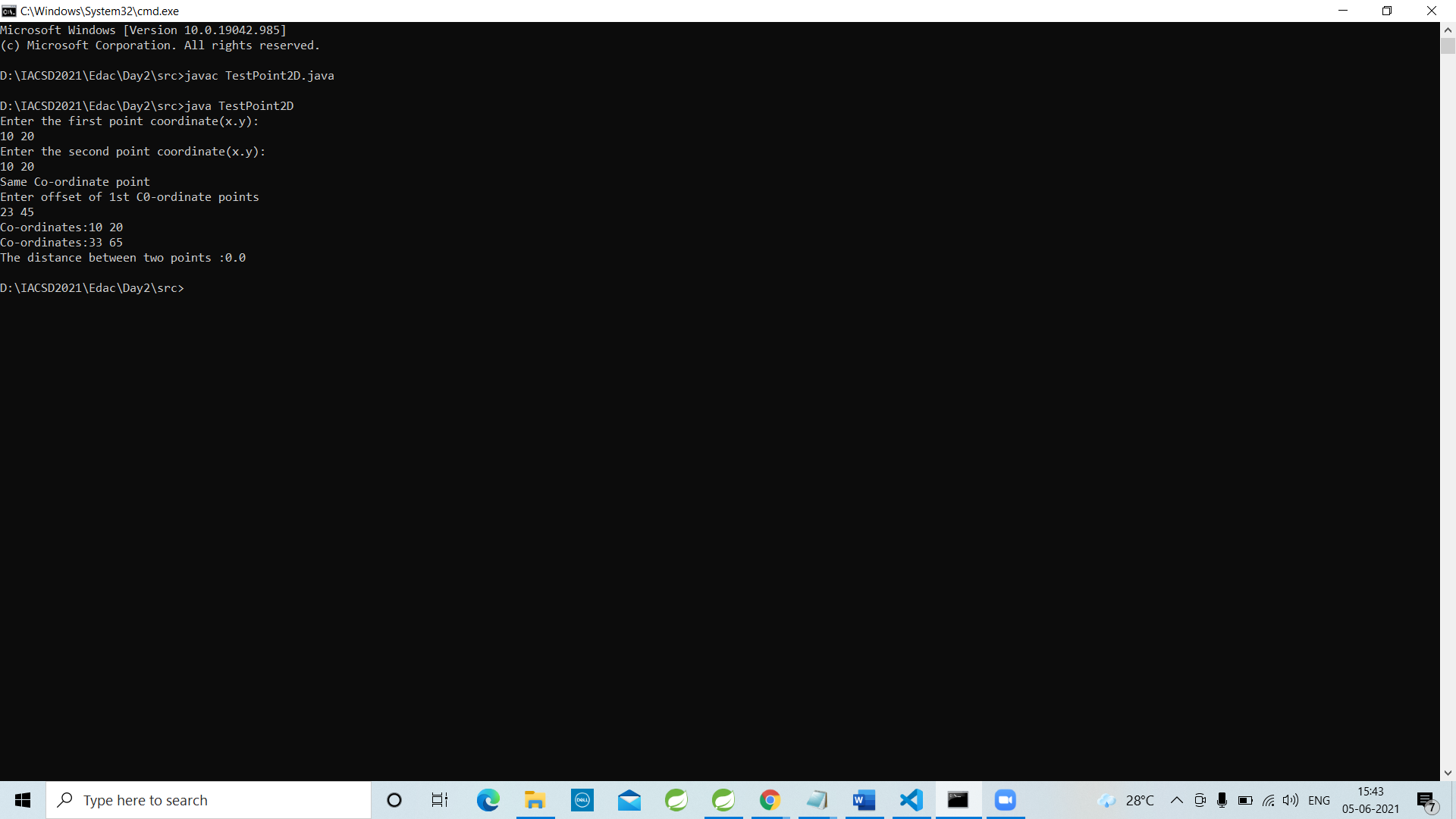
        System.out.println("The distance between two points :" + p1.calculateDistance(p2) );

        sc.close();

    }

}

**Output:**



**Assignment 3**

Day 3

1. Revise entire class work.

1.5

Copy earlier created Point2D class n place it under the package "com.geometry"

Copy earlier created TestPoint class n place it under "tester" package.

Solve compiler errors n run it without IDE (set CLASSPATH , so that you can run it from anywhere)

eg : day3\_lab/src

javac -d ..\bin tester\TestPoint.java

cd ..\bin

java tester.TestPoint

import java.lang.Math;

class Point2D

{

    private int x,y;

    //parameterized constructor

    Point2D(int x,int y )

    {

        this.x=x;

        this.y=y;

    }

    //return co-ordinate details in String form

    public String getdetails()

    {

        return "Co-ordinates:" + this.x + " " + this.y ;

    }

    //check equality of 2 point co-ordinate

    boolean isEqual(Point2D  anotherPoint)

    {

        return this.x == anotherPoint.x && this.y == anotherPoint.y;

    }

    //offset values method

    Point2D createNewPoint(int xOff,int yOff)

    {

        Point2D NewPoint=new Point2D(this.x + xOff ,this.y + yOff);

        return NewPoint;

    }

    double calculateDistance(Point2D p2)

    {

        double diff1 = p2.x - this.x;

        double diff2 = p2.y - this.y;

        double pow1 = Math.pow(diff1,2);

        double pow2 =Math.pow(diff2,2);

        double sum= pow1 +pow2;

        double squareroot = Math.sqrt(sum);

        return squareroot;

    }

}

import java.util.Scanner;

class TestPoint2D

{

    public static void main(String[] args)

    {

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter the first point coordinate(x.y):" );

        Point2D p1=new Point2D(sc.nextInt(),sc.nextInt());

          System.out.println("Enter the second point coordinate(x.y):" );

          Point2D p2=new Point2D(sc.nextInt(),sc.nextInt());

           System.out.println(p1.isEqual(p2) ? "Same Co-ordinate point" : "Different Co-ordinate point " );

            System.out.println("Enter offset of 1st C0-ordinate points" );

             Point2D p3 =p1.createNewPoint(sc.nextInt(),sc.nextInt());//b1 value is add in b3 new offset value

            System.out.println(p1.getdetails() );

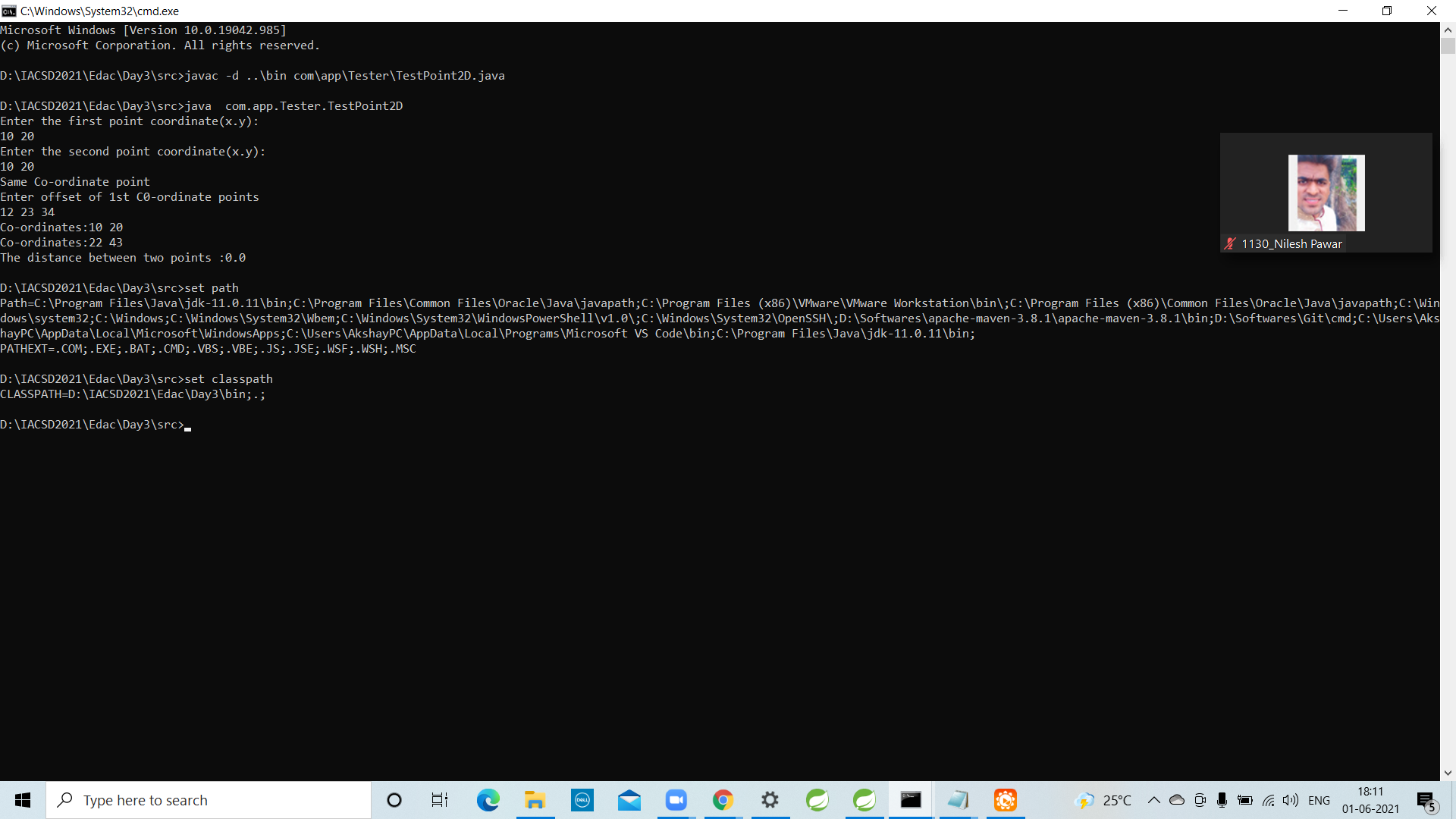
              System.out.println(p3.getdetails() );

        //Distance between two points p1 and p2

        System.out.println("The distance between two points :" + p1.calculateDistance(p2) );

        sc.close();

    }



2. Solve

Create new eclipse project "day3\_assgnment"

Copy Point2D class n TestPoints classalong with the packages in IDE.

2.1 Create a new tester in package "com.tester" : TestPointArray

Prompt user , how many points to plot?

Create suitable array , to store Point2D type of references.

eg : Point2D[] points=new Point2D[sc.nextInt()];//10 --- 1 obj --filled with 10 nulls

2.2 Add a menu , Run the application till user chooses option 10 (exit)

1 Plot a new point

I/P --index , x & y

eg : If user supplies 5 50 100

Create a Point2D with x, y of (50,100) & its reference should be stored at the 5th index position in array.

eg : case 1 : sop("Enter index x & y");

read index , x, y

boundary checking (0---points.length-1)

points[index]=new Point2D(x,y);

break;

2 Display all points plotted so far. (Use for-each)

Will null checking be required? : YESSS

eg : for(Point2D p : points)//p=points[0]

if(p != null)

sop(p.getDetails());

3. Test equality of 2 points

I/P : index1 , index 2

eg If user enters 1 7 , you have to check if point at 1st n 7th index is same.

If same , print "SAME" , otherwise print "DIFFERENT"

4. Calculate distance

I/P strt , end point indexes.

eg : If user enters 2 6

Find out the distance between 2nd & 6th point. (array indexes start from 0)

10. Exit

package com.tester;

import java.lang.Math;

public class Point2D

{

private int x,y;

//parameterized constructor

Point2D( int x,int y )

{

this.x=x;

this.y=y;

}

//return co-ordinate details in String form

public String getdetails()

{

return "Co-ordinates:" + this.x + " " + this.y ;

}

//check equality of 2 point co-ordinate

public boolean isEqual(Point2D anotherPoint)

{

return this.x == anotherPoint.x && this.y == anotherPoint.y;

}

//offset values method

public Point2D createNewPoint(int xOff,int yOff)

{

Point2D NewPoint=new Point2D(this.x+xOff ,this.y+yOff);

return NewPoint;

}

public double calculateDistance(Point2D P2)

{

double d1=Math.pow((P2.x-this.x),2);

double d2=Math.pow((P2.y-this.y),2);

d2=d2+d1;

return Math.sqrt(d2);

}

}

package com.tester;

import java.util.Scanner;

import com.tester.Point2D;//import java.awt.geom.Point2D;import java.util.Arrays;import java.util.Scanner;

public class TestPointArrAY {

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

System.out.println("enter index limit");

Point2D[] points=new Point2D[sc.nextInt()];

//points=new Point2D[sc.nextInt()];

boolean flag =false;

System.out.println("1. plot a new point");

System.out.println("2. Display plotted points");

System.out.println("3. test equality");

System.out.println("4. calculate distance");

System.out.println("10. exit");

while(flag!=true){

//System.out.println("enter a choice");

//int n=sc.nextInt();

while(flag!=true)

{

System.out.println("enter a choice number");

int n=sc.nextInt();

switch (n)

{

case 1:

for (int i=0;i<points.length;i++)

{

System.out.println("enter the coprdinate of point");

points[i]=new Point2D(sc.nextInt(),sc.nextInt());

}

break;

case 2:

System.out.println("printing the points");

for (Point2D p:points)

{

if(p!=null)

System.out.println(p.getdetails());

}break;

case 3:

System.out.println("To test equality");

System.out.println("Enter first index: ");

int i=sc.nextInt();

System.out.println("Enter second index: ");

int j=sc.nextInt();

if((i < points.length && i >=0) && (j < points.length && j >=0))

{

if(points[i].isEqual(points[j])) {

System.out.println("Same Co-ordinate");

}

else {

System.out.println("Different Co-ordinate");

}

}

else

System.out.println("not valid index");

break;

case 4:

System.out.println("for calculating distance");

System.out.println("enter 1st index");

int k=sc.nextInt();

System.out.println("enter 2nd index");

int l=sc.nextInt();

if((k < points.length && k >=0) && (l < points.length && k >=0))

System.out.println(points[k].calculateDistance(points[l]));

else

System.out.println("not valid index");

break;

case 10:

flag=true;

System.out.println("exit");

break;

}

} }

}}

**Output:**

enter index limit

2

1. plot a new point

2. Display plotted points

3. test equality

4. calculate distance

10. exit

enter a choice number

1

enter the cordinate of point

10 20

enter the cordinate of point

10 20

enter a choice number

2

printing the points

Co-ordinates:10 20

Co-ordinates:10 20

enter a choice number

3

To test equality

Enter first index:

1

Enter second index:

1

Same Co-ordinate

enter a choice number