

**Федеральное агентство связи**  
**Ордена Трудового Красного Знамени**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**«Московский технический университет связи и информатики»**

Кафедра Математической кибернетики и информационных технологий

**Отчет по лабораторной работе №2**  
по дисциплине «Технологии разработки программного обеспечения»

Выполнил: студент группы  
БВТ1801

Клюшкин Дмитрий Алексеевич

Руководитель:

Мосева Марина Сергеевна

Москва 2019

**Цель работы:** В данной лабораторной работе необходимо использовать классы по одному на файл, чтобы описать, как эти объекты работают.

**Ход работы:**

```
import java.util.Scanner;

public class Lab1 {
    public static void main(String[] args){
        Scanner input = new Scanner(System.in);
        System.out.println("Enter point 1: ");
        double x1 = input.nextDouble();
        double y1 = input.nextDouble();
        double z1 = input.nextDouble();
        Point3d point1 = new Point3d(x1,y1,z1);
        System.out.println("Enter point 2: ");
        double x2 = input.nextDouble();
        double y2 = input.nextDouble();
        double z2 = input.nextDouble();
        Point3d point2 = new Point3d(x2,y2,z2);
        System.out.println("Enter point 3: ");
        double x3 = input.nextDouble();
        double y3 = input.nextDouble();
        double z3 = input.nextDouble();
        Point3d point3 = new Point3d(x3,y3,z3);

        System.out.println(computeArea(point1, point2, point3));
    }

    public static double computeArea(Point3d point1, Point3d point2, Point3d point3){
        if (!point1.equil(point2) && !point2.equil(point3) && !point3.equil(point1)) {
            double a = point1.distanceTo(point2);
            double b = point2.distanceTo(point3);
            double c = point3.distanceTo(point1);
            double p = (a + b + c)/2;
            double S = Math.sqrt(p * (p - a) * (p - b) * (p - c));
            return (double) Math.round(S*100)/100;
        }
        System.out.println("2 или 3 точки равны!");
        return 0;
    }
}
```

```

public class Point3d extends Point2d{

    private double zCoord;

    public Point3d(double x, double y, double z){
        xCoord = x;
        yCoord = y;
        zCoord = z;
    }

    public Point3d(){
        this(0.0,0.0,0.0);
    }

    public double getZ(){
        return zCoord;
    }

    public void setZ(double zCoord) {
        this.zCoord = zCoord;
    }

    public boolean equil(Point3d point){
        if (xCoord==point.getX() && yCoord==point.getY() &&
zCoord==point.getZ()){
            return true;
        }
        return false;
    }

    public double distanceTo(Point3d point){
        double res = Math.sqrt(Math.pow((point.getX()-
xCoord),2)+Math.pow((point.getY()-yCoord),2)+Math.pow((point.getZ()-
zCoord),2));
        res = (double)Math.round(res*100)/100;
        return res;
    }
}

```

Enter point 1:

0  
0  
0

Enter point 2:

3  
0  
0

Enter point 3:

0  
3  
0  
4.5