МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное

бюджетное образовательное учреждение высшего образования

«Московский технический университет связи и информатики»

(МТУСИ)

Кафедра «Математической кибернетики и информационных технологий»

Лабораторная работа №2

По дисциплине: «Java - программирование»

По теме: «Основы объектно-ориентированного   
программирования»

Выполнил студент

группы БСУ1901

Иванов Р.А.

Москва, 2021

# Цель работы

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

# Выполнение

1. Создайте новый класс Point3d для представления точек в трехмерном   
   Евклидовом пространстве. Необходимо реализовать:   
   • создание нового объекта Point3d с тремя значениями с плавающей   
   точкой (double);   
   • создание нового объекта Point3d со значениями (0.0, 0.0, 0.0) по   
   умолчанию,   
   • возможность получения и изменения всех трех значений по   
   отдельности;   
   • метод для сравнения значений двух объектов Point3d.   
   Нельзя предоставлять непосредственный доступ к внутренним элементам   
   объекта класса Point3d .   
   2. Добавьте новый метод distanceTo, который в качестве параметра   
   принимает другой объект Point3d, вычисляет расстояние между двумя точками с точность двух знаков после запятой и возвращает полученное значение.

package com.company;  
  
import java.util.Scanner;  
  
public class point3d {  
  
 private double xCoord;  
 private double yCoord;  
 private double zCoord;  
  
 public point3d (double x, double y, double z)  
 {  
 xCoord = x;  
 yCoord = y;  
 zCoord = z;  
 }  
 public point3d ()  
 {  
 xCoord = 0;  
 yCoord = 0;  
 zCoord = 0;  
 }  
 public boolean isEaqual2 (point3d test)  
 {  
 if ((xCoord == test.getX())&&(yCoord == test.getY())&&(zCoord == test.getZ())){  
 return true;  
 }else  
 return false;  
 }  
  
 public double getX()  
 {  
 return xCoord;  
 }  
 public double getY()  
 {  
 return yCoord;  
 }  
 public double getZ()  
 {  
 return zCoord;  
 }  
 public void setX (double val)  
 {  
 xCoord = val;  
 }  
 public void setY (double val)  
 {  
 yCoord = val;  
 }  
 public void setZ (double val)  
 {  
 zCoord = val;  
 }  
 public static double distance(point3d a, point3d b)  
 {  
 double rez, rezX, rezY, rezZ;  
 rezX = Math.*pow*(a.getX()-b.getX(),2);  
 rezY = Math.*pow*(a.getY()-b.getY(),2);  
 rezZ = Math.*pow*(a.getZ()-b.getZ(),2);  
 rez = Math.*sqrt*(rezX+rezY+rezZ);  
 return rez;  
 }  
 public static double computeArea()  
 {  
 Scanner scan = new Scanner(System.*in*);  
 System.*out*.println("input X1,Y1,Z1; X2,Y2,Z2; X3,Y3,Z3");  
 String tmp = scan.nextLine();  
 scan.close();  
 String[] mas = tmp.split(" ");  
 point3d[] newMass = new point3d[mas.length];  
 for (int i=0;i<=mas.length-1;i++){  
 String[] TMP = mas[i].split(",");  
 newMass[i] = new point3d(Integer.*parseInt*(TMP[0]),Integer.*parseInt*(TMP[1]),Integer.*parseInt*(TMP[2]));  
  
 }  
 double halfP = (point3d.*distance*(newMass[0],newMass[1]) + point3d.*distance*(newMass[1],newMass[2])+point3d.*distance*(newMass[2],newMass[0]));  
 return Math.*sqrt*(halfP \* (halfP - point3d.*distance*(newMass[0],newMass[1]))\*(halfP - point3d.*distance*(newMass[1],newMass[2]))+(halfP - point3d.*distance*(newMass[2],newMass[0])));  
 }  
  
}

3. Создайте другой класс под названием Lab1, который будет содержать   
статический метод main. Помните, что метод main должен быть   
общедоступным (public) с возвращаемым значением void, а в качестве   
аргумента должен принимать строку (String). Этот класс должен иметь   
следующую функциональность:   
• Ввод координат трех точек, находящихся в трехмерном пространстве.   
Создание трех объектов типа Point3d на основании полученных данных.   
(Предполагается, что пользователь вводит корректные данные.)

• Создайте второй статический метод computeArea, который принимает   
три объекта типа Point3d и вычисляет площадь треугольника, образованного   
этими точками. (Вы можете использовать формулу Герона.) Верните   
получившееся значение площади в формате типа double.   
• На основе полученных данных и с использованием реализованного   
алгоритма посчитайте площадь и выведите полученное значение пользователю.   
Перед вызовом метода computeArea проверьте на равенство значений   
всех трех объектов Point3d. Если одна из точек равна другой, то выведите   
соответствующее сообщение пользователю и не вычисляйте площадь.

package com.company;  
  
import com.company.point3d;  
  
  
public class Main {  
  
 public static void main(String[] args) {  
  
 point3d myPoint = new point3d(3,6,5);  
 System.*out*.println(myPoint.getX());  
 System.*out*.println(myPoint.getY());  
 System.*out*.println(myPoint.getZ());  
 myPoint.setX(100);  
 System.*out*.println(myPoint.getX());  
 myPoint.setY(74);  
 System.*out*.println(myPoint.getY());  
 myPoint.setZ(274);  
 System.*out*.println(myPoint.getZ());  
 point3d newPoint = new point3d(1064,374,-4);  
 System.*out*.println(myPoint.isEaqual2(newPoint));  
 System.*out*.println(point3d.*distance*(myPoint,newPoint));  
 System.*out*.println(point3d.*computeArea*());  
 }  
  
  
}

4. Скомпилируйте оба исходных файла вместе

Изображение выглядит как текст

Автоматически созданное описание