**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**

Факультет РиТ

**ЛАБОРАТОРНАЯ РАБОТА №2**

по дисциплине «Java-программирование»

на тему:

«Основы объектно-ориентированного программирования»

Выполнил: студ. гр.БПЗ1901

Неживлева Ксения

Проверил: Харрасов Камиль Раисович

Москва 2021

Цель работы:

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

Выполнение:

Сохраним код для простого класса, который представляет двумерную точку в файле с именем Point2d.java, согласно требованиям Java к именам классов и именам файлов. Экземпляр класса можно также создать, вызвав любой из реализованных конструкторов. Для того, чтобы проверить равны ли сами значения, а не ссылки, создадим метод в классе Point2d, который будет сравнивать значения соответствующих полей объектов класса Point2d.

Класс Point2d:

package com.company;  
public class Point2d {  
 */\*\* координата X \*\*/* private double xCoord;  
 */\*\* координата Y \*\*/* private double yCoord;  
 private Object obj;  
 */\*\* Конструктор инициализации \*\*/* public Point2d ( double x, double y) {  
 xCoord = x;  
 yCoord = y;  
 }  
 */\*\* Конструктор по умолчанию. \*\*/* public Point2d () {  
//Вызовите конструктор с двумя параметрами и определите источник.  
 this(0, 0);  
 }  
 */\*\* Возвращение координаты X \*\*/* public double getX () {  
 return xCoord;  
 }  
 */\*\* Возвращение координаты Y \*\*/* public double getY () {  
 return yCoord;  
 }  
 */\*\* Установка значения координаты X. \*\*/* public void setX ( double val) {  
 xCoord = val;  
 }  
 */\*\* Установка значения координаты Y. \*\*/* public void setY ( double val) {  
 yCoord = val;  
 }  
 */\*\* Cравнение двух точек. \*\*/* public boolean equals(Object obj)  
 {  
 if (this == obj)  
 return true;  
 if (obj == null)  
 return false;  
 if ( !getClass().equals(obj.getClass()) )  
 return false;  
 Point2d point2d = (Point2d) obj;  
 return this.xCoord == point2d.xCoord && this.yCoord == point2d.yCoord;  
 }  
}

Класс main:

package com.company;  
  
public class Main {  
  
 public static void main(String[] args) {  
 Point2d myPoint = new Point2d ();//создает точку (0,0)  
 Point2d myOtherPoint = new Point2d (5,3);//создает точку (5,3)  
 Point2d aThirdPoint = new Point2d ();//создает точку (0,0)  
 System.*out*.println(myPoint.equals(myOtherPoint));  
 System.*out*.println(myPoint.equals(aThirdPoint));  
 System.*out*.println(myOtherPoint.equals(aThirdPoint));  
 }  
}

Проверим работу программы:

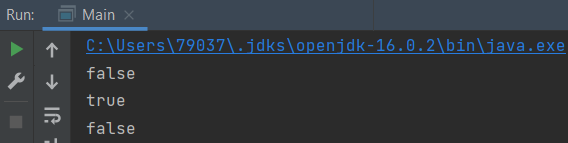


Рис.1-Проверка равенства двумерных точек

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 .

1. 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.

Класс Point3d:

package com.company;  
  
public class Point3d {  
 */\*\* координата X \*\*/* private double xCoord;  
 */\*\* координата Y \*\*/* private double yCoord;  
 */\*\* координата Z \*\*/* private double zCoord;  
 */\*\* Конструктор инициализации \*\*/* public Point3d ( double x, double y,double z) {  
 xCoord = x;  
 yCoord = y;  
 zCoord = z;  
 }  
 */\*\* Конструктор по умолчанию. \*\*/* public Point3d () {  
//Вызовите конструктор с тремя параметрами и определите источник.  
 this(0.0, 0.0, 0.0);  
 }  
 */\*\* Возвращение координаты X \*\*/* public double getX () {  
 return xCoord;  
 }  
 */\*\* Возвращение координаты Y \*\*/* public double getY () {  
 return yCoord;  
 }  
 */\*\* Возвращение координаты Z \*\*/* public double getZ () {  
 return zCoord;  
 }  
 */\*\* Установка значения координаты X. \*\*/* public void setX ( double val) {  
 xCoord = val;  
 }  
 */\*\* Установка значения координаты Y. \*\*/* public void setY ( double val) {  
 yCoord = val;  
 }  
 */\*\* Установка значения координаты Z. \*\*/* public void setZ ( double val) {  
 zCoord = val;  
 }  
 */\*\* Cравнение двух точек. \*\*/* public boolean equals(Object obj)  
 {  
 if (this == obj)  
 return true;  
 if (obj == null)  
 return false;  
 if ( !getClass().equals(obj.getClass()) )  
 return false;  
 Point3d point3d = (Point3d) obj;  
 return this.xCoord == point3d.xCoord && this.yCoord == point3d.yCoord && this.zCoord == point3d.zCoord;  
 }  
*/\*\* Поиск расстояния между точками \*\*/* public double distanceTo(Point3d p) {  
 return Math.*sqrt*(Math.*pow*(xCoord - p.getX(), 2) + Math.*pow*(yCoord - p.getY(), 2) + Math.*pow*(zCoord - p.getZ(), 2));  
 }  
}

Класс main:

package com.company;  
  
public class Main {  
  
 public static void main(String[] args) {  
   
 Point3d point3d\_1 = new Point3d(0.8,1.0,0.2);  
 Point3d point3d\_2 = new Point3d(0.8,1.0,0.2);  
 Point3d point3d\_3 = new Point3d(5.8,1.8,0.5);  
 */\*\*Вывод равенства координат между двумя точками \*\*/* System.*out*.println(point3d\_1.equals(point3d\_2));  
 System.*out*.println(point3d\_1.equals(point3d\_3));  
 */\*\*Вывод расстояния с точностью до двух знаков после запятой \*\*/* System.*out*.printf("%.2f",point3d\_1.distanceTo(point3d\_3));  
 }  
}

Проверим работу программы:

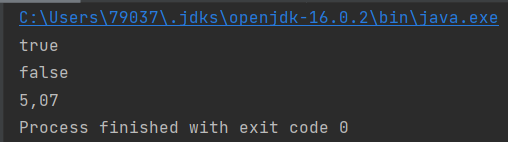


Рис.1-Проверка равенства трехмерных точек и нахождение расстояния между ними

1. 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.

Класс Lab1:

package com.company;  
import java.util.Scanner;  
public class Lab1 {  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Введите координаты первой точки: ");  
 Point3d point1 = new Point3d(in.nextDouble(), in.nextDouble(), in.nextDouble());  
 System.*out*.println("Введите координаты второй точки: ");  
 Point3d point2 = new Point3d(in.nextDouble(), in.nextDouble(), in.nextDouble());  
 System.*out*.println("Введите координаты третьей точки: ");  
 Point3d point3 = new Point3d(in.nextDouble(), in.nextDouble(), in.nextDouble());  
 if ((point1.equals(point2)) && (point1.equals(point3)) && (point2.equals(point3))) {  
 System.*out*.println("Вы ввели три одинаковые точки!"); }  
 else if ((point1.equals(point2)) || (point1.equals(point3)) || (point2.equals(point3))) {  
 System.*out*.println("Вы ввели две одинаковые точки!"); }  
 else  
 { System.*out*.println("Площадь треугольника = " + *computeArea*(point1,point2,point3)); }}  
 public static double computeArea(Point3d point1,Point3d point2,Point3d point3) {  
 */\*\* Стороны а, b, c для нахождения полупериметра \*\*/* double a = point1.distanceTo(point2);  
 double b = point2.distanceTo(point3);  
 double c = point3.distanceTo(point1);  
 */\*\* Полупериметр для нахождения площди по формуле Герона \*\*/* double p = (a + b + c) / 2.0;  
 */\*\* Нахождение площади \*\*/* double area = Math.*sqrt*(p \* (p - a) \* (p - b) \* (p - c));  
 return area;  
 }  
}

Проверим работу программы:

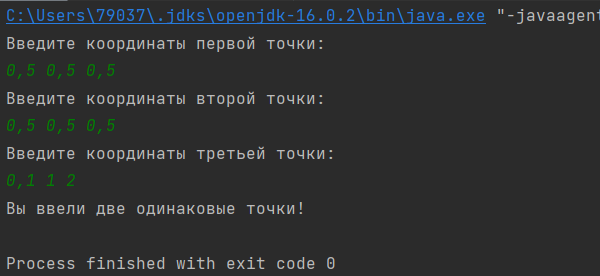


Рис.3-Ввод координат двух одинаковых точек

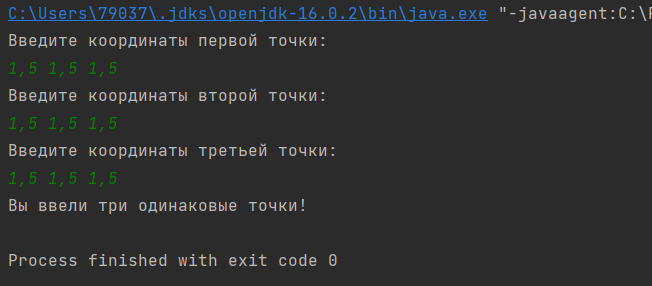


Рис.3-Ввод координат трех одинаковых точек

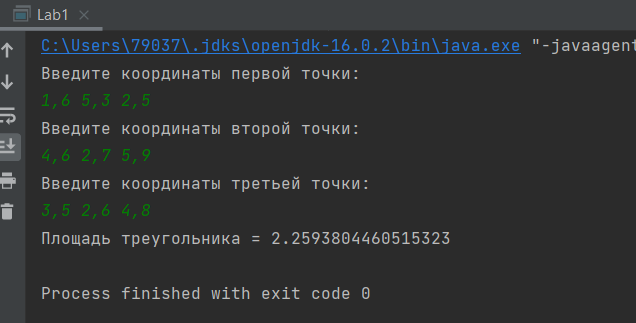


Рис.3- Ввод разных координат и подсчет площади