



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційні систем та технологій

Лабораторна робота №8
із дисципліни *«Основи програмування»*
Тема: «Основи ООП»

Виконали:
Студенти групи ІА-24
(бригада №1)
Чайка А.П.
Котлярчук М.С.
Коханчук М.М.

Перевірив:
Колеснік Валерій
Миколайович

Хід роботи:

1. Повторити теоретичні відомості
2. Проаналізувати предметну область завдання свого варіанту (табл.1)
3. Розробити базовий клас (відповідно до завдання можливо абстрактний клас або інтерфейс), клас-нащадок, а також допоміжні класи та/або інтерфейси за необхідністю. Відповідно до предметної області завдання передбачити відповідні методи бізнес-логіки, а також конструктори, сетери та/або гетери, методи equals() та toString(). Продемонструвати використання:
 - this;
 - super;
 - перевантаження (overloading) та заміщення(overriding) методів;
 - перевантаження (overloading) конструкторів.
4. Відповісти на контрольні запитання.

№	Базовий клас	Класи-нащадки	Мутабельність об'єктів
1	Координати пікселя	Піксель, що має колір	mutable

Результати + код:

```
8.85751658197714
26.608457302143616
```

```
Process finished with exit code 0
```

```
Main.java x Color.java x Coordinates.java x
1 public class Main {
2     public static void main(String[] args)
3     {
4         Coordinates coordinates = new Coordinates();
5         coordinates.findVectorLength();
6         Coordinates coordinates1 = new Coordinates( x: 2.34, y: 5.3, z: 6.7);
7         System.out.println(coordinates1.findVectorLength( x: 2.34, y: 5.3, z: 6.7));
8         Color color = new Color();
9         color.findVectorLength();
10        Color color1 = new Color( x: 2.34, y: 5.3, z: 6.7, r: 23.4, g: 12.6, b: 1.3);
11        System.out.println(color1.findVectorLength( x: 2.34, y: 5.3, z: 6.7));
12    }
13 }
```

```
Main.java x Color.java x Coordinates.java x
4 usages
1 public class Color extends Coordinates {
    3 usages
2     private double r;
    3 usages
3     private double g;
    3 usages
4     private double b;
    1 usage
5     Color() { super(); }
    1 usage
8     Color(double x, double y, double z, double r, double g, double b)
9     {
10         super(x, y, z);
11         this.r = r;
12         this.g = g;
13         this.b = b;
14     }
    3 usages
15     @Override
16     void findVectorLength() {
17         super.findVectorLength();
18     }
    2 usages
19     @Override
20     double findVectorLength(double x, double y, double z) {
21         if (x < 0 || x > 255 || y < 0 || y > 255 || z < 0 || z > 255) {
22             throw new IllegalArgumentException();
23         }
24         double colorlen = Math.sqrt(r*r + g*g + b*b);
25         return colorlen;
26     }
27 }
```

```
1 5 usages 1 inheritor
public class Coordinates {
2 1 usage
    private double x,y,z;
3 2 usages
    Coordinates(){
4
    }
5 2 usages
    Coordinates(double x,double y,double z){
6        this.x = x;
7        this.y = y;
8        this.z = z;
9    }
10 3 usages 1 override
    void findVectorLength(){
11
    }
12 2 usages 1 override
    double findVectorLength(double x, double y, double z){
13        double len = Math.sqrt(x*x + y*y + z*z);
14        return len;
15    }
16 }
```

Висновок:

У цій роботі ми узагальнили та повторили знання зі створення класів, їхніх об'єктів, методів та класів-нащадків; навчилися роботі з операторами `this` і `super` й реалізували перевантаження методів в Java.