



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

Сіденко Д.Д.

Філімонов Є.А.

Іскандаров Е. Е. огли

Яблонський Д.Б.

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

```
import javax.print.Doc;

public class Main {
    public static void main(String[] args) {
        Document document = new Document();
        System.out.println(document);

        Document document1 = new Document( name: "Lab8", size: 1024);
        System.out.println(document1);
        document1.setName("p1-lab-08");
        System.out.println("Renamed to "+document1.getName());

        TextDocument textDocument = new TextDocument();
        System.out.println( textDocument);
        TextDocument textDocument1 = new TextDocument( name: "Lab9", info: "My task",
            text: "A constructor in Java is a special method that is used to initialize objects");
        System.out.println(textDocument1);
    }
}
```

```

5 usages 1 inheritor
public class Document {
    7 usages
    String name;
    4 usages
    int size;

    1 override
    @Override
    public String toString(){
        return "Name: " + name + ", Size: " + size;
    }

    3 usages
    public Document() {
        name = "New file";
    }

    1 usage
    public String getName() { return name; }

    1 usage
    public void setName(String name) { this.name = name; }

    1 usage
    public Document(String name, int size){
        this.name = name;
        this.size = size;
    }
}

```

```

4 usages
public class TextDocument extends Document{

    2 usages
    String info;
    2 usages
    String text;

    1 usage
    public TextDocument(String name, String info, String text ){
        super.name = name;
        this.size = text.toCharArray().length *2;
        this.text = text;
        this.info = info;
    }

    1 usage
    public TextDocument(){

    }

    @Override
    public String toString() { return "Name: " + name + ", Size: " + size + ", Contents: " + info + ", Text: " + text; }
}

```

```

Name: New file, Size: 0
Name: Lab8, Size: 1024
Renamed to p1-lab-08
Name: New file, Size: 0, Contents: null, Text: null
Name: Lab9, Size: 152, Contents: My task, Text: A constructor in Java is a special method that is used to initialize objects

Process finished with exit code 0

```

Висновки: Завдяки даній лабораторній роботі ми навчилися працювати з класами та класами-нащадками, визначати предметну область завдання та розробляти програму відповідно до неї, на практиці спробували

використовувати перевантаження (overloading) та заміщення(overriding) методів, а також перевантаження (overloading) конструкторів.