



Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота № 7

З дисципліни «Технології розроблення програмного забезпечення»

Тема: «Шаблони «MEDIATOR», «FACADE»,
«BRIDGE», «TEMPLATE METHOD»»

Виконала: Лапа Руслана Ігорівна

студент групи ІА-11

Дата здачі

Захищено з балом

Перевірів:

ст. вик. кафедри ІСТ

Колеснік В. М.

Київ 2023

Тема: Шаблони «MEDIATOR», «FACADE», «BRIDGE», «TEMPLATE METHOD».

Мета: реалізувати один з розглянутих шаблонів.

Хід роботи:

..8 Powershell terminal (strategy, command, factory method, template method, interpreter, client-server)

Термінал для powershell повинен нагадувати типовий термінал з можливістю налаштування кольорів синтаксичних конструкцій, розміру вікна, фону вікна, а також виконання команд powershell і виконуваних файлів, а також працювати в декількох вікнах терміналу (у вкладках або одночасно шляхом розділення вікна).

1. Реалізувати не менше 3-х класів відповідно до обраної теми

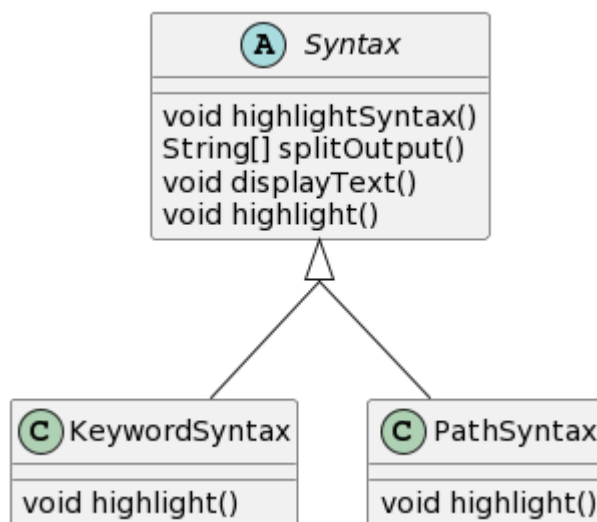


Рис. 1 – Діаграма класів для поданого коду

2. Реалізація шаблону «Template Method»

```
package com.example.terminal.Model.Execution.Style;

import ...
4 usages 2 inheritors
public abstract class Syntax {
    1 usage
    public final void highlightSyntax(String output, TextFlow textFlow) {
        String[] lines = splitOutput(output);
        for (String line : lines) {
            String[] words = line.split(regex: " ");

            for (String word : words) {
                Text text = new Text(s: word + " ");

                highlight(word, text, textFlow);
            }
            textFlow.getChildren().add(new Text(s: "\n"));
        }
        displayText(textFlow);
    }

    1 usage
    private String[] splitOutput(String output) {
        return output.split(regex: "\n");
    }

    1 usage
    private void displayText(TextFlow textFlow) {
        Text separator = new Text(s: "\n");
        separator.setFill(Color.BLACK);
        textFlow.getChildren().add(separator);
    }

    1 usage 2 implementations
    protected abstract void highlight(String word, Text text, TextFlow textFlow);
}
```

```

package com.example.terminal.Model.Execution.Style;

import ...

1 usage
public class KeywordSyntax extends Syntax {
    1 usage
    @Override
    protected void highlight(String word, Text text, TextFlow textFlow) {

        if (word.equals("if") || word.equals("else")) {
            text.setFill(Color.RED);
        }

        textFlow.getChildren().add(text);
    }
}

package com.example.terminal.Model.Execution.Style;

import ...

2 usages
public class PathSyntax extends Syntax {
    1 usage
    @Override
    protected void highlight(String word, Text text, TextFlow textFlow) {

        if (word.matches(regex: "[A-Z]:\\\\\\\\.*")) {
            text.setFill(Color.GREEN);
        }

        textFlow.getChildren().add(text);
    }
}

```

sample.fxml

```

<ScrollPane>
    <TextFlow fx:id="textFlow1" prefWidth="600.0" prefHeight="400.0"/>
</ScrollPane>

```

Висновок: у цій лабораторній роботі я ознайомилась з різними шаблонами проектування і реалізувала шаблон “Template Method” .