

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій

Кафедра комп'ютерної інженерії та інформаційних систем

## **Лабораторна робота № 4**

з дисципліни «Об'єктно-орієнтовані технології програмування»

на тему:

**«Поведінкові патерни»**

**Виконав:**

студент 1 курсу, групи КІ2м-23-2

\_\_\_\_\_ Мариняк В.М.

**Перевірив:**

\_\_\_\_\_ Лисенко С.М.

## Хід роботи

Розглянемо варіант реалізації патерну Шаблонний метод на прикладі застосунку для системи обробки даних, де будуть представлені різні типи обробників даних, що слідуватимуть спільному алгоритму, але певні кроки реалізовуватимуть по своєму.

### Технічне завдання:

1. Визначити абстрактний клас `DataProcessor`, що оголошує шаблонний метод.
2. Створити конкретні класи, що реалізують `DataProcessor`, визначаючи власні способи обробки даних.

### Реалізація:

1. Визначення абстрактного класу `DataProcessor`

```
1 namespace TemplateMethodExample.Maryniak.Models;
2
3 public abstract class DataProcessor
4 {
5     // Template method
6     public void ProcessData()
7     {
8         ReadData();
9         ProcessDataDetails();
10        WriteData();
11    }
12
13    protected abstract void ReadData();
14    protected abstract void ProcessDataDetails();
15    protected abstract void WriteData();
16 }
```

2. Реалізація конкретних класів обробників даних

```
1 namespace TemplateMethodExample.Maryniak.Models;
2
3 public class CsvDataProcessor : DataProcessor
4 {
5     protected override void ReadData()
6         => Console.WriteLine("Reading data from CSV file...");
7
8     protected override void ProcessDataDetails()
9         => Console.WriteLine("Processing data from CSV file...");
10
11    protected override void WriteData()
12        => Console.WriteLine("Writing processed data to CSV file...");
13 }
```

```

1 namespace TemplateMethodExample.Maryniak.Models;
2
3 public class XmlDataProcessor : DataProcessor
4 {
5     protected override void ReadData()
6         => Console.WriteLine("Reading data from XML file...");
7
8     protected override void ProcessDataDetails()
9         => Console.WriteLine("Processing data from XML file...");
10
11     protected override void WriteData()
12         => Console.WriteLine("Writing processed data to XML file...");
13 }

```

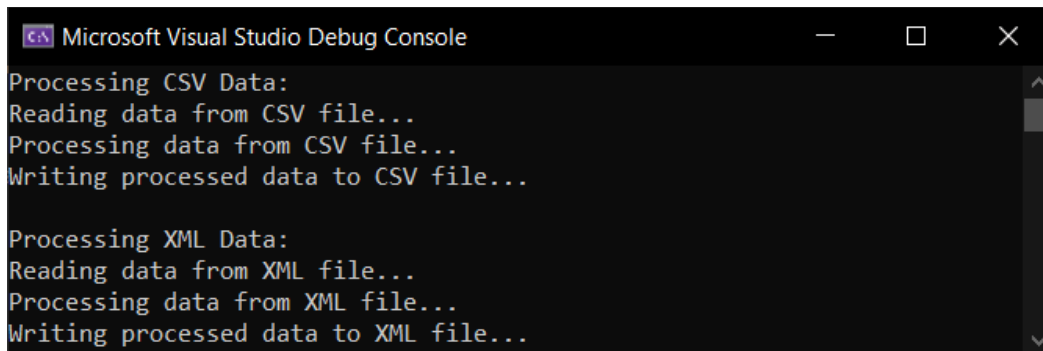
### 3. Клієнтський код

```

1 using TemplateMethodExample.Maryniak.Models;
2
3 namespace TemplateMethodExample.Maryniak;
4
5 internal class Program
6 {
7     static void Main()
8     {
9         // Create data processors
10         DataProcessor csvProcessor = new CsvDataProcessor();
11         DataProcessor xmlProcessor = new XmlDataProcessor();
12
13         // Process data
14         Console.WriteLine("Processing CSV Data:");
15         csvProcessor.ProcessData(); // Template method call for CSV
16
17         Console.WriteLine("\nProcessing XML Data:");
18         xmlProcessor.ProcessData(); // Template method call for XML
19     }
20 }

```

### 4. Результат виконання програми



```

Microsoft Visual Studio Debug Console
Processing CSV Data:
Reading data from CSV file...
Processing data from CSV file...
Writing processed data to CSV file...

Processing XML Data:
Reading data from XML file...
Processing data from XML file...
Writing processed data to XML file...

```

**Висновок.** Під час лабораторної роботи я навчився реалізовувати патерн Шаблонний метод та створив консольний додаток, який демонструє принцип його роботи.