****

Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут імені Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

Лабораторна робота №7  
ШАБЛОН «MEDIATOR», «FACADE»,

«BRIDGE», «TEMPLATE METHOD»

Виконав Перевірив:

студент групи ІА – 12: Колеснік В. М

Глухов Іван

**Завдання:**

1. Ознайомитися з короткими теоретичними відомостями.

2. Реалізувати частину функціоналу робочої програми у вигляді класів та їх взаємодій для

досягнення конкретних функціональних можливостей.

3. Застосування одного з розглянутих шаблонів при реалізації програми.

**Хід роботи**

Патерн Міст (Bridge) є структурним патерном, який відокремлює абстракцію від її реалізації, дозволяючи їм змінюватися незалежно одне від одного. Патерн Міст використовується для відокремлення абстракції обробки архіву (визначеної ArchiveProcessor) від її реалізації (зокрема, ZipArchiveProcessor). Ця відокремленість дозволяє незалежно міняти або розширювати реалізації.

Ось розбір використання патерну Міст у коді:

1. Абстракція (ArchiveProcessor та DirectoryProcessor):
   * ArchiveProcessor - це абстрактний клас, який визначає інтерфейс для обробки архівів.
   * DirectoryProcessor - ще один абстрактний клас, який визначає інтерфейс для обробки каталогу.
2. Реалізації (ZipArchiveProcessor та EmailProcessor):
   * ZipArchiveProcessor - це конкретна реалізація ArchiveProcessor, яка виконує розпаковку ZIP-архіву.
   * EmailProcessor - це конкретна реалізація DirectoryProcessor, яка обробляє каталог для електронних листів.
3. Міст (ArchiveProcessorBridge):
   * ArchiveProcessorBridge виступає містом між абстракцією та її реалізаціями.
   * У ньому є посилання як на ArchiveProcessor, так і на DirectoryProcessor.
   * Метод process\_archive моста використовує ArchiveProcessor для розпакування архіву, а потім делегує обробку каталогу DirectoryProcessor.
4. Використання у маршруті FastAPI (process\_directory):
   * Маршрут визначає точку входу для обробки каталогу з вивантаженого архіву.
   * Створюється екземпляр ZipArchiveProcessor для обробки архіву та екземпляр EmailProcessor для обробки каталогу.
   * Ці екземпляри використовуються для створення екземпляра ArchiveProcessorBridge.
   * Викликається метод process\_archive моста для виконання повного процесу обробки каталогу, використовуючи реалізації ArchiveProcessor та DirectoryProcessor.

Ця структура дозволяє легко розширювати код, додаючи нові реалізації ArchiveProcessor або DirectoryProcessor, не змінюючи існуючий код, який використовує міст. Це сприяє більш гнучкому та зручному дизайну.

