

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

**Лабораторна робота № 8**

з дисципліни «Теорія розробки ПЗ»

Тема: ШАБЛОНИ «COMPOSITE», «FLYWEIGHT»,  
«INTERPRETER», «VISITOR»

Варіант №3

Виконав:

студент групи ІА-13

Губенко Єгор Олександрович

Київ, 2023

Тема: ШАБЛОНИ «COMPOSITE», «FLYWEIGHT», «INTERPRETER», «VISITOR».

### Завдання.

1. Ознайомитися з короткими теоретичними відомостями.
2. Реалізувати частину функціоналу робочої програми у вигляді класів їх взаємодій для досягнення конкретних функціональних можливостей.
3. Застосування одного з розглянутих шаблонів при реалізації програми.

### Варіант.

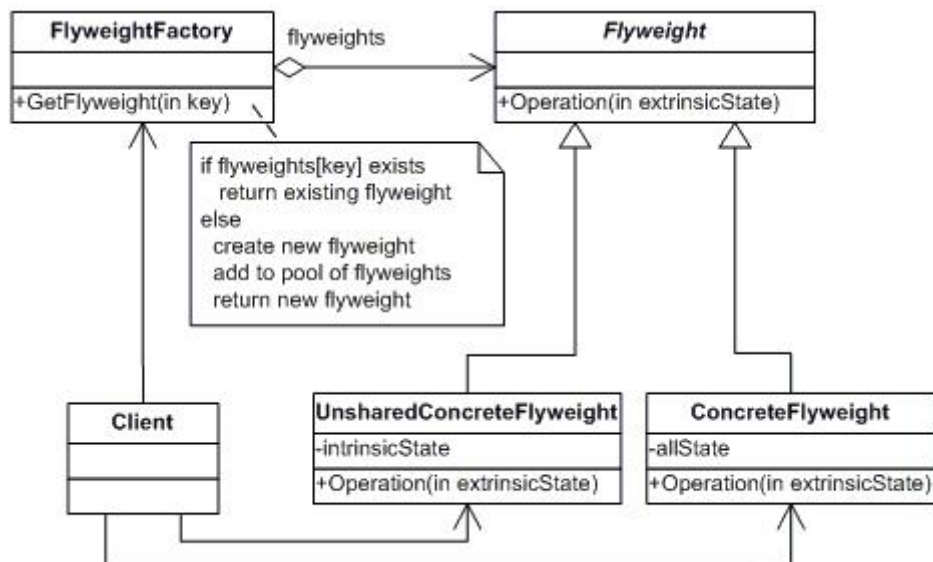
3. Текстовий редактор (strategy, command, observer, template method, flyweight, SOA)

Текстовий редактор повинен вміти розпізнавати текстові файли в будь-якій кодуванні, мати розширені функції редагування: макроси, сніппети, підказки, закладки, перехід на рядок / сторінку, підсвічування синтаксису (для однієї мови програмування або розмітки на розсуд студента).

### Хід роботи

#### Шаблон «Flyweight»

Шаблон використовується для зменшення кількості об'єктів в додатку шляхом поділу цих об'єктів між ділянками додатку. Flyweight являє собою поділюваний об'єкт.



4 usages Yehor Hubenko

```
1 class Snippet:
2     def __init__(self, id, name, content):
3         self.id = id
4         self.name = name
5         self.content = content
```

2 usages Yehor Hubenko

```
8 class SnippetFlyweightFactory:
9     _snippets = {}
```

1 usage Yehor Hubenko

```
11 @classmethod
12 def get_snippet(cls, name, content, database_strategy=None):
13     if name not in cls._snippets:
14         # If snippet is not in the local cache, try fetching it from the database
15         if database_strategy:
16             db_snippet = database_strategy.get_snippet_by_name(name)
17             if db_snippet:
18                 snippet = Snippet(name=name, content=db_snippet.content)
19                 cls._snippets[name] = snippet
20                 return snippet
21
22         # If snippet is not found in the database, create a new one
23         snippet = Snippet(name=name, content=content)
24         cls._snippets[name] = snippet
25         return snippet
26     else:
27         return cls._snippets[name]
```

```

109 1 usage (1 dynamic)  Yehor Hubenko
110 def create_snippet(self, name, content):
111     try:
112         if not self.connection:
113             raise Exception("Not connected to the database.")
114
115         with self.connection.cursor() as cursor:
116             cursor.execute("INSERT INTO snippets (name, content) VALUES (%s, %s) RETURNING id", (name, content))
117             snippet_id = cursor.fetchone()[0]
118             self.connection.commit()
119             return snippet_id
120
121     except Exception as e:
122         print(f"An error occurred during snippet creation: {e}")
123         return None
124
125 1 usage (1 dynamic)  Yehor Hubenko
126 def delete_snippet(self, snippet_id):
127     with self.connection.cursor() as cursor:
128         cursor.execute("DELETE FROM snippets WHERE id = %s", (snippet_id,))
129         self.connection.commit()
130
131 1 usage (1 dynamic)  Yehor Hubenko
132 def get_all_snippets(self):
133     try:
134         if not self.connection:
135             raise Exception("Not connected to the database.")
136
137         with self.connection.cursor() as cursor:
138             cursor.execute("SELECT id, name, content FROM snippets")
139             snippets_data = cursor.fetchall()
140
141             snippets = [Snippet(id=snippet[0], name=snippet[1], content=snippet[2]) for snippet in snippets_data]
142             return snippets
143
144     except Exception as e:
145         print(f"An error occurred while fetching snippets: {e}")
146         return None

```

PostgreSQLDatabaseStrategy > get\_file\_id()

**Висновок:** у даній лабораторній я додав функціонал сніппетів до проєкту, використовуючи шаблон Flyweight.