



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 №4  
З дисципліни «**Технології розроблення програмного  
забезпечення**»

Тема: «**ШАБЛони «SINGLETON», «ITERATOR», «PROXY»,  
«STATE», «STRATEGY»»**

Варіант №6

Виконав  
студент групи ІА–13:  
Костенко П.С.

Перевірив:  
Мягкий М. Ю.

Київ 2023

## Тема:

### ..6 Web-browser (proxy, chain of responsibility, factory method, template method, visitor, p2p)

Веб-браузер повинен мати можливість зробити наступне: мати адресний рядок для введення адреси сайту, переміщатися і відображати структуру html документа, переглядати підключений javascript та css файли, перегляд всіх підключених ресурсів (зображень), коректна обробка відповідей з сервера (коди відповідей HTTP) - переходи при перенаправленнях, відображення сторінок 404 і 502/503.

## Завдання:

1. Реалізувати не менше 3-х класів відповідно до обраної теми.
2. Реалізувати один з розглянутих шаблонів за обраною темою.

## Хід роботи:

За завданням необхідно реалізувати шаблон Проху. За цим шаблоном для об'єкту створюється заглушка, яка може розширювати або мати вузчий функціонал для використання замість об'єкту.

Для виконання завдання я обрав створити об'єкт-заклушку для класу Errors, яка має назву ProxyErrors. Мета - при виникненні помилки, замість виклику класу Errors та помилки буде викликатись ErrorsProxy з більш детальним та зрозумілим для користувача описом.

```
class ErrorsProxy:
    addressline = ''

    def error_404(self):
        ErrorsProxy.description_404(self)
        pass

    def error_502(self):
        ErrorsProxy.description_502(self)
        pass

    def error_503(self):
        ErrorsProxy.description_503(self)
        pass

    def description_404(self):
```

```

        print("Виникла помилка на пристрої,
спробуйте...")

    def description_502(self):
        print("Виникла проблема на сервері...")
        pass

    def description_503(self):
        print("Сервер зараз не може прийняти
запит, зачекайте...")
        pass
# методи для обробки помилок(проксі)

```

У класі ErrorsProxy можна побачити розширений функціонал класу Errors, а саме: при виклику методів помилок викликаються також методи опису.

```

class HTTPResponseHandler:
    addressline = ''

    # основний метод для респонс
    1 usage new *
    def send_get_response(self, url):
        try:
            HTTPRequestHandler.HTTPRequestHandler.send_get_request(url)
        except urllib.error.HTTPError as err:
            if err.code == 404:
                erpr.ErrorsProxy.error_404()
            elif err.code == 502:
                erpr.ErrorsProxy.error_502()
            elif err.code == 503:
                erpr.ErrorsProxy.error_503()

```

У класі HTTPResponseHandler відбувається отримання відповіді з HTTPRequestHandler та перевірка на помилки. При виникненні помилки викликається проксі.

**Висновок:** на цій лабораторній роботі я познайомився з шаблоном Proxu, засвоїв знання на практиці, продовжив розробку проекту.