**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Чорноморський національний університет   
імені Петра Могили**

**Факультет комп’ютерних наук**

**Кафедра інженерії програмного забезпечення**

**ЗВІТ**

*з лабораторної роботи №5*

«**HTTP Protocols and status codes**»

Дисципліна «Професійна практика програмної інженерії»

Спеціальність: **Інженерія програмного забезпечення**

121–ЛР.05–309.22011007

***Cтудент*** *\_\_\_\_\_\_\_\_\_****Н. Д. Кляцький***

*(підпис)*

*\_\_07.04.2023 \_\_*

*(дата)*

***Викладач*** *\_\_\_\_\_\_\_\_\_****С. В. Фаленков***

*(підпис)*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(дата)*

Миколаїв – 2023

**Лабораторна робота №5**

## *Тема:* HTTP Protocols and status codes

**Практичне завдання:**

1. 1. Створити простий API клієнт за допомогою HttpClient. Обрати будь-який API на порталі https://any-api.com/?tag=text. HttpClient повинен бути реалізован, як окремий клас в окремому файлі.
2. Створити модель для відповідей, яка має наступні поля:

Поле з повідомленням;

поле з кодом HTTP запиту (використати StatusCodes)

Generic List для зберігання отриманих даних з API

1. Реалізувати два методи для отримання даних:

GET() - отримання даних без будь-яких параметрів;

POST(…) - отримання даних з використанням параметрів для запиту;

1. Створити методи для отримання даних по API з формуванням моделі відповіді. Використати конструкцію try…catch. У разі помилки повернути статус код 500.
2. Усі методи повинні бути реалізовані, як асинхронні.

**Хід роботи:**

Код створеної програми можна побачити у репозиторії github: [https://github.com/klyatskiinikita/prof\_pract\_labs/tree/main/lab5/code](https://github.com/klyatskiinikita/prof_pract_labs/tree/main/lab2/code)

Результат виконання програми:

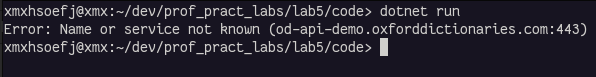


Рисунок 1 – Результат виконання програми (api не працює)

**Теоретичне завдання:**

1. Що таке HTTP?

HTTP (Hypertext Transfer Protocol) є протоколом передачі даних в мережі Інтернет, який використовується для передачі гіпертекстових документів, таких як веб-сторінки.

1. Які групи статус кодів існують? Привести приклади з StatusCodes.

1xx (Інформаційні): сервер прийняв запит клієнта, але процес його обробки ще триває (наприклад, 100 Continue).

2xx (Успішні): запит клієнта був успішно оброблений сервером (наприклад, 200 OK).

3xx (Перенаправлення): сервер відправляє клієнта на іншу адресу, зазвичай для перенаправлення на нову версію веб-сторінки або на іншу сторінку (наприклад, 301 Moved Permanently).

4xx (Помилки клієнта): сервер не зміг обробити запит клієнта через невірний запит або відсутність прав доступу (наприклад, 404 Not Found).

5xx (Помилки сервера): сервер зазнав помилки при обробці запиту клієнта (наприклад, 500 Internal Server Error).

1. Для яких протоколів HTTP використовується, як базовий «транспортний» протокол?

HTTP використовується як базовий «транспортний» протокол для протоколів, які використовують його як основу для обміну даними. Наприклад, протоколи REST (Representational State Transfer) та WebSocket використовують HTTP для передачі даних між клієнтом та сервером.

1. На які групи поділяють програмне забезпечення для роботи з протоколом HTTP?

Програмне забезпечення для роботи з протоколом HTTP можна поділити на кілька груп:

* Веб-сервери: програмне забезпечення, яке використовується на стороні сервера для обробки запитів від клієнтів і відправлення відповідей. Наприклад, Apache, Nginx, IIS.
* Веб-фреймворки: програмне забезпечення, яке дозволяє створювати веб-додатки на стороні сервера, використовуючи HTTP для обміну даними з клієнтами. Наприклад, Flask, Django, Ruby on Rails.
* Клієнтські бібліотеки: програмне забезпечення, яке дозволяє розробникам створювати клієнтські додатки, які взаємодіють з веб-серверами через HTTP. Наприклад, jQuery, axios, fetch.

1. Який з методів HTTP запитів не може мати body?

Метод HTTP запиту "GET" не може мати тіло (body). Цей метод використовується для запиту ресурсу з сервера і повернення його відповіді клієнту. Його параметри передаються в URL запиту. Інші методи HTTP запитів, такі як "POST", "PUT", "DELETE", можуть мати тіло, яке містить додаткові дані для обробки на сервері.

**Висновки:**

Протягом виконання завдань лабораторної роботи було покращено розуміння технології .NET. Покращено навички створення .NET-застосунків. Розглянуто реалізацію та застосування http-клієнта. Здобуто навички використання модулю http-клієнта.