МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЧЕРНІВЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ЮРІЯ ФЕДЬКОВИЧА

Навчально-науковий інститут фізико-технічних та комп’ютерних наук

Кафедра комп’ютерних наук

**ЗВІТ**  
**про виконання лабораторної роботи № 7**

**з дисципліни**

**«Сучасні технології програмування»**

**на тему: «Розробка RESTful за допомогою технологій WCF та ASP.NET Web API»**

Виконав

студент 5-го курсу

544м групи

Вікнянський Руслан

Чернівці, 2025

**Мета:** Навчитися створювати RESTful сервіси за допомогою WCF та Web API.

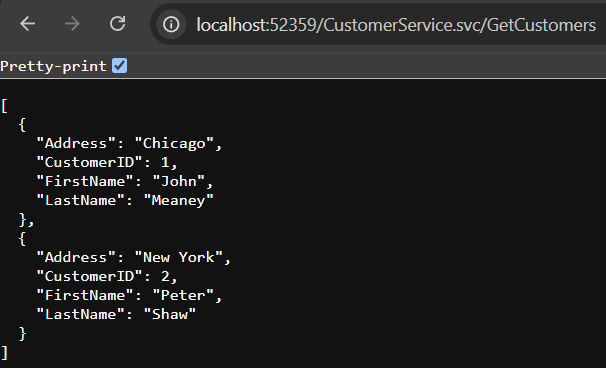
Посилання на Github - <https://github.com/RuslanCHNU/STP_Labs/tree/main/Lab7>

Обране індивідуальне завдання №6:  
 “*Студенти (номер студентської карти, ПІБ, курс)*”

**Порядок виконання роботи**

**Частина 1: WCF RESTful Service**

1. **Створення проекту WCF Service Application**
   1. У Visual Studio: **File → New → Project → WCF Service Application** (назва проекту: CustomerService).
   2. Видалення стандартних файлів IService1.cs та Service1.svc.
2. **Реалізація моделі та сервісу**
   1. Додано клас Customer з атрибутами [DataContract] та [DataMember].
   2. Створено інтерфейс ICustomerService з методом GetCustomerList(), позначеним атрибутами [OperationContract] та [WebInvoke].
   3. Реалізовано клас CustomerService, який повертає тестові дані клієнтів.
3. **Налаштування конфігурації**
   1. В файлі Web.config додано конфігурацію для webHttpBinding та поведінки служби.
4. **Тестування**
   1. Запуск проекту: **F5**.
   2. Перевірка роботи через URL: <http://localhost:52359/CustomerService.svc/GetCustomers>.

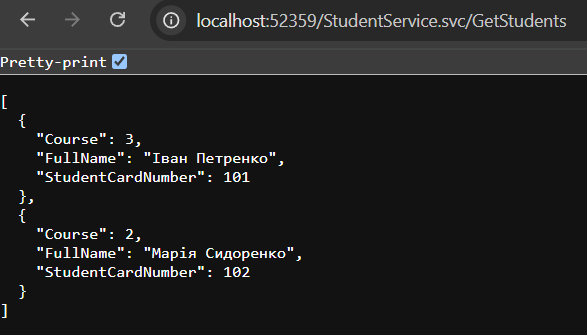


**Індивідуальне завдання №6 (Студенти)**

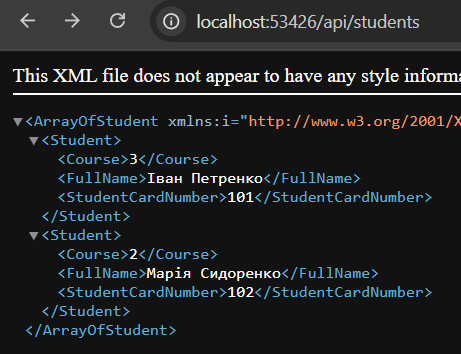
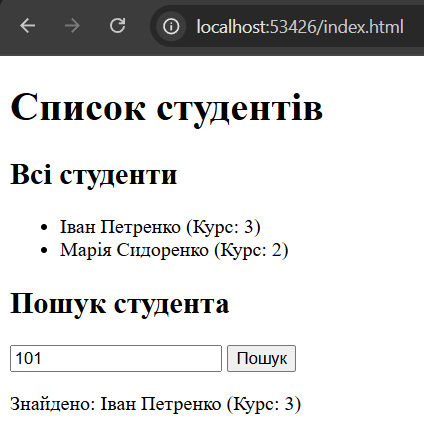
1. **Створення моделі** Student
   1. Клас з полями: StudentCardNumber, FullName, Course.

[DataContract(Namespace = "")]  
public class Student  
{  
 [DataMember] public int StudentCardNumber { get; set; }  
 [DataMember] public string FullName { get; set; }  
 [DataMember] public int Course { get; set; }  
}

1. **Реалізація WCF-сервісу**
   1. Додано файл StudentService.svc з директивою ServiceHost.
   2. Реалізовано методи GetStudentList() та PopulateStudentData() для повернення тестових даних.
2. **Тестування**
   1. URL: http://localhost:52359/StudentService.svc/GetStudents.



**Частина 2: ASP.NET Web API**

1. **Створення проекту**
   1. **File → New → Project → ASP.NET Web Application** (шаблон: **Empty** з відміткою **Web API**).
2. **Додавання моделі та контролера**
   1. Клас Student у папці Models.
   2. Контролер StudentsController з методами GetAllStudents() та GetStudent(int id).
3. **Тестування**
   1. Для всіх студентів: <http://localhost:52426/api/students>.  
      
   2. Для конкретного студента: <http://localhost:53426/api/students/101>.
4. **HTML-сторінка**
   1. Додано index.html з AJAX-запитами для відображення даних.  
      

**Контрольні питання**

1. **Що таке ASP.NET Web API?**  
   Це фреймворк для створення HTTP-сервісів, які підтримують RESTful архітектуру. Він дозволяє повертати дані у форматі JSON/XML та обробляти HTTP-методи (GET, POST тощо).
2. **Чим ASP.NET Web API відрізняється від ASP.NET MVC?**
   1. **Web API** призначений для побудови сервісів, повертає дані (JSON/XML).
   2. **MVC** використовується для веб-додатків з інтерфейсом (HTML/CSS).
3. **Що таке контролер в ASP.NET Web API і як його створити?**  
   Контролер — це клас, який обробляє HTTP-запити. Створюється шляхом успадкування від ApiController.  
   **Приклад створення:**

public class StudentsController : ApiController  
{  
 // Методи для обробки запитів  
}