**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**



**ІКНІ**

Кафедра **ПЗ**

ЗВІТ

До лабораторної роботи № 1

**На тему:** *“* *Структура DOM та методи доступу до вузлів дерева”*

**З дисципліни:** *“Програмування в Інтернет”*

**Лектор:**

асистент каф. ПЗ

Степанов Д. С.

**Виконав:**

ст. гр. ПЗ-22

Гамела Б. А.

**Перевірила:**

старший викладач каф. ПЗ

Грицай О. Д.

«\_\_\_»  \_\_\_\_\_\_\_\_\_  2025 р.

∑ = \_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_

Львів – 2025

**Тема роботи:** Структура DOM та методи доступу до вузлів дерева.

**Мета роботи:** Ознайомитись з ієрархічною структурою об’єктів JavaScript та об’єктами документа і браузера.

# Теоретичні відомості

**Основні поняття:**

1. **Модель DOM (Document Object Model)**

DOM представляє веб-сторінку як дерево об'єктів, де кожен вузол відповідає елементу, тексту чи атрибуту HTML-документа. Це дозволяє програмно взаємодіяти з вмістом сторінки через JavaScript.

1. **Глобальні об’єкти DOM:**
   * **window** – верхній об’єкт ієрархії, представляє вікно браузера (методи: alert, setTimeout; властивості: document, location).
   * **document** – поточна веб-сторінка (методи: getElementById, createElement; властивості: body, title).
   * **navigator** – інформація про браузер (властивості: userAgent, platform).
   * **screen** – дані про екран (властивості: width, height).
   * **history** – історія перегляду (методи: back, forward).
   * **location** – URL сторінки (властивості: href, protocol).
2. **Типи вузлів DOM:**
   * Елементи (HTML-теги).
   * Текст (вміст тегів).
   * Атрибути (пари «атрибут=значення»).
3. **Методи доступу до вузлів:**
   * getElementById – пошук елемента за ідентифікатором.
   * getElementsByTagName – пошук елементів за тегом.
   * appendChild, removeChild, replaceChild – додавання, видалення, заміна вузлів.
   * Властивості вузлів: firstChild, parentNode, nextSibling.
4. **Обробка подій:** Події (наприклад, click, mouseover) ініціюються діями користувача. Обробники подій (наприклад, onclick) дозволяють виконувати JavaScript-код у відповідь на ці дії.

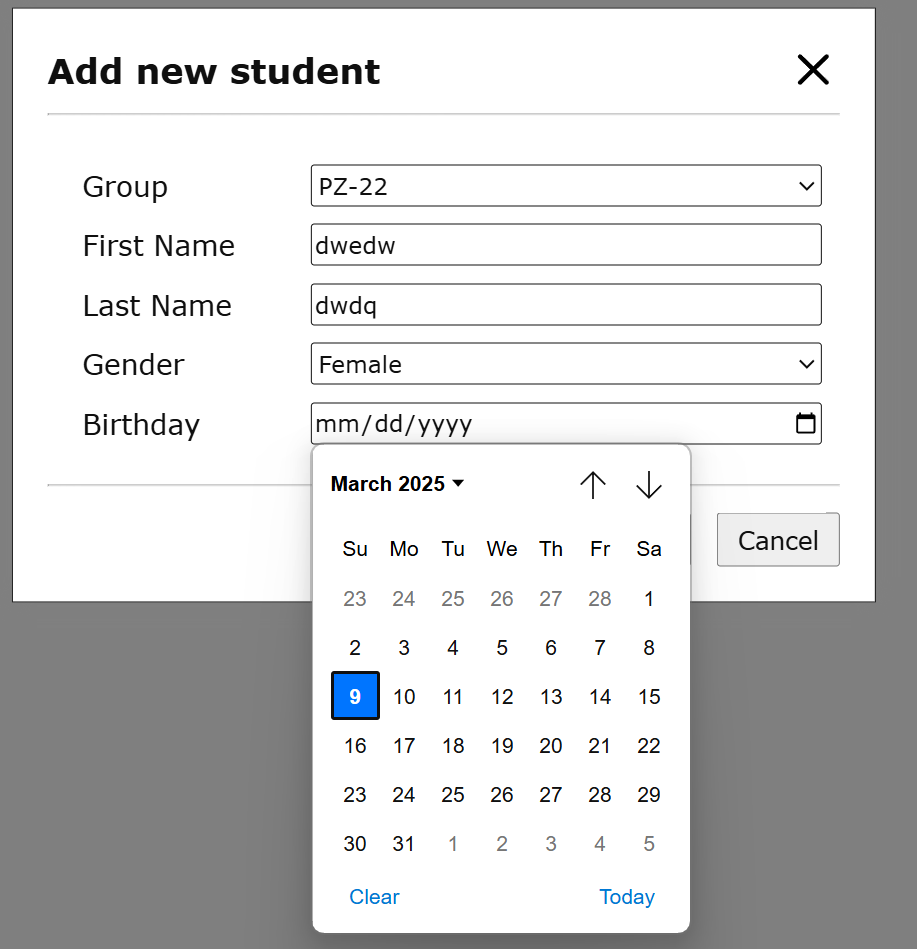
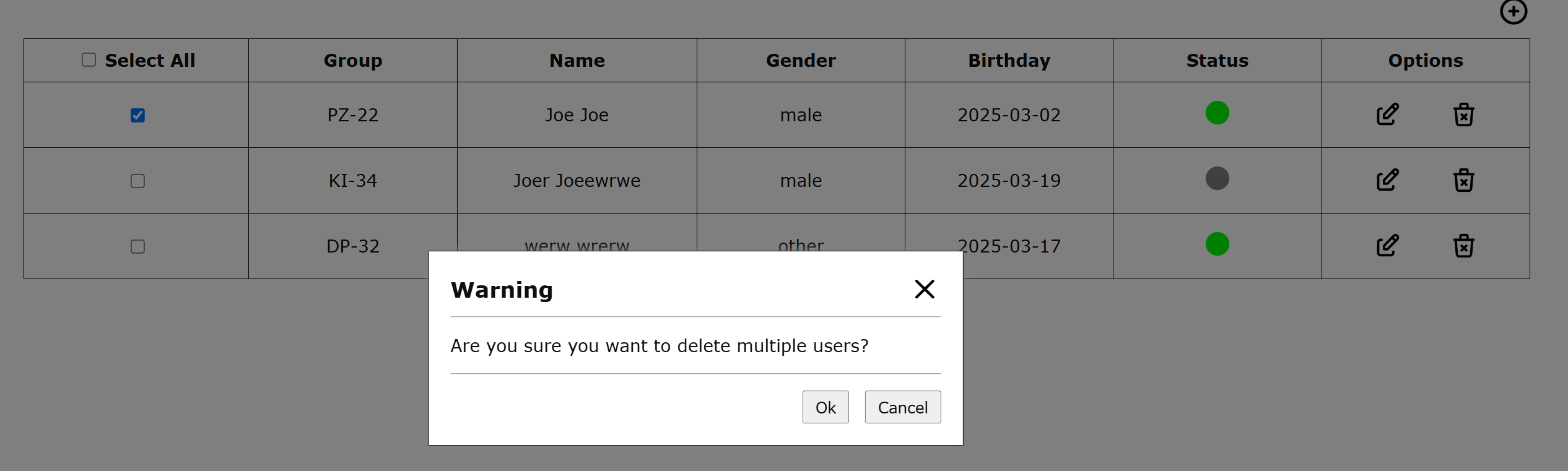
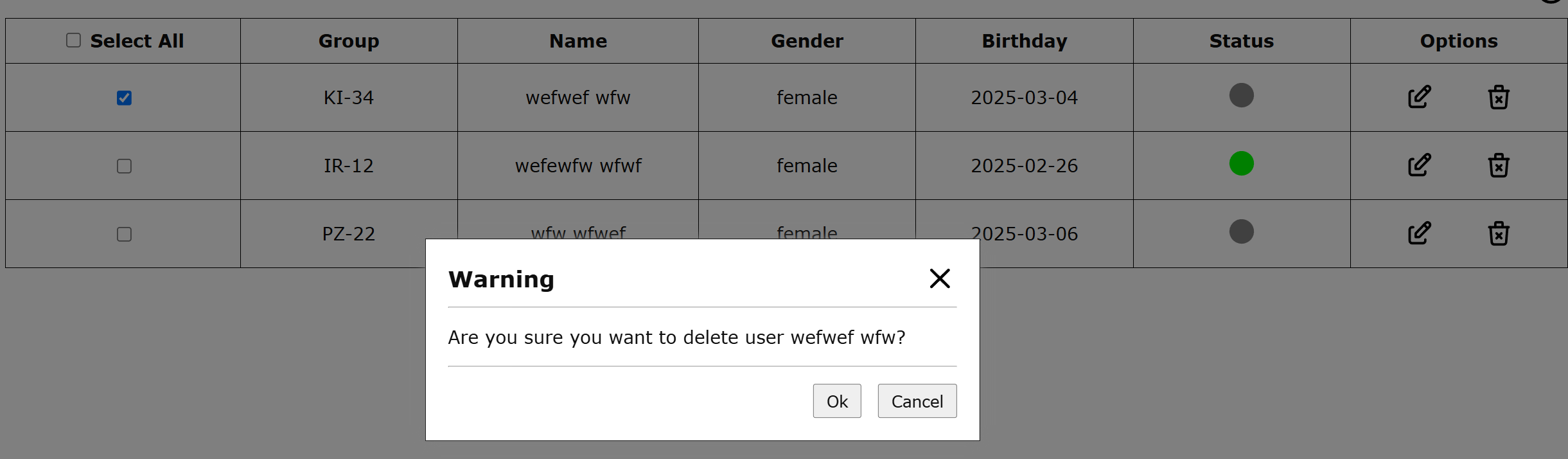
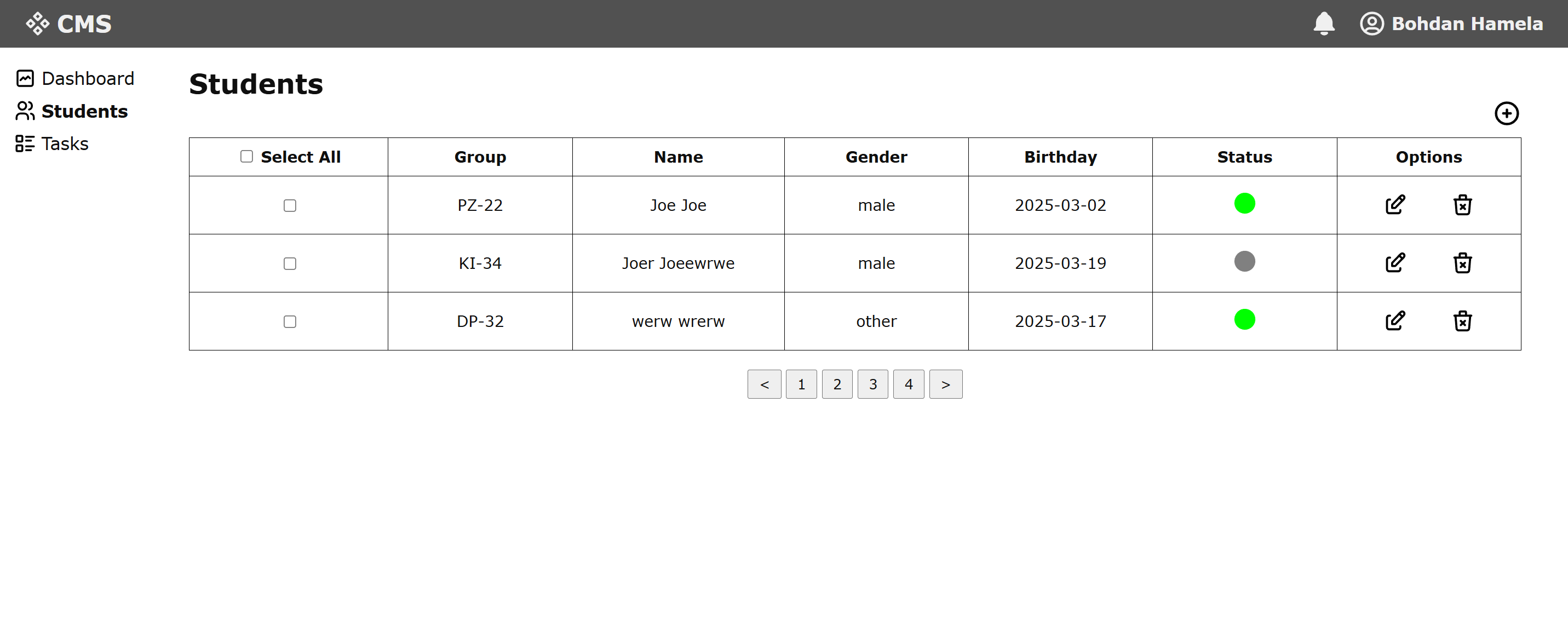
# Формулювання завдання

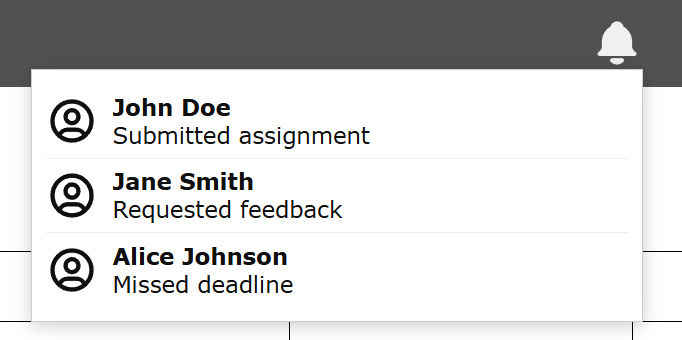
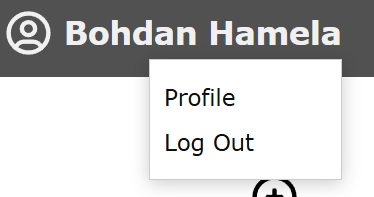
1. Розробити web-сторінку згідно [макета](https://cacoo.com/diagrams/ZvVhYS3UpG5PdbBy/EDE3A) (wireframe).

2. Задання Сторінка повинна відповідати наступним [вимогам](https://docs.google.com/document/d/1Lqwpmc2RlPVP-4p0pOTFw4pHGf_1tCLdpdMZKlAMBZQ/edit?tab=t.0).

# Результати виконання

Репозиторій проекту на github.com: [PI-Labs](https://github.com/thething1001/PI-Labs)



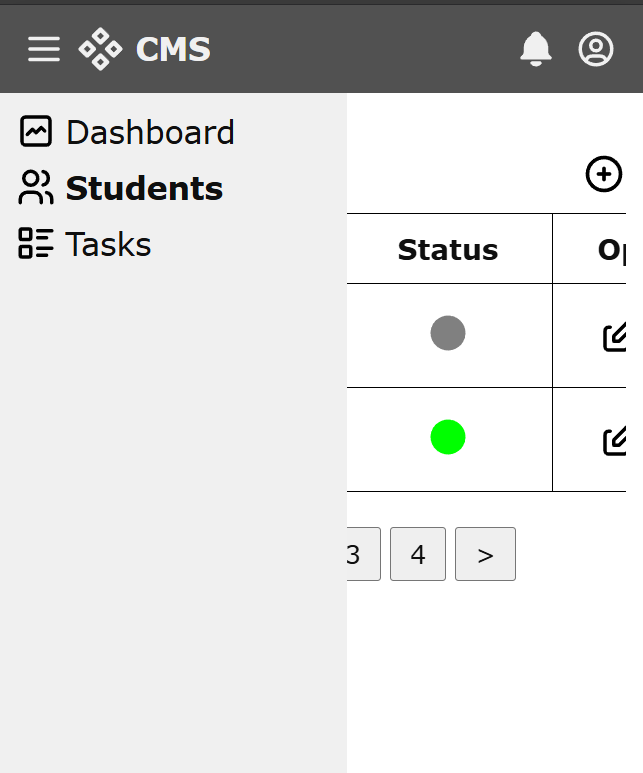
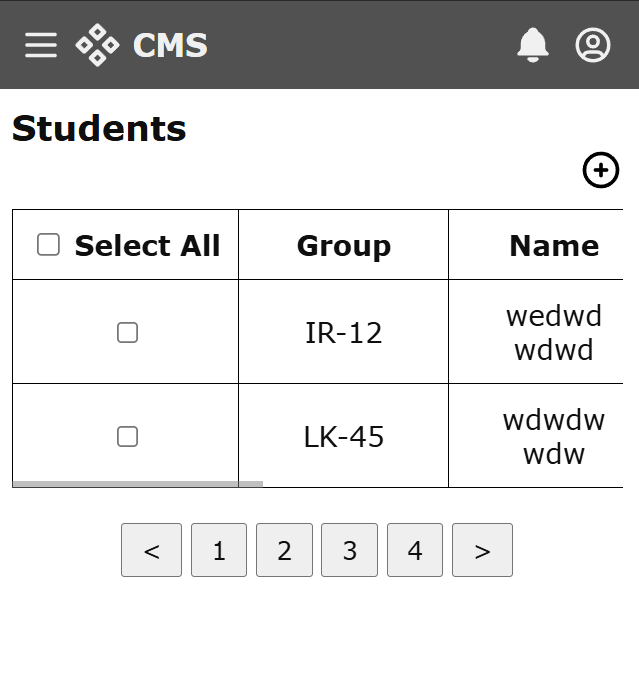


Рис. 1-8. Створений сайт

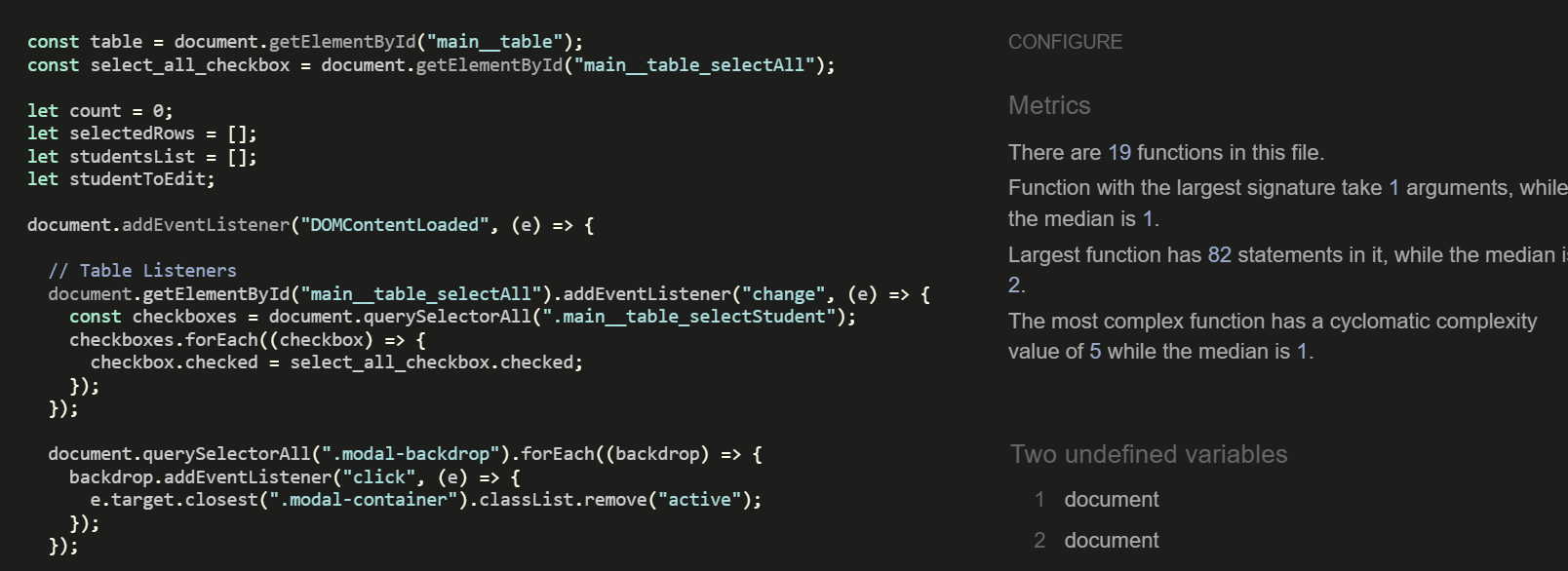
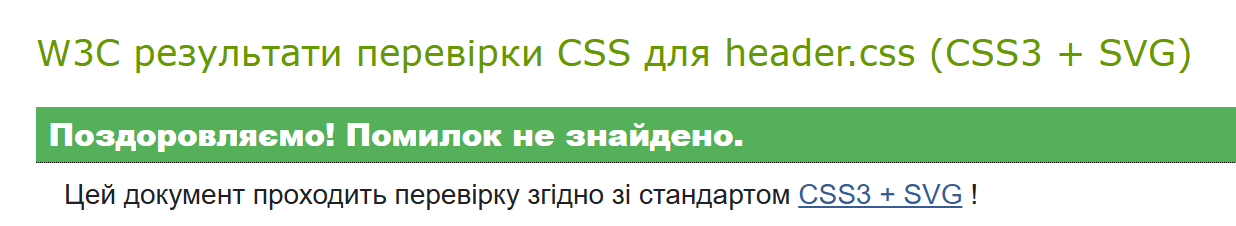
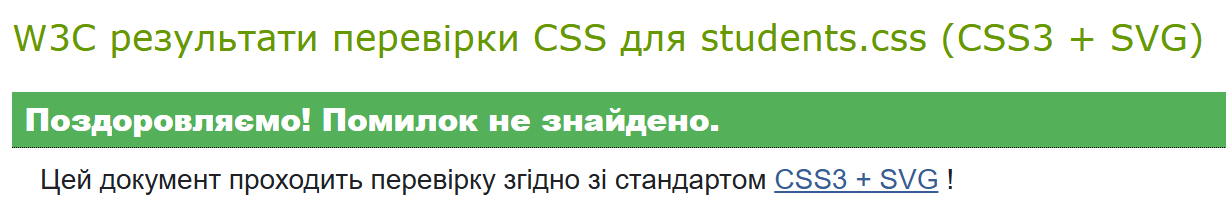
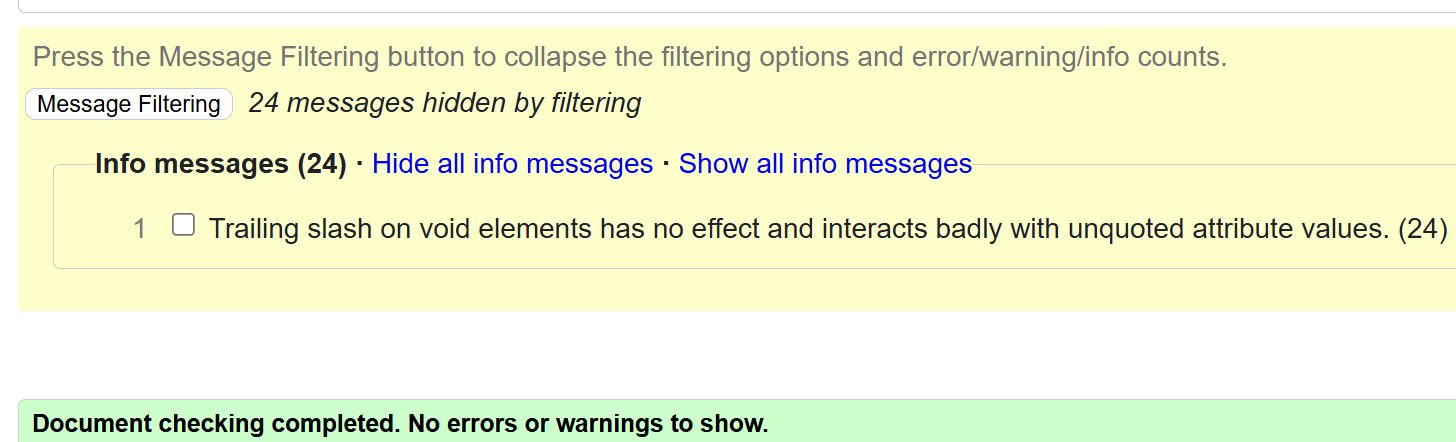
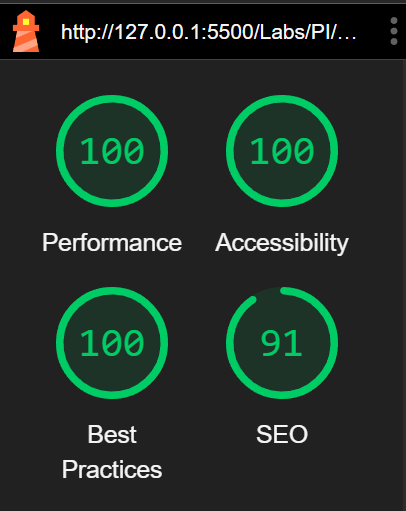
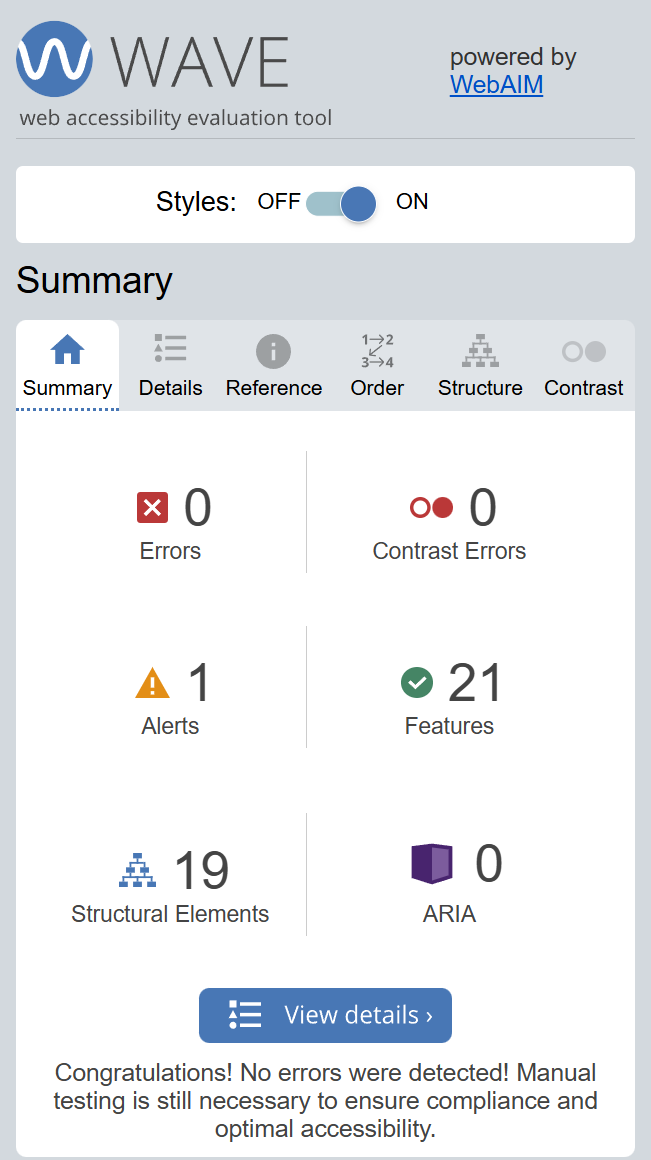


Рис. 9-15. Результати перевірок

# Висновки

Під час виконання лабораторної роботи №1 з дисципліни «Комп’ютерна графіка» я покращив свої знання про мову програмування Python та бібліотеку Tkinter, а саме про те, як правильно створювати двовимірні графічні об’єкти з використанням графічних примітивів і систем координат. Використовуючи можливості Tkinter, я розробив програму для побудови паралелограмів у другій координатній чверті з автоматичним відображенням діагоналей, однієї висоти та можливістю вибору кольору діагоналей. Провів детальний аналіз вимог завдання, реалізував необхідні функції для перетворення координат, обчислення четвертої вершини та висоти, а також забезпечив зручний/