**Міністерство освіти і науки України  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки**

**Лабораторна робота №4**

«Доступ до об'єктів і сценаріям. Доступ до властивостей і методів об'єктів»

з дисципліни  
«Основи Front-end технологій»

Виконала:

студентка групи ІМ-11   
Бащак Ярина Володимирівна

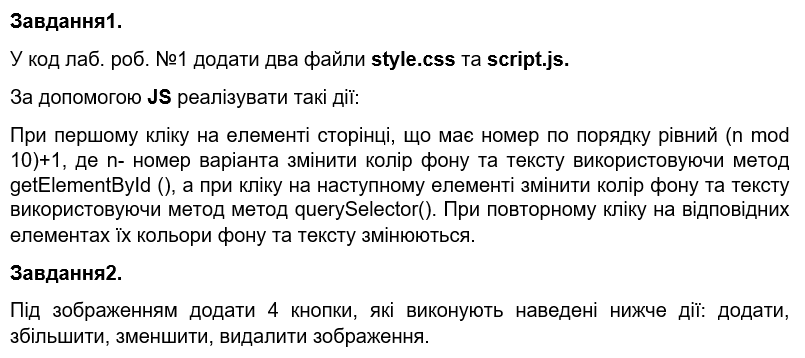
варіант 1

Перевірив:

доц. Жереб К.А.

Київ 2023

**Завдання.**



**Хід роботи.**

* Копіюємо код лабораторної №1 у лабораторну №4.
* Переносимо всі стилі в окремий файл style.css.
* Тим елементами, яким потрібно буде змінювати колір, я додала ідентифікатори (мій номер 31, тому це (31 mod 10)+1 = 2-й і 3-й елементти):

<p id="first-element">Дата народження: … </p>

<p id="second-element">Освіта:… </p>

* Тепер у файлі script.js можна отримати до них доступ за допомогою getElementById() чи querySelector():

const firstElement = document.getElementById('first-element')

const secondElement = document.querySelector('#second-element')

* Пишемо функції, щоб отримати випадковий колір, контрастний до нього (білий або чорний) і задати нові кольори фону і тексту. Додаємо виконання останньої функції при кліку на елемент:

const changeColor = element => () => {

  const newBackgroundColor = getRandomColor()

  element.style.backgroundColor = getRandomColor()

  element.style.color = getContrastingColor(newBackgroundColor)

}

firstElement.onclick = changeColor(firstElement)

secondElement.onclick = changeColor(secondElement)

* Тепер щодо виконання завдання 2. В html-файлі під фото додаємо 4 кнопки з відповідним тесктом, а у js-файлі пишемо функції для кожної з них. Особливості цих функцій:
  + фунція для додавання фото addImage створює новий елемент img за допомогою [createElement()](https://developer.mozilla.org/ru/docs/Web/API/Document/createElement) і додає його в кінець тегу <a> за допомогою [appendChild()](https://developer.mozilla.org/ru/docs/Web/API/Node/appendChild).

const addImage = () => {

  const img = document.createElement('img')

  img.src = 'https://…'

  img.alt = 'Нове фото'

  imageContainer.appendChild(img)

}

* + а функції для збільшення, зменшення і видалення фото для пошуку зображення, над яким роботи дію, треба отримати останнє фото і виконати над ним відповідну дію. Для цього я використовую querySelector з псевдо-класом [:last-of-type](https://developer.mozilla.org/ru/docs/Web/CSS/:last-of-type).

const img = document.querySelector('img:last-of-type')

**Результати:**

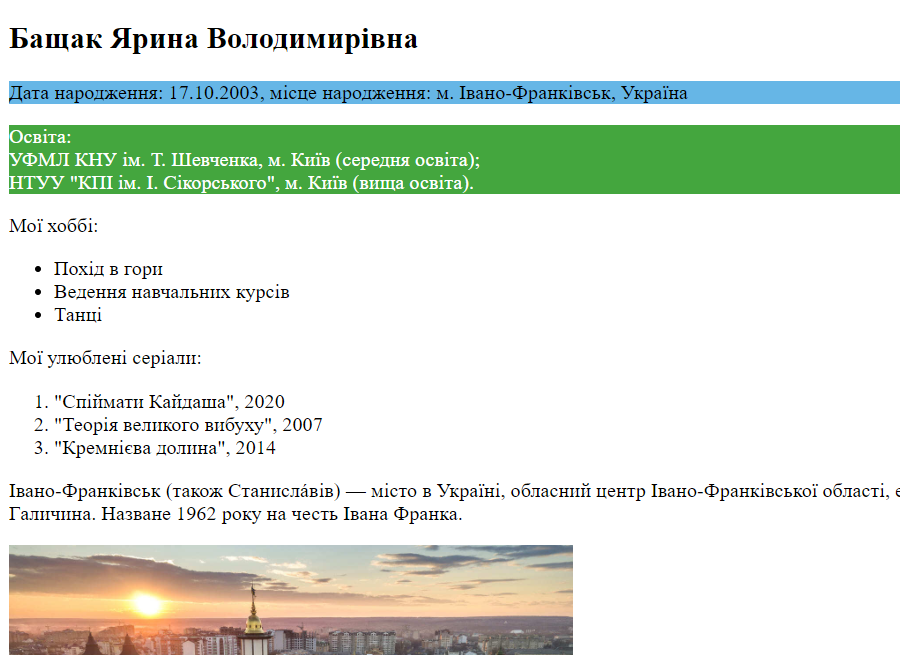
[Посилання на репозиторій в Github.](https://github.com/yaryna-bashchak/basics-of-frontend/tree/main/lab4)

[Посилання на сайт.](https://yaryna-bashchak.github.io/basics-of-frontend/lab4/index.html)

Щоб запустити в себе цю роботу, достаньо відкрити файл index.html у своєму браузері.

Скріншоти сайту:

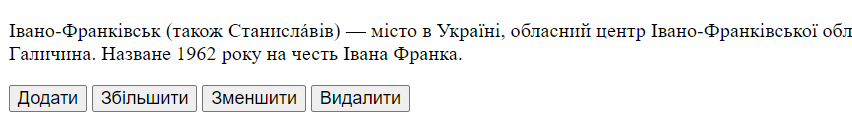
*Завдання 1*



*Завдання 2*







Висновок: Під час цієї лабораторної роботи я попрактикувалася у доступі до html-елементів, щоб вносити певні зміни за допомогою JS.