Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

**Лабораторна робота № 7**

з дисципліни «Основи клієнтської розробки»

Тема: «JavaScript. Події. Обробники подій. Спливання. Делегування подій.»

|  |  |
| --- | --- |
| Виконала:  студентка групи ІП-22  Дзюба Марія  Дата здачі \_\_\_\_\_\_\_\_\_\_\_\_  Захищено з балом \_\_\_\_\_\_\_\_\_\_\_ | Перевірила:  ст. вик. кафедри ІСТ  Хмелюк Марина Сергіївна |

Обрана тема: «Книги»

**Завдання:**

1. Використати одну з подій миші. Написати фукцію-обробник. Призначити функцію-обробник події через атрибут і через властивість.

Використати метод addEventListener, приначити одній події різні обробники (написати функції-обробники).

Призначити обробником події об’єкт за допомогою addEventListener, застосувати метод handleEvent, вивести елемент, на якому спрацював обробник, використовуючи event.currentTarget

Видалити об’єкт, використовуючи removeEventListener

1. Створити списк або використати існуючий. Реалізувати підсвічування елементів списку при кліку миші. Використовавати event.target. Обробник onclick застоувати для списку, а не для кожного елементу.

Створити меню (кілька кнопок), додати одни обробник для всього меню і атрибути data-\* для кожної кнопки, в відповідності з методами, які вони викликають.

Застосувати прийом проектування «Поведінка» (додавання елементам поведінки behavior за допомогою атрибута data-\*.

3) Завантажити проєкт на віддалений репозиторій (на GitHub).

Виконання:

1. Використаємо подію миші, напищемо функцію обробник та призначимо її через атрибут.

<input class="btn" type="button" value="Змінити колір кнопки" onclick="changeBtnColor(this)">

function changeBtnColor(btn){

    if(btn.style.backgroundColor == 'rgb(171, 81, 17)'){

        btn.style.backgroundColor = 'ivory';

        btn.style.color = 'rgb(171, 81, 17)'

    }

    else{

        btn.style.backgroundColor = 'rgb(171, 81, 17)';

        btn.style.color = 'ivory'

    }

}

1. Призначимо функцію обробник для кнопки через властивість

                <input class="btn" id="increase\_btn" type="button" value="Збільшити кнопкy">

                <script>

                    var increase\_btn = document.getElementById("increase\_btn");

                    increase\_btn.onmouseover = function(){

                        increase\_btn.style.height = '100px';

                        increase\_btn.style.fontSize = '1.2rem';

                    }

                    increase\_btn.onmouseout = function(){

                        increase\_btn.style.height = '50px';

                        increase\_btn.style.fontSize = '1rem';

                    }

                </script>

1. Використаємо метод addEventListener, щоб приначити одній події різні обробники.

let two\_events\_btn = document.getElementById('two\_events\_btn');

two\_events\_btn.addEventListener("click", function(){

    this.value = "A я додав нову кнопку!";

});

two\_events\_btn.addEventListener("click", function(){

    document.getElementById('new-btn').style.display = 'inline';

});

1. Призначимо обробником події об’єкт за допомогою addEventListener, застосуємо метод handleEvent, вивести елемент, на якому спрацював обробник, використовуючи event.currentTarget. Видалимо об’єкт, використовуючи removeEventListener.

var eventHandler = {

    handleEvent(event) {

        alert(event.type + " on " + event.currentTarget + '\nCoordinates :' + event.clientX + ':' + event.clientY);

        if (confirm('Видалити обробник події для цієї кнопки?')) {

            document.getElementById('new-btn').removeEventListener('click', eventHandler);

        }

    }

};

document.getElementById('new-btn').addEventListener('click', eventHandler );

1. Створимо список та реалізуємо підсвічування елементів списку при кліку миші. Для цього використаємо event.target.

                <ol id="tips-list">

                    <li>Знайдіть книгу яка вас зацікавила.</li>

                    <li>

                        Розпитайте своїх друзів і родину про їхні улюблені книги.

                    </li>

                    <li>

                        Прочитайте книгу про свій улюблений фільм.

                    </li>

                    <li>

                        Знайдіть зручніше місце для читання.

                    </li>

                    <li>

                        Розслабтеся, створіть атмосферу за допомогою чашечки кави, свічок або приємної музики

                    </li>

                    <li>

                        Зробіть читання приємним, якщо книга вам незрозуміла чи нецікава, не дочитуйте її.

                    </li>

                    <li>

                        Плануйте час для читання, виділіть спочатку 10 хвилин день, цього буде достатньо щоб просто почати.

                    </li>

                </ol>

let selectedLi = null;

let tipsList = document.getElementById('tips-list');

tipsList.addEventListener('click', function(event){

    let targetEl = event.target;

    if(targetEl.tagName === 'LI'){

        highlight(targetEl);

    }

});

function highlight(item){

    if (selectedLi) {

        selectedLi.classList.remove('highlight');

        }

    selectedLi = item;

    selectedLi.classList.add('highlight');

}

1. Створимо меню та додамо один обробник для всього меню і атрибути data-\* для кожної кнопки, в відповідності з методами, які вони викликають.

            <div id="menu">

                <button class="btn" data-act="info">Інформація про розробника</button>

                <button class="btn" data-act="sources">Посилання на джерела</button>

                <button class="btn" data-act="contact">Контакти</button>

            </div>

class Menu{

    constructor(element){

        this.elem = element;

        element.onclick = this.onClick.bind(this);

    }

    onClick(event){

        let action = event.target.dataset.act;

        if(action){

            this[action]();

        }

    }

    info(){

        alert("Виконала студентка групи ІП-22 Дзюба Марія");

    }

    sources(){

        alert("Джерело 1: https://book-ye.com.ua/ \nДжерело 2: https://coma.in.ua/19375 \nДжерело 3: https://nashkiev.ua/culture/tse-10-kafe-knigaren-kieva-yaki-varto-vidvidati");

    }

    contact(){

        alert("Гітхаб: dzmashaa \nТелеграм: @RaCC00ns")

    }

}

new Menu(document.getElementById('menu'));

1. Застосувати прийом проектування «Поведінка» (додавання елементам поведінки behavior за допомогою атрибута data-\*.

<button class="btn" id="hide-btn" data-hidden-id="tips-list">Сховати список</button>

document.addEventListener('click', function(event){

    let elementId = event.target.dataset.hiddenId;

    if(elementId){

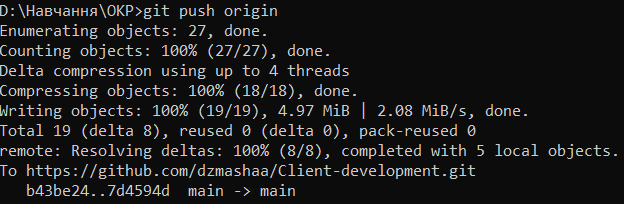
        let el = document.getElementById(elementId);

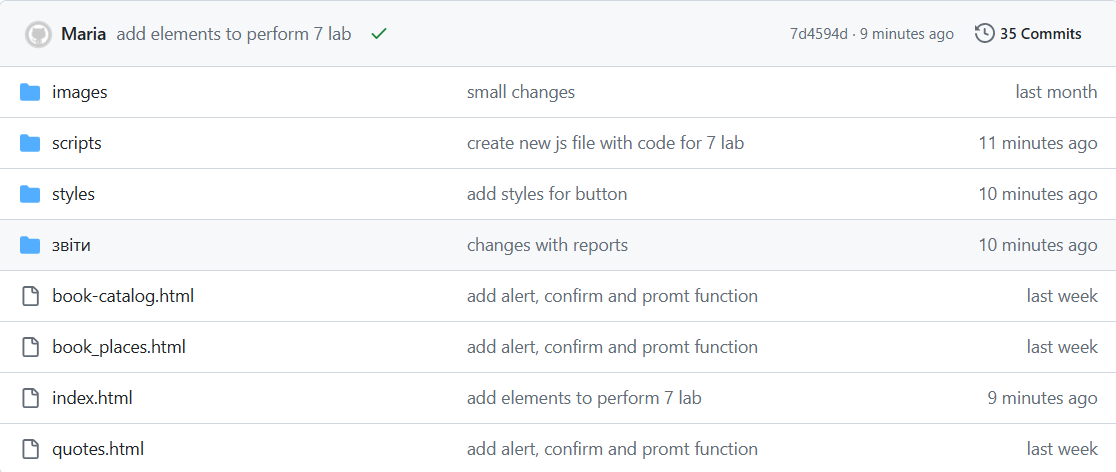
        el.hidden = !el.hidden;

    }

})

1. Завантажити проєкт на віддалений репозиторій (на GitHub).





Висновок

В результаті виконання лабораторної роботи я використала події миші. Написала фукцію-обробник та призначила функцію-обробник події через атрибут і через властивість. Використала метод addEventListener та приначила одній події різні обробники. Також призначила обробником події об’єкт за допомогою addEventListener, застосувала метод handleEvent, вивести елемент, на якому спрацював обробник, використовуючи event.currentTarget, а також видалила об’єкт, використовуючи removeEventListener. Створила список та реалізувала підсвічування елементів списку при кліку миші. Використала event.target. Обробник onclick застосувала для списку, а не для кожного елементу. Створила меню, додала один обробник для всього меню і атрибути data-\* для кожної кнопки, в відповідності з методами, які вони викликають. А також застосувала прийом проектування «Поведінка».