**Лабораторна робота № 6**

**Події***.*

**Хід роботи:**

**Завдання 1:**

Дана таблиця з користувачами: номер, ім'я та прізвище. Під таблицею зробіть форму, з допомогою якої можна буде додати нового користувача в таблицю.

**Лістинг програми:**

HTML file:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <table>

        <tr>

            <td style="background-color: gray; color: white;">№</td>

            <td style="background-color: gray; color: white;">First name</td>

            <td style="background-color: gray; color: white;">Last name</td>

        </tr>

        <tr>

            <td>1</td>

            <td>Darci</td>

            <td>Malovelence</td>

        </tr>

        <tr>

            <td>2</td>

            <td>Tusk</td>

            <td>Radiance</td>

        </tr>

    </table>

    <p>First Name:</p>

    <input id="name" type="text">

    <p>Last Name:</p>

    <input id="lastname" type="text">

    <p><button>Add</button></p>

    <script src="script.js"></script>

</body>

</html>

Java Script file:

document.querySelector("button").addEventListener('click', (event) => {

    let name = document.getElementById("name")

    let lastname = document.getElementById("lastname")

    let table = document.querySelector("table")

    let tds = table.querySelectorAll("td")

    let newTr = document.createElement("tr")

    let newIndex = document.createElement("td")

    newIndex.textContent = (parseInt(tds[tds.length - 3].textContent) + 1).toString()

    let newName = document.createElement("td")

    newName.innerText = name.value

    console.log(newName.textContent)

    let newLastname = document.createElement("td")

    newLastname.textContent = lastname.value

    newTr.appendChild(newIndex)

    newTr.appendChild(newName)

    newTr.appendChild(newLastname)

    table.appendChild(newTr)

})

// Добавляю td и tr із значенням input

CSS file:

td {

    border: 1px black solid;

    padding: 3px;

}

table {

    border-collapse: collapse;

    border-spacing: 0;

}

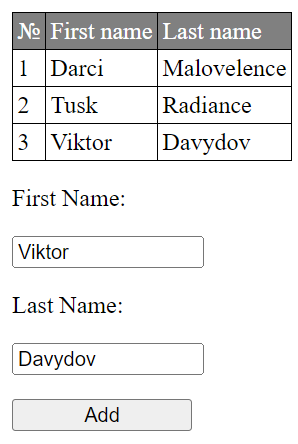
button,

input {

    width: 120px;

}

**Результати виконання:**

****

**Завдання 2:**

В елемент textarea вводяться нулі та одиниці і з них потрібно сформувати квадратики чорного і білого кольору.

**Лістинг програми:**

HTML file:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <script defer src="script.js"></script>

    <div>

        <textarea rows="4" cols="20"></textarea>

        <table></table>

    </div>

</body>

</html>

Java Script file:

document.querySelector("textarea").addEventListener("input", (event) => {

    let table = document.querySelector("table");

    let splitedArea = event.target.value.split('\n');

    table.innerHTML = "";

    for (let i = 0; i < splitedArea.length; ++i) {

        let newRow = document.createElement("tr");

        for (let j = 0; j < splitedArea[i].length; ++j) {

            let newCol = document.createElement("td");

            if (splitedArea[i][j] == "1") {

                newCol.classList.add("sketchedCol");

            }else{

                newCol.classList.add("sketchedCol2");

            }

            newRow.appendChild(newCol)

        }

        table.appendChild(newRow)

    }

})

// Якщо стоїть 1, то квадратик = синій, якщо ні = не замальовую

CSS file:

td {

    border: 1px black solid;

    width: 80px;

    height: 80px;

}

.sketchedCol {

    background-color: yellow;

}

.sketchedCol2{

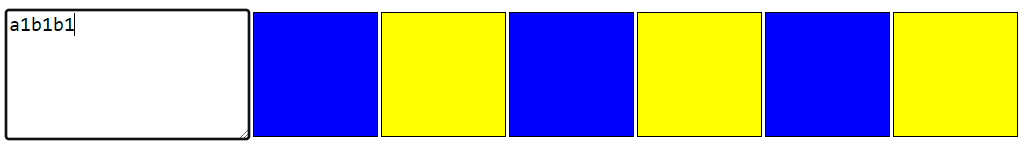
    background-color: blue;;

}

div {

    display: flex;

**Результати виконання:**

****

**Завдання 3:**

Вибрати потрібні кольори з палітри кольорів та залити ними даний прямокутник (блок).

**Лістинг програми:**

HTML File:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <script defer src="script.js"></script>

    <div>

        <table>

            <tr>

                <td style="background-color: brown;"></td>

                <td style="background-color: red;"></td>

                <td style="background-color: orange;"></td>

            </tr>

            <tr>

                <td style="background-color: yellow;"></td>

                <td style="background-color: greenyellow;"></td>

                <td style="background-color: green;"></td>

            </tr>

            <tr>

                <td style="background-color: lightskyblue;"></td>

                <td style="background-color: blue;"></td>

                <td style="background-color: darkblue;"></td>

            </tr>

        </table>

        <canvas class="block"></canvas>

    </div>

</body>

</html>

JavaScript file:

// Визначення функції обчислення позиції колірної зупинки в градієнті

let positionOfGradient = function (index, length) {

    return parseFloat(index \* 100 / length / 100);

}

// Виберіть всі елементи td на сторінці

let tds = document.querySelectorAll("td");

tds.forEach((td) => {

    // Додавання слухача події кліка до кожного осередку

    td.addEventListener('click', (event) => {

        event.target.classList.toggle('chosen');

        let chosen\_tds = document.querySelectorAll("td.chosen");

        if (chosen\_tds.length > 1) {

            // Якщо вибрано два або більше осередки

            // Вибір елемента canvas

            let canvas = document.querySelector("canvas");

            let ctx\_canvas = canvas.getContext("2d");

            // Створення лінійного градієнта

            let gradient = ctx\_canvas.createLinearGradient(0, 0, 0, 170);

            // Додавання колірних зупинок у градієнт на основі вибраних осередків

            chosen\_tds.forEach((chosen\_td, index) => {

                console.log(positionOfGradient(index, chosen\_tds.length - 1));

                gradient.addColorStop(positionOfGradient(index, chosen\_tds.length - 1), chosen\_td.style.backgroundColor);

            })

            ctx\_canvas.fillStyle = gradient;

            // Заливка всього canvas

            ctx\_canvas.fillRect(0, 0, canvas.width, canvas.height);

        } else if (chosen\_tds.length === 1) {

            // Якщо вибрана лише одна комірка

            // Завдання кольору заливки контексту canvas кольором вибраної комірки

            document.querySelector("canvas").getContext("2d").fillStyle = chosen\_tds[0].style.backgroundColor;

            // Заливка всього canvas

            document.querySelector("canvas").getContext("2d").fillRect(0, 0, document.querySelector("canvas").width, document.querySelector("canvas").height);

        } else {

            // Если ячейки не выбраны

            // Завдання кольору заливки контексту canvas білим кольором

            document.querySelector("canvas").getContext("2d").fillStyle = "white";

            // Заливка всього canvas

            document.querySelector("canvas").getContext("2d").fillRect(0, 0, document.querySelector("canvas").width, document.querySelector("canvas").height);

        }

    })

})

// Вибір усіх вибраних комірок після обробки подій

let chosen\_tds = document.querySelectorAll("td.chosen")

chosen\_tds.forEach((event) => {

    console.log(chosen\_tds.style.backgroundColor)

})

CSS file:

table {

    border-collapse: collapse;

    border-spacing: none;

}

td {

    width: 60px;

    height: 60px;

    border: 1px solid black;

}

.chosen {

    border: 5px solid black;

}

div {

    display: flex;

    justify-content: center;

}

.block {

    border: 1px solid black;

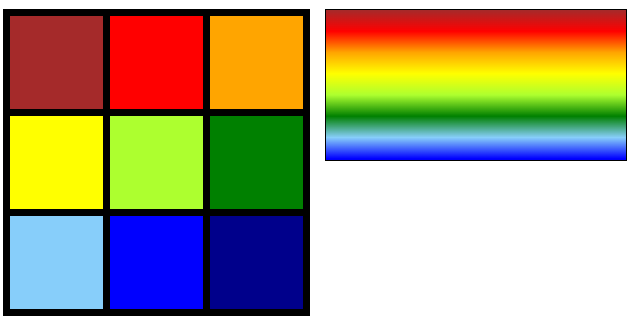
    width: 200px;

    height: 100px;

    margin-left: 10px;

}

**Результати виконання:**



**Завдання 4:**

Є блоки з текстом. В кожному з блоків є кнопки *Видалити* та *Редагувати*

При натисканні на кнопку *Видалити*, даний блок зникає. При натисканні на *Редагувати* блок переходить в стан редагування (наприклад, з допомогою *textarea*). При знятті фокуса з *textarea*, блок повертається в звичайний стан.

**Лістинг програми:**

HTML file:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <script defer src="script.js"></script>

    <div>

        <button id="del"></button>

        <button id="edit"></button><br>

        <textarea cols="30" rows="10"

            disabled>Lorem ipsum dolor sit amet consectetur adipisicing elit. Impedit at sint doloribus itaque consectetur natus, sit culpa suscipit! Voluptate quas nulla quam, iure a ipsam nemo recusandae et culpa quasi.</textarea>

    </div>

    <div>

        <button id="del"></button>

        <button id="edit"></button><br>

        <textarea cols="30" rows="10"

            disabled>Lorem ipsum dolor sit amet consectetur adipisicing elit. Eaque error praesentium nemo magni distinctio dolore aspernatur at consequatur soluta perferendis voluptatibus ratione accusantium, earum nesciunt molestias omnis enim quam! Sed.</textarea>

    </div>

    <div>

        <button id="del"></button>

        <button id="edit"></button><br>

        <textarea cols="30" rows="10"

            disabled>Lorem ipsum dolor sit amet consectetur adipisicing elit. Unde voluptates mollitia voluptate voluptatem voluptatum eveniet autem soluta quae accusamus distinctio sint esse architecto dolorem eos voluptas dolores perspiciatis, assumenda ab.</textarea>

    </div>

</body>

</html>

JavaScript file:

document.querySelectorAll("#del").forEach((but) => {

    but.addEventListener('click', (event) => {

        event.target.parentElement.remove();

    })

})

document.querySelectorAll("#edit").forEach((but) => {

    but.addEventListener('click', (event) => {

        event.target.parentElement.querySelector("textarea").disabled = !event.target.parentElement.querySelector("textarea").disabled

    })

})

CSS file:

button {

    width: 20px;

    height: 20px;

    padding: 0px;

    cursor: pointer;

    background-size: 100% 100%;

}

#edit {

    background-image: url("pen.png");

}

#del {

    background-image: url("del.png");

}

div {

    background-color: gray;

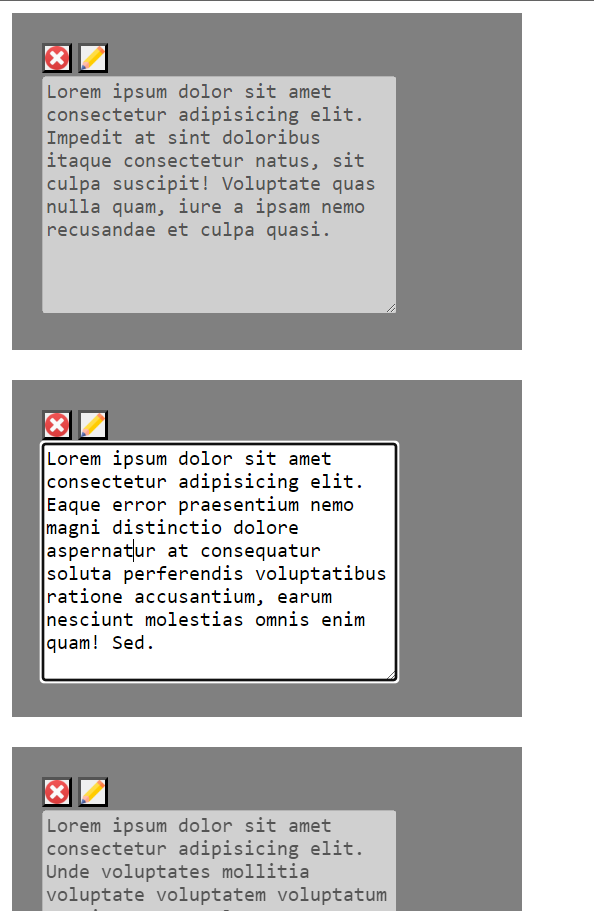
    width: 300px;

    padding: 20px;

    margin-bottom: 20px;

}

**Результати виконання:**

****

**Завдання 5:**

**Задати можливість заливати квадратики вибраним кольором.**

Є палітра кольорів, в якій клацанням можна вибрати лише один колір. І є прямокутне полотно (контейнер, блок), яке заповнене маленькими блоками-квадратиками. При клацанні курсором на ці квадратики, вони повинні заливатись вибраним кольором.

**Лістинг програми:**

HTML file:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <script defer src="script.js"></script>

    <table class="palette">

        <tr>

            <td style="background-color: brown;"></td>

            <td style="background-color: red;"></td>

            <td style="background-color: orange;"></td>

        </tr>

        <tr>

            <td style="background-color: yellow"></td>

            <td style="background-color: lightgreen;"></td>

            <td style="background-color: green;"></td>

        </tr>

        <tr>

            <td style="background-color: lightblue;"></td>

            <td style="background-color: blue;"></td>

            <td style="background-color: darkblue;"></td>

        </tr>

    </table>

    <table class="paint">

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

    </table>

</body>

</html>

Java Script file:

let paletteTds = document.querySelectorAll("table.palette tr td")

paletteTds.forEach((paletteTd) => {

    paletteTd.addEventListener('click', (event) => {

        paletteTds.forEach((td) => {

            if (event.target !== td && td.classList.contains("selected")) {

                td.classList.remove("selected")

            }

        })

        selectedTd = paletteTd

        paletteTd.classList.add("selected")

    })

})

//

let paintTds = document.querySelectorAll("table.paint tr td")

paintTds.forEach((paintTd) => {

    paintTd.addEventListener('mousedown', (event) => {

        event.target.style.backgroundColor = selectedTd.style.backgroundColor

    })

})

CSS file:

td {

    width: 40px;

    height: 40px;

    border: 1px solid black;

}

.selected {

    outline: 4px solid black;

}

table {

    border-collapse: collapse;

    border-spacing: 0px;

    margin-bottom: 10px;

}

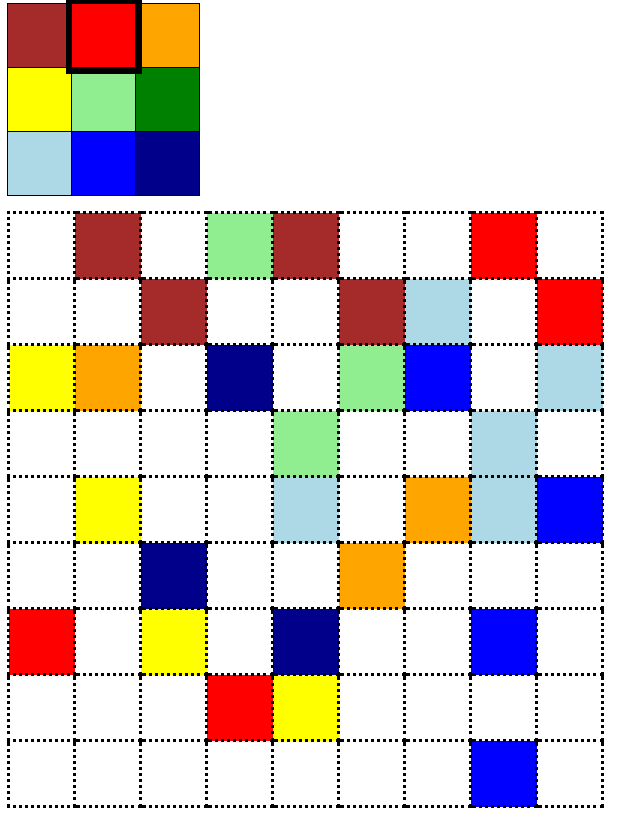
table.paint td {

    border: 2px black solid;

    border-style: dotted;

}

**Виконання програми:**

****

**Завдання 6:**

Є ряд гральних карт, повернутих до нас «сорочкою». При клацанні на карту, вона повертається до нас  лицьовим боком протягом 1-2 секунд (повинен бути саме поворот). При повторному клацанні, карта знову повертається «сорочкою».

**Лістинг програми:**

HTML File:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <script defer src="script.js"></script>

    <div id="flip-container">

        <div><img id="backSide" src="back.jpg"></div>

        <div><img id="frontSide" src="front.jpg"></div>

    </div>

</body>

</html>

Java Script file:

let front = document.getElementById("frontSide")

let back = document.getElementById("backSide")

document.getElementById("flip-container").addEventListener('click', (event) => {

    front.classList.toggle("rotate")

    back.classList.toggle("rotate")

})

CSS file:

div {

    position: absolute;

    perspective: 1000px;

    top: 50%;

    left: 50%;

    transform: translate(-50%, -50%);

}

img {

    transition: transform 0.6s;

    height: 500px;

}

.rotate {

    transform: rotateY(180deg);

}

#frontSide {

    backface-visibility: hidden;

}

**Виконання програми:**

****

****

**Завдання 7:**

Створити html-сторінку зі списком повідомлень на форумі та формою для додавання нового повідомлення. Після заповнення форми додати повідомлення до списку на екрані:

**Лістинг програми:**

HTML file:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <script defer src="script.js"></script>

    <div class="paragraph">

        <div class="head">

            <div class="name">Ann</div>

            <div class="date">Wed Nov 02 2022 17:26</div>

        </div>

        <div class="content">Lorem ipsum dolor sit amet consectetur adipisicing elit. Accusantium nihil commodi ipsam

            repellendus cum quasi ipsum pariatur tempore exercitationem nisi blanditiis quia veritatis reprehenderit,

            necessitatibus autem reiciendis perferendis architecto adipisci!</div>

    </div>

    <div class="paragraph">

        <div class="head">

            <div class="name">Bob</div>

            <div class="date">Wed Nov 02 2022 17:26</div>

        </div>

        <div class="content">Lorem ipsum dolor sit amet consectetur adipisicing elit. Pariatur reiciendis dignissimos

            expedita odio, nemo minus provident temporibus voluptas cumque, harum ipsa optio quaerat architecto adipisci

            a minima dicta quod velit.</div>

    </div>

    <div class="paragraph">

        <div class="head">

            <div class="name">Alice</div>

            <div class="date">Wed Nov 02 2022 17:26</div>

        </div>

        <div class="content">Lorem ipsum dolor sit amet consectetur adipisicing elit. Ullam enim aliquam molestiae

            facilis reiciendis. Ducimus incidunt dolores explicabo officia ab nostrum deserunt libero, perferendis

            praesentium itaque eos autem error provident?</div>

    </div>

    <h1>Add comment</h1>

    <div>

        <label>Name: </label>

        <input type="text" id="name"><br>

        <label>Comment: </label>

        <textarea cols="30" rows="2" id="comment"></textarea><br>

        <button id="send">Send</button>

    </div>

</body>

</html>

Java Script file:

const months = [

    "Jan", "Feb", "Mar", "Apr", "May", "Jun",

    "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"];

const days = ["Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"];

let send = document.getElementById("send");

send.addEventListener("click", (event) => {

    let date = new Date();

    let newParagraph = document.createElement("div");

    let newHead = document.createElement("div");

    let newName = document.createElement("div");

    let newDate = document.createElement("div");

    let newContent = document.createElement("div");

    newName.textContent = document.getElementById("name").value;

    console.log(document.getElementById("comment").value)

    newContent.textContent = document.getElementById("comment").value;

    newDate.textContent = `${days[date.getDay()]} ${months[date.getMonth()]} ${date.getDate()} ${date.getFullYear()} ${date.getHours()}:${date.getMinutes()}`;

    newHead.appendChild(newName)

    newHead.appendChild(newDate)

    newHead.classList.add("head")

    newContent.classList.add("content")

    newParagraph.appendChild(newHead)

    newParagraph.appendChild(newContent)

    newParagraph.classList.add("paragraph")

    document.querySelector("h1").insertAdjacentElement("beforebegin", newParagraph)

})

CSS file:

.head {

    display: flex;

    justify-content: space-between;

    height: 30px;

}

.head,

.content {

    border: 1px solid black;

}

.paragraph {

    margin-bottom: 10px;

}

label,

input,

textarea,

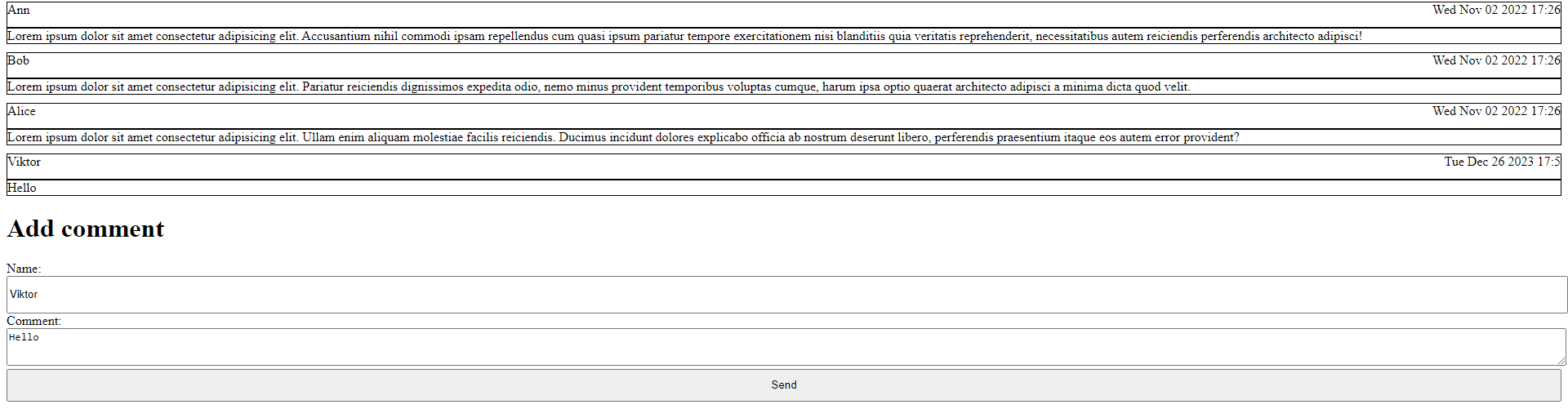
button {

    width: 100%;

    height: 40px;

}

**Виконання програми:**

****

**Завдання 8:**

Створити html-сторінку зі списком книг. При натисканні на книгу, колір фону повинен змінюватися на зелений. Врахуйте, що при повторному натисканні на іншу книгу, попередньої – необхідно повертати колишній колір.

**Лістинг програми:**

HTML file:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <script defer src="script.js"></script>

    <div>1 <span>«Володар кілець» Джон Р. Р. Толкін</span></div>

    <div>2 <span>«Гордість і упередження» Джейн Остін</span></div>

    <div>3 <span>«Темні початки» Філіп Пулман</span></div>

    <div>4 <span>«Автостопом по галактиці» Дуглас Адамс</span></div>

    <div>5 <span>«Гаррі Поттер і Кубок вогню» Джоан Роулінг</span></div>

    <div>6 <span>«Вбити пересмішника» Харпер Лі</span></div>

</body>

</html>

Java Script file:

console.log(document.querySelectorAll("span"))

document.querySelectorAll("span").forEach((span) => {

    span.addEventListener('click', () => {

        document.querySelectorAll("span").forEach((otherSpan) => {

            if (span.textContent !== otherSpan.textContent) {

                otherSpan.classList.remove("selected")

            }

        })

        span.classList.toggle("selected")

    })

})

});

CSS file:

div,

label {

    color: rgb(138, 37, 138);

    cursor: pointer;

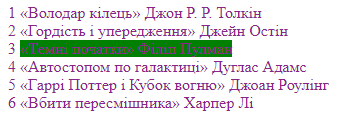
}

.selected {

    background-color: green;

}

**Виконання програми:**

****

GitHub:

https://github.com/djproblematic/frontend