**Лабораторна робота № 2**

*Знайомство з базовими алгоритмічними конструкціями мови програмування JavaScript та основами роботи з HTML-елементами та стилями.*

**Хід роботи:**

**Завдання 1:**

**1.1:**

Додати на сторінку 3 абзаци тексту на власний розсуд.

Написати JS-код, який:

- задасть для тега **body** шрифт тексту **Arial**

- додасть **inline-стилі** до кожного з абзаців (у даному завданні потрібно додати саме **inline-стилі**, а не класи):

- перший абзац:

- колір тексту: помаранчевий

- фон абзацу: жовтий

- розмір шрифту - 16 пікселів

- внутрішній відступ - 5 пікселів

- вирівнювання тексту - по центру

- другий абзац:

- колір тексту: синій

- фон абзацу: світло синій

- розмір шрифту: 14 пікселів

- внутрішній відступ - 5 пікселів

- вирівнювання тексту - по правій стороні

- третій абзац:

- колір тексту: червоний

- фон абзацу: світло червоний

- розмір шрифту: 12 пікселів

- внутрішній відступ - 5 пікселів

- вирівнювання тексту - по лівій стороні

**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>

    <p id="first">Lorem ipsum dolor sit amet consectetur adipisicing elit. Deleniti ab laudantium illum obcaecati cupiditate culpa ipsum qui magni similique quasi, voluptatum commodi recusandae ipsam soluta nesciunt earum eveniet, officia nobis?</p>

    <p id="second">Lorem ipsum, dolor sit amet consectetur adipisicing elit. Iusto maiores architecto, enim corporis repudiandae atque quae vero exercitationem qui libero explicabo iste mollitia nemo dolores odio? Accusantium explicabo nostrum fugit.</p>

    <p id="third">Lorem ipsum, dolor sit amet consectetur adipisicing elit. Modi ex magnam id, quaerat itaque obcaecati recusandae consequatur numquam dicta tempore nostrum earum sint quos expedita blanditiis? Odit eveniet cum commodi!</p>

    <script src="script.js"></script>

</body>

</html>

Java Script file:

function changestyle(element, color, backgroundColor, fontsize, padding, textAlign) {

    element.style.color = color;

    element.style.backgroundColor = backgroundColor;

    element.style.fontSize = fontsize;

    element.style.padding = padding;

    element.style.textAlign = textAlign;

}

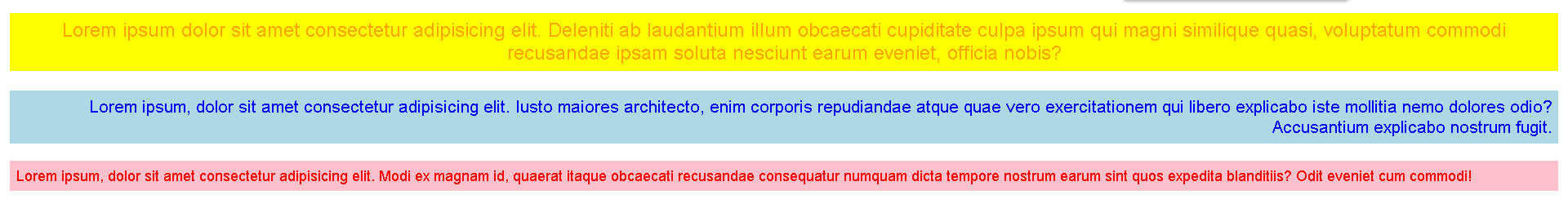
changestyle(document.getElementById("first"), "orange", "yellow", "16px", "5px", "center");

changestyle(document.getElementById("second"), "blue", "lightblue", "14px", "5px", "right");

changestyle(document.getElementById("third"), "red", "pink", "12px", "5px", "left");

document.body.style.fontFamily = "arial";

**Результати виконання:**

****

**1.2:**В HTML-файлі після 3-х абзаців тексту додати табличку розміром 5х5.

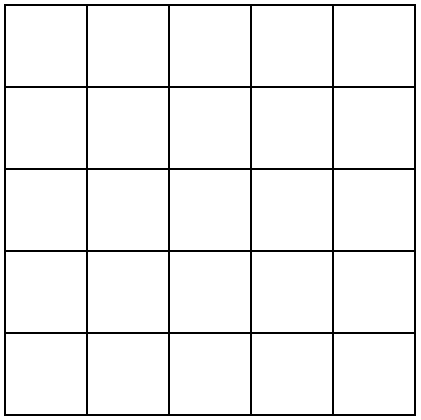
У файлі стилів розмістити клас для зафарбування фону елемента жовтим кольором:

.selected {

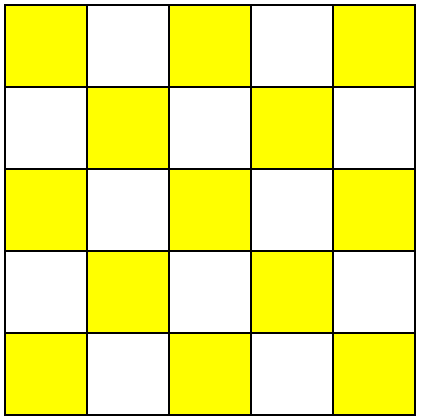
   background-color: yellow;

}

та стилі для оформлення таблиці за прикладом:



Напишіть JS-код, який буде знаходити комірки таблиці та додавати клас “**selected**” до клітинок таблиці таким чином, щоб отримати такий результат:



Змінити розмір таблиці на інший та перевірити, чи буде працювати написаний код для таблиці з іншим розміром (без необхідності внесення змін у JS-код):

**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></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

        <tr>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

            <td></td>

        </tr>

    </table>

    <script src="script.js"></script>

</body>

</html>

CSS file:

td {

    border: 1px black solid;

    width: 100px;

    height: 100px;

}

.selected {

    background-color: yellow;

}

Java Script file:

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

tds.forEach((kkk, index) => {

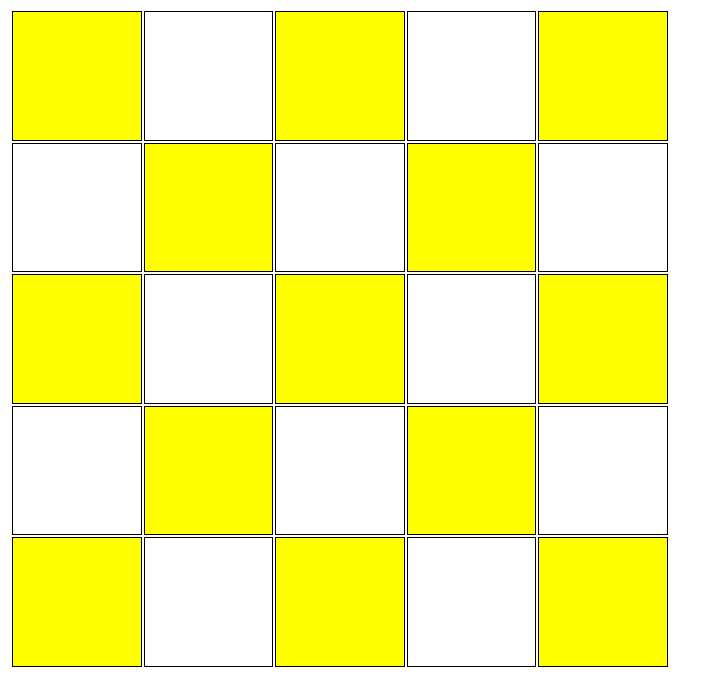
    if (index % 2 == 0) {

        kkk.classList.add("selected");

    }

})

**Результати виконання:**



**Завдання 2:**

Дано рядок тексту:

let text = `Lorem ipsum dolor sit amet, consectetur adipisicing elit. Deserunt ea earum incidunt minima nihil. Blanditiis consectetur cumque debitis delectus dolor dolorem dolorum, eos laboriosam, modi nemo, quasi quos sint voluptatibus!`;

1. Для заданого тексту утворити новий рядок, який буде сформований із символів рядка ***text***, які мають індекси 12, 6, 18, 25. Виведіть в консоль сформований рядок.
2. Виведіть у консоль рядок, утворений з рядка ***text*** шляхом приведення до нижнього регістру.
3. Виведіть всі позиції, в яких зустрічається послідовність символів***“in”***
4. Утворіть з рядка ***text*** масив підрядків шляхом розділення тексту на частини за роздільником ***“пробіл”*** та виведіть його.
5. Утворіть та виведіть рядок, який є інвертованим рядком ***text*** (текст, літери якого розташовуються у зворотному порядку).

* Допишіть код функції, яка приведе першу літеру тексту до верхнього регістру:  
  function ucFirst(txt) {  
     let newTxt =   
     return newTxt;  
  }  
  console.log(ucFirst(‘some text’));

**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>

</head>

<body>

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

</body>

</html>

Java Script file:

let text = `Lorem ipsum dolor sit amet, consectetur adipisicing elit. Deserunt ea earum incidunt minima nihil. Blanditiis consectetur cumque debitis delectus dolor dolorem dolorum, eos laboriosam, modi nemo, quasi quos sint voluptatibus!`;

let newStr = text[12] + text[6] + text[18] + text[25]

console.log(newStr)

console.log(text.toLocaleLowerCase())

let currentIndex = -1;

while ((currentIndex = text.indexOf('in', currentIndex + 1)) != -1) {

    console.log(currentIndex);

}

console.log(text.split(" "))

console.log(text.split('').reverse().join(''))

function ucFirst(txt) {

    let indexOfUpperLetter = 0

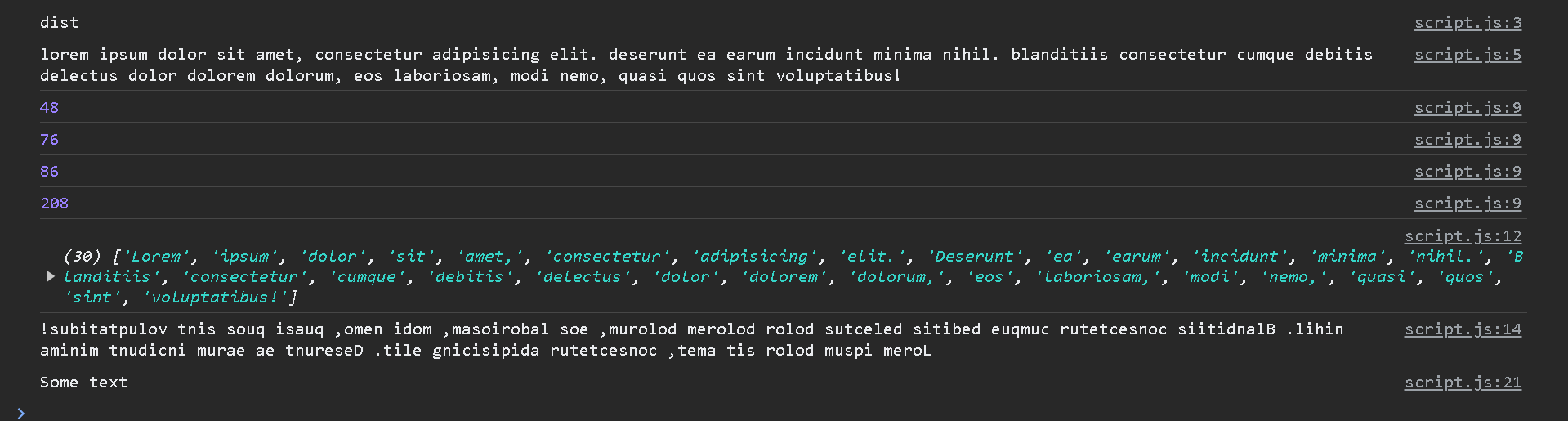
    let newTxt = txt[indexOfUpperLetter].toUpperCase() + txt.slice(indexOfUpperLetter + 1, txt.lenght)

    return newTxt;

}

console.log(ucFirst('some text'));

**Результати виконання:**

****

**Завдання 3:**

Створити HTML-список з потрійною вкладеністю та стилізувати його з використанням селекторів CSS.

Вимоги:

* Використовуйте CSS-селектори для стилізації різних рівнів вкладеності без використання селекторів **.class, .id**.
* Додайте переходи та анімації для покращення візуальної привабливості списку
* Cтворити додаткові стилі для активних і наведених елементів списку.

**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>Lab 1 Task 3</title>

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

</head>

<body>

    <div>

        <ul>

            <li>Рівень 1 Element 1

                <ul>

                    <li>Рівень 2 Елемент 1

                        <ul>

                            <li>Рівень 3 Елемент 1</li>

                            <li>Рівень 3 Елемент 2</li>

                        </ul>

                    </li>

                    <li>Рівень 2 Елемент 2

                        <ul>

                            <li>Рівень 3 Елемент 1</li>

                            <li>Рівень 3 Елемент 2</li>

                        </ul>

                    </li>

                </ul>

            </li>

            <li>Рівень 1 Елемент 2</li>

        </ul>

    </div>

</body>

</html>

CSS file:

li {

    margin-top: 10px;

}

div>ul>li:nth-child(1) {

    background-color: #F4EBDB;

    padding: 10px;

    width: 50%;

    min-width: 300px;

    list-style: none;

    border-left: 4px solid #2C4A52;

    border-radius: 5px;

}

div>ul>li:nth-child(2) {

    list-style: none;

    background-color: #8E9B97;

    border-radius: 5px;

}

ul ul>li:nth-child(2) {

    border-left: 4px solid #2C4A52;

    border-radius: 5px;

}

ul ul>li:nth-child(1),

ul ul ul>li:nth-child(2) {

    list-style: none;

    border-left: 4px solid #8E9B97;

    border-radius: 5px;

}

div>ul>li:nth-child(1),

div>ul>li:nth-child(2),

ul ul>li:nth-child(1),

ul ul>li:nth-child(2) {

    padding: 10px;

    width: 50%;

    min-width: 300px;

    list-style: none;

}

GitHub:

https://github.com/djproblematic/frontend