**Лабораторна робота № 10**

**LocalStorage. JSON. Наслідування класу. Статичні властивості та методи***.*

**Хід роботи:**

**Завдання 1:**

Дана форма з **input**, **textarea**, **checkbox**, **radio**, **select** і тп. Користувач вводить якісь дані і закриває сторінку (не факт, що він заповнив всю форму). Зробіть так, щоб при наступному заході на сторінку введені ним раніше дані стояли на своїх місцях

**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>Task 1</title>

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

</head>

<body>

    <form id="myForm">

        <label for="name">Ім'я:</label>

        <input type="text" id="name" name="name" placeholder="Введіть Ім'я">

        <label for="email">Email:</label>

        <input type="text" id="email" name="email" placeholder="Введіть пошту">

        <label for="message">Повідомлення:</label>

        <textarea id="message" name="message" placeholder="Необов'язково"></textarea>

        <label>Стать:</label>

        <div class="gender-wrapper">

            <div>

                <input type="radio" id="male" name="gender" value="male">

                <label for="male">Чоловіча</label>

            </div>

            <div>

                <input type="radio" id="female" name="gender" value="female">

                <label for="female">Жіноча</label>

            </div>

        </div>

        <label for="subscribe">Підписатися:</label>

        <input type="checkbox" id="subscribe" name="subscribe">

        <label for="country">Країна:</label>

        <select id="country" name="country">

            <option value="ukraine">Україна</option>

            <option value="germany">Германія</option>

            <option value="usa">США</option>

        </select>

        <button type="button" onclick="saveData()">Зберегти дані</button>

    </form>

    <script src="https://code.jquery.com/jquery-3.6.4.min.js"></script>

    <script src="script.js"></script>

</body>

</html>

Java Script file:

$(document).ready(function () {

    loadData();

});

$("#myForm").on("input change", "input, textarea, select", function () {

    saveData();

});

$(window).on("beforeunload", function () {

    saveData();

});

function saveData() {

    var form = $("#myForm");

    form.find("input, textarea, select").each(function () {

        var id = $(this).attr("id");

        var value;

        if ($(this).is(":checkbox"))

            value = $(this).prop("checked");

        else

            if ($(this).is(":radio"))

                value = $(this).prop("checked");

            else

                value = $(this).val();

        localStorage.setItem(id, value);

    });

}

function loadData() {

    var form = $("#myForm");

    form.find("input, textarea, select").each(function () {

        var id = $(this).attr("id");

        var savedValue = localStorage.getItem(id);

        if (savedValue !== null) {

            if ($(this).is(":checkbox"))

                $(this).prop("checked", savedValue === "true");

            else

                if ($(this).is(":radio"))

                    $(this).prop("checked", savedValue === "true");

                else

                    $(this).val(savedValue);

        }

    });

}

CSS file:

body {

    font-family: 'Franklin Gothic Medium', 'Arial Narrow', Arial, sans-serif;

    background-color: #c659bd;

    margin: 0;

    padding: 0;

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

}

form {

    background-color: #fff;

    padding: 20px;

    border-radius: 30px;

    box-shadow: 0 0 15px rgba(0, 0, 0, 0.1);

    width: 500px;

    box-sizing: border-box;

}

label {

    display: block;

    margin-bottom: 8px;

}

input,

textarea,

select,

button {

    width: 100%;

    padding: 15px;

    margin-bottom: 15px;

    box-sizing: border-box;

}

.gender-wrapper {

    display: flex;

    justify-content: space-around;

    margin-bottom: 15px;

}

button {

    background-color: #4c63af;

    color: #fff;

    border: none;

    border-radius: 30px;

    cursor: pointer;

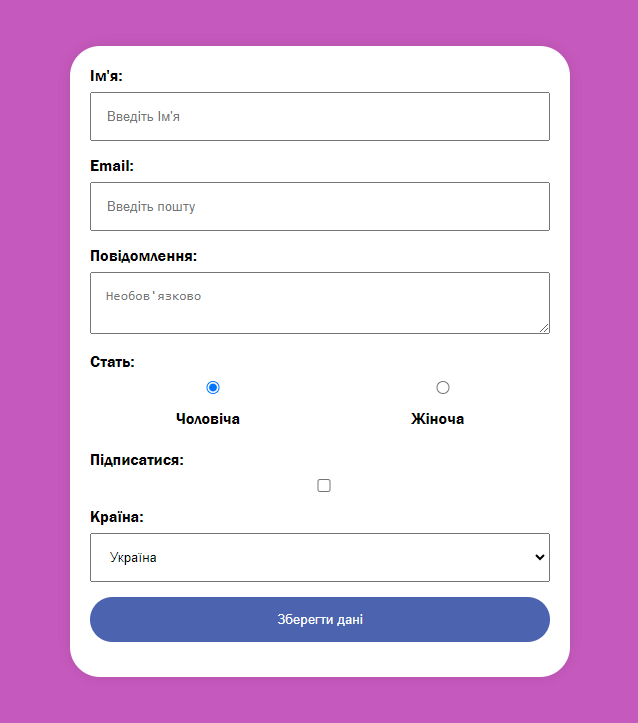
}

button:hover {

    background-color: #f3e712;

}

**Результати виконання:**

****

**Завдання 2:**

Створіть на сторінці текстовий input, який буде приймати імена файлів як масив в форматі JSON (наприклад, [" file1.jpg "," file2.gif "," file34.gif "] )

* По натисканню на кнопку повинні з'явитися зображення в маленькому розмірі, по натисканню на які відкриватися ці зображення в реальному розмірі
* Якщо в текстовому полі введені дані не відповідають правильному формату JSON, необхідно повідомити про це користувача

**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>Task 2</title>

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

</head>

<body>

    <input type="text" id="jsonInput" placeholder="Введіть JSON масив файлів"

        value='["1.jpg", "2.jpg", "3.jpg", "4.jpg"]'>

    <button onclick="showImages()">Показати зображення</button>

    <div id="errorMessage"></div>

    <div id="imageContainer"></div>

    <div id="fullImageContainer">

        <img id="fullImage" onclick="hideFullImage()" alt="Full Image">

    </div>

    <script src="script.js"></script>

</body>

</html>

Java Script file:

function showImages() {

    var jsonInput = document.getElementById("jsonInput").value;

    try {

        var filesArray = JSON.parse(jsonInput);

        document.getElementById("errorMessage").innerHTML = "";

        document.getElementById("imageContainer").innerHTML = "";

        for (var i = 0; i < filesArray.length; i++) {

            var thumbnail = document.createElement("img");

            thumbnail.src = filesArray[i].trim();

            thumbnail.className = "thumbnail";

            thumbnail.title = "Натисніть, щоб відкрити в реальному розмірі";

            thumbnail.addEventListener("click", function () {

                showFullImage(this.src);

            });

            document.getElementById("imageContainer").appendChild(thumbnail);

        }

    } catch (error) {

        document.getElementById("errorMessage").innerHTML = "Помилка у форматі JSON";

    }

}

function showFullImage(src) {

    var fullImage = document.getElementById("fullImage");

    fullImage.src = src;

    document.getElementById("fullImageContainer").style.display = "flex";

}

function hideFullImage() {

    document.getElementById("fullImageContainer").style.display = "none";

}

CSS file:

body {

    font-family: 'Trebuchet MS', 'Lucida Sans Unicode', 'Lucida Grande', 'Lucida Sans', Arial, sans-serif;

    margin: 0;

    padding: 100px;

    display: flex;

    flex-direction: column;

    align-items: center;

}

#imageContainer {

    display: flex;

    flex-wrap: wrap;

    justify-content: center;

}

.thumbnail {

    margin: 15px;

    cursor: pointer;

    width: 50px;

    height: 50px;

    object-fit: cover;

    transition: transform 0.3s ease-in-out;

}

.thumbnail:hover {

    transform: scale(1.2);

}

#jsonInput {

    margin-bottom: 10px;

}

#errorMessage {

    color: red;

    margin-bottom: 10px;

}

#fullImageContainer {

    display: none;

}

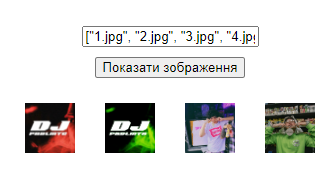
#fullImage {

    max-width: 30%;

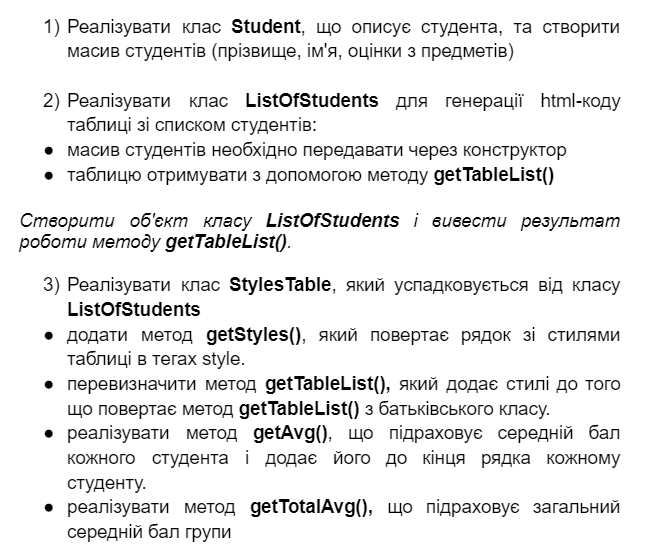
    max-height: 30%;

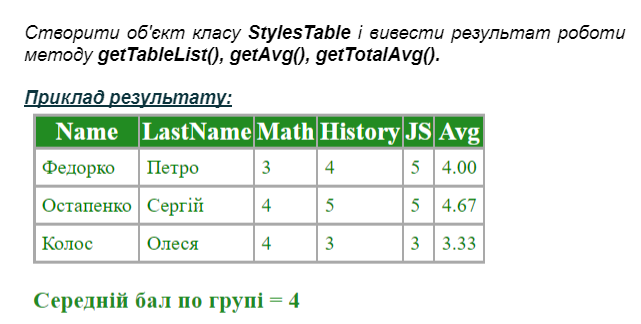
}

**Результати виконання:**

****

****

**Завдання 3:  
**

****

**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>Task 3</title>

</head>

<body>

    <script src="script.js"></script>

</body>

</html>

JavaScript file:

class Student {

    constructor(lastName, firstName, grades) {

        this.lastName = lastName;

        this.firstName = firstName;

        this.grades = grades;

    }

    getAverageGrade() {

        if (this.grades.length === 0) return 0;

        const sum = this.grades.reduce((acc, grade) => acc + grade, 0);

        return sum / this.grades.length;

    }

}

class ListOfStudents {

    constructor(students) {

        this.students = students;

    }

    getTableList() {

        let tableHtml = '<table border="1"><tr><th>Last Name</th><th>First Name</th><th>Math</th><th>History</th><th>Sport</th><th>Average Grade</th></tr>';

        this.students.forEach(student => {

            tableHtml += `<tr><td>${student.lastName}</td><td>${student.firstName}</td><td>${student.grades[0]}</td><td>${student.grades[1]}</td><td>${student.grades[2]}</td><td>${student.getAverageGrade().toFixed(2)}</td></tr>`;

        });

        tableHtml += '</table>';

        return tableHtml;

    }

}

class StylesTable extends ListOfStudents {

    getStyles() {

        return '<style>table { border-collapse: collapse; width: 50%; } th, td { padding: 10px; text-align: left; }</style>';

    }

    getTotalAvg() {

        const totalAvg = (this.students.reduce((acc, student) => acc + student.getAverageGrade(), 0) / this.students.length).toFixed(2);

        return `<div style="background-color: lightgreen; padding: 10px; margin-top: 10px;"><strong>Середній бал по групі:</strong> ${totalAvg}</div>`;

    }

}

const studentsArray = [

    new Student('Федорко', 'Петро', [3, 4, 5]),

    new Student('Остапенко', 'Сергій', [4, 5, 5]),

    new Student('Колос', 'Олеся', [4, 3, 3]),

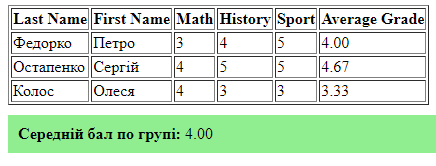
];

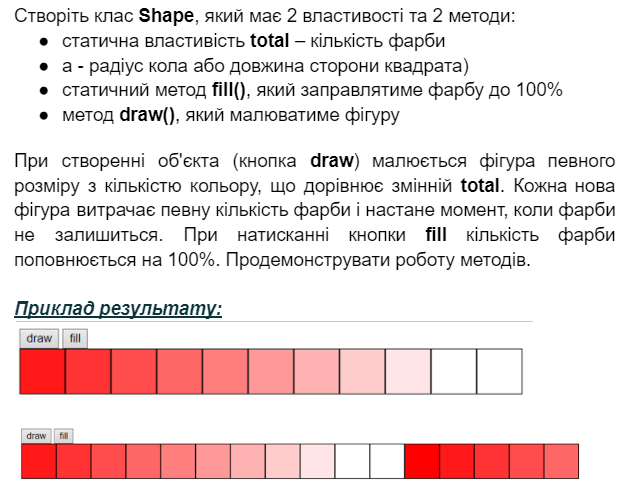
const styledTable = new StylesTable(studentsArray);

document.body.innerHTML += styledTable.getTableList();

document.body.innerHTML += styledTable.getTotalAvg();

**Результати виконання:**



**Завдання 4:  
**

**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>Task 4</title>

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

</head>

<body>

    <button onclick="drawShape()">Draw</button>

    <button onclick="fillPaint()">Fill</button>

    <div class="shape-container"></div>

    <script src="script.js"></script>

</body>

</html>

JavaScript file:

class Shape {

    static total = 100;

    static fill() {

        this.total = 100;

        console.log('Paint filled to 100%');

    }

    static draw() {

        this.total -= 10;

    }

}

function drawShape() {

    const shapeElement = document.createElement('div');

    shapeElement.classList.add('shape');

    const transparency = Shape.total / 100;

    const color = `rgba(255, 0, 255, ${transparency})`;

    shapeElement.style.backgroundColor = color;

    Shape.draw()

    document.querySelector('.shape-container').appendChild(shapeElement);

}

function fillPaint() {

    Shape.fill();

}

CSS file:

.shape-container {

    display: flex;

}

.shape {

    width: 50px;

    height: 50px;

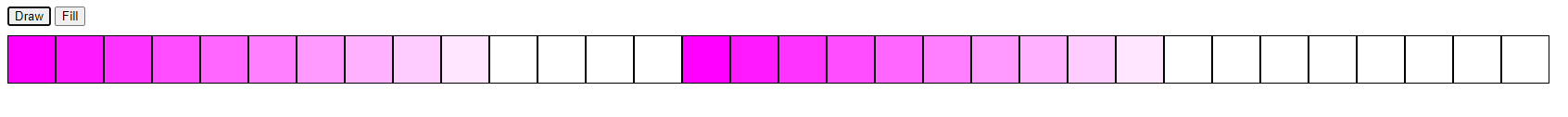
    border: 1px solid black;

    margin-block: 10px;

    transition: background-color 0.5s;

}

**Результати виконання:**



GitHub:

https://github.com/djproblematic/frontend