**Лабораторна робота №5**

**Завдання 1**

Дано чекбокси і кнопка. По натисканню на кнопку вивести в рядок значення вибраних елементів.

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="UTF-8">

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

        <title>Tusk 1</title>

    </head>

    <body>

        <form>

            <input type="checkbox" id="1">

            <label for="1">Ukrainian</label><br>

            <input type="checkbox" id="2">

            <label for="2">English</label><br>

            <input type="checkbox" id="3">

            <label for="3">Spanish</label><br>

            <p></p>

            <input type="button" value="Ok" onclick="check()">

        </form>

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

    </body>

</html>

JS

let input = document.querySelectorAll('input');

function check(){

    let text = "";

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

        if(input[i].checked){

            text += input[i].nextElementSibling.textContent + " ";

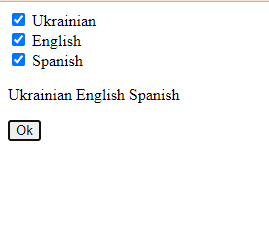
        }

    const p = document.querySelector('p');

    p.textContent = text;

}

Результат



**Завдання 2**

На сторінці розміщені електронні адреси з прапорцями (чекбоксами). Реалізуйте можливість формування рядку з вибраними електронними адресами. При позначенні електронної адреси, вона додається в div-елемент. При знятті чекбокса, адреса видаляється з елементу

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="UTF-8">

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

        <title>Tusk 2</title>

    </head>

    <body>

        <form>

            <input type="checkbox" id="1" onclick="check()">

            <label for="1">mountain@gmail.com</label><br>

            <input type="checkbox" id="2" onclick="check()">

            <label for="2">tree@gmail.com</label><br>

            <input type="checkbox" id="3" onclick="check()">

            <label for="3">world@gmail.com</label><br>

            <input type="checkbox" id="4" onclick="check()">

            <label for="4">cloud@gmail.com</label><br>

            <input type="checkbox" id="6" onclick="check()">

            <label for="5">river@gmail.com</label><br>

            <input type="checkbox" id="6" onclick="check()">

            <label for="6">wood@gmail.com</label><br>

            <input type="checkbox" id="7" onclick="check()">

            <label for="7">field@gmail.com</label><br>

            <input type="checkbox" id="8" onclick="check()">

            <label for="8">snow@gmail.com</label><br>

            <input type="checkbox" id="9" onclick="check()">

            <label for="9">rain@gmail.com</label><br>

            <input type="checkbox" id="10" onclick="check()">

            <label for="10">wind@gmail.com</label><br><br>

            <div></div>

        </form>

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

    </body>

</html>

JS

let input = document.querySelectorAll('input');

function check(){

    let text = "";

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

        if(input[i].checked){

            text += input[i].nextElementSibling.textContent + "; ";

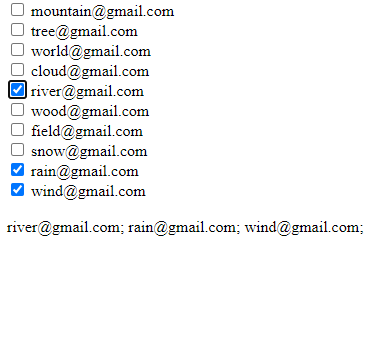
        }

    const p = document.querySelector('div');

    p.textContent = text;

}

Результат



**Завдання 3**

Створити JavaScript, який перевіряє знання таблиці множення. Веб-сторінка текстовий напис для показу загального рахунку, кнопку «наступне завдання», текстовий напис для показу завдання, радіокнопки для вибору правильного варіанту відповіді, та текстовий напис для виводу результатів перевірки. Вибір варіанту відповіді є сигналом до початку перевірки. Для кожного завдання користувач має лише одна спробу вибору.

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Tusk 3</title>

</head>

    <body>

        <h3>Питання №0/10</h3>

        <input type="button" value="Наступне завдання." onclick="checkAnswer(); action()">

        <p>2 x 3 = </p>

        <form action="">

            <input type="radio" name="myRadioGroup" id="1">

            <label for="1"></label><br>

            <input type="radio" name="myRadioGroup" id="2">

            <label for="2"></label><br>

            <input type="radio" name="myRadioGroup" id="3">

            <label for="3"></label><br>

            <input type="radio" name="myRadioGroup" id="4">

            <label for="4"></label><br>

        </form>

        <div> </div>

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

    </body>

</html>

JS

function getRandomInt(max) {

    return Math.floor(Math.random() \* max);

}

let answer = 0;

let input = document.querySelectorAll('input[type="radio"]');

let Count = 0;

let div = document.querySelector('div');

let rightAnswer = 0;

function action(){

    if(Count == 10)

        return;

    let first = getRandomInt(10);

    let second = getRandomInt(10);

    answer = first\*second;

    let argumentText = document.querySelector('p');

    argumentText.innerText = `${first} x ${second} = `;

    let trueAnswer = getRandomInt(input.length - 1)

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

        input[i].nextElementSibling.textContent = `${getRandomInt(60)}`;

    input[trueAnswer].nextElementSibling.textContent = `${first\*second}`

    console.log(`${trueAnswer} : ${first\*second}`)

}

function checkAnswer(){

    if(Count < 10)

        Count++;

    else{

        div.textContent = `Загальний рахунок ${rightAnswer \* 10}%(${rightAnswer} правильних відповідей з 10)`

        return

    }

    let maintext = document.querySelector('h3')

    let userAnswer = "";

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

        if(input[i].checked)

            userAnswer = input[i].nextElementSibling.textContent;

    if(userAnswer == `${answer}`){

        div.textContent = "Правильна відповідь!"

        rightAnswer++;

    }

    else

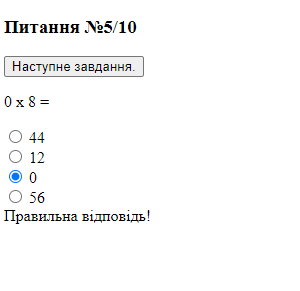
        div.textContent = `Помилка правильна відповідь ${answer}!`

    maintext.textContent = `Питання №${Count}/10`

}

action()

Результат



**Завдання 4**

Створити JavaScript, який перевіряє знання таблиці множення. Веб-сторінка містить текстовий напис для показу загального рахунку, кнопку «наступне завдання», текстовий напис для показу завдання, текстове поле для вводу відповіді, кнопку «перевірити» та текстовий напис для виводу результатів перевірки.

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Tusk 4</title>

</head>

    <body>

        <input type="button" value="Наступне завдання." onclick="action()"><br><br>

        <label for="answer">2 x 3 = </label> <input id="answer" type="text"> <input type="button" value="перевірити" onclick="check();">

        <div></div>

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

    </body>

</html>

JS

function getRandomInt(max) {

    return Math.floor(Math.random() \* max);

}

let answer = 0;

let div = document.querySelector('div');

function action(){

    let first = getRandomInt(10);

    let second = getRandomInt(10);

    answer = first\*second;

    let argumentText = document.querySelector('label');

    argumentText.innerText = `${first} x ${second} = `;

}

function check(){

    let input = document.querySelector('input[type="text"]')

    if(input.value == `${answer}`)

        div.textContent = "Правильна відповідь!"

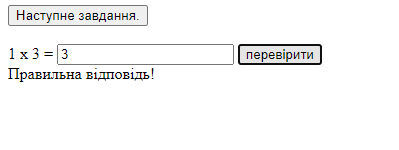
    else

        div.textContent = `Помилка правильна відповідь ${answer}!`

}

action();

Результат



**Завдання 5**

Створіть галерею зображень, кожне з яких неактивне. При клацанні по зображенню, воно стає активним. При наступному клацанні по ньому, воно знову стає неактивним. Для двох станів задайте наступні стилі:

o   *Активний стан:* фільтр *відтінки сірого(0%)*; непрозорість *1.*

o   *Неактивний стан:* фільтр *відтінки сірого(100%)*; непрозорість *0.5.*

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="UTF-8">

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

        <title>Tusk 5</title>

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

    </head>

    <body>

        <div class="table">

            <table>

                <tr>

                    <td><img src="galery/img1.jpg" class="inactive" onclick="Switch(this)"></td>

                    <td><img src="galery/img2.jpg" class="inactive" onclick="Switch(this)"></td>

                    <td><img src="galery/img3.jpg" class="inactive" onclick="Switch(this)"></td>

                    <td><img src="galery/img4.jpg" class="inactive" onclick="Switch(this)"></td>

                </tr>

                <tr>

                    <td><img src="galery/img5.jpg" class="inactive" onclick="Switch(this)"></td>

                    <td><img src="galery/img6.jpg" class="inactive" onclick="Switch(this)"></td>

                    <td><img src="galery/img7.jpg" class="inactive" onclick="Switch(this)"></td>

                    <td><img src="galery/img8.jpg" class="inactive" onclick="Switch(this)"></td>

                </tr>

                <tr>

                    <td><img src="galery/img9.jpg" class="inactive" onclick="Switch(this)"></td>

                    <td><img src="galery/img10.jpg" class="inactive" onclick="Switch(this)"></td>

                    <td><img src="galery/img11.jpg" class="inactive" onclick="Switch(this)"></td>

                    <td><img src="galery/img12.jpg" class="inactive" onclick="Switch(this)"></td>

                </tr>

            </table>

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

        </div>

    </body>

</html>

CSS

table, td{

    border: 1px solid black;

}

table{

    border-collapse: collapse;

}

img{

    width: 90px;

    height: 100px;

}

.active {

    filter: grayscale(0%);

    opacity: 1;

}

.inactive {

    filter: grayscale(100%);

    opacity: 0.5;

}

JS

function Switch(element){

    if(element.classList.contains('inactive')){

        element.classList.remove('inactive');

        element.classList.add('active')

    }else{

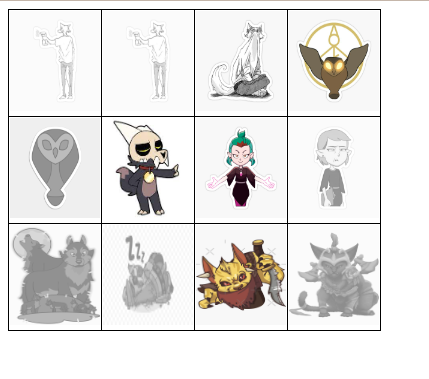
        element.classList.remove('active');

        element.classList.add('inactive')

    }

}

Результат



**Завдання 6**

Реалізуйте елементи форми. При фокусуванні на елемент, секція, в якому він знаходиться, стає активною (синій колір).

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Tusk 6</title>

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

</head>

    <body>

        <div class="block">

            <div><input type="text" placeholder="login"></div>

            <div><input type="password" placeholder="password"></div>

            <div><textarea placeholder="description"></textarea></div>

        </div>

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

    </body>

</html>

CSS

div > div{

    border: 2px solid rgb(205, 231, 254);

    margin-bottom: 10px;

    transition: 1s;

}

.block{

    width: 400px;

}

input, textarea{

    margin-block: 10px;

    margin-inline: 20px;

    border: 2px solid rgb(205, 231, 254);

}

textarea{

    height: 35px;

}

JS

let input = document.querySelectorAll('input');

let textarea = document.querySelector('textarea');

function on\_focused(event){

    let focusedElement = event.target;

    let parent = focusedElement.parentNode;

    parent.style.backgroundColor = 'rgb(170, 190, 252)';

}

function not\_focused(event){

    let focusedElement = event.target;

    let parent = focusedElement.parentNode;

    parent.style.backgroundColor = '';

}

input[0].addEventListener('focus', on\_focused);

input[0].addEventListener('blur', not\_focused);

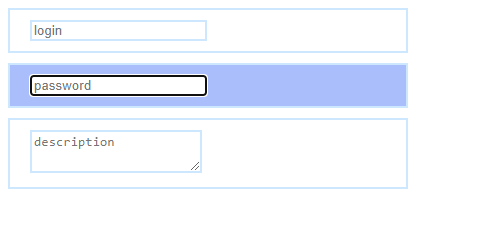
input[1].addEventListener('focus', on\_focused);

input[1].addEventListener('blur', not\_focused);

textarea.addEventListener('focus', on\_focused);

textarea.addEventListener('blur', not\_focused);

Результат



**Завдання 7**

Реалізувати можливість змінювати розмір блока, а також його кут повороту:

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="UTF-8">

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

        <title>Tusk 7</title>

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

    </head>

    <body>

        <div class="flexbox">

            <form oninput="">

                <label>width: 0<input type="range" id="WidthRang" value="0" max="500">500</label>

                <output id="First">0</output><br><br>

                <label>height: 0<input type="range" id="HeightRang" value="0" max="500">500</label>

                <output id="Second">0</output><br><br>

                <label>rotate: 0<input type="range" id="RotateRang" value="0" max="500">500</label>

                <output id="Third">0</output>

            </form><br><br>

            <div class="block"></div>

        </div>

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

    </body>

</html>

CSS

.block{

    border: 2px solid rgb(68, 113, 116);

    height: 100px;

    width: 100px;

    margin-left: 50px;

    background-color: rgb(79, 129, 189);

}

output{

    padding: 5px;

    font-weight: bolder;

    border: 2px solid black;

}

.flexbox{

    margin-top: 100px;

    display: flex;

}

JS

function changeWidth(){

    let value = document.querySelector('#First');

    value.textContent = inputWidth.value;

    square.style.width = `${inputWidth.value}px`;

}

function changeHeight(){

    let value = document.querySelector('#Second');

    value.textContent = inputHeight.value;

    square.style.height = `${inputHeight.value}px`;

}

function rotate(){

    let value = document.querySelector('#Third');

    value.textContent = inputRotate.value;

    square.style.transform = `rotate(${inputRotate.value}deg)`;

}

let square = document.querySelector('.block');

let inputWidth = document.querySelector('#WidthRang');

let inputHeight = document.querySelector('#HeightRang');

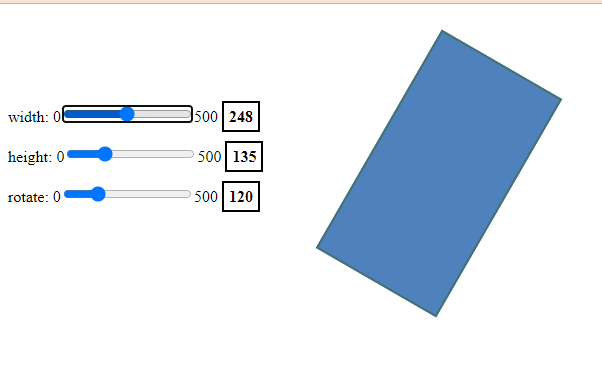
let inputRotate = document.querySelector('#RotateRang')

inputWidth.addEventListener('input', changeWidth);

inputHeight.addEventListener('input', changeHeight);

inputRotate.addEventListener('input', rotate);

Результат



**Завдання 8**

Дана картинка в тегу img та кнопка. Зробіть кнопку, за натисканням на яку вперше ширина і висота картинки буде збільшуватися в 2 рази, вдруге – приходити у вихідний стан.

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Tusk 1</title>

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

</head>

    <body>

        <div class="flexbox">

            <img src="img.jpg">

            <input type="button" value="Action" onclick="ScaleImg(this);">

        </div>

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

    </body>

</html>

CSS

img{

    width: 290px;

    height: 350px;

}

.flexbox{

    display: flex;

    flex-direction: column;

}

.action{

    transform: scale(2);

    position:static;

}

JS

let img = document.querySelector('img');

let div = document.querySelector('div')

let imgWidth = img.width;

let imgHeight = img.height;

div.style.width = `${imgWidth}px`;

function ScaleImg(element){

    let width = element.previousElementSibling.width;

    let height = element.previousElementSibling.height;

    if(imgWidth == width && imgHeight == imgHeight){

        element.previousElementSibling.style.width = `${width\*2}px`;

        element.previousElementSibling.style.height = `${height\*2}px`;

        div.style.width = `${width\*2}px`;

    }else{

        element.previousElementSibling.style.width = `${width - imgWidth}px`;

        element.previousElementSibling.style.height = `${height - imgHeight}px`;

        div.style.width = `${width - imgWidth}px`;

    }

}

Результат



**Завдання 9**

Є список товарів з кнопками *Замовити*. При натисканні на неї, одиниця даного товару заноситься в кошик.

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Tusk 9</title>

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

</head>

    <body>

        <p>Список товарів</p>

        <table>

            <tr>

                <td><div>Брюки</div></td>

                <td><input type="button" value="Замовити" name="trousers" onclick="add(this)"></td>

            </tr>

            <tr>

                <td><div>Сорочка</div></td>

                <td><input type="button" value="Замовити" name="shirt" onclick="add(this)"></td>

            </tr>

            <tr>

                <td><div>Шорти</div></td>

                <td><input type="button" value="Замовити" name="shorts" onclick="add(this)"></td>

            </tr>

            <tr>

                <td><div>Туфлі</div></td>

                <td><input type="button" value="Замовити" name="shoes" onclick="add(this)"></td>

            </tr>

        </table>

        <p>Кошик</p>

        <table>

            <tr>

                <td><div>Товар</div></td>

                <td><div>Кількість</div></td>

            </tr>

            <tr>

                <td><div>Брюки</div></td>

                <td><div id="trousers">0</div></td>

            </tr>

            <tr>

                <td><div>Сорочка</div></td>

                <td><div id="shirt">0</div></td>

            </tr>

            <tr>

                <td><div>Шорти</div></td>

                <td><div id="shorts">0</div></td>

            </tr>

            <tr>

                <td><div>Туфлі</div></td>

                <td><div id="shoes">0</div></td>

            </tr>

        </table>

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

    </body>

</html>

CSS

table{

    border-collapse: collapse;

}

table, td{

    border: 1.5px dotted black;

}

div{

    margin-right: 15px;

    margin-bottom: 10px;

}

input{

    margin-right: 15px;

    padding-block: 10px;

    padding-inline: 15px;

    background-color: rgb(164, 188, 230);

    border: 1px solid rgb(102, 154, 217);

    background: linear-gradient(to top, rgb(217, 226, 246) ,rgb(164, 188, 230));

}

input:active{

    background: linear-gradient(to top, rgb(196, 204, 222),rgb(139, 160, 196));

}

td:nth-child(2){

    width: 110px;

}

td:nth-child(2) > div{

    margin-left: 5px;

}

p{

    font-weight: bold;

    color: red;

}

JS

function add(element){

    let input;

    switch(element.name){

        case 'trousers':

            input = document.getElementById(element.name);

            input.textContent = `${parseInt(input.textContent) + 1}`;

            break;

        case 'shirt':

            input = document.getElementById(element.name);

            input.textContent = `${parseInt(input.textContent) + 1}`;

            break;

        case 'shorts':

            input = document.getElementById(element.name);

            input.textContent = `${parseInt(input.textContent) + 1}`;

            break;

        case 'shoes':

            input = document.getElementById(element.name);

            input.textContent = `${parseInt(input.textContent) + 1}`;

            break;

    }

}

Результат



**Завдання 10**

Реалізувати в контейнері можливість переглядати світлини одна за одною при клацанні курсора. Світлини змінюють одна одну з ефектом слайдингу.

Лістининг коду

HTML

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Tusk 10</title>

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

</head>

    <body>

        <div class="image-container">

            <button id="prevButton" onclick="changeImage(-1)"><p>◀</p></button>

            <div>

                <img src="./galery/image1.jpg">

                <img src="./galery/image2.jpg">

                <img src="./galery/image3.jpg">

                <img src="./galery/image4.jpg">

                <img src="./galery/image5.jpg">

                <img src="./galery/image6.jpg">

                <img src="./galery/image7.jpg">

                <img src="./galery/image8.jpg">

            </div>

            <button id="nextButton" onclick="changeImage(1)"><p>▶</p></button>

        </div>

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

    </body>

</html>

CSS

div {

    width: 330px;

    height: 260px;

    overflow: hidden;

    position: relative;

}

img {

    width: 100%;

    height: 100%;

    position: absolute;

    transition: transform 0.5s;

}

.image-container {

    display: flex;

    margin: 0 auto;

}

#nextButton, #prevButton{

    margin-block: 34%;

    border-radius: 20px;

}

p{

    padding: 5px;

    margin: 0px;

    font-weight: bolder;

}

JS

const images = document.querySelectorAll('img');

let currentIndex = 0;

function showImage(index) {

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

        if(i == index)

            images[i].style.transform = 'translateX(0)';

        else

            images[i].style.transform = 'translateX(100%)';

    }

}

function changeImage(step) {

    currentIndex += step;

    if(currentIndex >= images.length)

        currentIndex = 0;

    if(currentIndex < 0)

        currentIndex = images.length - 1;

    showImage(currentIndex);

}

showImage(currentIndex);

Результат

