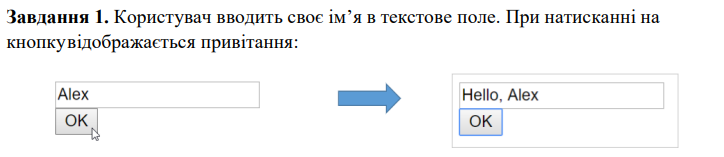
***Лабораторна робота №3***

***Тема:*** [***Обробка подій JavaScript.***](https://docs.google.com/document/d/1V-oyC79BzNa3ZJG6dM_eCYi12XwGPqp4vkJYtDp_-ds/edit)

***Хід роботи:***

***Завдання на лабораторну роботу:***

1. ***Завдання на лабораторну роботу:***

******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 1</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_1.css">5  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <table *class*="button\_section\_\_table">  
 <tr>  
 <td>  
 <label *for*="textGreeting" *class*="table\_\_label\_greeting">Enter your name:</label>  
 </td>  
 <td>  
 <input *type*="text" *name*="textGreeting" *id*="textGreeting" *class*="table\_\_text\_greeting" *placeholder*="Your name">  
 </td>  
 </tr>  
 <tr>  
 <td *colspan*="2">  
 <input *type*="button" *value*="Greeting" *id*="buttonGreeting" *class*="table\_\_button\_greeting">  
 </td>  
 </tr>  
 </table>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_1.js"></script>  
</div>  
</body>  
</html>

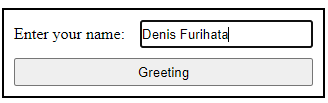
* ***SASS:***

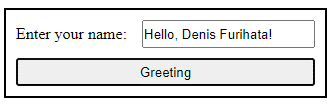
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid black  
 margin: 10px 0  
 padding: 10px  
 max-width: 30vw  
  
.*button\_section\_\_table* border-collapse: collapse  
  
.*table\_\_text\_greeting* margin-left: 15px  
 height: 28px  
  
.*table\_\_button\_greeting* height: 28px  
 width: 100%  
 margin: 10px 0 0 0

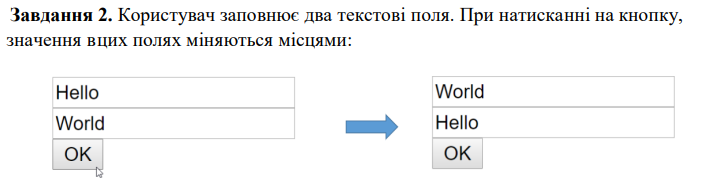
* ***JS:***

'use strict';  
  
*const* textGreeting = document.querySelector('#textGreeting');  
*const* buttonGreeting = document.querySelector('#buttonGreeting');  
buttonGreeting.addEventListener('click', () => {  
 textGreeting.value = 'Hello, ' + textGreeting.value + '!';  
})

***Результат програми:***

******



******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 2</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_2.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <table *class*="button\_section\_\_table">  
 <tr>  
 <td>  
 <label *for*="textFirstText" *class*="table\_\_label\_first\_text">Enter first text:</label>  
 </td>  
 <td>  
 <input *type*="text" *name*="textFirstText" *id*="textFirstText" *class*="table\_\_text\_first\_text" *placeholder*="Some text">  
 </td>  
 </tr>  
 <tr>  
 <td>  
 <label *for*="textSecondText" *class*="table\_\_label\_second\_text">Enter second text:</label>  
 </td>  
 <td>  
 <input *type*="text" *name*="textSecondText" *id*="textSecondText" *class*="table\_\_second\_text" *placeholder*="Some text">  
 </td>  
 </tr>  
 <tr>  
 <td *colspan*="2">  
 <input *type*="button" *value*="Swap values in two text inputs" *id*="buttonSwap" *class*="table\_\_button\_swap">  
 </td>  
 </tr>  
 </table>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_2.js"></script>  
</div>  
</body>  
</html>

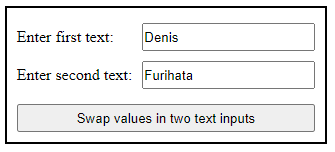
* ***SASS:***

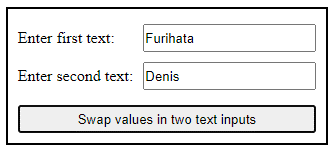
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid black  
 margin: 10px 0  
 padding: 10px  
 max-width: 30vw  
  
.*button\_section\_\_table* border-collapse: collapse  
  
.*table\_\_text\_first\_text*, .*table\_\_second\_text* margin: 5px 0 5px 10px  
 height: 28px  
  
.*table\_\_button\_swap* height: 28px  
 width: 100%  
 margin: 10px 0 0 0

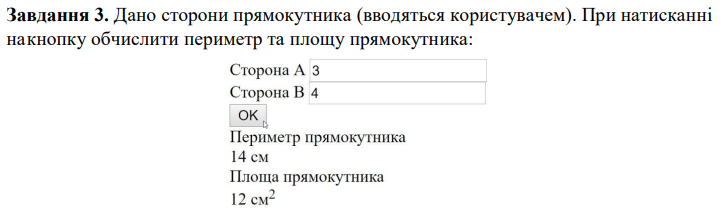
* ***JS:***

'use strict';  
  
*let* textFirstText = document.querySelector('#textFirstText');  
*let* textSecondText = document.querySelector('#textSecondText');  
*const* buttonSwap = document.querySelector('#buttonSwap');  
buttonSwap.addEventListener('click', () => {  
 [textFirstText.value, textSecondText.value] = [textSecondText.value, textFirstText.value]  
});

***Результат програми:***





******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 3</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_3.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <table *class*="button\_section\_\_table">  
 <tr>  
 <td>  
 <label *for*="textSideA" *class*="table\_\_label\_side\_a">Enter side A:</label>  
 </td>  
 <td>  
 <input *type*="text" *name*="textSideA" *id*="textSideA" *class*="table\_\_text\_side\_a" *placeholder*="Rectangle side A">  
 </td>  
 </tr>  
 <tr>  
 <td>  
 <label *for*="textSideB" *class*="table\_\_label\_side\_b">Enter side B:</label>  
 </td>  
 <td>  
 <input *type*="text" *name*="textSideB" *id*="textSideB" *class*="table\_\_text\_side\_b" *placeholder*="Rectangle side B">  
 </td>  
 </tr>  
 <tr>  
 <td *colspan*="2">  
 <input *type*="button" *value*="Get perimeter and area of rectangle" *id*="buttonPerimeterArea" *class*="table\_\_button\_perimeter\_area">  
 </td>  
 </tr>  
 </table>  
 <div *class*="button\_section\_\_output\_perimeter\_area"></div>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_3.js"></script>  
</div>  
</body>  
</html>

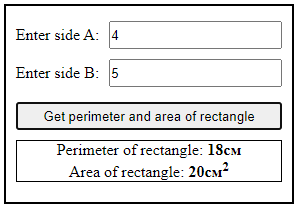
* ***SASS:***

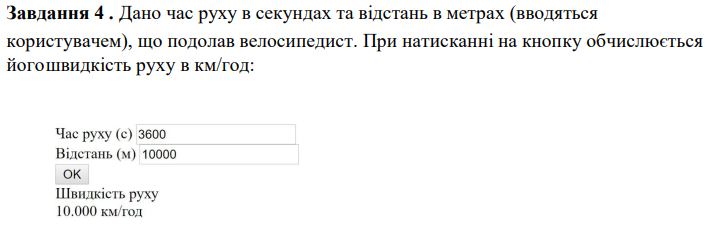
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid black  
 margin: 10px 0  
 padding: 10px  
 max-width: 30vw  
  
.*button\_section\_\_table* border-collapse: collapse  
  
.*table\_\_text\_side\_a*, .*table\_\_text\_side\_b* margin: 5px 0 5px 10px  
 height: 28px  
  
.*table\_\_button\_perimeter\_area* height: 28px  
 width: 100%  
 margin: 10px 0 0 0  
  
.*output\_perimeter\_area\_\_section* border: 1px solid black  
 margin: 10px 0  
 text-align: center

* ***JS:***

'use strict';  
  
*const* buttonPerimeterArea = document.querySelector('#buttonPerimeterArea');  
buttonPerimeterArea.addEventListener('click', () => {  
 *const* textSideA = *parseInt*(document.querySelector('#textSideA').value);  
 *const* textSideB = *parseInt*(document.querySelector('#textSideB').value);  
 *if* (*isNaN*(textSideA) || *isNaN*(textSideB)) {  
 document.querySelector('.*button\_section\_\_output\_perimeter\_area*').innerHTML =  
 `<section class="output\_perimeter\_area\_\_section">  
 <p>Values were entered <strong>incorrectly</strong>!</p>  
 </section>`;  
 } *else* {  
 *const* rectanglePerimeter = Math.round((textSideA \* 2 + textSideB \* 2) \* 100) / 100;  
 *const* rectangleArea = textSideA \* textSideB;  
 document.querySelector('.*button\_section\_\_output\_perimeter\_area*').innerHTML =  
 `<section class="output\_perimeter\_area\_\_section">  
 <p>Perimeter of rectangle: <strong>${rectanglePerimeter}см</strong></p>  
 <p>Area of rectangle: <strong>${rectangleArea}см<sup>2</sup></strong></p>  
 </section>`;  
 }  
});

***Результат програми:***

******

******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 4</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_4.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <table *class*="button\_section\_\_table">  
 <tr>  
 <td>  
 <label *for*="textTimeOfMovement" *class*="table\_\_label\_time\_of\_movement">Enter time of movement(sec.):</label>  
 </td>  
 <td>  
 <input *type*="text" *name*="textTimeOfMovement" *id*="textTimeOfMovement" *class*="table\_\_text\_time\_of\_movement" *placeholder*="Time of movement">  
 </td>  
 </tr>  
 <tr>  
 <td>  
 <label *for*="textDistance" *class*="table\_\_label\_distance">Enter distance(m.):</label>  
 </td>  
 <td>  
 <input *type*="text" *name*="textDistance" *id*="textDistance" *class*="table\_\_text\_distance" *placeholder*="Distance">  
 </td>  
 </tr>  
 <tr>  
 <td *colspan*="2">  
 <input *type*="button" *value*="Get speed of movement" *id*="buttonSpeedMovement" *class*="table\_\_button\_speed\_movement">  
 </td>  
 </tr>  
 </table>  
 <div *class*="button\_section\_\_output\_speed\_movement"></div>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_4.js"></script>  
</div>  
</body>  
</html>

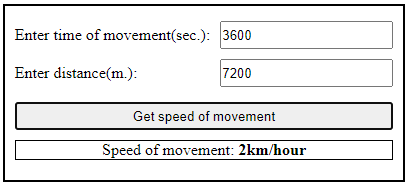
* ***SASS:***

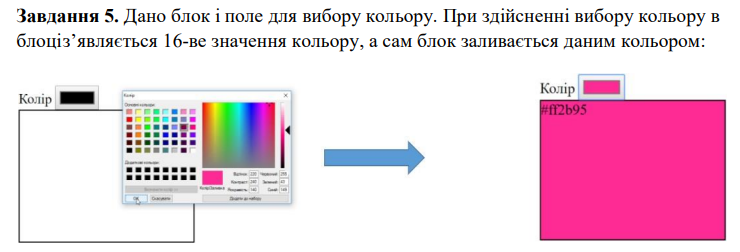
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid black  
 margin: 10px 0  
 padding: 10px  
 max-width: 30vw  
  
.*button\_section\_\_table* border-collapse: collapse  
  
.*table\_\_text\_time\_of\_movement*, .*table\_\_text\_distance* margin: 5px 0 5px 10px  
 height: 28px  
  
.*table\_\_button\_speed\_movement* height: 28px  
 width: 100%  
 margin: 10px 0 0 0  
  
.*output\_speed\_movement\_\_section* border: 1px solid black  
 margin: 10px 0  
 text-align: center

* ***JS:***

'use strict';  
  
document.querySelector('#buttonSpeedMovement').addEventListener('click', () => {  
 *const* timeOfMovement = *parseFloat*(document.querySelector('#textTimeOfMovement').value);  
 *const* distance = *parseFloat*(document.querySelector('#textDistance').value);  
 *if* (*isNaN*(timeOfMovement) || *isNaN*(distance)) {  
 document.querySelector('.*button\_section\_\_output\_speed\_movement*').innerHTML =  
 `<section class="button\_section\_\_output\_speed\_movement">  
 <p>Incorrect type of entered values!</p>  
 </section>`;  
 } *else* {  
 console.log(timeOfMovement, distance);  
 *const* speedOfMovement = Math.round((distance / timeOfMovement) \* 100) / 100;  
 document.querySelector('.*button\_section\_\_output\_speed\_movement*').innerHTML =  
 `<section class="output\_speed\_movement\_\_section">  
 <p>Speed of movement: <strong>${speedOfMovement}km/hour</strong></p>  
 </section>`;  
 }  
});

***Результат програми:***

******

******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 5</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_5.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <table *class*="button\_section\_\_table">  
 <tr>  
 <td>  
 <label *for*="colorChooseColor" *class*="table\_\_label\_choose\_color">Choose some color:</label>  
 </td>  
 <td>  
 <input *type*="color" *name*="colorChooseColor" *id*="colorChooseColor" *class*="table\_\_color\_choose\_color">  
 </td>  
 </tr>  
 </table>  
 <div *class*="button\_section\_\_output\_color"></div>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_5.js"></script>  
</div>  
</body>  
</html>

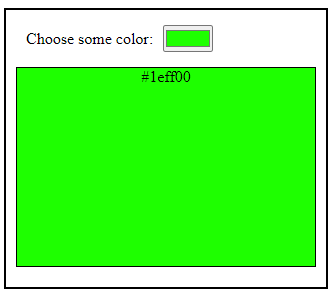
* ***SASS:***

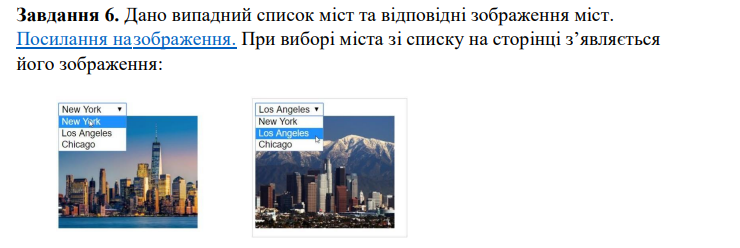
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid black  
 margin: 10px 0  
 padding: 10px  
 max-width: 30vw  
  
.*button\_section\_\_table* border-collapse: collapse  
  
.*table\_\_label\_choose\_color*, .*table\_\_color\_choose\_color* margin: 5px 0 5px 10px  
  
.*button\_section\_\_output\_color* border: 1px solid black  
 margin: 10px 0  
 text-align: center  
 width: 300px  
 height: 200px

* ***JS:***

'use strict';  
  
*const* colorChooseColor = document.querySelector('#colorChooseColor');  
  
colorChooseColor.addEventListener('input', () => {  
 *const* color = colorChooseColor.value;  
 *const* outputColor = document.querySelector('.*button\_section\_\_output\_color*');  
   
 outputColor.innerHTML = `<p class="output\_color\_\_p">${color}</p>`;  
 outputColor.style.backgroundColor = color;  
});

***Результат програми:***

******

******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 6</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_6.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <table *class*="button\_section\_\_table">  
 <tr>  
 <td>  
 <label *for*="selectChooseImg" *class*="table\_\_label\_choose\_img">Choose some image to see it:</label>  
 </td>  
 <td>  
 <select *name*="selectChooseImg" *id*="selectChooseImg" *class*="table\_\_select\_choose\_img">  
 <option *value*="default">--> Select image <--</option>  
 <option *value*="new\_york">New York</option>  
 <option *value*="los\_angeles">Los Angeles</option>  
 <option *value*="chicago">Chicago</option>  
 </select>  
 </td>  
 </tr>  
 </table>  
 <div *class*="button\_section\_\_output\_img"></div>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_6.js"></script>  
</div>  
</body>  
</html>

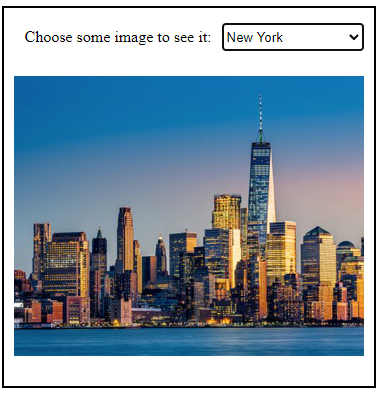
* ***SASS:***

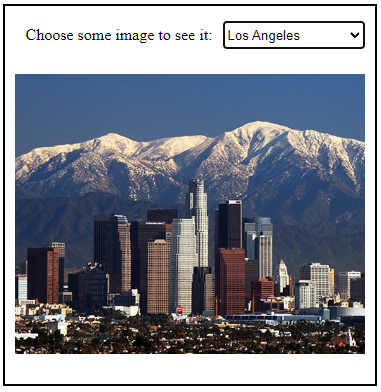
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid black  
 margin: 10px 0  
 padding: 10px  
 max-width: 30vw  
  
.*button\_section\_\_table* border-collapse: collapse  
  
.*table\_\_label\_choose\_img*, .*table\_\_select\_choose\_img* margin: 5px 0 5px 10px  
 height: 28px  
  
.*button\_section\_\_output\_img* margin: 10px 0  
 width: 100%  
 height: 300px

* ***JS:***

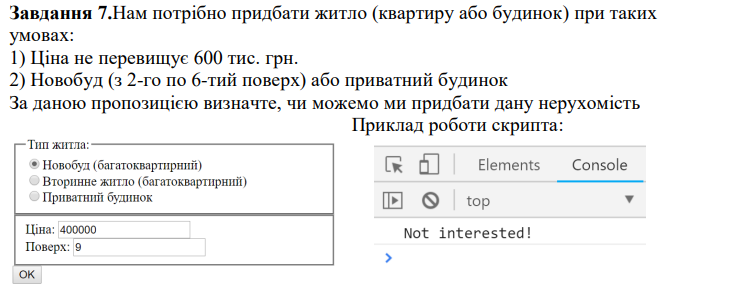
'use strict';  
  
*const* selectChooseImg = document.querySelector('#selectChooseImg');  
  
selectChooseImg.addEventListener('input', () => {  
 *const* img = selectChooseImg.value;  
 *const* outputImg = document.querySelector('.*button\_section\_\_output\_img*');  
   
 *switch* (img) {  
 *case* 'los\_angeles': {  
 outputImg.style.background = "url('../../.././img/los-angeles.jpg') no-repeat center";  
 outputImg.style.backgroundSize = 'contain';  
 *break*;  
 }  
 *case* 'new\_york': {  
 outputImg.style.background = "url('../../.././img/new-york.jpg') no-repeat center";  
 outputImg.style.backgroundSize = 'contain';  
 *break*;  
 }  
 *case* 'chicago': {  
 outputImg.style.background = "url('../../.././img/chicago.jpg') no-repeat center";  
 outputImg.style.backgroundSize = 'contain';  
 *break*;  
 }  
 }  
});

***Результат програми:***







******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 7</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_7.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <fieldset *class*="button\_section\_\_dwelling\_fieldset">  
 <legend>Type of dwelling:</legend>  
 <table *class*="button\_section\_\_table">  
 <tr>  
 <td>  
 <input *type*="radio" *name*="radioBuildingType" *id*="radioNewBuilding"  
 *class*="table\_\_number\_new\_building" *value*="New building">  
 </td>  
 <td>  
 <label *for*="radioNewBuilding" *class*="table\_\_label\_new\_building">New  
 building(multi-apartment)</label>  
 </td>  
 </tr>  
 <tr>  
 <td>  
 <input *type*="radio" *name*="radioBuildingType" *id*="radioSecondaryHousing"  
 *class*="table\_\_number\_secondary\_housing" *value*="Secondary housing">  
 </td>  
 <td>  
 <label *for*="radioSecondaryHousing" *class*="table\_\_label\_secondary\_housing">Secondary  
 housing(multi-apartment)</label>  
 </td>  
 </tr>  
 <tr>  
 <td>  
 <input *type*="radio" *name*="radioBuildingType" *id*="radioPrivateHouse"  
 *class*="table\_\_number\_private\_house" *value*="Private house">  
 </td>  
 <td>  
 <label *for*="radioPrivateHouse" *class*="table\_\_label\_private\_house">Private house</label>  
 </td>  
 </tr>  
 </table>  
 </fieldset>  
 <fieldset *class*="button\_section\_\_actions\_fieldset">  
 <table *class*="button\_section\_\_table">  
 <tr *class*="table\_\_price\_tr">  
 <td>  
 <label *for*="numberPrice" *class*="table\_\_label\_price">Price:</label>  
 </td>  
 <td>  
 <input *type*="number" *name*="numberPrice" *id*="numberPrice" *class*="table\_\_number\_price">  
 </td>  
 </tr>  
 <tr *class*="table\_\_floor\_tr">  
 <td>  
 <label *for*="numberFloor" *class*="table\_\_label\_floor">Floor:</label>  
 </td>  
 <td>  
 <input *type*="number" *name*="numberFloor" *id*="numberFloor" *class*="table\_\_number\_floor">  
 </td>  
 </tr>  
 <tr *class*=table\_\_check\_suggestions\_tr"">  
 <td *colspan*="2">  
 <input *type*="button" *value*="Check suggestions" *id*="buttonCheckSuggestions"  
 *class*="table\_\_button\_check\_suggestions">  
 </td>  
 </tr>  
 </table>  
 </fieldset>  
 <div *class*="button\_section\_\_output\_status"></div>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_7.js"></script>  
</div>  
</body>  
</html>

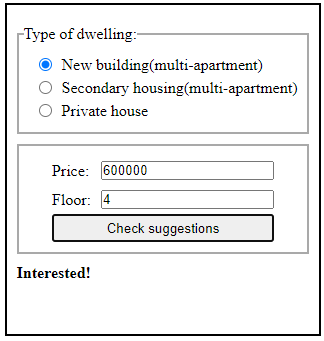
* ***SASS:***

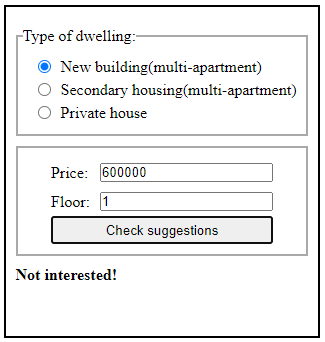
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid black  
 margin: 10px 0  
 padding: 10px  
 max-width: 35vw  
  
.*button\_section\_\_dwelling\_fieldset*, .*button\_section\_\_actions\_fieldset* border: 2px solid darkgrey  
 padding: 5px  
 margin: 10px 0  
 width: 100%  
  
.*button\_section\_\_table* border-collapse: collapse  
 margin: 5px  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*table\_\_label\_new\_building*, .*table\_\_label\_secondary\_housing*, .*table\_\_label\_private\_house* margin: 0 0 0 10px  
 height: 28px  
  
.*table\_\_button\_check\_suggestions* width: 100%  
 height: 28px  
  
.*table\_\_number\_price*, .*table\_\_number\_floor*, .*table\_\_number\_new\_building*, .*table\_\_number\_secondary\_housing*, .*table\_\_number\_private\_house* margin: 5px 0 5px 10px  
  
.*button\_section\_\_output\_status* margin: 10px 0  
 width: 100%  
 height: 50px

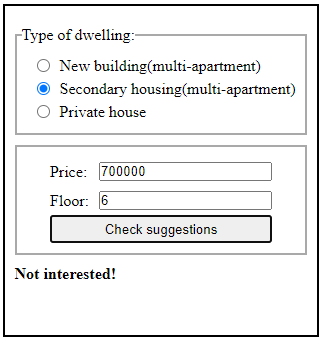
* ***JS:***

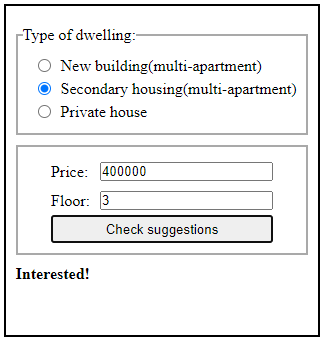
'use strict';  
*const* radioBuildingType = document.querySelectorAll('input[*name*="radioBuildingType"]');  
  
*const hiddenFloor* = () => {  
 document.querySelector('.*table\_\_floor\_tr*').setAttribute('hidden', '');  
};  
  
*const unHiddenFloor* = () => {  
 document.querySelector('.*table\_\_floor\_tr*').removeAttribute('hidden');  
};  
  
radioBuildingType[0].addEventListener('change', *unHiddenFloor*)  
radioBuildingType[1].addEventListener('change', *unHiddenFloor*)  
radioBuildingType[2].addEventListener('change', *hiddenFloor*)  
  
*const* buttonCheckSuggestions = document.querySelector('#buttonCheckSuggestions');  
  
buttonCheckSuggestions.addEventListener('click', () => {  
 *const* numberPrice = document.querySelector('#numberPrice').value;  
 *const* numberFloor = document.querySelector('#numberFloor').value;  
 *const* outputField = document.querySelector('.*button\_section\_\_output\_status*');  
  
 *if* (!radioBuildingType[0].checked && !radioBuildingType[1].checked && !radioBuildingType[2].checked) {  
 outputField.innerHTML = `<p>Choose type of dwelling!</p>`;  
 console.log('Choose type of dwelling!');  
 } *else* {  
 *if* (numberPrice > 600000 || numberPrice < 0) {  
 outputField.innerHTML = `<p><strong>Not interested!</strong></p>`;  
 console.log('Not interested!');  
 } *else* {  
 *if* (radioBuildingType[0].checked || radioBuildingType[1].checked) {  
 *if* (numberFloor < 1 || numberFloor > 47) {  
 outputField.innerHTML = `<p><strong>Not interested!</strong></p>`;  
 console.log('Not interested!');  
 } *else* {  
 *if* (numberFloor >= 2 && numberFloor <= 6) {  
 outputField.innerHTML =  
 `<p><strong>Interested!</strong></p>`;  
 console.log('Interested!');  
 } *else* {  
 outputField.innerHTML =  
 `<p><strong>Not interested!</strong></p>`;  
 console.log('Not interested!');  
 }  
 }  
 } *else* {  
 outputField.innerHTML =  
 `<p><strong>Interested!</strong></p>`;  
 console.log('Interested!');  
 }  
 }  
 }  
});

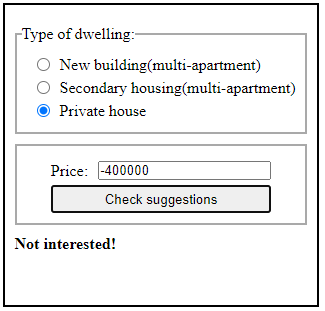
***Результат програми:***

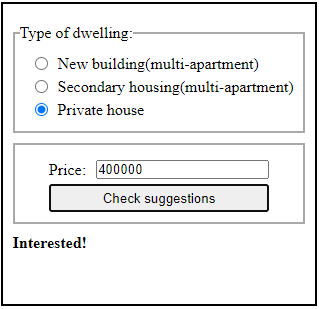
******

******

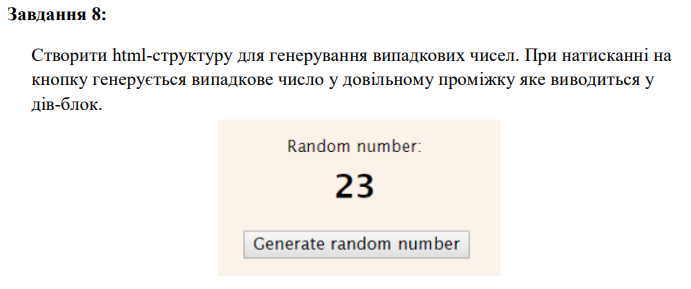
******

******

******

******

1. ***Домашнє завдання:***

******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 8</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_8.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <fieldset *class*="button\_section\_\_random\_number\_fieldset">  
 <label *for*="buttonRandomNumber" *class*="table\_\_label\_random\_number">Random number</label>  
 <div *class*="button\_section\_\_div\_output\_number"><h3 *class*="div\_output\_number\_\_h3">0</h3></div>  
 <input *type*="button" *value*="Generate random number[-100; 100]" *name*="buttonRandomNumber" *id*="buttonRandomNumber" *class*="table\_\_button\_random\_number">  
 </fieldset>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_8.js"></script>  
</div>  
</body>  
</html>

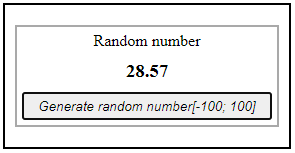
* ***SASS:***

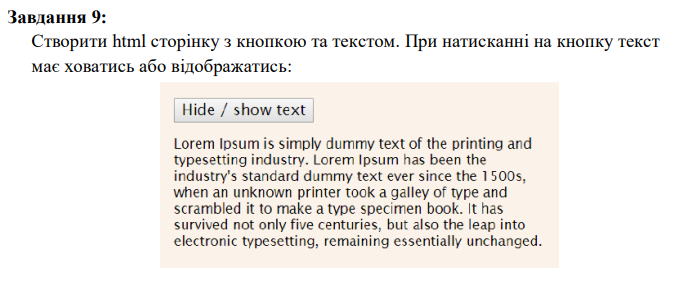
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid black  
 margin: 10px 0  
 padding: 10px  
 max-width: 35vw  
  
.*button\_section\_\_random\_number\_fieldset* text-align: center  
 border: 2px solid darkgrey  
 padding: 5px  
 margin: 10px 0  
 width: 100%  
  
.*table\_\_button\_random\_number* width: 250px  
 height: 28px  
 font-style: italic  
  
.*button\_section\_\_div\_output\_number* margin: 10px 0  
 width: 100%

* ***JS:***

'use strict';  
  
*const getRandomFloat* = (*fromNumber*, *toNumber*) => Math.round((Math.random() \* (*toNumber* - *fromNumber*) + *fromNumber*) \* 100) / 100;  
  
*const* buttonRandomNumber = document.querySelector('#buttonRandomNumber');  
  
buttonRandomNumber.addEventListener('click', () => {  
 document.querySelector('.*div\_output\_number\_\_h3*').innerHTML = *getRandomFloat*(-100, 100);  
})

***Результат програми:***

******

******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 9</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_9.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <fieldset *class*="button\_section\_\_show\_hide\_fieldset">  
 <input *type*="button" *value*="Show/hide text" *name*="buttonShowHideText"  
 *id*="buttonShowHideText" *class*="table\_\_button\_show\_hide\_text">  
 <div *class*="button\_section\_\_div\_text">  
 <p>  
 Lorem ipsum dolor sit amet, consectetur adipisicing elit. Accusantium, animi  
 blanditiis consequuntur dicta eaque earum eius enim et eum facere, id ipsam  
 laboriosam officiis perspiciatis, quidem rem saepe unde vitae.  
 </p>  
 <p>  
 Adipisci aperiam, cumque distinctio dolorem ducimus, eum eveniet expedita fuga illum  
 in inventore, labore minima nam nobis non nostrum placeat quam quibusdam quisquam  
 rerum totam ullam veniam. A, aut, cupiditate.  
 </p>  
 <p>  
 Ad, consectetur dicta doloribus et iure laudantium ratione vitae! Animi dolores  
 excepturi labore neque officia porro quam sapiente tempora voluptatibus. Autem  
 cupiditate dignissimos ea facere incidunt iure omnis, reprehenderit unde.  
 </p>  
 <p>  
 Aspernatur cum debitis dignissimos esse id, illo itaque iusto laboriosam maiores  
 natus, nihil odit officia quibusdam quis quod recusandae repudiandae tempora totam  
 velit veniam? Aliquam pariatur quaerat rem sit suscipit.  
 </p>  
 <p>  
 Architecto, aspernatur beatae commodi cum deserunt distinctio ea error eum ex  
 excepturi impedit inventore ipsa ipsum iusto laborum modi molestiae nemo odit optio  
 praesentium quam repellat saepe sed voluptatibus voluptatum!  
 </p>  
 </div>  
 </fieldset>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_9.js"></script>  
</div>  
</body>  
</html>

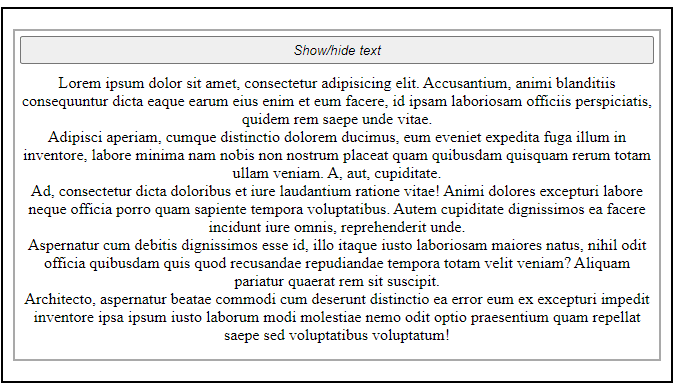
* ***SASS:***

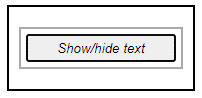
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid black  
 margin: 10px 0  
 padding: 10px  
 max-width: 35vw  
  
.*button\_section\_\_show\_hide\_fieldset* text-align: center  
 border: 2px solid darkgrey  
 padding: 5px  
 margin: 10px 0  
 width: 100%  
  
.*table\_\_button\_show\_hide\_text* width: 100%  
 min-width: 150px  
 height: 28px  
 font-style: italic  
  
.*button\_section\_\_div\_text* margin: 10px 0  
 width: 100%

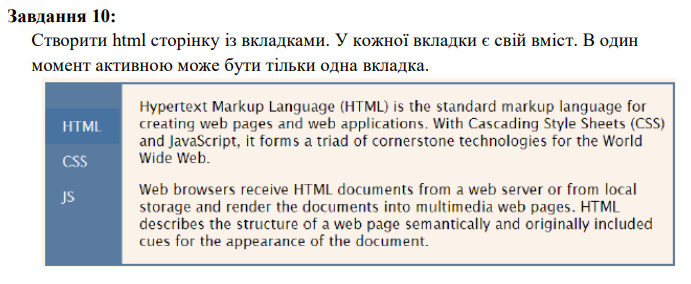
* ***JS:***

'use strict';  
  
*const* buttonShowHideText = document.querySelector('#buttonShowHideText');  
  
buttonShowHideText.addEventListener('click', () => {  
 *const* text = document.querySelector('.*button\_section\_\_div\_text*');  
   
 *if*(text.hasAttribute('hidden')){  
 text.removeAttribute('hidden');  
 } *else*{  
 text.setAttribute('hidden', '');  
 }  
});

***Результат програми:***





******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 10</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_10.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <aside *class*="button\_section\_\_aside">  
 <div *class*="aside\_\_HTML" *id*="asideHTML">HTML</div>  
 <div *class*="aside\_\_CSS" *id*="asideCSS">CSS</div>  
 <div *class*="aside\_\_JS" *id*="asideJS">JS</div>  
 </aside>  
 <div *class*="button\_section\_\_div\_HTML\_text" *id*="divHTMLText">  
 <h1>HTML</h1>  
 <p>  
 The HyperText Markup Language or HTML is the standard markup  
 language for documents designed to be displayed in a web browser. It can be assisted by technologies  
 such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.  
 </p>  
 <p>  
 Web browsers receive HTML documents from a web server or from local storage and render the documents  
 into multimedia web pages. HTML describes the structure of a web page semantically and originally  
 included cues for the appearance of the document.  
 </p>  
 <p>  
 HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects  
 such as interactive forms may be embedded into the rendered page. HTML provides a means to create  
 structured documents by denoting structural semantics for text such as headings, paragraphs, lists,  
 links, quotes and other items. HTML elements are delineated by tags, written using angle brackets.  
 Tags such as "img" and "input" directly introduce content into the page. Other tags such as "p"  
 surround and provide information about document text and may include other tags as sub-elements.  
 Browsers do not display the HTML tags but use them to interpret the content of the page.  
 </p>  
 <p>  
 HTML can embed programs written in a scripting language such as JavaScript, which affects the  
 behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The  
 World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS  
 standards, has encouraged the use of CSS over explicit presentational HTML since 1997.[2] A form of  
 HTML, known as HTML5, is used to display video and audio, primarily using the  
 "canvas" element, in collaboration with javascript.  
 </p>  
 </div>  
 <div *class*="button\_section\_\_div\_CSS\_text" *id*="divCSSText">  
 <h1>CSS</h1>  
 <p>  
 Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a  
 document written in a markup language such as HTML or XML (including XML dialects such as SVG,  
 MathML or XHTML).[1] CSS is a cornerstone technology of the World Wide Web, alongside HTML and  
 JavaScript.[2]  
 </p>  
 <p>  
 CSS is designed to enable the separation of presentation and content, including layout, colors, and  
 fonts.[3] This separation can improve content accessibility; provide more flexibility and control in  
 the  
 specification of presentation characteristics; enable multiple web pages to share formatting by  
 specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the  
 structural content; and enable the .css file to be cached to improve the page load speed between the  
 pages that share the file and its formatting.  
 </p>  
 <p>  
 Separation of formatting and content also makes it feasible to present the same markup page in  
 different  
 styles for different rendering methods, such as on-screen, in print, by voice (via speech-based  
 browser  
 or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting  
 if  
 the content is accessed on a mobile device.[4]  
 </p>  
 <p>  
 The name cascading comes from the specified priority scheme to determine which style rule applies if  
 more than one rule matches a particular element. This cascading priority scheme is predictable.  
 </p>  
 <p>  
 The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type  
 (MIME  
 type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS  
 validation service for CSS documents.[5]  
 </p>  
 <p>  
 In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG,  
 and  
 XUL.  
 </p>  
 </div>  
 <div *class*="button\_section\_\_div\_JS\_text" *id*="divJSText">  
 <h1>JS</h1>  
 <p>  
 JavaScript (/ˈdʒɑːvəskrɪpt/), often abbreviated as JS, is a programming language that is one of the  
 core technologies of the World Wide Web, alongside HTML and CSS. As of 2022, 98% of websites use  
 JavaScript on the client side for webpage behavior, often incorporating third-party libraries. All  
 major web browsers have a dedicated JavaScript engine to execute the code on users' devices.  
 </p>  
 <p>  
 JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript  
 standard.[10] It has dynamic typing, prototype-based object-orientation, and first-class functions.  
 It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It has  
 application programming interfaces (APIs) for working with text, dates, regular expressions,  
 standard data structures, and the Document Object Model (DOM).  
 </p>  
 <p>  
 The ECMAScript standard does not include any input/output (I/O), such as networking, storage, or  
 graphics facilities. In practice, the web browser or other runtime system provides JavaScript APIs  
 for I/O.  
 </p>  
 <p>  
 JavaScript engines were originally used only in web browsers, but are now core components of some  
 servers and a variety of applications. The most popular runtime system for this usage is Node.js.  
 </p>  
 <p>  
 Although Java and JavaScript are similar in name, syntax, and respective standard libraries, the two  
 languages are distinct and differ greatly in design.  
 </p>  
 </div>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_10.js"></script>  
</div>  
</body>  
</html>

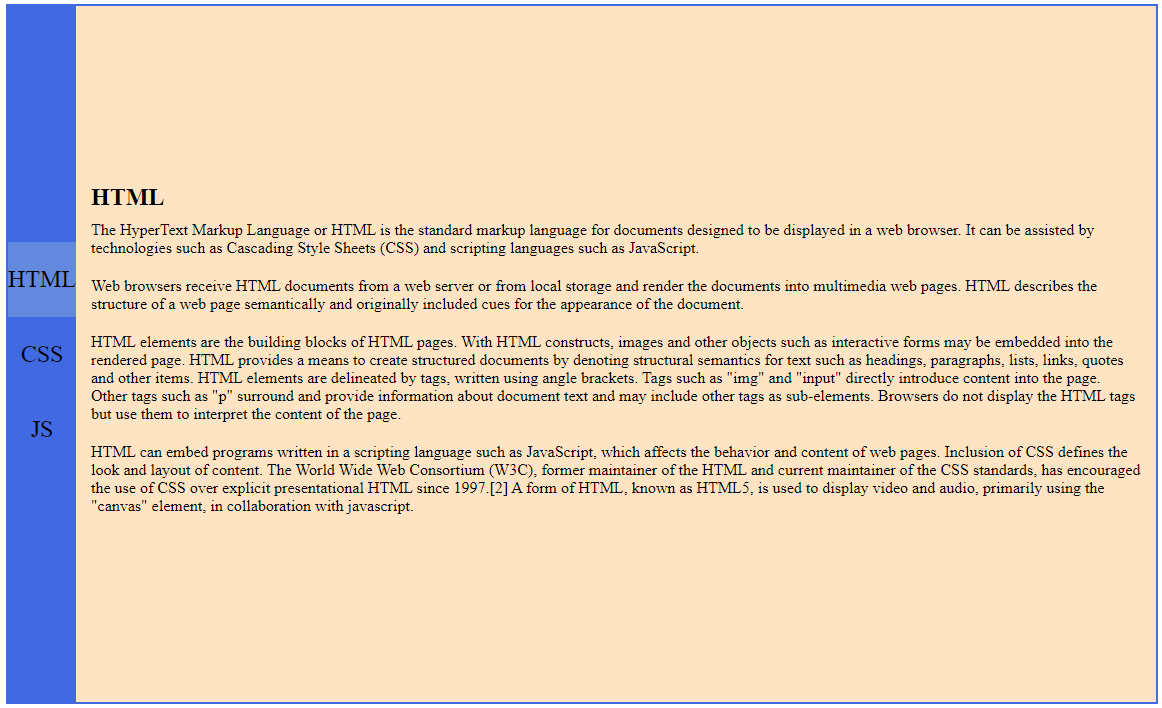
* ***SASS:***

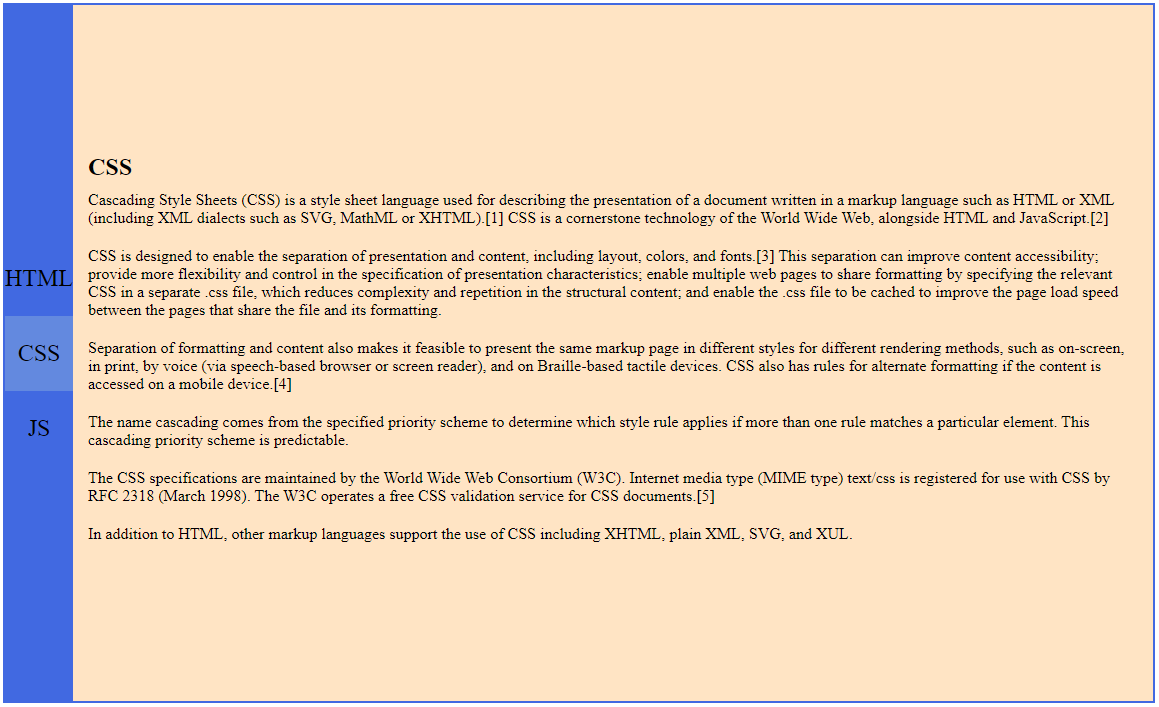
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid royalblue  
 margin: 10px 0  
 max-width: 60vw  
 display: flex  
 align-items: center  
 height: 700px  
  
.*button\_section\_\_aside* background-color: royalblue  
 height: 100%  
 max-width: 30vw  
 display: flex  
 flex-direction: column  
 align-items: center  
 justify-content: center  
  
.*aside\_\_HTML*, .*aside\_\_CSS*, .*aside\_\_JS* height: 75px  
 width: 100%  
 font-size: 24px  
 display: flex  
 align-items: center  
 justify-content: center  
 &:*hover* background-color: #6389df  
 cursor: pointer  
  
.*button\_section\_\_div\_HTML\_text*, .*button\_section\_\_div\_CSS\_text*, .*button\_section\_\_div\_JS\_text* background-color: bisque  
 height: 100%  
 padding: 15px  
 display: flex  
 flex-direction: column  
 justify-content: center  
 p  
 margin: 10px 0

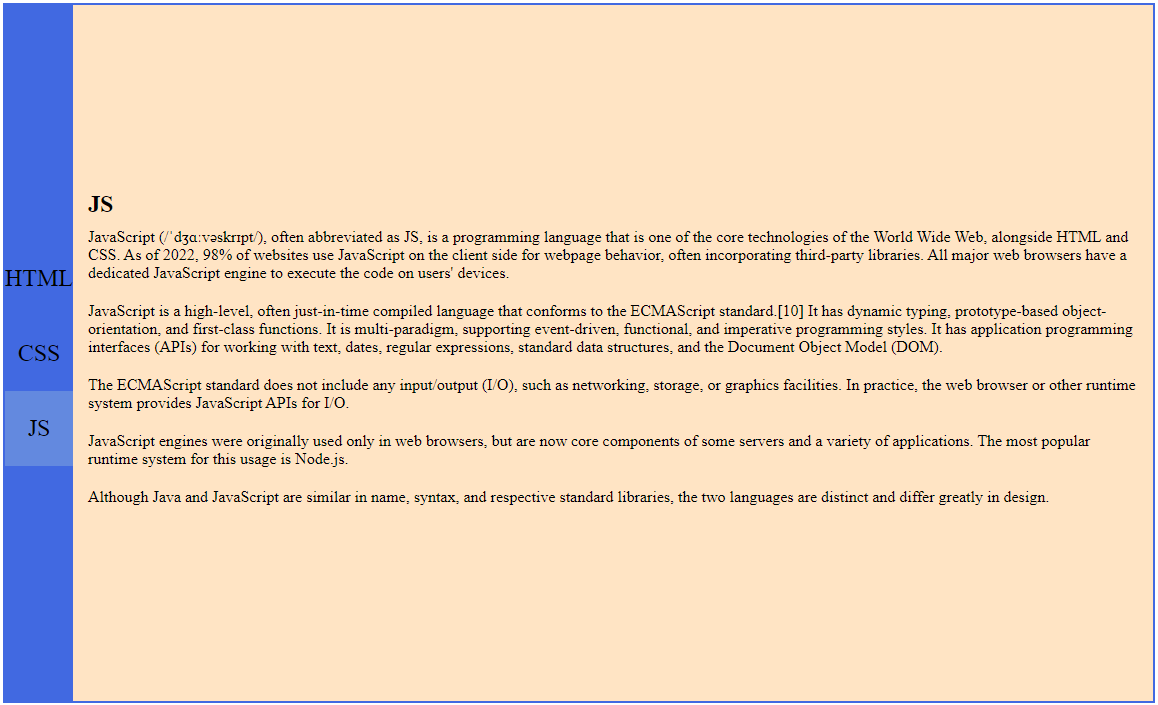
* ***JS:***

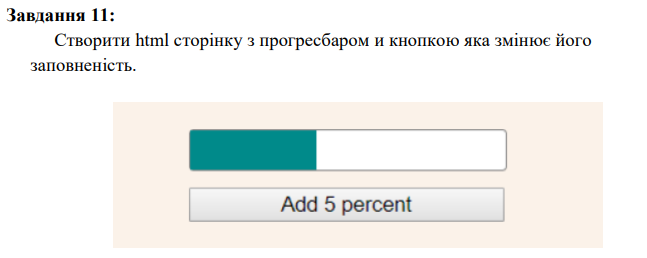
'use strict';  
  
*const chooseSelector* = (*chosenElem*) => *chosenElem*.style.backgroundColor = '#6389df';  
  
*const unChooseSelector* = (*chosenElem*) => *chosenElem*.style.backgroundColor = '#4169E1FF';  
  
*const hiddenSelector* = (*selector*) => *selector*.style.display = 'none';  
*// const hiddenSelector = (selector) => selector.setAttribute('hidden', '');  
  
const unhiddenSelector* = (*selector*) => *selector*.style.display = 'flex';  
*// const unhiddenSelector = (selector) => selector.removeAttribute('hidden');  
  
const* divHTMLText = document.querySelector('#divHTMLText');  
*const* divCSSText = document.querySelector('#divCSSText');  
*const* divJSText = document.querySelector('#divJSText');  
  
*const* asideHTML = document.querySelector('#asideHTML');  
*const* asideCSS = document.querySelector('#asideCSS');  
*const* asideJS = document.querySelector('#asideJS');  
  
*chooseSelector*(asideHTML);  
*hiddenSelector*(divCSSText);  
*hiddenSelector*(divJSText);  
  
asideHTML.addEventListener('click', () => {  
 *chooseSelector*(asideHTML);  
 *unChooseSelector*(asideCSS);  
 *unChooseSelector*(asideJS);  
  
 *unhiddenSelector*(divHTMLText);  
 *hiddenSelector*(divCSSText);  
 *hiddenSelector*(divJSText);  
});  
  
asideCSS.addEventListener('click', () => {  
 *chooseSelector*(asideCSS);  
 *unChooseSelector*(asideHTML);  
 *unChooseSelector*(asideJS);  
  
 *unhiddenSelector*(divCSSText);  
 *hiddenSelector*(divHTMLText);  
 *hiddenSelector*(divJSText);  
});  
  
asideJS.addEventListener('click', () => {  
 *chooseSelector*(asideJS);  
 *unChooseSelector*(asideHTML);  
 *unChooseSelector*(asideCSS);  
  
 *unhiddenSelector*(divJSText);  
 *hiddenSelector*(divHTMLText);  
 *hiddenSelector*(divCSSText);  
});

***Результат програми:***

******





******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 11</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_11.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <progress *class*="button\_section\_\_progress\_bar" *id*="progressBar" *max*="100" *value*="5">5%</progress>  
 <div *class*="button\_section\_\_buttons">  
 <input *class*="button\_section\_\_button\_add" *id*="buttonAddProgressBar" *type*="button" *value*="Add 5 percents">  
 <input *class*="button\_section\_\_button\_remove" *id*="buttonRemoveProgressBar" *type*="button" *value*="Remove 5 percents">  
 </div>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_11.js"></script>  
</div>  
</body>  
</html>

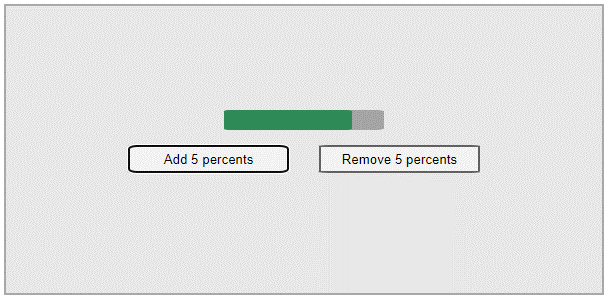
* ***SASS:***

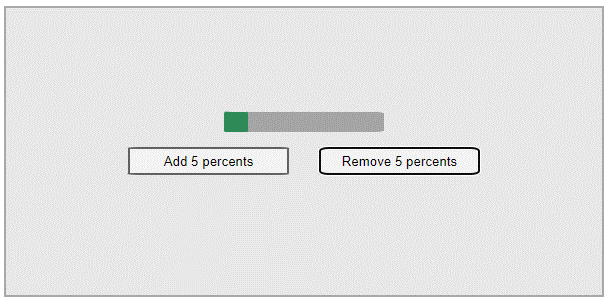
\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 2px solid darkgrey  
 background-color: #e8e8e8  
 margin: 10px 0  
 max-width: 60vw  
 display: flex  
 flex-direction: column  
 justify-content: center  
 align-items: center  
 width: 600px  
 height: 30vh  
  
.*button\_section\_\_progress\_bar* height: 20px  
 background-color: white  
  
 &::*-webkit-progress-bar* background-color: #a5a5a5  
 border-radius: 5%  
  
 &::*-webkit-progress-value* background-color: seagreen  
 border-radius: 5%  
  
 &[*value*]::*-webkit-progress-value*::*before* content: '80%'  
 position: absolute  
 right: 0  
 top: -125%  
  
.*button\_section\_\_buttons* width: 90%  
 display: flex  
 justify-content: center  
  
  
.*button\_section\_\_button\_add*, .*button\_section\_\_button\_remove* margin: 15px  
 height: 28px  
 width: 30%  
 background-color: #f3f3f3  
 border: 2px solid #626262  
 border-radius: 5%  
 cursor: pointer

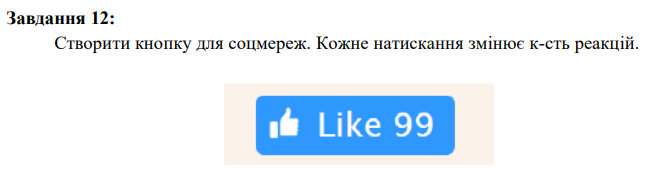
* ***JS:***

'use strict';  
  
*const* progressBar = document.querySelector('#progressBar');  
*const* buttonAddProgressBar = document.querySelector('#buttonAddProgressBar');  
  
buttonAddProgressBar.addEventListener('click', () => {  
 progressBar.value += 5;  
});  
  
*const* buttonRemoveProgressBar = document.querySelector('#buttonRemoveProgressBar');  
  
buttonRemoveProgressBar.addEventListener('click', () => {  
 progressBar.value -= 5;  
});

***Результат програми:***





******

***Виконання:***

* ***HTML:***

<!DOCTYPE *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <title>Task 12</title>  
 <link *rel*="stylesheet" *href*="./styles/css/task\_12.css">  
 <link *rel*="icon" *href*="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
</head>  
<body>  
<div *class*="wrapper">  
 <div *class*="container">  
 <section *class*="button\_section">  
 <button *class*="button\_section\_\_button\_like" *id*="buttonLike" *value*="0">  
 <img *src*="../../../../img/icons/racoon\_animal\_icon\_123571.ico" *alt*="raccoon\_ico"  
 *class*="button\_section\_\_raccoon\_ico">  
 Like 0  
 </button>  
 </section>  
 </div>  
  
 <script *src*="./js/task\_12.js"></script>  
</div>  
</body>  
</html>

* ***SASS:***

\*  
 margin: 0  
 padding: 0  
 box-sizing: border-box  
  
.*wrapper* width: 100%  
 height: 100vh  
 display: flex  
 align-items: center  
 flex-direction: column  
  
.*button\_section* border: 1.5px solid darkgrey  
 background-color: #e8e8e8  
 margin: 10px 0  
 max-width: 20vw  
 display: flex  
 flex-direction: column  
 justify-content: center  
 align-items: center  
 width: 250px  
 height: 10vh  
  
.*button\_section\_\_button\_like* width: 150px  
 height: 50px  
 font-size: 20px  
 color: white  
 background-color: #c1c1c1  
 cursor: pointer  
 border: 2px solid darkgrey  
 border-radius: 10px  
  
 &:*hover* background-color: #a2a2a2  
  
 display: flex  
 justify-content: space-evenly  
 align-items: center  
  
.*button\_section\_\_raccoon\_ico* width: 40px  
 height: 40px

* ***JS:***

'use strict';  
  
*const* buttonLike = document.querySelector('#buttonLike');  
  
buttonLike.addEventListener('click', () => {  
 buttonLike.value = `${*parseInt*(buttonLike.value) + 1}`;  
 buttonLike.innerHTML = `<img src="../../../../img/icons/racoon\_animal\_icon\_123571.ico" alt="raccoon\_ico"   
 class="button\_section\_\_raccoon\_ico">  
 Like ${buttonLike.value}`;  
});

***Результат програми:***



***Репозиторій лабораторної роботи:***

* https://github.com/AndriiBabushko/Frontend

***Висновок:*** під час виконання лабораторної роботи було отримано навички обробки подій в мові JavaScript.