

Лабораторна робота №5

Хід роботи:

Завдання на лабораторну роботу:

1. Завдання на лабораторну роботу:

					«Житомирська політехніка».22. 121.01.000–Лр5		
Змн.	Арк.	№ докум.	Підпис	Дата	Звіт з лабораторної роботи ФІКТ Гр. ВТ-21-1[1]		
Розроб.	Бабушко А.С.						
Перевір.	Фуріхата Д.В.						
Керівник							
Н. контр.							
Зав. каф.							
					Літ.	Арк.	Аркушів
						1	31

1. Завдання

Створіть HTML-сторінку з `div`, який займає всю ширину та висоту екрану. При русі мишки всередині `div` виведіть поточні координати мишки. При натисканні кнопкою миші туди ж виводить, якою саме кнопкою було здійснено клік (правою чи лівою).

X = 123, Y = 50



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">
  <link rel="icon" href="../../img/icons/racoon_animal_icon_123571.ico">
</head>
<body>
<div class="wrapper">
  <div class="container"><p class="coordinates"></p><p class="mouse_click"></p></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
justify-content: center

.container
display: flex
flex-direction: column
justify-content: center
align-items: center
width: 150px
height: 100px
font-size: 14px
border: 2px solid #a5a5a5
```

- **JS:**

```
'use strict'

const findOutButtonClick = (button) => {
  switch (button) {
    case 0:
      return 'Left button clicked.';
  }
}
```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				2
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    case 1:
        return 'Middle button clicked.';
    case 2:
        return 'Right button clicked.';
    case 3:
        return 'Back button clicked.';
    case 4:
        return 'Forward button clicked.';
    default:
        return `Unknown button code: ${button}`;
}

const coordinates = document.querySelector('.coordinates');
const mouse_click = document.querySelector('.mouse_click');

document.querySelector('.wrapper').addEventListener('mousemove', (e) => {
    const coordinateX = e.clientX;
    const coordinateY = e.clientY;
    coordinates.innerHTML = `X: ${coordinateX}; Y: ${coordinateY}`;
});

window.addEventListener('mouseup', (e) => {
    mouse_click.innerHTML = `${findOutButtonClick(e.button)}`
})

```

Результат програми:

X: 1176; Y: 423
Left button clicked.

X: 1142; Y: 409
Right button clicked.

Назад
Вперед
Перезагрузить

Сохранить как.
Печать...
Трансляция...

2. Завдання

Перенесіть червоний квадрат у місце натискання курсору



Виконання:

- **HTML:**

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				3
Змн.	Арк.	№ докум.	Підпис	Дата		

```

<!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/raccoon_animal_icon_123571.ico">
</head>
<body>
<div class="wrapper">
  <div class="square"></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

.square
display: block
border: 2px solid #a5a5a5
width: 50px
height: 50px

```

- **JS:**

```

'use strict';

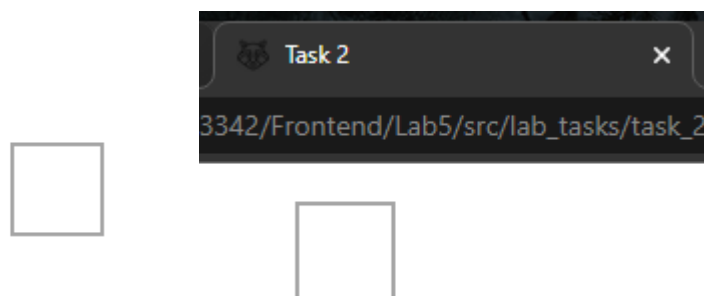
const wrapper = document.querySelector('.wrapper');
const square = document.querySelector('.square');

wrapper.addEventListener('click', (e) => {
  const coordinateX = e.clientX;
  const coordinateY = e.clientY;

  square.style.position = 'absolute';
  square.style.left = coordinateX + 'px';
  square.style.top = coordinateY + 'px';
});

```

Результат програми:



		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

3. Завдання

При редагуванні інпута, дублюємо останній символ у текстовий рядок.



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="section">
      <table class="section__table">
        <tr>
          <td>
            <label for="inputText" class="table__label_text">Enter
text:</label>
          </td>
          <td>
            <input type="text" name="inputText" id="inputText"
class="table__input_text" maxlength="40">
          </td>
        </tr>
        <tr>
          <td colspan="2">
            <div class="table__output_text"></div>
          </td>
        </tr>
      </table>
    </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
justify-content: center
flex-direction: column

.section
border: 2px solid black
margin: 10px 0
```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```
padding: 10px

.section_table
border-collapse: collapse

.table_label_text
margin-right: 10px
```

- **JS:**

```
'use strict';

const outputField = document.querySelector('.table_output_text');
const inputText = document.querySelector('#inputText');

inputText.addEventListener('input', (event) => {
    outputField.innerHTML = `<p>Entered input text:</p><p>${event.target.value}</p>`;
})
```

Результат програми:

Enter text:

Entered input text: 123123543

Enter text:

Entered input text: 123

4. Завдання

При натисканні на елемент, він замальовується білим кольором. При кліку мишкою і одночасному натисканні CTRL – замальовується червоним.

Відповідно: ALT+click – зелений, SHIFT+click – синій.

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="section"></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
```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

```
align-items: center
justify-content: center
flex-direction: column

.section
border: 2px solid black
border-radius: 15px
width: 100px
height: 100px
cursor: pointer
```

- **JS:**

```
'use strict';

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

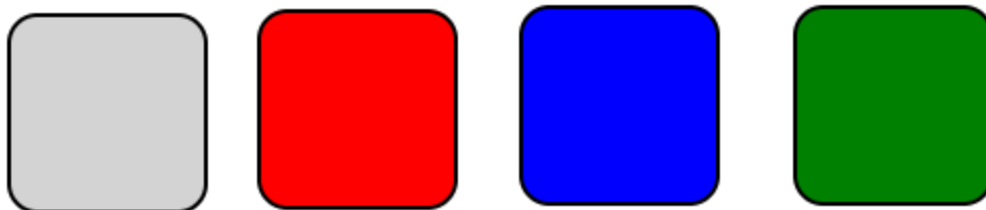
section.addEventListener('click', (event) => {
  if (event.button === 0)
    section.style.backgroundColor = 'lightgrey';

  if ((event.ctrlKey || event.metaKey) && event.isTrusted)
    section.style.backgroundColor = 'red';

  if ((event.altKey) && event.isTrusted)
    section.style.backgroundColor = 'green';

  if (event.shiftKey && event.isTrusted)
    section.style.backgroundColor = 'blue';
});
```

Результат програми:



5. Завдання

Створіть список `` з декількома ``. Додайте кнопку, при натисканні якої, в кінець списку додається ще один ``

При натисканні на будь-який(включно з створеними) ``, змінити довільну css властивість(background, margin, font і тд.).

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">
```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

```

<section class="section">
  <table class="section__table">
    <tr>
      <td>
        <label for="olList" class="table__label_ol_list"
          id="labelOlList">Ol elements list:</label>
      </td>
    </tr>
    <tr>
      <td>
        <ol class="table__ol_list" id="olList">
          <li class="ol_list__li_element">1</li>
          <li class="ol_list__li_element">2</li>
          <li class="ol_list__li_element">3</li>
          <li class="ol_list__li_element">4</li>
          <li class="ol_list__li_element">5</li>
        </ol>
      </td>
    </tr>
    <tr>
      <td>
        <input type="button" value="Add element to ol list"
          id="buttonAddElemList" class="table__button_add_elem_list">
      </td>
    </tr>
  </table>
</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
justify-content: center
flex-direction: column

.section
border: 2px solid black
margin: 10px 0
padding: 10px
text-align: center

.section__table
border-collapse: collapse

.table__label_ol_list
font-size: 24px
font-weight: bold

.table__ol_list
display: flex
flex-direction: column
align-items: center

.ol_list__li_element
cursor: pointer

.table__button_add_elem_list
margin: 10px 0

```



```
width: 100%
height: 28px
```

- JS:

```
'use strict';

const getRandomInt = (fromNumber, toNumber) => Math.round((Math.random() * (toNumber - fromNumber) + fromNumber));
const generateRandomColor = () => Math.floor(Math.random() * 16777215).toString(16);

function addEventListenersToList(nodeList) {
  for (let i = 0; i < nodeList.length; i++) {
    nodeList[i].addEventListener('click', (event) => {
      const chosenCss = getRandomInt(0, 2);
      const target = event.target;
      console.log(chosenCss);
      switch (chosenCss) {
        case 0:
          target.style.backgroundColor = '#' + generateRandomColor();
          break;
        case 1:
          target.style.fontSize = getRandomInt(8, 28) + 'px';
          break;
        case 2:
          target.style.padding = `${getRandomInt(0, 25)}px, ${getRandomInt(0, 25)}px, ${getRandomInt(0, 25)}px, ${getRandomInt(0, 25)}px`;
          break;
      }
    })
  }
}

const buttonAddElemList = document.querySelector('#buttonAddElemList');
const olList = document.querySelector('#olList');
let olListElements = document.querySelectorAll('.ol_list__li_element');

let elemsList = []
for (let i = 0; i < olListElements.length; i++) {
  elemsList.push(olListElements[i].textContent);
}
addEventListenersToList(olListElements);

buttonAddElemList.addEventListener('click', () => {
  const element = document.createElement('li');
  element.innerHTML = `${elemsList.length + 1}`;
  element.classList.add('ol_list__li_element');

  elemsList.push((elemsList.length + 1).toString());
  olList.appendChild(element);

  olListElements = document.querySelectorAll('.ol_list__li_element');
  addEventListenersToList(olListElements);
});
```

Результат програми:

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

Ol elements list:

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

Add element to ol list

Ol elements list:

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15

Add element to ol list

6. Завдання

Створіть таблицю `<table>` з двома колонками(прізвище та імя)

Створіть форму для додавання нових рядків в таблицю(2 інпута та 1 кнопку)

Кожне поле у таблиці має містити наступну подію – при кліку, випадає prompt, і введені дані переносяться у дане поле.

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="section">
      <table class="section__table">
        <tr>
          <td>
            <table class="table__output_table" id="outputNameSurnameTable">
              <tr>
                <td>
                  Surname
                </td>
                <td>

```

```

        Name
      </td>
    </tr>
  </table>
</td>
</tr>
<tr>
  <td>
    <input type="button" value="Add element to table"
id="buttonAddElemTable" class="table__button_add_elem_table">
  </td>
</tr>
</table>
</section>
</div>
<script src="./js/task_6.js"></script>
</div>
</body>
</html>

```

• SASS:

```

*
margin: 0
padding: 0
box-sizing: border-box
// outline: 1px solid red

.wrapper
width: 100%
height: 100vh
display: flex
align-items: center
justify-content: center
flex-direction: column

.section
border: 2px solid black
margin: 10px 0
padding: 10px
text-align: center
width: 600px
max-width: 60vw

.section__table
width: 100%
border-collapse: collapse

.table__button_add_elem_table
margin: 10px 0
width: 100%
height: 28px
cursor: pointer

.table__output_table
border-collapse: collapse
width: 100%
tr
  &:first-child
    td
      border: 2px solid black
      cursor: auto
    td
      border: 1.5px solid black
      cursor: pointer

```

• JS:

```

'use strict';

function addEventListenersToTable(tableNodeList) {
  for (let i = 0; i < tableNodeList.length; i++) {

```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        tableNodeList[i].addEventListener('click', (event) => {
            const target = event.target;
            const surname = target.parentElement.firstChild.textContent;
            const name = target.parentElement.lastChild.textContent;
            alert(`Clicked row:\nSurname: '${surname}'; Name: '${name}'`);
        });
    }
}

const buttonAddElemTable = document.querySelector('#buttonAddElemTable');

buttonAddElemTable.addEventListener('click', (event) => {
    const surname = prompt('Enter surname to add to table:');
    const name = prompt('Enter name to add to table:');

    const outputNameSurnameTable = document.querySelector('#outputNameSurnameTable')

    const newTableRow = document.createElement('tr');
    const newSurnameColumn = document.createElement('td');
    const newNameColumn = document.createElement('td');

    newSurnameColumn.innerHTML = surname;
    newNameColumn.innerHTML = name;

    newTableRow.appendChild(newSurnameColumn);
    newTableRow.appendChild(newNameColumn);
    outputNameSurnameTable.appendChild(newTableRow);

    const outputTableElems = document.querySelectorAll('#outputNameSurnameTable tr');
    console.log(outputTableElems);
    addEventListenersToTable(outputTableElems);
});

```

Результат програми:

Surname	Name
Add element to table	

Підтвердіть дійство на сторінці localhost:63342

Enter surname to add to table:

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

Surname	Name
Babushko	Andrii

Add element to table

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				13
Змн.	Арк.	№ докум.	Підпис	Дата		

```

</head>
<body>
<div class="wrapper">
  <div class="container">
    <section class="section">
      <table class="section__table">
        <tr>
          <td>
            <label for="inputName" id="labelName"
class="table__label_name">Enter name(can't contain digits):</label>
          </td>
          <td>
            <input type="text" name="inputName" id="inputName"
class="table__input_name" required>
          </td>
        </tr>
      </table>
    </section>
  </div>
  <script src="./js/task_7.js"></script>
</div>
</body>
</html>

```

- **SASS:**

```

*
margin: 0
padding: 0
box-sizing: border-box
// outline: 1px solid red

.wrapper
width: 100%
height: 100vh
display: flex
align-items: center
justify-content: center
flex-direction: column

.section
border: 2px solid black
margin: 10px 0
padding: 10px
text-align: center
width: 450px
max-width: 60vw

.section__table
width: 100%
border-collapse: collapse

```

- **JS:**

```

'use strict';

const containsNumbers = (str) => /\d/.test(str);

const inputName = document.querySelector('#inputName');

inputName.addEventListener('input', (event) => {
  if (containsNumbers(event.target.value)) {
    console.log('CONTAIN DIGITS!')
    console.log(`Target value: ${event.target.value}`);
    event.target.value = event.target.value.replace(/\d/g, '');
  }
});

```

Результат програми:

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				14
Змн.	Арк.	№ докум.	Підпис	Дата		

Enter name(can't contain digits):

Andrii

Target value: Andrii

CONTAIN DIGITS!

Target value: Andrii1

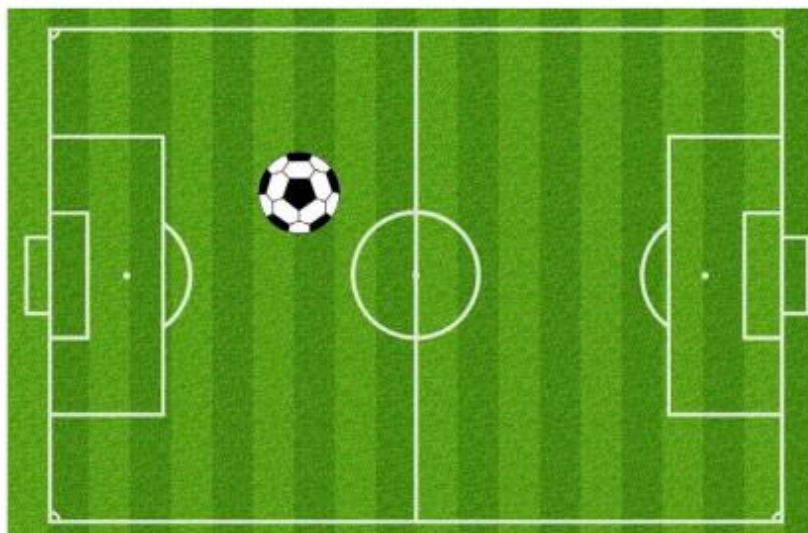
task 7.js:9

task 7.js:10

8. Завдання

Створіть HTML-сторінку з футбольним полем, яке займає всю ширину та висоту екрану, і м'ячем розміром 100 на 100 пікселів.

- зробіть так, щоб клікнувши мишкою по полю, м'яч плавно переміщався на місце кліка.
- враховуйте, щоб центр м'яча зупинявся саме там, де було здійснено клік мишкою.
- також передбачте, аби м'яч не вилітав за межі поля.



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="football_stadium">
      
    </div>
  </div>
```

Бабушко А.С.

Фуріхата Д.В.

№ докум.

Підпис

Дата

«Житомирська політехніка».22.121.01.000 – Лр5

Арк.

15

```
<script src="./js/task_8.js"></script>
</body>
</html>
```

• SASS:

```
*
margin: 0
padding: 0
box-sizing: border-box
// outline: 1px solid red

body
  overflow-y: hidden

.wrapper
  width: 100%
  height: 100vh
  display: flex
  justify-content: center

.football_stadium
  background: url("../img/football_stadium_2d.png") no-repeat
  background-size: contain
  width: 70%
  height: 95%
  cursor: pointer

.football_stadium_ball
  width: 100px
  height: 100px
  position: absolute
  top: 43%
  left: 47%
  transition: top .5s ease, left .5s ease
```

• JS:

```
'use strict';

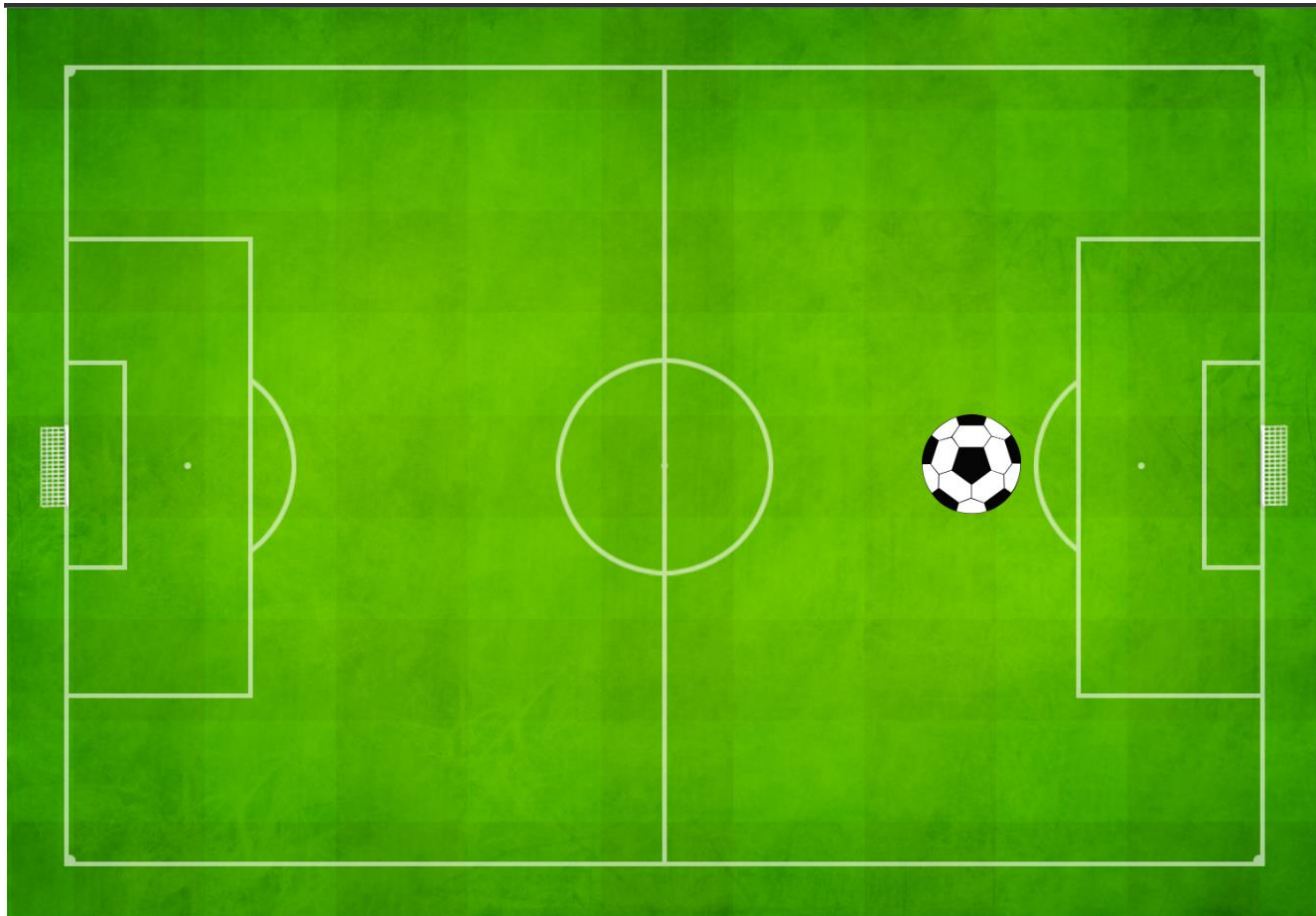
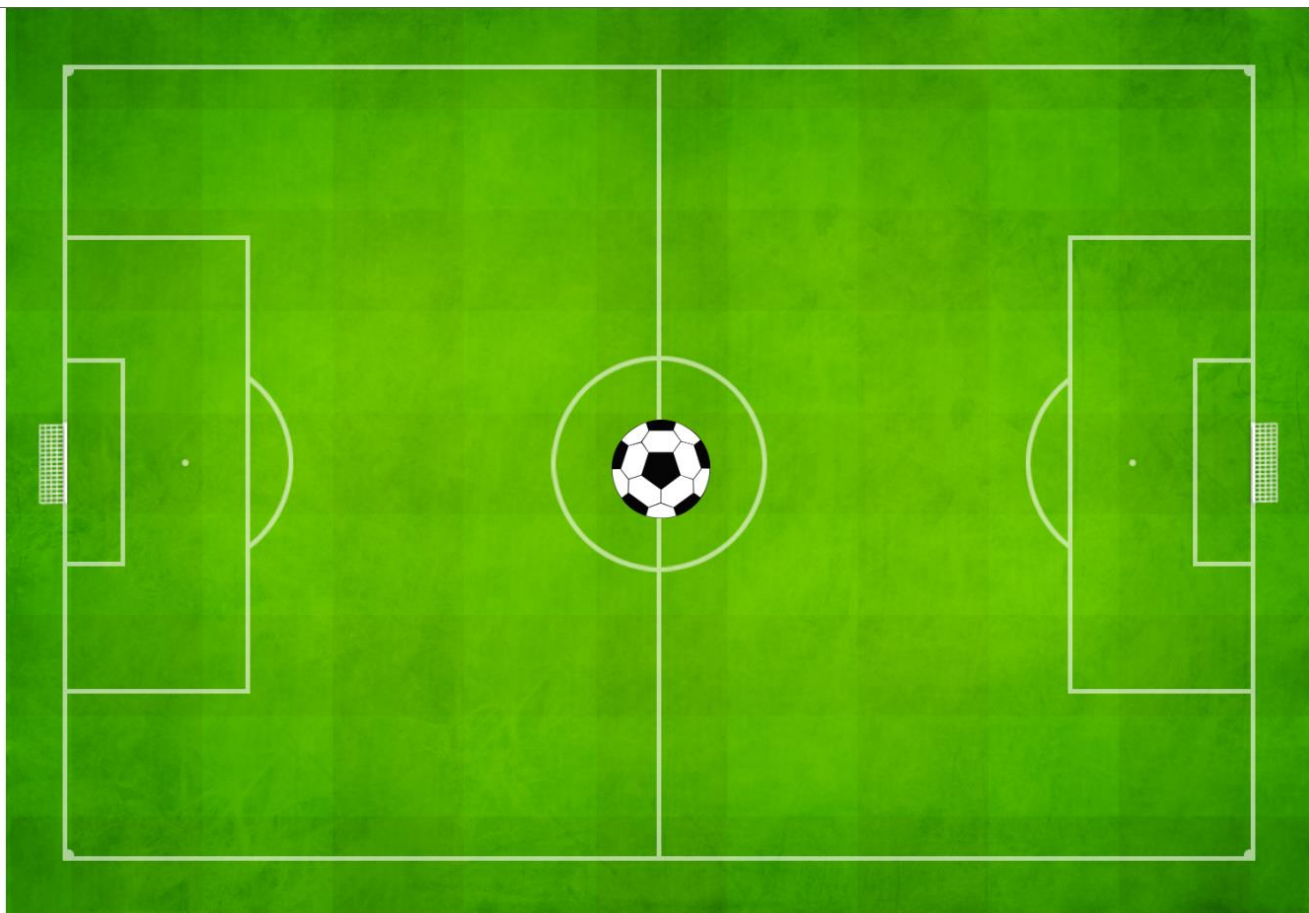
const footballStadium = document.querySelector('.football_stadium');
const ball = document.querySelector('#ball');

const ballTiming = {
  duration: 2000,
  iterations: 1,
}

footballStadium.addEventListener('click', (event) => {
  const FinishX = event.clientX;
  const FinishY = event.clientY;
  const footballStadiumRect = footballStadium.getBoundingClientRect()
  if (FinishY < footballStadiumRect.height + window.innerHeight * 0.05 && FinishX <
  footballStadiumRect.width + window.innerWidth * 0.2) {
    ball.style.top = FinishY - 50 + 'px';
    ball.style.left = FinishX - 50 + 'px';
  }
}, false);
```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				16
Змн.	Арк.	№ докум.	Підпис	Дата		

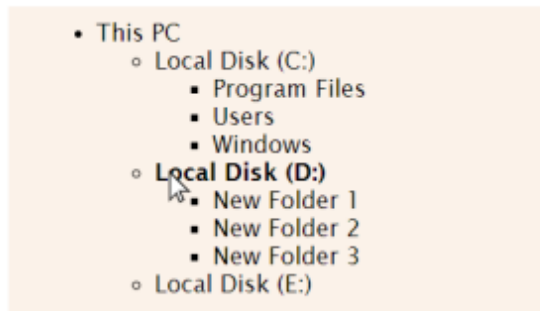
Результат програми:



		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				17
Змн.	Арк.	№ докум.	Підпис	Дата		

9. Завдання

Створіть HTML-сторінку з деревом вкладених директорій. При натисканні на елемент списку, він має згортатися або розгортатися.



або

Створіть HTML-сторінку з меню, яке має випадаючі списки. Список елементів підменю має з'являтися по клацанню на відповідний елемент меню.



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="nav_bar">
    <div class="nav_bar_item" id="home">Home</div>
    <div class="nav_bar_item" id="categories">Categories
      <div class="nav_bar_sub_nav_bar">
        <div class="sub_nav_bar_item">Dresses</div>
        <div class="sub_nav_bar_item">Shirts</div>
        <div class="sub_nav_bar_item">Jeans</div>
        <div class="sub_nav_bar_item">Shoes</div>
      </div>
    </div>
    <div class="nav_bar_item" id="about">About</div>
    <div class="nav_bar_item" id="contact">Contact</div>
  </div>
</div>
<script src="../js/task_9.js"></script>
</body>
</html>
```

- **SASS:**

```
*
margin: 0
padding: 0
box-sizing: border-box
// outline: 1px solid red

.wrapper
width: 100%
height: 100vh
display: flex
justify-content: center

.nav_bar
width: 30%
height: 40px
background-color: steelblue
display: flex
justify-content: space-evenly

.nav_bar__item, .sub_nav_bar__item
font-size: 18px
padding: 10px
width: 100%
color: white
background-color: steelblue
display: flex
align-items: center
justify-content: center
cursor: pointer

&:hover
animation-name: nav_item_color
animation-duration: 3s
animation-timing-function: ease-in-out

@keyframes nav_item_color
from
background-color: steelblue
50%
background-color: #6b94b7
to
background-color: steelblue

.nav_bar__sub_nav_bar
&.active
display: flex
position: relative
display: none
flex-direction: column
justify-content: space-evenly
align-items: center
width: 90px

.nav_bar__sub_nav_bar
position: absolute
top: 4.1%
background-color: steelblue
```

- **JS:**

```
'use strict';

const navBarItem = document.querySelectorAll('.nav_bar__item');

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

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				19
Змн.	Арк.	№ докум.	Підпис	Дата		

```
navBarItems[i].addEventListener('click', (event) => {
  const subNavBarElement = event.target.firstChild
  if (subNavBarElement) {
    console.log('SUB NAV BAR!');
    subNavBarElement.classList.toggle('active');
  }
});
}
```

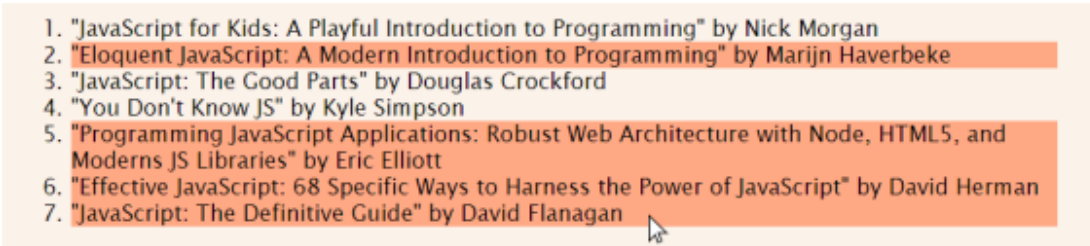
Результат програми:



10. Завдання

Створіть HTML-сторінку зі списком книг.

- При натисканні на елемент, колір тексту має змінюватися на помаранчевий. При повторному натисканні на іншу книгу, попередня має повернути початковий колір.
- Якщо при натисканні мишкою була затиснута клавіша Ctrl, то елемент додається / видаляється із виділених.
- Якщо при натисканні мишкою була затиснута клавіша Shift, то до виділення додаються всі елементи в проміжку від попереднього натиснутого до поточного.



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/raccoon_animal_icon_123571.ico">
</head>
```

```

<body>
<div class="wrapper">
  <div class="books_container">
    <ol class="books_container_ol_list">
      <li class="ol_list_li_book">Robert Cecil Martin - Clean Code</li>
      <li class="ol_list_li_book">Andy Hunt and Dave Thomas - The Pragmatic
Programmer</li>
      <li class="ol_list_li_book">Erich Gamma, John Vlissides, Ralph Johnson,
Richard Helm - Design Patterns: Elements of Reusable Object-Oriented Software</li>
      <li class="ol_list_li_book">Robert Cecil Martin - The Clean Coder: A Code of
Conduct for Professional Programmers</li>
      <li class="ol_list_li_book">Kent Beck and Martin Fowler - Refactoring:
Improving the Design of Existing Code</li>
      <li class="ol_list_li_book">Steve McConnell - Code Complete</li>
      <li class="ol_list_li_book">Eric Freeman, Elisabeth Robson, Bert Bates, Kathy
Sierra - Head First Design Patterns</li>
    </ol>
  </div>
</div>
<script src="./js/task_10.js"></script>
</body>
</html>

```

• SASS:

```

*
margin: 0
padding: 0
box-sizing: border-box
// outline: 1px solid red

.wrapper
width: 100%
height: 100vh
display: flex
justify-content: center
align-items: center

.books_container
width: 46%
height: 18%
position: relative
display: flex
justify-content: center
align-items: center
background-color: burlywood

.books_container_ol_list
cursor: pointer

.ol_list_li_book
  &.active
background-color: orange

```

• JS:

```

'use strict';

const booksToggle = (event, books, booksStatus, index) => {
  event.target.classList.toggle('active');

  clickedBooksIndexes.push(index)

  if (booksStatus[index] === 'active') {
    booksStatus[index] = 'inactive';
  } else {
    booksStatus[index] = 'active'
  }

  console.log(booksStatus);
};

```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				21
Змн.	Арк.	№ докум.	Підпис	Дата		

```

const setBooks = (event, bookIndex, books, booksStatus, clickedBooksIndexes) => {
  const index = parseInt(event.target.getAttribute('value'));
  if (!booksStatus.includes('active') || event.ctrlKey) {
    booksToggle(event, books, booksStatus, index);
    return;
  }
  if (event.shiftKey) {
    console.log('SHIFT');
    clickedBooksIndexes.push(index)
    const startIndex = clickedBooksIndexes[clickedBooksIndexes.length - 2];
    const finishIndex = clickedBooksIndexes[clickedBooksIndexes.length - 1];
    console.log(startIndex, finishIndex);
    if (startIndex > finishIndex) {
      for (let i = startIndex; i >= finishIndex; i--) {
        if (i !== startIndex)
          books[i].classList.toggle('active');
        booksStatus[i] = 'active';
      }
    }
    if (startIndex < finishIndex) {
      for (let i = startIndex; i <= finishIndex; i++) {
        if (i !== startIndex)
          books[i].classList.toggle('active');
        booksStatus[i] = 'active';
      }
    }
    console.log(booksStatus);
    return;
  }
  if (booksStatus.includes('active')) {
    console.log(clickedBooksIndexes);
    const activeIndex = clickedBooksIndexes[clickedBooksIndexes.length - 1];
    books[activeIndex].classList.toggle('active');
    booksStatus[activeIndex] = 'inactive';
    booksToggle(event, books, booksStatus, index);
  }
}

const books = document.querySelectorAll('.ol_list__li_book');

let booksStatus = [];
let clickedBooksIndexes = []
for (let i = 0; i < books.length; i++) {
  books[i].classList.add('inactive');
  books[i].setAttribute('value', i.toString());
  booksStatus.push('inactive');

  books[i].addEventListener('click', (event) => {
    console.log('EVENT!');
    setBooks(event, i, books, booksStatus, clickedBooksIndexes)
  });
}

```

Результат програми:

0. Robert Cecil Martin - Clean Code

1. Andy Hunt and Dave Thomas - The Pragmatic Programmer
2. Erich Gamma, John Vlissides, Ralph Johnson, Richard Helm - Design Patterns: Elements of Reusable Object-Oriented Software
3. Robert Cecil Martin - The Clean Coder: A Code of Conduct for Professional Programmers
4. Kent Beck and Martin Fowler - Refactoring: Improving the Design of Existing Code
5. Steve McConnell - Code Complete
6. Eric Freeman, Elisabeth Robson, Bert Bates, Kathy Sierra - Head First Design Patterns

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				22
Змн.	Арк.	№ докум.	Підпис	Дата		

0. Robert Cecil Martin - Clean Code
1. Andy Hunt and Dave Thomas - The Pragmatic Programmer
2. Erich Gamma, John Vlissides, Ralph Johnson, Richard Helm - Design Patterns: Elements of Reusable Object-Oriented Software
3. Robert Cecil Martin - The Clean Coder: A Code of Conduct for Professional Programmers
4. Kent Beck and Martin Fowler - Refactoring: Improving the Design of Existing Code
5. Steve McConnell - Code Complete
6. Eric Freeman, Elisabeth Robson, Bert Bates, Kathy Sierra - Head First Design Patterns

0. Robert Cecil Martin - Clean Code
1. Andy Hunt and Dave Thomas - The Pragmatic Programmer
2. Erich Gamma, John Vlissides, Ralph Johnson, Richard Helm - Design Patterns: Elements of Reusable Object-Oriented Software
3. Robert Cecil Martin - The Clean Coder: A Code of Conduct for Professional Programmers
4. Kent Beck and Martin Fowler - Refactoring: Improving the Design of Existing Code
5. Steve McConnell - Code Complete
6. Eric Freeman, Elisabeth Robson, Bert Bates, Kathy Sierra - Head First Design Patterns

11. Завдання

Створіть HTML-сторінку для відображення / редагування тексту.

При відкритті сторінки, текст відображається за допомогою тегу div. При натисканні на Ctrl + E, замість div з'являється textarea з тим самим текстом, який тепер можна редагувати. При натисканні на Ctrl + S, замість textarea з'являється div із вже зміненим текстом.

Вимкніть поведінку за замовчуванням для цих комбінацій клавіш або виберіть інші комбінації (Ctrl + E – Alt + E, і т.д.).

Press Ctrl+E for edit text.
Press Ctrl+S for save text.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.

Press Ctrl+E for edit text.
Press Ctrl+S for save text.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.....

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/raccoon_animal_icon_123571.ico">
</head>
```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				23
Змн.	Арк.	№ докум.	Підпис	Дата		

```

<body>
<div class="wrapper">
  <div class="container">
    <div class="container__hotkeys">
      <p class="hotkeys__p_ctrl_e" id="ctrl_e">Press <strong>CTRL+E</strong> for edit
text</p>
      <p class="hotkeys__p_ctrl_s" id="ctrl_s">Press <strong>CTRL+S</strong> for save
text</p>
    </div>
    <div class="container__div_text">
      <div class="div_text__text">
        <p>
          Lorem ipsum dolor sit amet, consectetur adipisicing elit. Aliquid
assumenda atque cum eaque enim eveniet incidunt laboriosam maiores nemo numquam odit,
praesentium quidem ratione totam unde velit veniam vero, voluptatibus!
At maiores minus nulla quis voluptatibus! Adipisci atque, aut consequatur culpa ea esse
et eum eveniet ex facilis, impedit ipsum maxime modi molestias pariatur
porro quo quod voluptatum? Enim, et. Ab excepturi libero nam non! Aut minima quis quod
rem soluta. Aperiam dignissimos exercitationem explicabo hic, ipsam
nemo possimus. A aspernatur enim natus obcaecati quibusdam, sed veritatis. Alias, illum,
laudantium. Consequatur delectus dignissimos expedita illum in laborum
laudantium mollitia nisi obcaecati omnis, porro quas reiciendis similique voluptatibus
voluptatum? Aliquam aspernatur commodi enim esse exercitationem illo
itaque laboriosam natus odit quos!
        </p>
      </div>
    </div>
  </div>
</div>
<script src="./js/task_11.js"></script>
</body>
</html>

```

- **SASS:**

```

*
margin: 0
padding: 0
box-sizing: border-box
// outline: 1px solid red

.wrapper
width: 100%
height: 100vh
display: flex
justify-content: center
align-items: center

.container
width: 45%
display: flex
flex-direction: column
justify-content: center
border: 1px solid #4a4a4a
border-radius: 15px
background-color: blanchedalmond
padding: 15px

.container__hotkeys
padding-bottom: 5px
margin-bottom: 15px
border-bottom: 1px solid grey

.div_text__text
width: 100%

&.active
display: inherit

.div_text__textarea
width: 100%

```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				24
Змн.	Арк.	№ докум.	Підпис	Дата		


```

&.active
display: inherit

.textarea__change_text
width: 100%
height: 150px
resize: none

```

• JS:

```

'use strict';

const deleteElemFromArr = (array, deleteElem) => {
  const index = array.indexOf(deleteElem);

  if (index > -1) { // only splice array when item is found
    array.splice(index, 1); // 2nd parameter means remove one item only
  }

  return array
}

const removeSpacesAndBreakLines = (splitText) => {
  for (let i = 0; i < splitText.length; i++) {
    if (splitText[i] === ' ' || /^[ \n\r]$/.test(splitText[i]))
      splitText = deleteElemFromArr(splitText, splitText[i]);
  }

  return splitText
}

document.addEventListener('keydown', (keyDownEvent) => {
  // if ((keyDownEvent.ctrlKey || keyDownEvent.metaKey) && (keyDownEvent.key === 'e' ||
keyDownEvent.key === 'E')) ||
  // (keyDownEvent.ctrlKey || keyDownEvent.metaKey) && (keyDownEvent.key === 's' ||
keyDownEvent.key === 'S')) {
    // keyDownEvent.preventDefault();
    // }
    if ((keyDownEvent.key === 'e' || keyDownEvent.key === 'E')) {
      console.log('ctrl+e')

      const divText = document.querySelector('.div_text__text');

      let splitText = divText.firstElementChild.innerHTML.replace(/\n/g, ' ').split(' ');
      for (let i = 0; i < 3; i++) {
        splitText = removeSpacesAndBreakLines(splitText);
      }
      const text = splitText.join(' ');

      const divTextarea = document.createElement('div');
      const textarea = document.createElement('textarea');

      divTextarea.setAttribute('class', 'div_text__textarea');
      textarea.setAttribute('name', 'textareaTextChange');
      textarea.setAttribute('id', 'textareaTextChange');
      textarea.setAttribute('class', 'textarea__change_text');

      textarea.value = text;

      divTextarea.appendChild(textarea);
      divText.parentElement.appendChild(divTextarea);

      divText.remove();

      if ((keyDownEvent.key === 's' || keyDownEvent.key === 'S')) {
        console.log('ctrl+s');

        const divTextarea = document.querySelector('.div_text__textarea');

```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				25
Змн.	Арк.	№ докум.	Підпис	Дата		

```

const textarea = document.querySelector('#textareaTextChange');
const textareaText = textarea.value;

const divText = document.createElement('div');
const p = document.createElement('p');

divText.setAttribute('class', 'div_text__text');

p.textContent = textareaText;

divText.appendChild(p);
divTextarea.parentElement.appendChild(divText);

divTextarea.remove();
}
}, false);

```

Результат програми:

Press **CTRL+E** for edit text
Press **CTRL+S** for save text

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Aliquid assumenda atque cum eaque enim eveniet incidunt laboriosam maiores nemo numquam odit, praesentium quidem ratione totam unde velit veniam vero, voluptatibus! At maiores minus nulla quis voluptatibus! Adipisci atque, aut consequatur culpa ea esse et eum eveniet ex facilis, impedit ipsum maxime modi molestias pariatur porro quo quod voluptatum? Enim, et. Ab excepturi libero nam non! Aut minima quis quod rem soluta. Aperiam dignissimos exercitationem explicabo hic, ipsam nemo possimus. A aspernatur enim natus obcaecati quibusdam, sed veritatis. Alias, illum, laudantium. Consequatur delectus dignissimos expedita illum in laborum laudantium mollitia nisi obcaecati omnis, porro quas reiciendis similique voluptatibus voluptatum? Aliquam aspernatur commodi enim esse exercitationem illo itaque laboriosam natus odit quos!

Press **CTRL+E** for edit text
Press **CTRL+S** for save text

12345

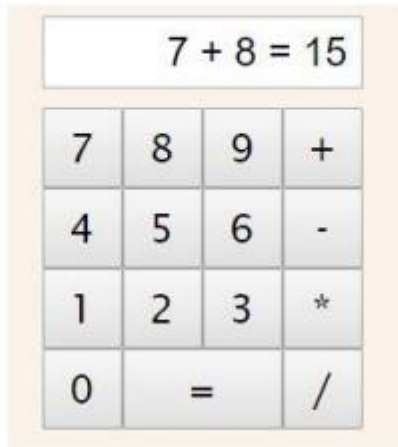
Press **CTRL+E** for edit text
Press **CTRL+S** for save text

12345

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				26
Змн.	Арк.	№ докум.	Підпис	Дата		

12. Завдання

Створіть HTML-сторінку «Калькулятор» і реалізуйте його функціональність.



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">
    <div class="container_calculator">
      <input type="text" name="calculatorField" id="calculatorField"
class="calculator_field" readonly>
      <button type="button" id="calculatorTotalClear"
class="calculator_total_clear">C</button>
      <button type="button" id="calculator7" class="calculator__number" data-
num="7">7</button>
      <button type="button" id="calculator8" class="calculator__number" data-
num="8">8</button>
      <button type="button" id="calculator9" class="calculator__number" data-
num="9">9</button>
      <button type="button" id="calculatorAdd" class="calculator__action" data-
action="Add">+</button>
      <button type="button" id="calculator4" class="calculator__number" data-
num="4">4</button>
      <button type="button" id="calculator5" class="calculator__number" data-
num="5">5</button>
      <button type="button" id="calculator6" class="calculator__number" data-
num="6">6</button>
      <button type="button" id="calculatorSubtract" class="calculator__action" data-
action="Subtract">-</button>
      <button type="button" id="calculator1" class="calculator__number" data-
num="1">1</button>
      <button type="button" id="calculator2" class="calculator__number" data-
num="2">2</button>
      <button type="button" id="calculator3" class="calculator__number" data-
num="3">3</button>
      <button type="button" id="calculatorMultiply" class="calculator__action" data-
action="Multiply">*</button>
      <button type="button" id="calculator0" class="calculator__number" data-
num="0">0</button>
```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				27
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        <button type="button" id="calculatorEqual" class="calculator__action" data-
equals="">=</button>
        <button type="button" id="calculatorDivide" class="calculator__action" data-
action="Divide">/</button>
    </div>
</div>
<script src="./js/task_12.js"></script>
</body>
</html>

```

• SASS:

```

*
margin: 0
padding: 0
box-sizing: border-box
// outline: 1px solid red

.wrapper
width: 100%
height: 100vh
display: flex
justify-content: center
align-items: center

.container
width: 20%
height: 33%
display: flex
flex-direction: column
justify-content: center
align-items: center
border: 1px solid #4a4a4a
border-radius: 10px
background-color: #fff0d8
padding: 15px

.container_calculator
width: 57%
height: 100%
display: flex
flex-wrap: wrap
justify-content: center
align-items: center

.calculator_field
width: 100%
height: 32px
font-size: 18px
margin-bottom: 10px

.calculator_number, .calculator__action
width: 50px
height: 50px
font-size: 24px

#calculatorEqual, #calculatorClear
width: 100px
height: 50px

#calculatorTotalClear
width: 100%
height: 50px

```

• JS:

```

'use strict';

const getElement = (element) => {
    if (element.charAt(0) === "#") {
        return document.querySelector(element);
    }
}

```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				28
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    return document.querySelectorAll(element);
};

const calculatorField = getElement("#calculatorField");
const calculatorEqual = getElement("#calculatorEqual");
const calculatorNumbers = getElement(".calculator__number");
const calculatorActions = getElement(".calculator__action");
let currentNumber = "", firstNumber = "", resultNumber, operator;

const setNumber = function () {
    if (resultNumber) {
        currentNumber = this.getAttribute("data-num");
        resultNumber = "";
    } else {
        currentNumber += this.getAttribute("data-num");
    }

    calculatorField.value = currentNumber;
};

const moveNumber = function () {
    firstNumber = currentNumber;
    currentNumber = "";
    operator = this.getAttribute("data-action");

    calculatorEqual.setAttribute("data-result", "");
};

const displayNumber = () => {
    firstNumber = parseFloat(firstNumber);
    currentNumber = parseFloat(currentNumber);

    switch (operator) {
        case "Add":
            resultNumber = firstNumber + currentNumber;
            break;

        case "Subtract":
            resultNumber = firstNumber - currentNumber;
            break;

        case "Multiply":
            resultNumber = firstNumber * currentNumber;
            break;

        case "Divide":
            resultNumber = firstNumber / currentNumber;
            break;

        default:
            resultNumber = currentNumber;
    }

    if (!isFinite(resultNumber)) {
        if (isNaN(resultNumber)) {
            resultNumber = "NaN!";
        } else {
            resultNumber = "Dividing by zero!";
        }
    }

    calculatorField.value = resultNumber;
    calculatorEqual.setAttribute("data-result", resultNumber);

    firstNumber = 0;
    currentNumber = resultNumber;
};

const clearAll = () => {

```

		Бабушко А.С.			«Житомирська політехніка».22.121.01.000 – Лр5	Арк.
		Фуріхата Д.В.				29
Змн.	Арк.	№ докум.	Підпис	Дата		

```

firstNumber = "";
currentNumber = "";
calculatorField.value = "0";
calculatorEqual.setAttribute("data-result", resultNumber);
};

for (let i = 0; i < calculatorNumbers.length; i++) {
    calculatorNumbers[i].onclick = setNumber;
}

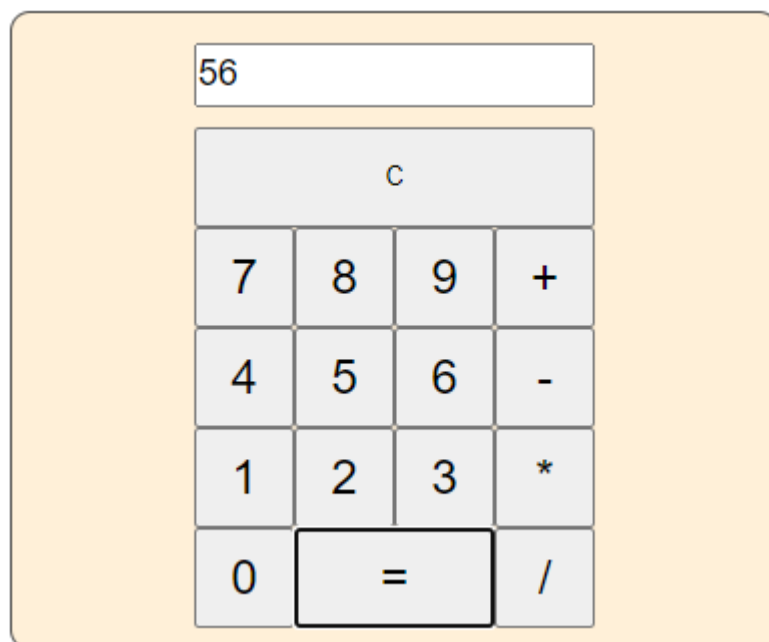
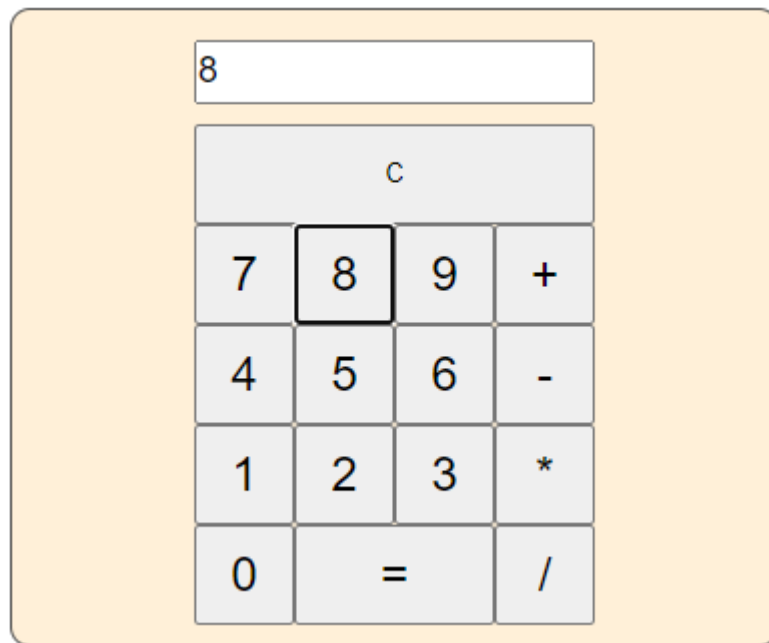
for (let i = 0; i < calculatorActions.length; i++) {
    calculatorActions[i].onclick = moveNumber;
}

calculatorEqual.onclick = displayNumber;

getElement("#calculatorTotalClear").onclick = clearAll;

```

Результат програми:



Репозиторій лабораторної роботи:

- <https://github.com/AndriiBabushko/Frontend>

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