# Solutions - Exercise: Intro to DOM

## 1.Cards

function solve() {

Array.from(document.getElementsByTagName('img')).forEach((img) => {

img.addEventListener('click', clickEvent);

});

function clickEvent(e) {

let card = e.target;

card.src = './images/whiteCard.jpg';

card.removeEventListener('click', clickEvent);

let parent = card.parentNode;

let spans = document.getElementById('result').children;

let leftSpan = spans[0];

let rightSpan = spans[2];

if(parent.id === 'player1Div'){

leftSpan.textContent = card.name;

} else if(parent.id === 'player2Div') {

rightSpan.textContent = card.name;

}

if(spans[0].textContent && spans[2].textContent){

let winner;

let looser;

if(+leftSpan.textContent > +rightSpan.textContent){

winner = document.querySelector(`#player1Div img[name='${leftSpan.textContent}']`);

looser = document.querySelector(`#player2Div img[name='${rightSpan.textContent}']`);

} else {

winner = document.querySelector(`#player2Div img[name='${rightSpan.textContent}']`);

looser = document.querySelector(`#player1Div img[name='${leftSpan.textContent}']`);

}

winner.style.border = '2px solid green';

looser.style.border = '2px solid darkred';

document.getElementById('history').textContent += `[${leftSpan.textContent} vs ${rightSpan.textContent}] `;

// let historyElement = document.getElementById('history');

// historyElement.textContent += `[${leftSpan.textContent}] vs [${rightSpan.textContent}] `;

leftSpan.textContent = '';

rightSpan.textContent = '';

// setTimeout(() => {

// leftSpan.textContent = '';

// rightSpan.textContent = '';

// }, 2000);

}

}

}

|  |
| --- |
| function solve() { |
|  | Array.from(document.getElementsByTagName('img')).forEach((img) => { |
|  | img.addEventListener('click', clickEvent); |
|  | }); |
|  |  |
|  | function clickEvent(e) { |
|  | let card = e.target; |
|  | card.src = './images/whiteCard.jpg'; |
|  | card.removeEventListener('click', clickEvent); |
|  |  |
|  | let parent = card.parentNode; |
|  | let spans = document.getElementById('result').children; |
|  |  |
|  | let leftSpan = spans[0]; |
|  | let rightSpan = spans[2]; |
|  |  |
|  | if(parent.id === 'player1Div'){ |
|  | leftSpan.textContent = card.name; |
|  | } else if(parent.id === 'player2Div') { |
|  | rightSpan.textContent = card.name; |
|  | } |
|  |  |
|  | if(spans[0].textContent && spans[2].textContent){ |
|  | let winner; |
|  | let looser; |
|  |  |
|  | if(+leftSpan.textContent > +rightSpan.textContent){ |
|  | winner = document.querySelector(`#player1Div img[name='${leftSpan.textContent}']`); |
|  | looser = document.querySelector(`#player2Div img[name='${rightSpan.textContent}']`); |
|  | } else { |
|  | winner = document.querySelector(`#player2Div img[name='${rightSpan.textContent}']`); |
|  | looser = document.querySelector(`#player1Div img[name='${leftSpan.textContent}']`); |
|  | } |
|  |  |
|  | winner.style.border = '2px solid green'; |
|  | looser.style.border = '2px solid darkred'; |
|  |  |
|  | document.getElementById('history').textContent += `[${leftSpan.textContent} vs ${rightSpan.textContent}] `; |
|  |  |
|  | leftSpan.textContent = ""; |
|  | rightSpan.textContent = ""; |
|  | } |
|  | } |
|  | } |

## 2.Chat room with Pesho

function solve() {

let buttons = Array.from(document.getElementsByTagName('button'));

let inputs = Array.from(document.getElementsByTagName('input'));

buttons.forEach((button) => {

button.addEventListener('click', function (event) {

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

let span = document.createElement('span');

let p = document.createElement('p');

let senderButton = event.target.name;

if (senderButton === 'myBtn'){

span.textContent = 'Me';

p.textContent = document.getElementById('myChatBox').value;

div.style.textAlign = 'left';

} else if (senderButton === 'peshoBtn'){

span.textContent = 'Pesho';

p.textContent = document.getElementById('peshoChatBox').value;

div.style.textAlign = 'right';

}

div.appendChild(span);

div.appendChild(p);

let chatChronology = document.getElementById('chatChronology');

chatChronology.appendChild(div);

inputs[0].value = '';

inputs[1].value = '';

})

})

}

|  |
| --- |
| function solve() { |
|  | // 0 is mine, 1 is pesho |
|  | let buttons = Array.from(document.getElementsByTagName('button')); |
|  | let inputFields = Array.from(document.getElementsByTagName('input')); |
|  |  |
|  | buttons.forEach((btn) => { |
|  | btn.addEventListener('click', function (event) { |
|  |  |
|  | let divElement = document.createElement('div'); |
|  | let spanElement = document.createElement('span'); |
|  | let pElement = document.createElement('p'); |
|  |  |
|  | let senderBtn = event.target.name; |
|  |  |
|  | if(senderBtn === 'myBtn'){ |
|  | spanElement.textContent = 'Me'; |
|  | pElement.textContent = document.getElementById('myChatBox').value; |
|  |  |
|  | divElement.style.textAlign = 'left'; |
|  |  |
|  | } else if(senderBtn === 'peshoBtn'){ |
|  | spanElement.textContent = 'Pesho'; |
|  | pElement.textContent = document.getElementById('peshoChatBox').value; |
|  |  |
|  | divElement.style.textAlign = 'right'; |
|  | } |
|  |  |
|  | divElement.appendChild(spanElement); |
|  | divElement.appendChild(pElement); |
|  |  |
|  |  |
|  | document.getElementById('chatChronology').appendChild(divElement); |
|  |  |
|  | inputFields[0].value = ""; |
|  | inputFields[1].value = ""; |
|  | }) |
|  | }) |
|  | } |

## 3.Encode and Decode Messages

function solve() {

    let buttons = document.getElementsByTagName('button');

    let textAreas = document.getElementsByTagName('textarea');

    buttons[0].addEventListener('click', encode);

    buttons[1].addEventListener('click', decode);

    function encode(){

        let icputMessage = textAreas[0].value;

        let encodedMessage = '';

        icputMessage.split('').forEach((char) =>{

            let asciiValue = char.charCodeAt(0) + 1;

            encodedMessage += String.fromCharCode(asciiValue);

        });

        textAreas[0].value = '';

        textAreas[1].value = encodedMessage;

    }

    function decode(){

        let encodedMessage = textAreas[1].value;

        let decodedMessage = '';

        encodedMessage.split('').forEach((char) => {

            let asciiValue = char.charCodeAt(0) - 1;

            decodedMessage += String.fromCharCode(asciiValue);

        });

        textAreas[1].value = decodedMessage;

    }

}

|  |
| --- |
| function solve() { |
|  | let buttons = document.getElementsByTagName('button'); |
|  | let textAreas = document.getElementsByTagName('textarea'); |
|  |  |
|  | buttons[0].addEventListener('click', encode); |
|  | buttons[1].addEventListener('click', decode); |
|  |  |
|  | function encode() { |
|  |  |
|  | let encodeMessage = textAreas[0].value; |
|  |  |
|  | let newMessage = ''; |
|  |  |
|  | encodeMessage.split('').forEach((char) => { |
|  | let asciiValue = char.charCodeAt(0) + 1; |
|  | newMessage += String.fromCharCode(asciiValue); |
|  | }); |
|  |  |
|  | textAreas[0].value = ""; |
|  | textAreas[1].value = newMessage; |
|  | } |
|  |  |
|  |  |
|  | function decode() { |
|  |  |
|  | let decodeMessage = textAreas[1].value; |
|  | let originalMessage = ""; |
|  |  |
|  | decodeMessage.split('').forEach((char) => { |
|  | let asciiValue = char.charCodeAt(0) - 1; |
|  | originalMessage += String.fromCharCode(asciiValue); |
|  | }); |
|  |  |
|  | textAreas[1].value = originalMessage; |
|  | } |
|  | } |

## 4.Locked Profile

function solve() {

let profiles = Array.from(document.getElementsByClassName('profile'))

for (let profile of profiles) {

let button = profile.getElementsByTagName('button')[0];

button.addEventListener('click', function () {

let radioUnlocked = profile.getElementsByTagName('input')[1];

let hidenFieldId = profile.getElementsByTagName('div')[1].id;

let hiddenFields = document.getElementById(`${hidenFieldId}`);

if (radioUnlocked.checked && button.textContent === 'Show more') {

hiddenFields.style.display = 'block';

button.textContent = 'Hide it';

} else if(radioUnlocked.checked && button.textContent === 'Hide it') {

hiddenFields.style.display = 'none';

button.textContent = 'Show more';

}

})

}

}

function solve() {

let user1BtnUnlock = document.getElementsByName('user1Locked')[1];

let button1 = document.getElementsByTagName('button')[0];

let user1Section = document.getElementById('user1HiddenFields');

let user2BtnUnlock = document.getElementsByName('user2Locked')[1];

let button2 = document.getElementsByTagName('button')[1];

let user2Section = document.getElementById('user2HiddenFields');

let user3BtnUnlock = document.getElementsByName('user3Locked')[1];

let button3 = document.getElementsByTagName('button')[2];

let user3Section = document.getElementById('user3HiddenFields');

button1.addEventListener('click', () => {

if (user1BtnUnlock.checked) {

user1Section.style.display = 'block';

// document.getElementsByName('user1Username')[0].disabled = false;

button1.textContent = 'Hide it';

hideIt(user1BtnUnlock, user1Section, button1);

}

});

button2.addEventListener('click', () => {

if (user2BtnUnlock.checked) {

user2Section.style.display = 'block';

// document.getElementsByName('user1Username')[0].disabled = false;

button2.textContent = 'Hide it';

hideIt(user2BtnUnlock, user2Section, button2);

}

});

button3.addEventListener('click', () => {

if (user3BtnUnlock.checked) {

user3Section.style.display = 'block';

// document.getElementsByName('user1Username')[0].disabled = false;

button3.textContent = 'Hide it';

hideIt(user3BtnUnlock, user3Section, button3);

}

});

function showMore(radioBtn, userSection, btn) {

btn.addEventListener('click', () => {

if (radioBtn.checked) {

userSection.style.display = 'block';

// document.getElementsByName('user1Username')[0].disabled = false;

btn.textContent = 'Hide it';

hideIt(radioBtn, userSection, btn);

}

});

}

function hideIt(radioBtn, userSection, btn){

btn.addEventListener('click', ()=>{

if (radioBtn.checked) {

userSection.style.display = 'none';

// document.getElementsByName('user1Username')[0].disabled = false;

btn.textContent = 'Show More';

showMore(radioBtn, userSection, btn)

}

});

}

}

|  |
| --- |
| function solve() { |
|  | const profiles = Array.from(document.getElementsByClassName('profile')); |
|  | for (const profile of profiles) { |
|  | let button = profile.getElementsByTagName('button')[0]; |
|  | button.addEventListener('click', function() { |
|  | let profileUnlocked = profile.getElementsByTagName('input')[1]; |
|  | let hiddenDivId = profile.getElementsByTagName('div')[1].id; |
|  | let hiddenDiv = document.getElementById(`${hiddenDivId}`); |
|  | if(profileUnlocked.checked && button.textContent === 'Show more') { |
|  | hiddenDiv.style.display = 'block'; |
|  | button.textContent = 'Hide it'; |
|  | } else if(profileUnlocked.checked && button.textContent === 'Hide it') { |
|  | hiddenDiv.style.display = 'none'; |
|  | button.textContent = 'Show more'; |
|  | } |
|  | }) |
|  | } |
|  | } |

## 5.Number convertor

function solve() {

document.getElementsByTagName('button')[0].addEventListener('click', convert);

let inputNumberElement = document.getElementById('input');

let menuElement = document.getElementById('selectMenuTo');

let optionBinary = document.createElement('option');

optionBinary.textContent = 'Binary';

optionBinary.value = 'binary';

menuElement.appendChild(optionBinary);

let optionHexadecimal = document.createElement('option');

optionHexadecimal.textContent = 'Hexadecimal';

optionHexadecimal.value = 'hexadecimal';

menuElement.appendChild(optionHexadecimal);

function convert(){

let selectedOption = menuElement.options[menuElement.selectedIndex].value;

let numberSystem = {binary:2, decimal:10, hexadecimal:16};

let convertedNumber = (inputNumberElement.value >>> 0).toString(numberSystem[selectedOption]).toUpperCase();

document.getElementById('result').value = convertedNumber;

inputNumberElement.value = '';

}

}

|  |
| --- |
| function solve() { |
|  | let decimalNumber = document.getElementById('input'); |
|  | let menuElement = document.getElementById('selectMenuTo'); |
|  | let binaryOption = document.createElement('option'); |
|  | binaryOption.textContent = 'Binary'; |
|  | binaryOption.value = 'binary'; |
|  |  |
|  | let hexadecimalOption = document.createElement('option'); |
|  | hexadecimalOption.textContent = 'Hexadecimal'; |
|  | hexadecimalOption.value = 'hexadecimal'; |
|  |  |
|  | menuElement.appendChild(binaryOption); |
|  | menuElement.appendChild(hexadecimalOption); |
|  |  |
|  | const convertValues = { |
|  | binary: 2, |
|  | decimal: 10, |
|  | hexadecimal: 16, |
|  | } |
|  |  |
|  | const convertButton = document.querySelector('button'); |
|  |  |
|  | convertButton.addEventListener('click', function(e) { |
|  | if(decimalNumber.value !== '') { |
|  | const selectedOption = menuElement.options[menuElement.selectedIndex].value; |
|  | const convertedNumber = (decimalNumber.value >>> 0).toString(convertValues[selectedOption]).toUpperCase(); |
|  |  |
|  | decimalNumber.value = ''; |
|  | document.getElementById('result').value = convertedNumber; |
|  | } |
|  | }) |
|  | } |

## 6.Softuni Quiz

function solve() {

    let exerciseElement = document.getElementById('exercise');

    let buttons = Array.from(document.getElementsByTagName('button'));

    let resultElement = document.getElementById('result');

    let rightAnswers = ['2013', 'Pesho', 'Nakov'];

    let rightAnswersCount = 0;

    let hiddenSectionsCount = 0;

    buttons.forEach(button => {

        button.addEventListener('click', event => {

            let parent = button.parentNode;

            let inputElements = Array.from(parent.getElementsByTagName('input'));

            inputElements.forEach(input => {

                if(input.checked === true){

                    hiddenSectionsCount++;

                    let answer = input.value;

                    if (rightAnswers.includes(answer)){

                        rightAnswersCount++;

                    }

                    if (button.textContent === 'Get the results'){

                        if (rightAnswersCount === 3){

                            resultElement.textContent = `You are recognized as top SoftUni fan!`;

                        } else {

                            resultElement.textContent = `You have ${rightAnswersCount} right answers`;

                        }

                    } else {

                        exerciseElement.children[hiddenSectionsCount].style.display = 'block';

                    }

                }

            });

        });

    });

}

function solve() {

let exerciseElement = document.getElementById('exercise');

let buttons = Array.from(document.getElementsByTagName('button'));

let resultElement = document.getElementById('result');

let allRightAnswers = ['2013', 'Pesho', 'Nakov'];

let hiddenSectionCounter = 0;

let rightAnswers = 0;

buttons.forEach(currentButton => {

currentButton.addEventListener('click', event => {

let parent = currentButton.parentNode;

let inputElements = Array.from(parent.getElementsByTagName('input'));

inputElements.forEach(currentInput => {

if (currentInput.checked === true) {

hiddenSectionCounter++;

let answer = currentInput.value;

if (allRightAnswers.includes(answer)) {

rightAnswers++;

}

// all question displayed

if (currentButton.textContent === 'Get the results') {

if (rightAnswers === 3) {

resultElement.textContent = `You are recognized as top SoftUni fan!`;

} else {

resultElement.textContent = `You have ${rightAnswers} right answers`;

}

} else {

exerciseElement.children[hiddenSectionCounter].style.display = 'block';

}

}

});

});

});

}

function solve() {

    let exerciseElement = document.getElementById('exercise');

    let buttons = Array.from(document.getElementsByTagName('button'));

    let resultElement = document.getElementById('result');

    let allRightAnswers = ['2013', 'Pesho', 'Nakov'];

    let hiddenSectionCounter = 0;

    let rightAnswers = 0;

    buttons.forEach(currentButton => {

        currentButton.addEventListener('click', event => {

            let parent = currentButton.parentNode;

            let inputElements = Array.from(parent.getElementsByTagName('input'));

            inputElements.forEach(currentInput => {

                if (currentInput.checked === true) {

                    hiddenSectionCounter++;

                  let answer = currentInput.value;

                    if (allRightAnswers.includes(answer)) {

                        rightAnswers++;

                    }

                    // all question displayed

                    if (currentButton.textContent === 'Get the results') {

                        if (rightAnswers === 3) {

                            resultElement.textContent = `You are recognized as top SoftUni fan!`;

                        } else {

                            resultElement.textContent = `You have ${rightAnswers} right answers`;

                        }

                    } else {

                        exerciseElement.children[hiddenSectionCounter].style.display = 'block';

                    }

                }

            });

        });

    });

}

|  |
| --- |
| function solve() { |
|  | const sections = Array.from(document.querySelectorAll('section')); |
|  | let result = 0; |
|  |  |
|  | for (let i = 0; i < sections.length; i++) { |
|  | const section = sections[i]; |
|  | const button = section.querySelector('button'); |
|  | button.addEventListener('click', function() { |
|  | const answers = Array.from(section.querySelectorAll('input')); |
|  | for (const answer of answers) { |
|  | if(answer.checked && i < sections.length - 1) { |
|  | sections[i + 1].removeAttribute('class'); |
|  | } |
|  | if(answer.checked && (answer.value === "2013" || |
|  | answer.value === "Pesho" || |
|  | answer.value === "Nakov")) { |
|  | result++; |
|  | } |
|  | } |
|  | }) |
|  |  |
|  | if(i === sections.length - 1) { |
|  | button.addEventListener('click', function() { |
|  | let resultDivElement = document.getElementById('result'); |
|  | if(result < 3) { |
|  | resultDivElement.textContent = `You have ${result} right answers`; |
|  | } else if(result === 3) { |
|  | resultDivElement.textContent = "You are recognized as top SoftUni fan!"; |
|  | } |
|  | }) |
|  | } |
|  | } |
|  | } |

## 7.Table – Search Engine

function solve() {

let studentsData = Array.from(document.querySelectorAll('table>tbody>tr'));

let searchButton = document.getElementById('searchBtn');

let searchField = document.getElementById('searchField');

searchButton.addEventListener('click', function() {

let searchedWord = searchField.value.toLowerCase();

studentsData.forEach(studentDataRow => {

studentDataRow.classList.remove('select');

let studentDataCells = Array.from(studentDataRow.cells);

studentDataCells.forEach(cell => {

if (cell.textContent.toLowerCase().includes(searchedWord)){

studentDataRow.classList.add('select');

}

})

searchField.value = '';

})

});

}

|  |
| --- |
| function solve() { |
|  | const studentsData = Array.from(document.querySelectorAll('table>tbody>tr')); |
|  |  |
|  | let searchFieldElement = document.getElementById('searchField'); |
|  | let searchButton = document.getElementById('searchBtn'); |
|  | searchButton.addEventListener('click', function (event) { |
|  | let formattedSearch = searchFieldElement.value.toLowerCase(); |
|  | studentsData.forEach(student => { |
|  | student.classList.remove('select'); |
|  | Array.from(student.cells).forEach(cell => { |
|  | if(cell.textContent.toLowerCase().includes(formattedSearch)) { |
|  | student.classList.add('select'); |
|  | } |
|  | }) |
|  | searchFieldElement.value = ''; |
|  | }) |
|  | }); |
|  | } |

## 8.Toto numbers

function solve() {

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

    button.addEventListener('click', addTotoNumbers);

    function addTotoNumbers () {

        let inputField = document.querySelector('input');

        let inputArray = Array.from(inputField.value.split(' '));

        let numbers = inputArray.map(Number);

        if (numbers.length !== 6){

            return;

        }

        for (let number of numbers){

            if (number < 1 || number > 49){

                return;

            }

        }

        //numbers.sort((a,b) => a - b);

        let allNumbersDiv = document.getElementById('allNumbers');

        for (let i = 1; i <= 49; i++) {

            let numberDiv = document.createElement('div');

            numberDiv.textContent = i;

            numberDiv.classList.add('numbers');

            if (numbers.includes(i)){

                numberDiv.style.backgroundColor = 'orange';

            }

            allNumbersDiv.appendChild(numberDiv);

        }

        button.setAttribute('disabled', true);

        inputField.setAttribute('disabled', true);

    }

}

|  |
| --- |
| function solve() { |
|  | let button = document.querySelector('button'); |
|  | button.addEventListener('click', addTotoNumbers); |
|  |  |
|  | function addTotoNumbers() { |
|  | let inputFieldElement = document.querySelector('input'); |
|  | let numbers = Array.from(new Set(inputFieldElement.value.split(' ').map(Number))); |
|  |  |
|  | let minNumber = Math.min(...numbers); |
|  | let maxNumber = Math.max(...numbers); |
|  |  |
|  | if(numbers.length === 6 && minNumber >= 1 && maxNumber <= 49) { |
|  | let allNumbersDivElement = document.getElementById('allNumbers'); |
|  |  |
|  | for (let i = 1; i <= 49; i++) { |
|  | let number = document.createElement('div'); |
|  | number.textContent = i; |
|  | number.classList.add('numbers'); |
|  | if(numbers.includes(i)) { |
|  | number.style.backgroundColor = 'orange'; |
|  | } |
|  | allNumbersDivElement.appendChild(number); |
|  | } |
|  | button.setAttribute('disabled', true); |
|  | inputFieldElement.setAttribute('disabled', true); |
|  | } |
|  | } |
|  | } |

## 9.Virus Installer

function solve() {

let buttons = document.querySelectorAll('button');

let nextButton = buttons[0];

let cancelButton = buttons[1];

let contentElement = document.getElementById('content');

let firstStepDiv = document.getElementById('firstStep');

let secondStepDiv = document.getElementById('secondStep');

let thirdStepDiv = document.getElementById('thirdStep');

cancelButton.addEventListener('click', hideExerciseContainer);

nextButton.addEventListener('click', showFirstStep);

function hideExerciseContainer(){

//let exerciseContainer = document.getElementById('exercise'); - one test in Judge didn't get with this

let exerciseContainer = document.querySelector('#exercise>section');

//exerciseContainer.style.visibility = 'hidden';

//exerciseContainer.style.visibility = 'none';

exerciseContainer.style.display = 'none';

}

function showFirstStep(){

contentElement.style.backgroundImage = 'none';

//contentElement.style.display = 'none';

firstStepDiv.style.display = 'block';

nextButton.removeEventListener('click', showFirstStep);

nextButton.addEventListener('click', showSecondStep);

}

function showSecondStep(){

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

let agreeInput = inputs[0];

let disagreeInput = inputs[1];

if (disagreeInput.checked){

} else if (agreeInput.checked){

firstStepDiv.style.display = 'none';

secondStepDiv.style.display = 'block';

nextButton.removeEventListener('click', showSecondStep);

nextButton.addEventListener('click', showThirdStep);

nextButton.style.display = 'none';

setTimeout(() => {

nextButton.style.display = 'inline';

}, 3000);

}

}

function showThirdStep(){

secondStepDiv.style.display = 'none';

thirdStepDiv.style.display = 'block';

nextButton.style.display = 'none';

cancelButton.value = 'Finish';

//cancelButton.textContent = 'Finish';

//cancelButton.addEventListener('click', hideExerciseContainer);

}

}

25/100 in Judge

function solve() {

let nextBtn = document.getElementById("buttons").children[0];

let cancelBtn = document.getElementById("buttons").children[1];

let agreeCheck = document.getElementsByName('license');

let firstPage = true;

let secondPage = false;

let thirdPage = false;

nextBtn.addEventListener('click', event);

cancelBtn.addEventListener('click', cancelEvent);

function event() {

if (firstPage) {

firstEvent();

} else if (agreeCheck[0].checked && secondPage) {

secondEvent();

} else if (thirdPage) {

thirdEvent();

}

}

function firstEvent() {

document.getElementById("content").style.backgroundImage = 'none';

document.getElementById("firstStep").style.display = 'inline';

firstPage = false;

secondPage = true;

}

function secondEvent() {

document.getElementById("firstStep").style.display = 'none';

document.getElementById("secondStep").style.display = 'inline';

document.getElementById("buttons").children[0].style.display = "none";

setTimeout(function() {

secondPage = false;

thirdPage = true;

document.getElementById("buttons").children[0].style.display = "inline";

}, 3000);

}

function thirdEvent() {

document.getElementById("secondStep").style.display = 'none';

document.getElementById("thirdStep").style.display = 'inline';

document.getElementById("buttons").children[0].style.display = "none";

document.getElementById("buttons").children[1].textContent = 'Finish';

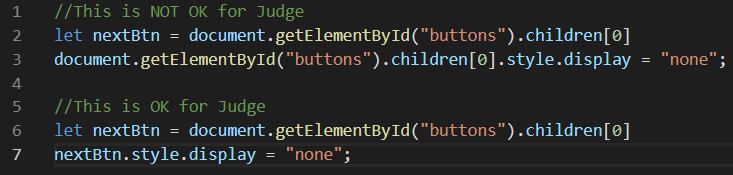
}

function cancelEvent() {

document.getElementById("exercise").innerHTML = '';

}

}



С не минаващ 4-ти тест:

function solve() {

let clicks = 0;

let cancelButton = document.getElementById("buttons").children[1];

cancelButton.addEventListener("click", () => {

document.querySelector("#exercise").style.display = "none";

document.querySelector("#softUni-person-img").style.float = "right";

});

let nextButton = document.getElementById("buttons").children[0];

nextButton.addEventListener("click", installingVirus);

function installingVirus(){

if(clicks == 0) {

document.querySelector("#content").style.backgroundImage = "none";

document.querySelector("#firstStep").style.display = "block";

clicks++;

} else if(clicks == 1 && radioButtons[0].checked) {

document.querySelector("#firstStep").style.display = "none";

document.querySelector("#secondStep").style.display = "block";

nextButton.style.display="none";

setTimeout(() => {

nextButton.style.display = "inline-block";

}, 3000);

clicks++;

} else if (clicks == 2) {

document.querySelector("#secondStep").style.display = "none";

document.querySelector("#thirdStep").style.display = "block";

nextButton.style.display = "none";

cancelButton.textContent = "Finish";

}

}

let radioButtons = document.querySelectorAll("#firstStep input");

radioButtons[0].addEventListener("click", () => {

radioButtons[0].checked = true;

radioButtons[1].checked = false;

});

radioButtons[1].addEventListener("click", () => {

radioButtons[0].checked = false;

radioButtons[1].checked = true;

});

}

Последния тест е проверка дали чистиш див-а при натискане на бутони cancel/finish

 Ред 8 ти е излишен, а в ред 7 му добави children[0] защото ти трябва див-а в този елемент който е първи

Гледа се дали имате изискваната функционалност за бутоните Cancel/Finish. Аз скривах елемента с (Element).style.visibility = "hidden"; (това ми излезе първо в Гугъл), което върши работа, но не и в Джъдж. Трябва да се използва (Element).style.display = 'none';

|  |
| --- |
| //75/100  function solve() { |
|  | const buttons = document.querySelectorAll('button'); |
|  | let nextButton = buttons[0]; |
|  | let cancelButton = buttons[1]; |
|  | let content = document.getElementById('content'); |
|  | let firstStepDivElement = document.getElementById('firstStep'); |
|  | let secondStepDivElement = document.getElementById('secondStep'); |
|  | let thirdStepDivElement = document.getElementById('thirdStep'); |
|  |  |
|  | cancelButton.addEventListener('click', hideContainer); |
|  | nextButton.addEventListener('click', showFirstStep); |
|  |  |
|  | function hideContainer() { |
|  | let container = document.getElementById('exercise'); |
|  | container.style.visibility = 'hidden'; |
|  | } |
|  |  |
|  | function showFirstStep() { |
|  | content.style.backgroundImage = 'none'; |
|  | firstStepDivElement.style.display = 'block'; |
|  | nextButton.removeEventListener('click', showFirstStep); |
|  | nextButton.addEventListener('click', showSecondStep); |
|  | } |
|  |  |
|  | function showSecondStep() { |
|  | const inputs = document.querySelectorAll('input'); |
|  | let agreeInput = inputs[0]; |
|  | let disagreeInput = inputs[1]; |
|  |  |
|  | if(disagreeInput.checked) { |
|  |  |
|  | } else if(agreeInput.checked) { |
|  | firstStepDivElement.style.display = 'none'; |
|  | secondStepDivElement.style.display = 'block'; |
|  |  |
|  | nextButton.removeEventListener('click', showSecondStep); |
|  | nextButton.addEventListener('click', showThirdStep); |
|  | //Hide Next button |
|  | nextButton.style.display = 'none'; |
|  | setTimeout(() => { |
|  | nextButton.style.display = 'inline'; |
|  | }, 3000); |
|  | } |
|  | } |
|  |  |
|  | function showThirdStep() { |
|  | secondStepDivElement.style.display = 'none'; |
|  | thirdStepDivElement.style.display = 'block'; |
|  | nextButton.style.display = 'none'; |
|  | cancelButton.textContent = "Finish"; |
|  | } |
|  | } |
| function solve() { |
|  | let button = document.querySelector('button'); |
|  | button.addEventListener('click', addTotoNumbers); |
|  |  |
|  | function addTotoNumbers() { |
|  | let inputFieldElement = document.querySelector('input'); |
|  | let numbers = Array.from(new Set(inputFieldElement.value.split(' ').map(Number))); |
|  |  |
|  | let minNumber = Math.min(...numbers); |
|  | let maxNumber = Math.max(...numbers); |
|  |  |
|  | if(numbers.length === 6 && minNumber >= 1 && maxNumber <= 49) { |
|  | let allNumbersDivElement = document.getElementById('allNumbers'); |
|  |  |
|  | for (let i = 1; i <= 49; i++) { |
|  | let number = document.createElement('div'); |
|  | number.textContent = i; |
|  | number.classList.add('numbers'); |
|  | if(numbers.includes(i)) { |
|  | number.style.backgroundColor = 'orange'; |
|  | } |
|  | allNumbersDivElement.appendChild(number); |
|  | } |
|  | button.setAttribute('disabled', true); |
|  | inputFieldElement.setAttribute('disabled', true); |
|  | } |
|  | } |
|  | } |

|  |
| --- |
| 100/100  function solve() { |
|  | const buttons = document.querySelectorAll('button'); |
|  | let nextButton = buttons[0]; |
|  | let cancelButton = buttons[1]; |
|  | let content = document.getElementById('content'); |
|  | let firstStepDivElement = document.getElementById('firstStep'); |
|  | let secondStepDivElement = document.getElementById('secondStep'); |
|  | let thirdStepDivElement = document.getElementById('thirdStep'); |
|  |  |
|  | cancelButton.addEventListener('click', hideContainer); |
|  | nextButton.addEventListener('click', showFirstStep); |
|  |  |
|  | function hideContainer() { |
|  | let container = document.querySelector('#exercise > section'); |
|  | container.style.display = 'none'; |
|  | } |
|  |  |
|  | function showFirstStep() { |
|  | content.style.backgroundImage = 'none'; |
|  | firstStepDivElement.style.display = 'block'; |
|  | nextButton.removeEventListener('click', showFirstStep); |
|  | nextButton.addEventListener('click', showSecondStep); |
|  | } |
|  |  |
|  | function showSecondStep() { |
|  | const inputs = document.querySelectorAll('input'); |
|  | let agreeInput = inputs[0]; |
|  | let disagreeInput = inputs[1]; |
|  |  |
|  | if(disagreeInput.checked) { |
|  |  |
|  | } else if(agreeInput.checked) { |
|  | firstStepDivElement.style.display = 'none'; |
|  | secondStepDivElement.style.display = 'block'; |
|  |  |
|  | nextButton.removeEventListener('click', showSecondStep); |
|  | nextButton.addEventListener('click', showThirdStep); |
|  | //Hide Next button |
|  | nextButton.style.display = 'none'; |
|  | setTimeout(() => { |
|  | nextButton.style.display = 'inline'; |
|  | }, 3000); |
|  | } |
|  | } |
|  |  |
|  | function showThirdStep() { |
|  | secondStepDivElement.style.display = 'none'; |
|  | thirdStepDivElement.style.display = 'block'; |
|  | nextButton.style.display = 'none'; |
|  | cancelButton.textContent = "Finish"; |
|  | } |
|  | } |

## 10.No Signal

function solve() {

    let exerciseDiv = document.getElementById('exercise');

    let noSignalDiv = exerciseDiv.getElementsByTagName('div')[0];

    // setTimeout(() => {//it change its position only once

    //  let horizontalRandomNumber = Math.floor(Math.random() \* 81) + 1;

    //  let verticalRandomNumber = Math.floor(Math.random() \* 45) + 1;

    //  noSignalDiv.style.position = `absolute`;

    //  noSignalDiv.style.top = `${horizontalRandomNumber}%`;

    //  noSignalDiv.style.left = `${verticalRandomNumber}vh`;

    // }, 2000);

    setInterval(function () {

        let horizontalRandomNumber = Math.floor(Math.random() \* 81) + 1;

        let verticalRandomNumber = Math.floor(Math.random() \* 45) + 1;

        noSignalDiv.style.position = `absolute`;

        noSignalDiv.style.top = `${horizontalRandomNumber}%`;

        noSignalDiv.style.left = `${verticalRandomNumber}vh`;

    }, 2000);

}