Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

**Лабораторна робота №6**

з дисципліни "Основи клієнтської розробки"

Тема: "JavaScript. Внутрішні, зовнішні скрипти. Змінні. Умови. Цикли. Функції.

DOM. BOM. Браузер: документ(document)"

Тематика:«Рецепти»

Виконав: Перевірила:

студент групи ІП-93 ст. вик. Кафедри ІСТ

Домінський Валентин Хмелюк Марина Сергіївна

Олексійович

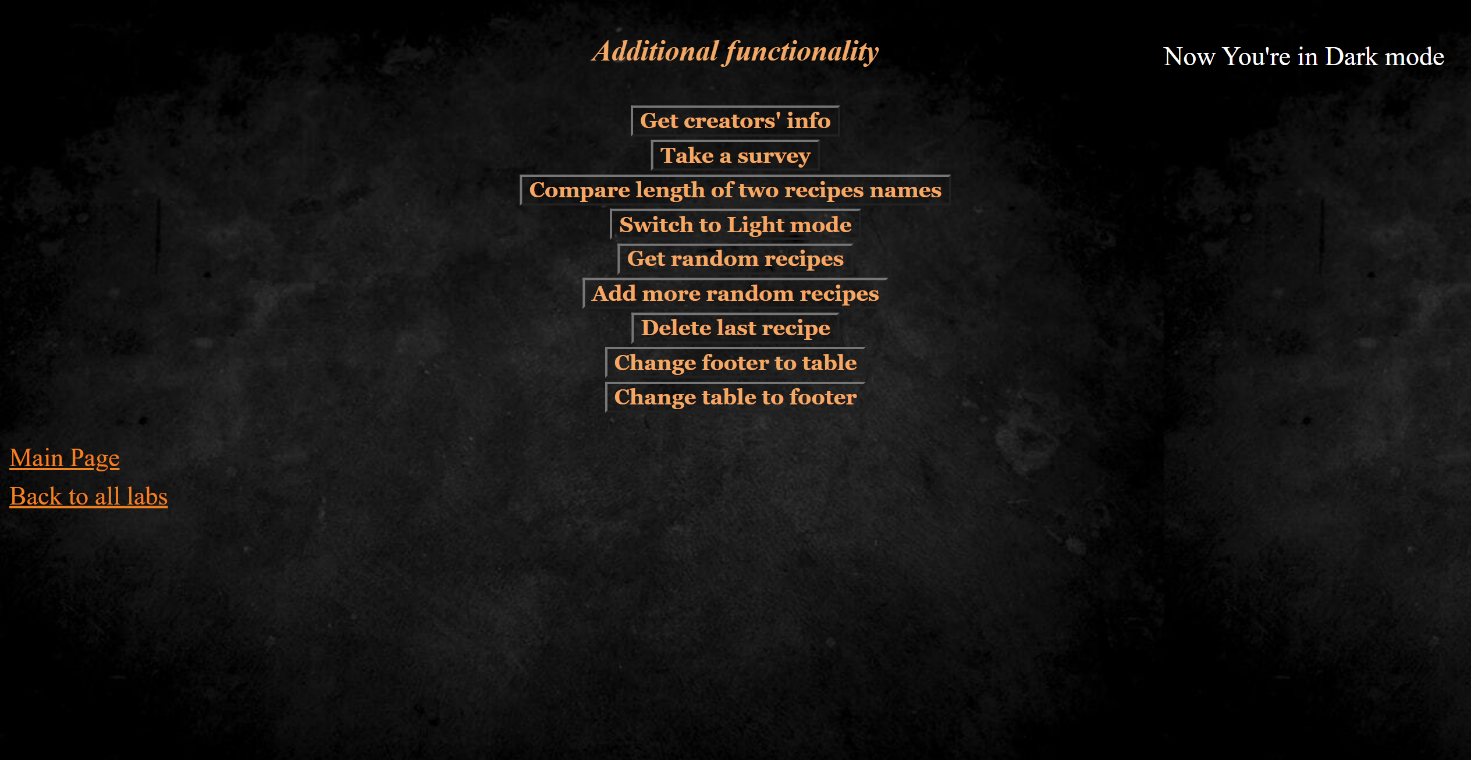
Київ 2021

Завдання:

1. Застосувати функції alert, prompt, confirm. Оформити внутрішніми скриптами. Написати та викликати власну функцію «Діалог з користувачем», застосувати змінні, умовне розгалуження, цикли. Функцію помістити в зовнішній файл-скрипт. Підключити до html-документу. Написати та викликати власну функцію виводу інформації про розробника сторінки з параметрами (прізвище, імя, посада). Параметру «посада» задати значення за замовчуванням. Написати функцію порівняння двох рядків, більший вивести на екран, використовуючи alert.
2. За допомогою об’єкта document змінити фон сторінки на 30 секунд. За допомогою об’єкта location перенаправити браузер на іншу сторінку. Викоритовуючи метод getElementById Використати метод querySelectorAll Використати наступні властивості DOM-вузла: innerHTML, outerHTML, nodeValue / data, textContent Внести зміни в документи/сторінку, використовуючи document.write, document.createElement(tag), document.createTextNode(text) та методи вставки node.append (... nodes or strings), node.prepend (... nodes or strings), node.after (... nodes or strings), node.replaceWith (... nodes or strings), метод видалення вузлів node.remove ().

Результати:

Додаткова сторінка, де використані всі методи, які є в завданні:



Вихідний код:

index.html:

<!DOCTYPE html>

<html lang="en">

<head>

<meta name="author" content="Valentyn Dominskyi">

<meta name="keywords" content="recipes, cookbook">

<meta name="description" content="cookbook site with recipes">

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

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

<title>Additional page</title>

</head>

<body>

<header>

<h3 id="AdditionalFunc">Additional functionality</h3>

<script>

document.write("<p id='ThemeMode'>Now You're in Dark mode</p>")

</script>

</header>

<div class="AdditionalPage">

<div class="AdditionalPageButtons">

<button type="button" onclick="showInfo(SURNAME, NAME)">Get creators' info</button>

<br>

<button type="button" onclick="startDialog()">Take a survey</button>

<br>

<button type="button" onclick="compareTwoTexts()">Compare length of two recipes names</button>

<br>

<button type="button" id="ThemeSwitcher" onclick="changeTheme()">Switch to Light mode</button>

<br>

<button type="button" onclick="insertRandomRecipes()">Get random recipes</button>

<br>

<button type="button" onclick="addRecipePlaceholder()">Add more random recipes</button>

<br>

<button type="button" onclick="deleteRecipePlaceholder()">Delete last recipe</button>

<br>

<button type="button" onclick="changeFooterToTable()">Change footer to table</button>

<br>

<button type="button" onclick="changeTableToFooter()">Change table to footer</button>

</div>

<p class="RandomRecipe"></p>

<p class="RandomRecipe"></p>

<p class="RandomRecipe"></p>

</div>

<footer>

<a href="index.html" target="\_blank">Main Page</a>

<br>

<a href="../index.html">Back to all labs</a>

</footer>

<script>

let pageName = document.getElementById("AdditionalFunc").firstChild;

setTimeout(() => alert(`Now You are on page with ${pageName.data}`), 1000);

</script>

<script>

function compareTwoTexts() {

let texts;

texts = getTwoTexts();

if (!texts[0] || !texts[1]) {

return;

}

if (isTextsChosen()) {

let textToShow = compareLengthes(texts[0], texts[1]);

showText(textToShow);

return;

}

return;

}

function getTwoTexts() {

let firstText = prompt("Write first recipes' name", "");

if (!firstText) {

return;

}

let secondText = prompt("And for the second one", "");

if (!secondText) {

return;

}

return [firstText, secondText];

}

function isTextsChosen() {

let isChosen = confirm("Are You sure, that You want to compare these strings?");

return isChosen;

}

function compareLengthes(firstText, secondtext) {

let text;

if (firstText.length > secondtext.length) {

text = firstText;

} else {

text = secondtext;

}

return text;

}

function showText(text) {

alert(text);

}

</script>

<script src="scripts/GetInfo.js"></script>

<script src="scripts/Dialog.js"></script>

<script src="scripts/ChangeTheme.js"></script>

<script src="scripts/GetRandomRecipe.js"></script>

<script src="scripts/ChangeFooter.js"></script>

<script src="scripts/ChangeRecipePlaceholder.js"></script>

</body>

</html>

Style.css:

header {

text-align: center

}

body {

font-size: 23px;

color: #ffffff;

font-style: normal;

line-height: 1.3;

text-align: left;

background-image: url("images/Background.jpg");

}

h3 {

font-size: 25px;

color: #f3a866;

font-style: italic;

line-height: 1.5;

text-align: center;

}

p#ParagraphWithScale {

text-align: center;

block-size: 250px;

}

p#Frame {

text-align: right !important;

margin-top: -506.5px !important;

}

#ImageWithScale {

transform: scale(0.5) translate(0%, -50%) !important;

}

p.SidebarHeader {

font-size: 5;

color: orange;

font-family: "Lucida Console", sans-serif;

text-align: center;

}

ul,

ol,

dl {

font-size: 20px;

color: #ffffff;

font-family: Georgia, serif;

margin: auto;

padding-left: 10%;

}

ul,

ol {

padding-left: 15%;

}

ul.CircleType {

list-style-type: circle;

}

ul.DiscType {

list-style-type: disc;

}

ul.SquareType {

list-style-type: square;

}

ol.UpperRomanType {

list-style-type: upper-roman;

}

ol.LowerRomanType {

list-style-type: lower-roman;

}

ol.AType {

list-style-type: upper-alpha;

}

ol.DecimalType {

list-style-type: decimal;

}

table {

font-size: x-large;

margin-left: auto;

margin-right: auto;

text-align: center;

padding: auto;

border: 5px solid #56018F;

vertical-align: middle;

color: rgb(255, 249, 240);

}

th {

background-color: purple;

}

tr {

vertical-align: middle;

text-align: center;

}

td {

border: 1.5px solid #6701ac;

}

a {

font-size: 22px;

font-style: normal;

line-height: 1.5;

text-align: center;

}

a:link {

color: #2E82B3;

}

a:visited {

color: #F08221;

}

a:active {

color: #C728F7;

}

div.RecipeDescription {

padding-left: 10%;

padding-right: 10%;

}

div.RecipeDescription>p:first-child:first-letter {

font-size: 200%;

border: white solid 1px;

padding: 4px;

margin: 4px;

}

div.Sidebar {

float: right;

width: 25%;

height: auto;

border: 2px solid white;

padding: 1%;

margin: auto 2%;

margin-bottom: 3%;

}

div.AdditionalPageButtons {

text-align: center;

}

div.AdditionalPageButtons button {

font-family: Georgia, serif;

background: transparent;

overflow: hidden;

font-weight: 800;

font-size: 1.15rem;

color: #f3a866;

}

p#ThemeMode {

position: absolute;

top: 10px;

left: 1000px;

}

ChangeFooter.js:

'use strict'

const FOOTER\_SELECTOR = "footer";

const FOOTER\_STRING = '<footer><a href="index.html" target="\_blank">Main Page</a><br><a href="../index.html">Back to all labs</a></footer>';

const TABLE\_SELECTOR = "table";

const TABLE\_STRING = '<table><tr><td><a href="index.html" target="\_blank">Main Page</a></td></tr><tr><td><a href="../index.html">Back to all labs</a></td></tr></table>';

function changeFooterToTable() {

let footer = document.querySelector(FOOTER\_SELECTOR);

footer.outerHTML = TABLE\_STRING;

}

function changeTableToFooter() {

let table = document.querySelector(TABLE\_SELECTOR);

let footer = document.createElement(FOOTER\_SELECTOR);

footer.innerHTML = FOOTER\_STRING;

table.replaceWith(footer);

}

ChangeRecipePlaceholder.js:

'use strict'

const PARAGRAPH\_TAG = "p";

const PARAGRAPH\_CLASS = "RandomRecipe";

const DIV\_SELECTOR = "div.AdditionalPage";

const LAST\_CHILD\_SELECTOR = ":last-child";

function addRecipePlaceholder() {

let paragraph = getParagraph();

let newParagraph = createNewParagraph();

addParagraph(paragraph, newParagraph);

}

function getParagraph() {

let paragraph = document.querySelector(RECIPE\_SELECTOR + LAST\_CHILD\_SELECTOR);

if (!paragraph) {

let startDiv = findStartDiv();

addParagraphToDiv(startDiv);

}

return paragraph;

}

function findStartDiv() {

let div = document.querySelector(DIV\_SELECTOR);

return div;

}

function addParagraphToDiv(div) {

let newParagraph = createNewParagraph();

div.append(newParagraph);

}

function createNewParagraph() {

let newParagraph = document.createElement(PARAGRAPH\_TAG);

newParagraph.className = PARAGRAPH\_CLASS;

return newParagraph;

}

function addParagraph(paragraph, newParagraph) {

paragraph.after(newParagraph);

}

function deleteRecipePlaceholder() {

let paragraph = getParagraph();

paragraph.remove();

}

ChangeTheme.js:

'use strict'

const DEFAULT\_BACKGROUND = "images/Background.jpg";

const NEW\_BACKGROUND = "images/BackgroundLight.jpg";

const ELEMENT = "ThemeMode";

const PREPEND\_TEXT = "Now You're in ";

const DARK\_TEXT = "Dark mode";

const LIGHT\_TEXT = "Light mode";

const DEFAULT\_ELEMENT\_TEXT = "Now You're in Dark mode";

const NEW\_ELEMENT\_TEXT = "Now You're in Light mode";

const BUTTON = "ThemeSwitcher";

const DEFAULT\_BUTTON\_TEXT = "Switch to Light mode";

const NEW\_BUTTON\_TEXT = "Wait 30 seconds";

const TIME = 30 \* 1000;

function changeTheme() {

changeBackground();

let themeModeText = changeThemeModeText();

let themeSwitcher = changeThemeSwitcher();

changeDefaultThemeAfterTime(themeModeText, themeSwitcher);

}

function changeBackground() {

document.body.style.backgroundImage = `url(${NEW\_BACKGROUND})`;

}

function changeThemeModeText() {

let themeModeText = document.getElementById(ELEMENT);

themeModeText.textContent = LIGHT\_TEXT;

themeModeText.prepend(PREPEND\_TEXT);

return themeModeText;

}

function changeThemeSwitcher() {

let themeSwitcher = document.getElementById(BUTTON);

themeSwitcher.textContent = NEW\_BUTTON\_TEXT;

return themeSwitcher;

}

function changeDefaultThemeAfterTime(themeModeText, themeSwitcher) {

setTimeout(() => (document.body.style.backgroundImage = `url(${DEFAULT\_BACKGROUND})`, themeModeText.textContent = DEFAULT\_ELEMENT\_TEXT, themeSwitcher.textContent = DARK\_TEXT, themeSwitcher.prepend(PREPEND\_TEXT)), TIME);

}

Dialog.js:

'use strict'

var questions = {

0: "How old are You?",

1: "Why do You attend Our site?",

2: "What do You like the most about this site?",

3: "How did you find out about Our site?",

};

const START\_SURVEY\_MESSAGE = "This button will start a survey. Do You want to start?";

const END\_SURVEY\_MESSAGE = "Thank You for Your answers!";

const CANCEL\_SURVEY\_MESSAGE = "No problem. Do what You want";

function startDialog() {

let confirmation = getConfirmation();

if (confirmation) {

startSurvey();

return;

}

cancelSurvey();

}

function getConfirmation() {

let confirmation = confirm(START\_SURVEY\_MESSAGE);

return confirmation;

}

function startSurvey() {

let answers = {};

for (let i = 0; i < 4; i++) {

answers[i] = prompt(questions[i]);

}

alert(END\_SURVEY\_MESSAGE);

}

function cancelSurvey() {

alert(CANCEL\_SURVEY\_MESSAGE);

}

GetInfo.js:

'use strict'

const SURNAME = "Dominskyi";

const NAME = "Valentyn";

const DEFAULT\_POSITION = "Student";

const GITHUB\_INVITATION = "Wanna see github profile?";

const GITHUB\_ACCOUNT = "https://github.com/VsIG-official";

function showInfo(surname, name, position = DEFAULT\_POSITION) {

alert(`Surname - ${surname}, name - ${name} and position - ${position}`);

goToGithub();

}

function goToGithub() {

let confirmation = confirm(GITHUB\_INVITATION);

if (confirmation) {

location.href = GITHUB\_ACCOUNT;

}

}

GetRandomRecipe.js:

'use strict'

var links = {

0: "Belgian-Waffles.html",

1: "Cracked-Cookies.html",

2: "Envelopes-With-Apples.html",

3: "Waffle-Tubes.html",

}

const RECIPE\_SELECTOR = "div > p.RandomRecipe";

function insertRandomRecipes() {

let paragraphs = getAllParagraphs();

let paragraphsNum = 0;

paragraphs.forEach(paragraph => {

paragraphsNum++;

changeHTML(paragraph, paragraphsNum);

});

}

function getAllParagraphs() {

let paragraphs = document.querySelectorAll(RECIPE\_SELECTOR);

return paragraphs;

}

function getRandomRecipe() {

let randomNumber = Math.round(Math.random() \* 3);

let randomRecipe = links[randomNumber];

return randomRecipe;

}

function changeHTML(paragraph, paragraphsNum) {

let randomRecipe = getRandomRecipe();

let number = document.createTextNode(paragraphsNum);

paragraph.innerHTML = `<a href="${randomRecipe}" target="\_blank">Random Recipe №${number.textContent}</a>`;

}

Висновки:

Я познайомився з javascript, внутрішніми та зовнішніми скриптами. Застосував змінні, умови, цикли. Написав власні функції. Дійзнався більше про DOM\BOM

Посилання:

[Github](https://github.com/VsIG-official/Basics-Of-Client-Development)

[Github Pages](https://vsig-official.github.io/Basics-Of-Client-Development/)