Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №3

По дисциплине: «Веб-технологии»

Выполнил:

Студент 3-го курса

Группы АС-63

Кухарчук И. Н.

Проверил:

Михняев А. Л.

Брест 2025

**Цель работы:** приобрести навыки работы с JavaScript

**Задание 1**

Задан бесконечный цикл. С помощью двух отдельных вызовов функции prompt() вводит два числа. Числа сравниваются между собой, после чего пользователю отображается одна из трёх фраз: «числа равны», «первое число меньше», «второе число меньше». При введении пользователем не числовых значений, в зависимости от введённых данных пользователь может увидеть фразы «первый или второй ввод – не число». Выполнение скрипта прекращается при вводе пользователем в первом или втором вводе значения q без учёта регистра.

<div>

<input type="text" id="num1" placeholder="Enter first number">

<input type="text" id="num2" placeholder="Enter second number">

<button onclick="compareNumbers()">Compare</button>

<p id="result"></p>

</div>

<script>

function compareNumbers() {

const num1 = document.getElementById('num1').value;

const num2 = document.getElementById('num2').value;

const result = document.getElementById('result');

if (num1.toLowerCase() === 'q' || num2.toLowerCase() === 'q') {

result.innerText = "Script terminated.";

return;

}

const number1 = parseFloat(num1);

const number2 = parseFloat(num2);

if (isNaN(number1) && isNaN(number2)) {

result.innerText = "Both inputs are not numbers.";

} else if (isNaN(number1)) {

result.innerText = "First input is not a number.";

} else if (isNaN(number2)) {

result.innerText = "Second input is not a number.";

} else if (number1 === number2) {

result.innerText = "Numbers are equal.";

} else if (number1 < number2) {

result.innerText = "First number is less.";

} else {

result.innerText = "Second number is less.";

}

}

</script>

**Задание 2**

Дан многоквартирный жилой дом. У ждома присутствуют три характеристики: количество этажей (1-20), число подъездов (1-15), количество квартир на лестничной площадке (1-5). Пользовательский скрипт запрашивает харакеристики, а также квартиры. Необходимо рассчитать и вывести на экран номер подъезда, в котором находится запрашиваемая квартира. Если пользователь вводит некорректиные значения предусмотреть генерацию исключения, с выводом информации об ошибке на экран.

function calculateEntrance() {

// Получаем значения из input

const floors = parseInt(document.getElementById('floors').value);

const entrances = parseInt(document.getElementById('entrances').value);

const flatsPerFloor = parseInt(document.getElementById('flatsPerFloor').value);

const flatNumber = parseInt(document.getElementById('flatNumber').value);

// Очищаем предыдущий результат

document.getElementById('result2').textContent = '';

try {

// Проверяем корректность введенных данных

if (isNaN(floors) || floors < 1 || floors > 20) {

throw new Error('Некорректное количество этажей. Введите число от 1 до 20.');

}

if (isNaN(entrances) || entrances < 1 || entrances > 15) {

throw new Error('Некорректное количество подъездов. Введите число от 1 до 15.');

}

if (isNaN(flatsPerFloor) || flatsPerFloor < 1 || flatsPerFloor > 5) {

throw new Error('Некорректное количество квартир на этаже. Введите число от 1 до 5.');

}

if (isNaN(flatNumber) || flatNumber < 1) {

throw new Error('Некорректный номер квартиры. Введите положительное число.');

}

// Рассчитываем общее количество квартир в одном подъезде

const flatsPerEntrance = floors \* flatsPerFloor;

// Рассчитываем номер подъезда

const entranceNumber = Math.ceil(flatNumber / flatsPerEntrance);

// Проверяем, существует ли такая квартира в доме

if (entranceNumber > entrances) {

throw new Error('Квартиры с таким номером нет в доме.');

}

// Выводим результат

document.getElementById('result2').textContent = `Квартира находится в подъезде №${entranceNumber}.`;

} catch (error) {

// Выводим сообщение об ошибке

document.getElementById('result2').textContent = error.message;

}

}

**Задание 3**

22 марта 2022 был вторник. Написать скрипт который будет спрашивать у пользователя номер месяца в интервале “1..12”, а так же число в этом месяце “1..31”. Результатом работы скрипта является вывод на экран названия дня недели (например, «среда») для введённого пользователем числа и месяца.

<div>

<input type="number" id="month" placeholder="Month (1-12)" min="1" max="12">

<input type="number" id="day" placeholder="Day (1-31)" min="1" max="31">

<button onclick="calculateDayOfWeek()">Calculate Day</button>

<p id="dayResult"></p>

</div>

<script>

function calculateDayOfWeek() {

const month = parseInt(document.getElementById('month').value);

const day = parseInt(document.getElementById('day').value);

const result = document.getElementById('dayResult');

if (isNaN(month) || isNaN(day)) {

result.innerText = "Please enter valid numbers.";

return;

}

const date = new Date(`2022-${month}-${day}`);

const days = ['Sunday', 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday'];

result.innerText = `Day of the week: ${days[date.getDay()]}`;

}

</script>

**Задание 4**

Вам необходимо создать функцию, которая будет получать на вход два любых числа. При условии, если оба введённых числа чётные - необходимо венуть их произведение. При нечётных числах возвращаем сумму. В варианте, при котором первое число чётное, а второе нечётное - возвращаем нечётное число, в обратном случае возвращаем чётное

function compareEvenOdd(a, b) {

if (a % 2 === 0 && b % 2 === 0) {

return a \* b;

} else if (a % 2 !== 0 && b % 2 !== 0) {

return a + b;

} else if (a % 2 === 0) {

return b;

} else {

return a;

}

}

**Задание 5**

Необходмо создать функцию, которая будет возвращать сумму любого введённого количества чисел. Обратите внимание на то как должна вызываться данная функция: result = sum(3)(1)(1); // результат 5

function sum(a) {

return function(b) {

return function(c) {

return a + b + c;

};

};

}

**Задание 6**

Реализуйте функцию range(), которая будет принимать на вход два аргумента: “старт” и “финиш” диапазона. Функция возвращать массив, содержащий все числа ввдеённого диапазона, включая начальное и конечное. При этом необходимо предусмотреть необязательный аргумент – шаг для построения массива который может быть как положительным так и отрицательным. Если шаг явно не задан – он равен 1. Результатом работы range(7, 4, -2) будет [7, 5].

function range(start, end, step = 1) {

const result = [];

if (step > 0) {

for (let i = start; i <= end; i += step) {

result.push(i);

}

} else {

for (let i = start; i >= end; i += step) {

result.push(i);

}

}

return result;

}

**Задание 7**

Необходимо реализовать функцию создания матрицы, принимающую в качестве аргументов количество строк и количество столбцов. Эта функция должна отображать на экране матрицу, заполненную случайными числами в диапазоне от 0 до 100. Так же необходимо написать другую функцию, которая будет выполнять суммирование двух различных по значению “первых матриц”.

function createMatrix(rows, cols) {

const matrix = [];

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

const row = [];

for (let j = 0; j < cols; j++) {

row.push(Math.floor(Math.random() \* 101));

}

matrix.push(row);

}

return matrix;

}

function sumMatrices(matrix1, matrix2) {

const result = [];

for (let i = 0; i < matrix1.length; i++) {

const row = [];

for (let j = 0; j < matrix1[i].length; j++) {

row.push(matrix1[i][j] + matrix2[i][j]);

}

result.push(row);

}

return result;

}

**Задание 8**

Создайте функцию, которая будет объединять уникальные элементы всех одномерных массивов, переданных ей в качестве параметра. Если вводимый пользователем массив многомерный, то необходимо выдать пользователю сообщение о том что функция работает только с одномерными массива и указать размерность введённого массива. MyUnionFuic([1, 5, 5, 3], [10, 5, 1, 10], [5, 1]) вернёт[1, 3, 5, 10].

function MyUnionFunc(...arrays) {

const uniqueElements = new Set();

for (const array of arrays) {

if (!Array.isArray(array)) {

console.log("Only one-dimensional arrays are supported.");

return;

}

for (const element of array) {

uniqueElements.add(element);

}

}

return Array.from(uniqueElements);

}

**Задание 9**

Создайте функцию, которая в качестве агрумента может принимать массив с уровнями вложенности элементов любой глубины, выполнит приведение его к “плоскому” виду. Если элементы массива не имеют вложенности, то функция просто выводит его на экран без изменений. MyArrayFlattenFunc([1, [2], [[5], [3, [4]]]]) вернёт [1, 2, 3, 4, 5]

function MyArrayFlattenFunc(arr) {

const result = [];

arr.forEach(item => {

if (Array.isArray(item)) {

result.push(...MyArrayFlattenFunc(item));

} else {

result.push(item);

}

});

return result;

}

**Задание 10**

Реализовать функцию которая принимает на вход в качестве параметра массив или объект, подобный массиву, а так же параметр который отвечает за количество повторений. Возвращаемое функцией значение это массив, в котором удалены повторяющиеся знаачения.

var furstResult= MyFunctionUnique([1, 2, 1, 5, 1, 5, 1, 4], 3);

// furstResult = [2, 4, 5]

var secondResult = MyFunctionUnique("miru mir", 2);

// secondResult = [“ ”,“u”];

function MyFunctionUnique(arr, count) {

const occurrences = {};

arr.forEach(item => {

occurrences[item] = (occurrences[item] || 0) + 1;

});

return Object.keys(occurrences)

.filter(key => occurrences[key] <= count)

.map(key => parseInt(key, 10));

}

**Вывод:** в ходе работы были приобретены навыки работы с JavaScript