//Ejercico 1

// 1. Tiene dos arreglos de String, ‘clientes’ y ‘empleados’, y deseas combinarlas para

// crear una arreglo de contactos. ¿Qué método sería el más adecuado para esta

// tarea? Prueba todas las opciones y describe el resultado.

// a. concat

// b. join

// c. push

// d. splice

*let* clientes = ["cliente1", "cliente2"];

*let* empleados = ["empleado1", "empleado2"];

*let* combinar = clientes.concat(empleados);

console.log(combinar);

//la respuesta es concat

//ejercicio 2

// Teniendo el siguiente arreglo:

// var numbers = [5, 32, 43, 4];

// Y la siguiente función:

// numbers.filter(function(n) { return n % 2 !== 0; });

// Agrega una variable para alojar el resultado de la función filter() y explica cual

// es el resultado.

*var* numbers = [5, 32, 43, 4];

*var* resultado = numbers.filter(*function*(*n*) { return *n* % 2 !== 0; });

console.log(resultado);

//la funcion filter verifica si los numeros son pares o impares

//dando como resultado el 5 y el 43

//ejercicio 3

// Dado el siguiente arreglo de objetos:

// var people = [{

//  id: 1,

//  name: "John",

//  age: 28

// }, {

//  id: 2,

//  name: "Jane",

//  age: 31

// }, {

// id: 3,

//  name: "Peter",

// age: 55 }];

// Genera la función en javascript que filtre a las personas menores de 35 años.

*var* people = [{

     id: 1,

     name: "John",

     age: 28

    }, {

     id: 2,

     name: "Jane",

     age: 31

    }, {

    id: 3,

     name: "Peter",

    age: 55 }];

*var* personasMenores = people.filter( *function*(*persona*) {

    return persona.age < 35;

});

console.log(personasMenores);

//ejercicio 4

// Dado el siguiente arreglo de objetos:

// let people = [

// { name: "bob", id=1 }, { name: "john", id=2 },

// { name: "alex", id=3 }, { name: "john", id=3 }

// ];

// Genera la función para obtener el numero de veces que aparece cada nombre.

*let* people = [

{ name: "bob", id: 1 }, { name: "john", id: 2 },

{ name: "alex", id: 3 }, { name: "john", id: 3 }

];

*function* contarNom(*people*) {

*let* nombreGuardado = {};

    people.forEach(*personas* *=>* {

        if (nombreGuardado[personas.name]) {

            nombreGuardado[personas.name]++

         } else {

            nombreGuardado[personas.name] = 1;

         }

    });

  return nombreGuardado;

}

*let* resultados = contarNom(people);

console.log(resultados);

// Ejercico 5

// Dado el siguiente arreglo:

// var myArray = [1, 2, 3, 4];

// Genera la función para imprimir en consola el numero máximo y el numero

// mínimo.

*var* myArray = [1, 2, 3, 4];

*function* maxMin(*array*) {

*var* maximo = Math.max(...*array*);

*var* minimo = Math.min(...*array*);

    console.log("numero maximo", maximo);

    console.log("numero minimo", minimo);

}

maxMin(myArray);

//ejercicio 6

// Teniendo el siguiente objeto:

// var object = {

//  key1: 10,

//  key2: 3,

//  key3: 40,

//  key4: 20

// };

// Generar la función para pasar cada elemento ‘key’ a un arreglo y ordenarlos

// según su valor.

*var* object = {

     key1: 10,

     key2: 3,

     key3: 40,

     key4: 20

    };

*function* ordenar(*obj*) {

*var* keys = *Object*.keys(*obj*);

keys.sort(*function*(*a*, *b*) {

    return *obj*[*a*] - *obj*[*b*];

});

    return keys;

}

*var* ordenarLlaves = ordenar(object);

console.log(ordenarLlaves);

Alerta

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Alert con Información del Elemento</title>

</head>

<body>

    <ul>

        <li id="c1" data-id="US" data-dial-code="1">USA</li>

        <li id="c2" data-id="CA" data-dial-code="1">Canada</li>

        <li id="c3" data-id="FR" data-dial-code="1">France</li>

    </ul>

    <script src="main.js"></script>

</body>

</html>

jsdocument.addEventListener('DOMContentLoaded', () *=>* {

    // Seleccionar todos los elementos 'li'

*var* listItems = document.querySelectorAll('li');

    // Añadir un evento de clic a cada 'li'

    listItems.forEach(*item* *=>* {

*item*.addEventListener('click', *function*() {

*var* country = this.textContent;

*var* id = this.getAttribute('data-id');

*var* dialCode = this.getAttribute('data-dial-code');

            alert(`Country: ${country}\nID: ${id}\nDial Code: ${dialCode}`);

        });

    });

});

Position

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Eventos de Ratón y Teclado</title>

    <style>

        body {

*font-family*: Arial, sans-serif;

*text-align*: center;

*margin-top*: 100px;

        }

        #info {

*margin-top*: 20px;

        }

    </style>

</head>

<body>

<h1>Eventos de Ratón y Teclado</h1>

<div id="info">

    <p>Posición del ratón (respecto del navegador): <span id="posicionNavegador"></span></p>

    <p>Posición del ratón (respecto de la página): <span id="posicionPagina"></span></p>

    <p>Último evento de teclado: <span id="eventoTeclado"></span></p>

</div>

<script>

    // Obtener elementos

*var* posicionNavegadorSpan = document.getElementById('posicionNavegador');

*var* posicionPaginaSpan = document.getElementById('posicionPagina');

*var* eventoTecladoSpan = document.getElementById('eventoTeclado');

    // Función para actualizar la posición del ratón

*function* actualizarPosicion(*event*) {

*var* xNavegador = *event*.clientX;

*var* yNavegador = *event*.clientY;

*var* xPagina = *event*.pageX;

*var* yPagina = *event*.pageY;

        posicionNavegadorSpan.textContent = xNavegador + ', ' + yNavegador;

        posicionPaginaSpan.textContent = xPagina + ', ' + yPagina;

    }

    // Función para manejar eventos de ratón

*function* manejarEventoRaton(*event*) {

        actualizarPosicion(*event*);

    }

    // Función para manejar eventos de teclado

*function* manejarEventoTeclado(*event*) {

        eventoTecladoSpan.textContent = 'Tecla pulsada: ' + *event*.key;

    }

    // Agregar listeners

    document.addEventListener('mousemove', manejarEventoRaton);

    document.addEventListener('keypress', manejarEventoTeclado);

</script>

</body>

</html>

Tablas

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Tabla de Personas</title>

    <style>

        table {

*width*: 100%;

*border-collapse*: collapse;

        }

        table, th, td {

*border*: 1px solid black;

        }

        th, td {

*padding*: 10px;

*text-align*: left;

        }

        th {

*background-color*: #cde69e;

        }

    </style>

</head>

<body>

    <h2>Tabla de Personas</h2>

    <div id="table-container"></div>

    <script src="main.js"></script>

</body>

</html>

*var* personArr = [

    {

        "personId": 123,

        "name": "Jhon",

        "city": "melbourne",

        "phoneNO": "1234567890"

    },

    {

        "personId": 124,

        "name": "Amelia",

        "city": "sydney",

        "phoneNO": "1234567890"

    },

    {

        "personId": 125,

        "name": "Emily",

        "city": "perth",

        "phoneNO": "1234567890"

    },

    {

        "personId": 126,

        "name": "Abraham",

        "city": "perth",

        "phoneNO": "1234567890"

    }

];

*function* generateTable(*data*) {

*var* table = document.createElement('table');

    // Crear la fila de encabezado

*var* thead = table.createTHead();

*var* headerRow = thead.insertRow();

*var* headers = *Object*.keys(*data*[0]);

    headers.forEach(*header* *=>* {

*var* th = document.createElement('th');

        th.textContent = *header*.charAt(0).toUpperCase() + *header*.slice(1);

        headerRow.appendChild(th);

    });

    // Crear las filas del cuerpo de la tabla

*var* tbody = document.createElement('tbody');

*data*.forEach(*item* *=>* {

*var* row = tbody.insertRow();

        headers.forEach(*header* *=>* {

*var* cell = row.insertCell();

            cell.textContent = *item*[*header*];

        });

    });

    table.appendChild(tbody);

    return table;

}

document.addEventListener('DOMContentLoaded', () *=>* {

*var* tableContainer = document.getElementById('table-container');

*var* table = generateTable(personArr);

    tableContainer.appendChild(table);

});

Calculadora

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Calculadora</title>

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

</head>

<body>

    <div class="calculator">

        <div class="display">

            <input type="text" id="display" disabled>

        </div>

        <div class="buttons">

            <button class="btn" onclick="clearDisplay()">C</button>

            <button class="btn" onclick="appendValue('1')">1</button>

            <button class="btn" onclick="appendValue('2')">2</button>

            <button class="btn" onclick="appendValue('3')">3</button>

            <button class="btn operator" onclick="appendOperator('+')">+</button>

            <button class="btn" onclick="appendValue('4')">4</button>

            <button class="btn" onclick="appendValue('5')">5</button>

            <button class="btn" onclick="appendValue('6')">6</button>

            <button class="btn operator" onclick="appendOperator('-')">-</button>

            <button class="btn" onclick="appendValue('7')">7</button>

            <button class="btn" onclick="appendValue('8')">8</button>

            <button class="btn" onclick="appendValue('9')">9</button>

            <button class="btn operator" onclick="appendOperator('')"></button>

            <button class="btn" onclick="appendValue('0')">0</button>

            <button class="btn operator" onclick="appendOperator('/')">/</button>

            <button class="btn" onclick="calculateResult()">=</button>

            <button class="btn" onclick="backspace()">←</button>

        </div>

    </div>

    <script src="main.js"></script>

</body>

</html>

body {

*display*: flex;

*justify-content*: center;

*align-items*: center;

*height*: 100vh;

*background-color*: #f0f0f0;

*margin*: 0;

*font-family*: Arial, sans-serif;

}

.calculator {

*background-color*: #fff;

*padding*: 20px;

*border-radius*: 10px;

*box-shadow*: 0 0 10px rgba(0, 0, 0, 0.1);

}

.display {

*margin-bottom*: 20px;

}

#display {

*width*: 100%;

*padding*: 10px;

*font-size*: 24px;

*text-align*: right;

*border*: 1px solid #ccc;

*border-radius*: 5px;

*box-sizing*: border-box;

}

.buttons {

*display*: grid;

*grid-template-columns*: repeat(4, 1fr);

*gap*: 10px;

}

.btn {

*padding*: 20px;

*font-size*: 18px;

*border*: none;

*border-radius*: 5px;

*cursor*: pointer;

*transition*: background-color 0.3s;

}

.btn.operator {

*background-color*: #85c165;

*color*: #fff;

}

.btn.operator:hover {

*background-color*: #fbc02d;

}

.btn:hover {

*background-color*: #e0e0e0;

}

*let* currentInput = '';

*let* stack = [];

*function* clearDisplay() {

    currentInput = '';

    stack = [];

    updateDisplay();

}

*function* appendValue(*value*) {

    currentInput += *value*;

    updateDisplay();

}

*function* appendOperator(*operator*) {

    if (currentInput !== '') {

        stack.push(parseFloat(currentInput));

        stack.push(*operator*);

        currentInput = '';

        updateDisplay();

    }

}

*function* calculateResult() {

    if (currentInput !== '') {

        stack.push(parseFloat(currentInput));

    }

*let* result = stack[0];

    for (*let* i = 1; i < stack.length; i += 2) {

*const* operator = stack[i];

*const* nextValue = stack[i + 1];

        if (operator === '+') {

            result += nextValue;

        } else if (operator === '-') {

            result -= nextValue;

        } else if (operator === '\*') {

            result \*= nextValue;

        } else if (operator === '/') {

            result /= nextValue;

        }

    }

    currentInput = result.toString();

    stack = [];

    updateDisplay();

}

*function* backspace() {

    currentInput = currentInput.slice(0, -1);

    updateDisplay();

}

*function* updateDisplay() {

*const* display = document.getElementById('display');

    display.value = currentInput;

}