

Dado un input y un boton, escribir nombres de personas (no controlamos nada de momento) y lo mostramos en una lista desordenada. Deberemos de guardar los nombres en un array y comprobar si existe el nombre en el array.

— index.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <meta http-equiv="X-UA-Compatible"
content="IE=edge">
  <title>Page Title</title>
  <meta name="viewport"
content="width=device-width, initial-scale=1">
  <link rel="stylesheet" type="text/css"
media="screen" href="main.css" />
  <script src="main.js"></script>
</head>
<body>
  <form>
    <input id="nombre" type="text"
placeholder="Insertar nombre..."/>
    <button
onclick="insertarNombre(event)">Insertar</bu
tton>

  </form>
  <div>
    <ul id="lista-nombres"></ul>
  </div>
</body>
</html>
```

```
nombres = []

function insertarNombre(evt) {
  evt.preventDefault();

  let nombre =
document.getElementById("nombre").
value;
```

```

        if (nombre == "") {
            alert("El nombre no puede
estar vacio");
        } else
if(!existeNombre(nombre)) {
    let opcion = "<li> " +
nombre + "</li>";

    let lista =
document.getElementById("lista-nom
bres");

    lista.innerHTML += opcion;

    nombres.push(nombre);

    alert("Se ha insertado el
nombre");
} else {
    alert("Existe el nombre");
}
}

function existeNombre(nombre) {

    // Busco el nombre
    const nombreEncontrado =
nombres.find(element =>
element.trim().toLowerCase() ===
nombre.trim().toLowerCase());

    if(nombreEncontrado){
        return true;
    } else {
        return false;
    }

    /*
    // Opcion clasica
    for(let i=0; i <
nombres.length; i++){
        const element =
nombres[i];

        if(element.trim().toLowerCase()
=== nombre.trim().toLowerCase()){

```

```

        return true;
    }

}

return false;
*/
}

```

-----Ejercicios array-----

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>EJ-Arrays</title>
</head>
<body>

    <script>
        /*---EJ1---*/
/*
function eraseElement(array) {
    // Muestra el array inicial
    console.log("Array inicial:", array);

    // Solicitamos la posición que vamos a eliminar
    let pos = parseInt(prompt("Introduce la posición a eliminar (entre 1 y 10): ")) - 1;

    // Guardamos el valor que se eliminará
    let valorBorrado = array[pos];

    // Eliminamos el elemento de la posición indicada
    array.splice(pos, 1);

    // Desplazamos todos los elementos a la izquierda
    for (let i = pos; i < array.length; i++) {
        array[i-1] = array[i];
    }

    // Mostramos el array final y el valor eliminado
    console.log("Array final:", array);
    console.log("Valor eliminado:", valorBorrado);
}

```

```

// Creamos un array de 10 posiciones con valores aleatorios entre 1 y 100
let array = Array.from({length: 10}, () => Math.floor(Math.random() * 100) + 1);

// Llamamos a la función para eliminar un elemento
eraseElement(array);
*/

/*---EJ2---*/

function rellenarArray() {
  let array = [];
  for (let i = 1; i < 11; i++) {
    let valor;

    if (i % 2 == 0) {

      // Si la posición es par
      valor = Math.floor(Math.random() * 6 + 14) * 2; // Generamos un valor par entre 14 y 25
    }

    else {

      // Si la posición es impar
      valor = Math.floor(Math.random() * 6 + 14) * 2 + 1; // Generamos un valor impar entre 14
y 25
    }
    array.push(valor);
  }
  return array;
}

// Ejemplo de uso
let array = rellenarArray();
console.log(array);

/*---EJ3---*/
/*
function unirArraysYEliminarRepetidos(array1, array2) {

// Unimos los dos arrays en uno solo
let arrayUnion = array1.concat(array2);
let valoresDeleted = 0;

for (let i = 0; i < arrayUnion.length; i++) {
  for (let j = i + 1; j < arrayUnion.length; j++) {
    if (arrayUnion[i] === arrayUnion[j]) {
      // Si encontramos un valor repetido

```

```

        arrayUnion.splice(j, 1);
        // Eliminamos el valor repetido del array

        j--;
        // Decrementamos j para evitar saltar valores al eliminarlos

        valoresDeleted++;
        // Incrementamos el contador de valores eliminados
    }
}
}

return { array: arrayUnion, valoresDeleted: valoresDeleted };
}

// Ejemplo de uso
let array1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];
let array2 = [5, 6, 7, 8, 9, 10, 11, 12, 13, 14];

let resultado = unirArraysYEliminarRepetidos(array1, array2);

console.log(resultado.array);
console.log(`Valores eliminados: ${resultado.valoresEliminados}`);
*/

/*---EJ4---*/
/*
let nombres = [];

while (true) {
    let nombre = prompt("Introduce un nombre (o pulsa cancelar para salir:");
    if (nombre === null) {
        break;
    }
    nombres.push(nombre.toUpperCase());
}

nombres.sort();

let numPalabrasVocal = 0;
for (let nombre of nombres) {
    if (nombre.startsWith("A") || nombre.startsWith("E") || nombre.startsWith("I") ||
nombre.startsWith("O") || nombre.startsWith("U")) {
        numPalabrasVocal++;
    }
}
}

```

```

console.log("Nombres introducidos:", nombres);
console.log("Número de palabras que empiezan por vocal:", numPalabrasVocal);
*/
    </script>

</body>
</html>

```

-----EJERCICIOS EVENTOS-----

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="DOM-ResaltarImg.css">
</head>

<body>

  <header>
    <input onclick="resaltarCasa()" type="button" id="botCasas" value="Casas">
    <input onclick="resaltarPisos()" type="button" id="botPisos" value="Pisos">
  </header>

  <main>
    <article>
      
      
      
    </article>
    <article>
      
      
      
    </article>
  </main>

  <script>
function resaltarCasa() {
  // Remover el borde de todas las casas
  let casas = document.getElementsByClassName("casas");
  for (let i = 0; i < casas.length; i++) {
    casas[i].style.border = "none";
  }
}

```

```

    // Agregar el borde rojo a la casa del medio
    let casaDelMedio = document.getElementsByClassName("casas")[1];
    casaDelMedio.style.border = "3px solid red";
  }

  function resaltarPisos() {
    // Agregar el borde azul a todos los pisos
    let pisos = document.getElementsByClassName("pisos");
    for (let i = 0; i < pisos.length; i++) {
      pisos[i].style.border = "3px dashed blue";
    }
  }
</script>
</body>
</html>

```

—PARTE CSS—

```

main{
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
}
header{
  width: 100%;
  text-align: center;
  margin-bottom: 2em;
}
input{
  margin: 1em;
}

article{
  width: 100vw;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-around;
  text-align: center;
  margin:.5em;
}
.casas {
  width: 30vw;
  height:80vh;
}

```

-----TEREA DOM1-----

<!DOCTYPE html>

```

<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tarea DOM</title>
  <link rel="stylesheet" type="text/css" href="css/DOM.css">

  <script type="text/javascript" src="js/DOM.js"></script>
</head>
<body>
  <header>
    <h1>Uso del DOM</h1>
  </header>
  <section>
    <div>
      <p id="amarillo">Jugamos el DOM con títulos</p>
    </div>

    <h2 id="adaptable"> Este es un título que vamos a hacer dinámico</h2>

    <button onclick="cambiarColor()">Cambiar color</button>
    <button onclick="cambiarTamano()">Cambiar tamaño de fuente</button>
  </section>

</body>
</html>

```

—JS—

```

function cambiarColor(){
  var titutlo = document.getElementById("adaptable");
  titutlo.style.color = "red";
}

```

```

function cambiarTamano(){
  var titulo = document.getElementById("adaptable");
  titulo.style.fontSize = "50px";
}

```

—CSS—

```

header{
  background-color: grey;
  color:darkblue;
  text-align: center;

}

p{
  background-color: yellow;

```



```

        text-align: center;
    }

    h2{
        color:#B63285;
        text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.5);
        text-align: center;
        font-family:Helvetica;
    }

```

-----TAREA DOM2-----

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>EJ2-DOM</title>
    <link rel="stylesheet" type="text/css" href="css\DOM2.css">
    <script type="text/javascript" src="js/DOM2.js"></script>
</head>
<body>
    <header>
        <h1>Dinamismo en HTML</h1>
    </header>
    <section>
        <h2> Jugando con DOM y Eventos</h2>

        <p>En este ejercicio vamos a validar un código postal que puede formar
perfectamente parte de un formulario html que construyamos para recojer datos de un
cliente, por tanto usaremos un input como elemento base html.</p>
        <hr>

        <div id="form">
            <input id="codigoPostal" type="text" name="Nombre"
placeholder="Código Postal" required>
            <button id="boton"
onclick="validarCodigoPostal()">Validar</button>
        </form>
    </div>

    </section>
</body>
</html>
—JS—
function validarCodigoPostal() {
    let codigoPostal = document.getElementById("codigoPostal").value;

    // Verificar que el código postal tiene 5 dígitos
    if (codigoPostal.length !== 5 || isNaN(codigoPostal)) {

```

```

    alert("Código Invalido!!");
    return;
}

// Verificar las provincias válidas
let provincia = codigoPostal.slice(0, 2);
let provinciasValidas = ["01", "11", "17", "28"];
if (!provinciasValidas.includes(provincia)) {
    alert("Código Invalido!!");
    return;
}

// Verificar el rango de códigos válidos
let codigoNumerico = parseInt(codigoPostal);
if (codigoNumerico < 1000 || codigoNumerico > 52999) {
    alert("Código Invalido!!");
    return;
}

alert("Código Válido");
}

```

-----TAREA STRINGS-----

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>EJ-String</title>
</head>
<body>
    <script type="text/javascript" src="js/strings.js"></script>
    <br>
    <script type="text/javascript">

```

```

/*---EJ1---*/

```

```

var texto = prompt("Escribe la siguiente cadena: \nDisfruto del día");
var separation = texto.split("").join("-");

```

```

document.write(separation);
document.write("<br>")

```

```

/*---EJ2---*/

```

```

var texto = prompt("Escriba una cadena de texto");
var particion = Math.ceil(texto.length / 2);
var contar = texto;
var part1 = texto.substring(0, particion);

```

```

var part2 = texto.substring(particion);

if(texto.length % 2 === 0){
    document.write(part1+ "-" +part2);
}

else
    document.write(part1.fontcolor("red") +part2);
</script>
</body>
</html>

```

EJERCICIOS FUNCIONES

EJERCICIO1

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Funciones en JavaScript</title>
    <script src="js/funciones.js"></script>
  </head>
  <body>
    <h1>Funciones en JavaScript</h1>

    <p>El resultado total es: <strong id="resultado"></strong></p>
    <p>La fecha del día es: <strong id="resultado2"></strong></p>

    <script>
let x = prompt("Introduce el número base:");
let n = prompt("Introduce la potencia:");

try {
    let resultado = potencia(x, n);
    alert(x + " elevado a la potencia " + n + " es igual a " + resultado);
    document.getElementById("resultado").innerHTML = resultado;
} catch (error) {
    alert(error.message);
}

let fecha = fechaHoy();
document.getElementById("resultado2").innerHTML = fecha;

    </script>
  </body>
</html>

```

—JS—

```
function potencia(x, n) {  
  if (n === 0) {  
    throw new Error("La potencia no puede ser cero");  
  } else {  
    let resultado = 1;  
    for (let i = 0; i < n; i++) {  
      resultado *= x;  
    }  
    return resultado;  
  }  
}
```

```
function fechaHoy() {  
  let fecha = new Date();  
  let dia = ("0" + fecha.getDate()).slice(-2);  
  let mes = ("0" + (fecha.getMonth() + 1)).slice(-2);  
  let anio = fecha.getFullYear();  
  return dia + "-" + mes + "-" + anio;  
}
```

EJERCICIO 2

```
<!DOCTYPE html>  
<html>  
  <head>  
    <meta charset="UTF-8" />  
    <title>Funciones en JavaScript</title>  
    <script src="js/funciones.js"></script>  
  </head>  
  <body>  
    <h1>Funciones en JavaScript</h1>  
  
    <p>El menor de dos números es: <strong id="resultado1"></strong></p>  
    <p>La fecha del día es: <strong id="resultado2"></strong></p>  
  
    <script>  
      // Llamamos a la función para obtener el menor de dos números y mostramos el  
      resultado en la página  
      let menor = menorDeDosNumeros();  
      document.getElementById("resultado1").innerHTML = menor;  
  
      // Llamamos a la función para obtener la fecha del día y mostramos el resultado en la  
      página  
      let fecha = fechaDelDia();  
      document.getElementById("resultado2").innerHTML = fecha;  
    </script>
```

```

</body>
</html>
—JS—
// Función para devolver el menor de dos números
function menorDeDosNumeros() {
  let a = parseInt(prompt("Ingresa el primer número:"));
  let b = parseInt(prompt("Ingresa el segundo número:"));

  if (isNaN(a) || isNaN(b)) {
    return "Error: Ingresa dos números válidos";
  } else if (a < b) {
    return a;
  } else {
    return b;
  }
}

// Función para devolver la fecha actual en formato "dd-mm-yyyy"
function fechaDelDia() {
  let fecha = new Date();
  let dia = ("0" + fecha.getDate()).slice(-2);
  let mes = ("0" + (fecha.getMonth() + 1)).slice(-2);
  let anio = fecha.getFullYear();
  return dia + "-" + mes + "-" + anio;
}

```

EXAMEN

```

—HTML—
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>ej4-Programación</title>
  <script type="text/javascript" src="js/EJERCICIO4.js"></script>
</head>
<body>

<section>

  <h2 onmouseover="cambiarColorA()" onmouseout="quitarColorA()"
id="amarillo" style=" color:darkblue; background: none;">Primer Texto de tipo enlace que
vamos a resaltar en amarillo</h2>

  <h2 onmouseover="cambiarColorG()" onmouseout="quitarColorG()" id="gris"
style="color:darkblue; background: none">Segundo Texto de tipo enlace que vamos a
resaltar en gris</h2>

```

```

        </section>

</body>
</html>

-JAVA-
function cambiarColorA(){
    var titutlo = document.getElementById("amarillo");
    titutlo.style.background = "yellow";

}

function cambiarColorG(){
    var titulo = document.getElementById("gris");
    titulo.style.background = "grey";

}

function quitarColorA(){
    var titutlo = document.getElementById("amarillo");
    titutlo.style.background = "none";

}

function quitarColorG(){
    var titulo = document.getElementById("gris");
    titulo.style.background = "none";

}

```

-----EXAMEN2-----

```

-HTML-
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Calculadora de Peso Ideal</title>
    <script src="js/calculadora.js"></script>
</head>
<body>
    <h1>Calculadora de Peso Ideal</h1>

    <form id="formulario">
        <label for="altura">Altura (cm):</label>

```

```

<input type="number" id="altura" required>

<label for="edad">Edad:</label>
<input type="number" id="edad" required>

<label for="sexo">Sexo:</label>
<select id="sexo" required>
  <option value="h">Hombre</option>
  <option value="m">Mujer</option>
</select>

<label for="pesoActual">Peso Actual (kg):</label>
<input type="number" id="pesoActual" required>

<button type="submit">Calcular Peso Ideal</button>
</form>

<div id="resultado"></div>

<script>
  // Capturar el evento submit del formulario
  document.getElementById("formulario").addEventListener("submit", function(event) {
    event.preventDefault(); // Evitar que el formulario se envíe

    // Obtener los valores ingresados por el usuario
    let altura = document.getElementById("altura").value;
    let edad = document.getElementById("edad").value;
    let sexo = document.getElementById("sexo").value;
    let pesoActual = document.getElementById("pesoActual").value;

    // Validar los datos ingresados

    // ...

    // Calcular el peso ideal y mostrar el resultado
    let pesoIdeal = calcularPesoIdealHombre(parseInt(altura), parseInt(edad));
    if (sexo === "m") {
      pesoIdeal *= 0.9;
    }

    document.getElementById("resultado").innerText = "Su peso ideal aproximado es: " +
    pesoIdeal.toFixed(2) + " kg";
  });
</script>
</body>
</html>

```

—JAVA—

```

function calcularPesoldeal() {
  let altura, edad, sexo, pesoActual;

  // Validar altura (entre 65 y 250 cm)
  while (true) {
    altura = parseInt(prompt("Ingrese su altura en cm:"));
    if (altura >= 65 && altura <= 250) {
      break;
    } else {
      alert("La altura ingresada no es válida. Debe estar entre 65 y 250 cm.");
    }
  }

  // Validar sexo (h/H para hombre, m/M para mujer)
  while (true) {
    sexo = prompt("Ingrese su sexo (Hombre: h/H, Mujer: m/M):");
    if (sexo.toLowerCase() === "h" || sexo.toLowerCase() === "m") {
      break;
    } else {
      alert("El sexo ingresado no es válido. Use h/H para hombre o m/M para mujer.");
    }
  }

  // Validar edad (entero positivo)
  while (true) {
    edad = parseInt(prompt("Ingrese su edad:"));
    if (!isNaN(edad) && edad > 0) {
      break;
    } else {
      alert("La edad ingresada no es válida. Debe ser un número entero positivo.");
    }
  }

  // Solicitar peso actual
  pesoActual = parseFloat(prompt("Ingrese su peso actual en kg:"));

  // Calcular peso ideal
  let pesoldeal = calcularPesoldealHombre(altura, edad);
  if (sexo.toLowerCase() === "m") {
    pesoldeal *= 0.9;
  }

  // Mostrar resultado
  alert("Su peso ideal aproximado es: " + pesoldeal.toFixed(2) + " kg");
}

function calcularPesoldealHombre(altura, edad) {

```



```
    return 50 + ((altura - 150) / 4) * 3 + (edad - 20) / 4;
}
```

```
// Ejecutar la función principal
calcularPesoldeal();
```

-----Datos-----

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Dado</title>
</head>
<body>
    <h1> Resultado de lanzar un dado 1000 veces</h1>
    <div id="resultados"></div>
    <script>
function lanzarDado() {
    return Math.floor(Math.random() * 6) + 1;
}

function contarCaras() {
    let resultados = {
        1: 0,
        2: 0,
        3: 0,
        4: 0,
        5: 0,
        6: 0
    };

    for (let i = 0; i < 1000; i++) {
        let cara = lanzarDado();
        resultados[cara]++;
    }

    return resultados;
}

let conteo = contarCaras();
let resultadosDiv = document.getElementById("resultados");

for (let cara in conteo) {
    let p = document.createElement("p");
    p.textContent = "Cara " + cara + ": " + conteo[cara];
    resultadosDiv.appendChild(p);
}
</script>
```

```
</body>
```

```
</html>
```

```
-----Notas-----
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

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

```
  <title>Calificaciones</title>
```

```
</head>
```

```
<body>
```

```
  <h1>Calificaciones</h1>
```

```
  <label for="nota">Introduce tu nota:</label>
```

```
  <input type="number" id="nota" min="0" max="10">
```

```
  <button onclick="calificar()">Calificar</button>
```

```
  <p id="mensaje"></p>
```

```
<script>
```

```
  function calificar() {
```

```
    var notaInput = document.getElementById("nota");
```

```
    var nota = parseFloat(notaInput.value);
```

```
    var mensaje = document.getElementById("mensaje");
```

```
    if (isNaN(nota)) {
```

```
      mensaje.textContent = "La nota ingresada no es válida.";
```

```
      mensaje.style.color = "red";
```

```
    } else if (nota >= 0 && nota < 5) {
```

```
      mensaje.textContent = "Calificación: Insuficiente";
```

```
      mensaje.style.color = "red";
```

```
    } else if (nota >= 5 && nota < 7) {
```

```
      mensaje.textContent = "Calificación: Suficiente";
```

```
      mensaje.style.color = "orange";
```

```
    } else if (nota >= 7 && nota < 9) {
```

```
      mensaje.textContent = "Calificación: Notable";
```

```
      mensaje.style.color = "green";
```

```
    } else if (nota >= 9 && nota <= 10) {
```

```
      mensaje.textContent = "Calificación: Sobresaliente";
```

```
      mensaje.style.color = "blue";
```

```
    } else {
```

```
      mensaje.textContent = "La nota ingresada no es válida.";
```

```
      mensaje.style.color = "red";
```

```
    }
```

```
  }
```

```
</script>
```

```
</body>
</html>
```

```
-----Arbol-----
// Solicitar al usuario el número de peldaños
var array = []
user_in = prompt(parseInt("Enter number: "));

while(array.length < user_in){
array.push(Math.floor(1000*Math.random()));
document.write ("El numero es "+array+"<br>");
}*/
/*
var numLineas = parseInt(prompt("Ingrese el número de líneas del árbol:"));

for (var i = 1; i <= numLineas; i++) {
    var linea = "";

    for (var j = 1; j <= numLineas - i; j++) {
        linea += " ";
    }

    for (var k = 1; k <= (2 * i) - 1; k++) {
        linea += "*";
    }

    document.write("<br>" + linea + "<br>");
}

var tronco = "";
for (var l = 1; l <= numLineas - 1; l++) {
    tronco += " ";
}
tronco += "|";
document.write(tronco);*/

function ejer3(ara){

    ara=[1,33,2,44,3,55];

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

        if(ara[i]%2==0){

            document.write('en la posicion '+i+' hay un '+ara[i]+'<br>'); }
    }
}
```

```
}  
}  
ejer3()
```

-----Carusel de fotos-----

```
[23:13, 14/6/2023] Abner FP: <div class="carousel">  
  <div class="carousel-container">  
      
      
      
  </div>  
</div>
```

```
[23:13, 14/6/2023] Abner FP: .carousel {  
  width: 100%;  
  overflow: hidden;  
}
```

```
.carousel-container {  
  display: flex;  
  transition: transform 0.3s ease-in-out;  
}
```

```
.carousel img {  
  width: 100%;  
  height: auto;  
}
```

```
[23:13, 14/6/2023] Abner FP: document.addEventListener("DOMContentLoaded", function()  
{  
  const carouselContainer = document.querySelector(".carousel-container");  
  const images = carouselContainer.getElementsByTagName("img");  
  const totalImages = images.length;  
  let currentIndex = 0;
```

```
  function showImage(index) {  
    carouselContainer.style.transform = `translateX(-${index * 100}%)`;  
  }
```

```
  function nextImage() {  
    currentIndex = (currentIndex + 1) % totalImages;  
    showImage(currentIndex);  
  }
```

```
  function prevImage() {  
    currentIndex = (currentIndex - 1 + totalImages) % totalImages;  
    showImage(currentIndex);  
  }
```

```
  setInterval(nextImage, 3000); // Cambiar la imagen cada 3 segundos
```

```
// Eventos de botones
document.getElementById("prevBtn").addEventListener("click", prevImage);
document.getElementById("nextBtn").addEventListener("click", nextImage);
});
```

-----Parte Teórica Comandos-----

Operadores aritméticos

Suma (+). Suma dos números: $3 + 2 = 5$. Si en lugar de números se suman cadenas, como por ejemplo "A" + "B" = "AB" se produce una concatenación. Si alguno de los elementos a concatenar no es una cadena de texto, queda convertido a cadena: "AB" + 123 = "AB123" . Resta (-). Diferencia entre dos números. Ejemplo A: $3 - 2 = 1$. Ejemplo B: $(-1) - (-2) = 1$.

Negativo (-). Indica el signo negativo de un número o una expresión: -3.

División (/). Divide dos números, obteniendo un cociente de coma flotante: $5 / 2 = 2.5$. Fíjate en que el separador decimal es el punto.

Módulo aritmético %. Divide dos números, obteniendo un resto entero: $5 \% 2 = 1$.

Multiplicación (*). Multiplica dos números: $3 * 2 = 6$.

Incrementos y decrementos (++ --). Incrementa o decrementa el valor de una variable numérica en una unidad. No puede aplicarse a un literal.

Pueden darse dos casos:

```
var A, B; B = 2;

A = ++B
A = B++
```

En el primer caso, $A = 3$ Es decir, que el incremento de B ha sido el correcto, pero no así el segundo, $A = 2$.

Si se usa como operador prefijo (++x), devuelve el valor de su operando después de agregar uno; si se usa como operador sufijo (x++), devuelve el valor de su operando antes de agregar uno

De igual forma, para decrementos:

```
var A, B; B = 2;

A = --B
A = B--
```

En el primer caso resulta: $A = 1$ y en el segundo: $A = 2$

Operadores lógicos

Estos operadores tienen unas aplicaciones un tanto especiales, ya que no funcionan como los aritméticos, conocidos por todos.

Para comprender el funcionamiento de los operadores lógicos, primero hay que saber qué son los estados lógicos de una expresión. Y te preguntarás que es eso de "estados lógicos"... Esta es una cuestión un tanto filosófica, ya que el estado lógico de algo depende de respecto a qué establecemos ese estado, y por tanto no siempre será el mismo. Dependerá del entorno o contexto en el que se define y utiliza la expresión.

En JavaScript hay tres posibles estados lógicos:

True (verdadero),

False (falso) y

Null (nulo, es decir que no se ha establecido estado alguno por la razón que sea)

Disyunción lógica || (Or). Establece una disyunción lógica de dos expresiones, es decir, que el resultado se dará evaluando una expresión u otra.

El hierro es un metal = True.

El hierro es duro = True.

El hierro es un metal || El hierro es duro = True.

| Operador | Descripción |
|------------------------------------------|------------------------------------------------------------------------------------------------------------|
| . [] () | Acceso a campos, índice de matrices y llamada a funciones. |
| ++ -- - ~ ! delete new typeof void | Incremento +1, decremento -1, negativo, NOT, NOT lógico borrado, crear objeto, mostrar tipo, indefinido |
| * / % | Multiplicación, división, módulo de división (resto) |
| + - + | Suma, resta, concatenación de cadenas |
| << >> >>> | Bit shifting |
| < <= > >= | menor que, menor que o igual, mayor que, mayor que o igual |
| == != === !== | Igualdad, desigualdad, identidad, no identidad |

| | |
|----|---------------------|
| & | AND |
| ^ | XOR |
| | OR |
| && | AND logico |
| | OR logico |
| ?: | Condicional |
| = | Asignación |
| , | Evaluación múltiple |

1 Bucle while

La sintaxis del **bucle while** es la siguiente:

```
while (condición) {
    bloque_de_sentencias
}
```

La condición se evalúa al principio de cada iteración: si el resultado es true se ejecuta el bloque de sentencias; si el resultado es false el bucle se termina



Cuando se programa un bucle while hay que tener cuidado en que la condición deje de cumplirse en algún momento, porque si no es así, el bucle no terminaría nunca.

En los bucles while más sencillos, antes del bucle se inicializa una variable que se evalúa en la condición y dentro del bucle se modifica la variable, como muestra el ejemplo siguiente:

```
i = 0; // Inicialización de la variable contador
```

```
// Condición: Mientras la variable contador sea menor de 5

while (i < 5) {

    console.log("Valor de i:", i);

    i = i + 1; // Incrementamos el valor de i

}
```

Ejemplo: Realizar un programa que imprima en pantalla los números del 1 al 100.

Sin conocer las estructuras repetitivas podemos resolver el problema empleando una estructura secuencial. Inicializamos una variable con el

valor 1, luego imprimimos la variable, incrementamos nuevamente la variable y así sucesivamente. Pero esta solución es muy larga.

La mejor forma de resolver este problema es emplear una estructura repetitiva:

```
<!DOCTYPE html>

<html>

<head>

    <title>Ejemplo de JavaScript</title>

    <meta charset="UTF-8">

</head>

<body>

<script>

    var x;

    x=1;

    while (x<=100)

    {

        document.write(x);

        document.write('<br>');

        x=x+1;

    }
```



```
</script>
```

```
</body>
```

```
</html>
```

2. Bucle do ... while

La sintaxis del **bucle do ... while** es la siguiente:

```
do {  
  
    bloque_de_sentencias  
  
} while (condición)
```

La condición se evalúa al final de cada iteración: si el resultado es true se ejecuta el bloque de sentencias; si el resultado es false el bucle se termina.



El bucle do ... while es muy similar al bucle while, la principal diferencia es que en el bucle do ... while el bloque de sentencias se ejecuta por lo menos una vez mientras que en el bucle while depende de si la condición es cierta o no la primera vez que se evalúa.

Cuando se programa un bucle do ... while hay que tener cuidado en que la condición deje de cumplirse en algún momento, porque si no es así, el bucle no terminaría nunca.

Problema: Escribir un programa que solicite la carga de un número entre 0 y 999, y nos muestre un mensaje de cuántos dígitos tiene el mismo. Finalizar el programa cuando se cargue el valor 0.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Ejemplo de JavaScript</title>
```

```
    <meta charset="UTF-8">
```

```
</head>
```

```
<body>
```

```
<script>

    var valor;

    do {

        valor=prompt('Ingrese un valor entre 0 y 999:', '');

        valor=parseInt(valor);

        document.write('El valor '+valor+' tiene ');

        if (valor<10)

        {

            document.write('Tiene 1 digitos');

        }

        else

        {

            if (valor<100)

            {

                document.write('Tiene 2 digitos');

            }

            else

            {

                document.write('Tiene 3 digitos');

            }

        }

        document.write('<br>');

    } while(valor!=0);

</script>

</body>

</html>
```

La estructura for estructura se emplea en aquellas situaciones en las cuales CONOCEMOS la cantidad de veces que queremos que se

La sintaxis más habitual del **bucle for** es la siguiente:



- Se establece el valor inicial de la variable de control definida en la asignación inicial.
- Evalúa la condición de continuación:
 - si el resultado es true se ejecuta el bloque de sentencias, se efectúa el cambio de la variable de control y se evalúa nuevamente la condición de continuación;
 - si el resultado es false el bucle se termina.

```
<!DOCTYPE html>

<html>

<head>

  <title>Ejemplo de JavaScript</title>

  <meta charset="UTF-8">

</head>

<body>

<script>

  var f;

  for(f=1;f<=10;f++)
```

```

{
    document.write(f+" ");
}

</script>

</body>

</html>

```

Inicialmente `f` se la inicializa con 1. Como la condición se verifica como verdadera se ejecuta el bloque del `for` (en este caso mostramos el contenido de la variable `f` y un espacio en blanco). Luego de ejecutar el bloque pasa al tercer argumento del `for` (en este caso con el operador `++` se incrementa en uno el contenido de la variable `f`, existe otro operador — que decreuenta en uno una variable), hubiera sido lo mismo poner `f=f+1` pero éste otro operador matemático nos simplifica las cosas.

Importante: Tener en cuenta que no lleva punto y coma al final de los tres argumentos del `for`.

El disponer un punto y coma provoca un error lógico y no sintáctico, por lo que el navegador no avisará.

Ejemplo: La siguiente función cuenta el número de pares de un array pasado por parámetro:

```

function pares(array){
    let nPares=0;
    for (let i=0;i<array.length;i++){
        if (array[i]%2==0)
            nPares++;
    }

    return nPares;
}

```

← solución 1

solución 2 →

```

function pares(array){
    let nPares=0;
    for (n of array){
        if (n%2===0){
            nPares++;
        }
    }
    return nPares;
}

```

Llamada: →

```

<body>
<script src="funciones.js"></script>
<script>
    document.write(pares([1,2,3,4,5,6,7,8,9]));
</script>
</body>

```

Ejemplo: La siguiente función devolverá la palabra 'par' si le enviamos por parámetro un número par y devolverá la palabra 'impar' en caso contrario.

```
function par(x){  
  return((x%2==0) ? "par" : "impar");  
}
```

← solución 1

solución 2 →

```
function parImpar(numero) {  
  if(numero % 2 == 0) {  
    return "par";  
  }  
  else {  
    return "impar";  
  }  
}
```

Llamada: →

```
<body>  
<script src="funciones.js"></script>  
<script>  
  document.write(par(11)+'<br>');  
  document.write(parImpar(7)+'<br>');  
</script>  
</body>
```

Ejemplo propuesto: Escribe 500 números aleatorios entre el 1 y el 10.000 y que al lado de cada número diga si es par o impar haciendo uso de la función creada en el ejemplo anterior (`function aleatoriosparImpar()`)

```
/*  
1 - Solicita un nombre, la edad y muestra por consola el mensaje "Hola  
____, tienes ____ años y el año que viene tendrás ____ años"  
Realiza el ejercicio con prompt(mensaje) y haz uso de los template strings  
*/  
  
let name = prompt('Introduce tu nombre')  
let age = parseInt(prompt('Introduce tu edad'))  
  
console.log(`Hola ${name} tienes ${age} años y el año que viene tendrás $  
{age+1} años`);
```

```

let figure = prompt('Introduce la figura geométrica de la que
quieres calcular el área: triangle, rectangle or circle')

let base;
let height;
let radius;

switch (figure) {
  case 'triangle':
    base = prompt('Introduce la base del triángulo')
    height = prompt('Introduce la altura del triángulo')
    console.log(`El área del triángulo es ${ (base*height)/2 }`);
    break
  case 'rectangle':
    base = prompt('Introduce la base del rectángulo')
    height = prompt('Introduce la altura del rectángulo')
    console.log(`El área del rectángulo es ${base*height}`);
    break
  case 'circle':
    radius = prompt('Introduce el radio del círculo')
    console.log(`El área del círculo es ${Math.PI * Math.pow(
radius,2)}`);
    break
}

```

```

/*
3 - Solicita un número e imprime todos los números pares e impares
desde 1 hasta ese número con el mensaje "es par" o "es impar"
si el número es 5 el resultado será:
    1 - es impar
    2 - es par
    3 - es impar
    4 - es par
    5 - es impar
*/

let num = parseInt(prompt('Introduce un número'))

for (let i=1; i<=num; i++){
  if (i%2==0){
    console.log(`${i} - es par`);
  } else {
    console.log(`${i} - es impar`);
  }
}

```

```

/*
4 - Escribe un programa que pida un número entero mayor que 1 y
que escriba si el número primo o no.
    Un número primo es aquel que solo es divisible por sí mismo y
    la unidad
*/

let num = parseInt(prompt('Introduce un número'))
let divisores = 0

if (num === 1) console.log('El número no es válido')
else {
    for (let i = 2; i < num; i++) {
        if (num % i === 0) {
            console.log(`${num} / ${i} = ${num / i} No es primo`)
            divisores++
            break
        }
    }
}

if(divisores===0) console.log(`${num} es primo`);

```

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Page Title</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <script src="main_6.js"></script>
</head>
<body>
    <form>
        <input type="text" id="num1" />
        <input type="text" id="num2" />
        <button type="button" onclick="mayorNum()">¿Quién es el mayor?</button>
    </form>
</body>
</html>

```

```

function mayorNum() {

    var num1 = parseInt(document.getElementById("num1").value);
    var num2 = parseInt(document.getElementById("num2").value);

    if(num1 && num2 ){
        if (num1 >= num2) {
            if (num1 == num2) {
                alert("Los numeros son iguales")
            }else{
                alert("El num1 es mayor")
            }
        }else{
            alert("El num2 es mayor")
        }
    }else{
        alert("Uno de los numeros no se relleno")
    }
}

```

```

6
7  const numbers = [3,43,21,20,56]
8  let pares = []
9  let impares = []
10
11  for(const number of numbers){
12      let random = Math.round(Math.random() * 10 + 1)
13      const result = number * random
14
15      console.log(`${number} x ${random} = ${result}`);
16      if(result % 2 == 0){
17          pares.push(result)
18      }else{
19          impares.push(result)
20      }
21  }
22
23  console.log(pares);
24  console.log(impares);

```



```

9 - Solicitar al usuario una palabra y mostrar por consola el
número de consonantes, vocales y longitud de la palabra.
*/

const palabra = prompt('Introduce una palabra').toLowerCase()

let consonantes=0
let vocales = 0

for(const letra of palabra){
  if(letra == 'a' || letra == 'e' || letra == 'i' || letra ==
'o' || letra == 'u') vocales++
  else consonantes++
}

console.log(`Tu palabra tiene ${vocales} vocales, ${consonantes}
consonantes y tiene un total de ${palabra.length} letras.`);

```

```

10 - Dado un array que contiene ["azul", "amarillo", "rojo",
"verde", "rosa"] determinar si un color introducido por el
usuario a través de un prompt se encuentra dentro del array o
*/

const colors = ["azul", "amarillo", "rojo", "verde", "rosa"]

const color = prompt('Introduce un color').toLowerCase()

if(colors.indexOf(color) !== -1){
  console.log('Tu color se encuentra en el array');
}else{
  console.log('Tu color no está definido');
}

```