**- \*\*Externo\*\*: En un archivo separado con extensión `.js` y enlazado en el HTML.**

html

<!DOCTYPE html>

<html>

<head>

<title>Mi página</title>

</head>

<body>

<h1>Hola Mundo</h1>

<script src="mi-script.js"></script>

</body>

</html>

**2. \*\*Sintaxis básica\*\***

JavaScript es un lenguaje de programación de alto nivel, interpretado y orientado a objetos.

**Variables**

Puedes declarar variables usando **`var`, `let` o `const`.**

let nombre = "Juan"; // Variable que puede cambiar

const edad = 25; // Constante, no puede cambiar

var ciudad = "Madrid"; // Antigua forma de declarar variables (menos recomendada)

**Tipos de datos**

JavaScript tiene varios tipos de datos:

**- \*\*String\*\*: Cadenas de texto.**

**- \*\*Number\*\*: Números.**

**- \*\*Boolean\*\*: `true` o `false`.**

**- \*\*Array\*\*: Listas.**

**- \*\*Object\*\*: Objetos.**

**- \*\*null\*\* y \*\*undefined\*\*: Valores especiales.**

let nombre = "Ana"; // String

let edad = 30; // Number

let esEstudiante = true; // Boolean

let hobbies = ["leer", "correr", "programar"]; // Array

let persona = { nombre: "Ana", edad: 30 }; // Object

**Funciones**

Las funciones son bloques de código que puedes llamar para ejecutar una tarea específica.

function saludar(nombre) {

return "Hola, " + nombre;

}

let saludo = saludar("Carlos");

console.log(saludo); // "Hola, Carlos"

**Condicionales**

Puedes usar `if`, `else if` y `else` para tomar decisiones.

let edad = 18;

if (edad >= 18) {

console.log("Eres mayor de edad");

} else {

console.log("Eres menor de edad");

}

**Bucles**

Los bucles te permiten repetir código.

**- \*\*for\*\*: Para iterar un número específico de veces.**

**for (let i = 0; i < 5; i++) {**

**console.log("Iteración: " + i);**

**}**

**- \*\*while\*\*: Para iterar mientras una condición sea verdadera.**

let i = 0;

while (i < 5) {

console.log("Iteración: " + i);

i++;

}

**3. \*\*Manipulación del DOM\*\***

El DOM (Document Object Model) es una representación del HTML que puedes manipular con JavaScript.

**Seleccionar elementos**

let titulo = document.getElementById("titulo"); // Selecciona por ID

let parrafos = document.getElementsByClassName("parrafo"); // Selecciona por clase

let boton = document.querySelector("button"); // Selecciona el primer elemento que coincida

**Cambiar contenido**

titulo.textContent = "Nuevo título"; // Cambia el texto del título

titulo.innerHTML = "<strong>Nuevo título</strong>"; // Cambia el HTML interno

**Eventos**

Puedes escuchar eventos como clics, movimientos del ratón, etc.

boton.addEventListener("click", function() {

alert("Botón clickeado!");

});

**4. \*\*Ejemplo completo\*\***

Aquí tienes un ejemplo completo que combina todo lo anterior:

html

<!DOCTYPE html>

<html>

<head>

<title>Mi página</title>

</head>

<body>

<h1 id="titulo">Hola Mundo</h1>

<button id="boton">Haz clic</button>

<script>

let titulo = document.getElementById("titulo");

let boton = document.getElementById("boton");

boton.addEventListener("click", function() {

titulo.textContent = "¡Botón clickeado!";

});

</script>

</body>

</html>

**3. Diferencias entre tipos primitivos y objetos**

**Primitivos**: Son inmutables y se almacenan por valor. Cuando asignas un primitivo a otra variable, se copia su valor.

let a = 10;

let b = a; // b es una copia de a

b = 20;

console.log(a); // 10 (a no cambia)

**Objetos:** Son mutables y se almacenan por referencia. Cuando asignas un objeto a otra variable, ambas apuntan al mismo objeto en memoria.

let obj1 = { nombre: "Juan" };

let obj2 = obj1; // obj2 apunta al mismo objeto que obj1

obj2.nombre = "Ana";

console.log(obj1.nombre); // "Ana" (obj1 también cambia)

**5. \*\*Recursos para aprender más\*\***

- \*\*MDN Web Docs\*\*: [JavaScript Guide](https://developer.mozilla.org/es/docs/Web/JavaScript/Guide)

- \*\*freeCodeCamp\*\*: [JavaScript Algorithms and Data Structures](https://www.freecodecamp.org/learn/javascript-algorithms-and-data-structures/)

- \*\*JavaScript.info\*\*: [The Modern JavaScript Tutorial](https://javascript.info/)