

Javascript

|                         |  |  |
|-------------------------|--|--|
| Características básicas | Interpretado   | No requiere compilación previa, se ejecuta directamente por el navegador.<br><br>console.log("Hola mundo");  |
|                         |  |  |
|                         |  |  |
|                         | Dinámico   | No se necesita declarar el tipo de dato; este se asigna en tiempo de ejecución.<br><br>let x = 5; x = "texto";   |
|                         |  |  |
|                         |  |  |
|                         | Multiparadigma   | Permite distintos estilos de programación: Imperativo (con instrucciones secuenciales), Funcional (uso de funciones como ciudadanos de primera clase), Orientado a objetos (mediante prototipos o clases). |
|                         |  |  |
|                         |  |  |
| Estructuras de control  | Condicionales  | if, else, else if, switch  |
|                         |  | if (edad >= 18) { console.log("Adulto"); } else {console.log("Menor de edad"); }   |
|                         | Bucles   | for, while, do...while, for...of, for...in   |
|                         |  | for (let i = 0; i < 5; i++) { console.log(i); }  |
| Tipos de datos          | Primitivos   | number, string, boolean, null, undefined, symbol, bigint   |
|                         |  | let nombre = "Luis"; // string<br>let edad = 22; // number   |
|                         | Referencias  | object, array, function  |
|                         |  | let persona = { nombre: "Luis", edad: 30 };<br>let frutas = ["manzana", "pera"];   |
| Funciones               | Declaradas   | function saludar() { console.log("Hola"); }  |
|                         |  |  |
|                         | Expresadas   | const saludar = function() { console.log("Hola"); };   |
|                         |  |  |
| Objetos y Arrays        | Flechas  | const saludar = () => console.log("Hola");   |
|                         |  |  |
|                         | Objetos  | let auto = { marca: "Toyota", modelo: 2023, arrancar: function() { return "Arrancando..."; } };  |
|                         |  |  |
| Arrays                  | Arrays   | let colores = ["rojo", "verde", "azul"];<br>console.log(colores[1]); // verde  |
|                         |  |  |
| Eventos del DOM         | DOM (Document Object Model):   | Representación estructurada del documento HTML que permite modificar contenido desde JavaScript.   |
|                         |  |  |
|                         | Eventos:   | document.getElementById("btn").addEventListener("click", () => { alert("¡Hola!"); });  |
|                         |  |  |
| Asincronia              | Callbacks  | Funciones que se ejecutan después de otra.<br><br>setTimeout(() => { console.log("Pasaron 2 segundos"); }, 2000);  |
|                         |  |  |
|                         | Promesas   | Manejan operaciones que pueden completarse en el futuro.<br><br>fetch("https://api.com").then(res => res.json()).then(data => console.log(data));  |
|                         |  |  |
| Async/Await:            | Sintaxis moderna y más legible para trabajar con promesas.   |  |
|                         |  |  |
|                         | async function cargarDatos() { const res = await fetch("https://api.com"); const data = await res.json(); console.log(data); } |  |
|                         |  |  |