

Tutorial de JavaScript Básico

Una guía completa para aprender los fundamentos de JavaScript necesarios para programar con confianza.

Tabla de Contenidos

1. [Introducción a JavaScript](#)
 2. [Variables y Tipos de Datos](#)
 3. [Operadores](#)
 4. [Funciones](#)
 5. [Estructuras de Control](#)
 6. [Arrays \(Arreglos\)](#)
 7. [Objetos](#)
 8. [Strings \(Cadenas de Texto\)](#)
 9. [La Consola del Navegador](#)
 10. [Conceptos Útiles](#)
-

Introducción a JavaScript

JavaScript es un lenguaje de programación que se ejecuta en navegadores web y servidores. Es el lenguaje que hace que las páginas web sean interactivas y dinámicas.

¿Dónde escribir JavaScript?

1. Consola del navegador: Presiona F12 → Pestaña "Console"
2. Archivo HTML: Entre etiquetas `<script> </script>`
3. Archivo .js externo: Vinculado al HTML

Tu primera línea de código

```
javascript  
  
console.log("¡Hola, mundo!");
```

Variables y Tipos de Datos

Las variables son contenedores que almacenan datos. En JavaScript moderno usamos `let` y `const`.

Declarando Variables

```
javascript  
  
let nombre = "María";           // Variable que puede cambiar  
const edad = 25;                // Variable constante (no cambia)  
let ciudad;                     // Variable sin valor inicial (undefined)
```

Reglas para nombres de variables:

- Empiezan con letra, _ o \$
- No pueden empezar con números

- No usar palabras reservadas (let, const, function, etc.)
- Usar camelCase: `miVariable`, `edadUsuario`

Tipos de Datos Primitivos

javascript

// String (cadena de texto)

```
let nombre = "Carlos";
```

```
let mensaje = 'Hola mundo';
```

```
let template = `Mi nombre es ${nombre}`; // Template string
```

// Number (número)

```
let edad = 30;
```

```
let precio = 19.99;
```

```
let negativo = -15;
```

// Boolean (verdadero/falso)

```
let esEstudiante = true;
```

```
let esMayor = false;
```

// Undefined (sin definir)

```
let valor; // undefined
```

// Null (nulo)

```
let dato = null;
```

Verificar Tipos de Datos

javascript

```
let texto = "Hola";  
console.log(typeof texto); // "string"  
  
let numero = 42;  
console.log(typeof numero); // "number"  
  
let booleano = true;  
console.log(typeof booleano); // "boolean"
```

Operadores

Operadores Aritméticos

javascript

```
let a = 10;  
let b = 3;  
  
console.log(a + b); // 13 - Suma  
console.log(a - b); // 7 - Resta  
console.log(a * b); // 30 - Multiplicación  
console.log(a / b); // 3.333... - División  
console.log(a % b); // 1 - Módulo (resto de la división)  
console.log(a ** b); // 1000 - Exponente
```

Operadores de Asignación

javascript

```
let x = 5;
```

```
x += 3; // x = x + 3 → x vale 8
```

```
x -= 2; // x = x - 2 → x vale 6
```

```
x *= 2; // x = x * 2 → x vale 12
```

```
x /= 3; // x = x / 3 → x vale 4
```

Operadores de Comparación

javascript

```
let a = 5;
```

```
let b = "5";
```

```
console.log(a == b); // true - Igualdad (convierte tipos)
```

```
console.log(a === b); // false - Igualdad estricta
```

```
console.log(a != b); // false - Diferente
```

```
console.log(a !== b); // true - Diferente estricto
```

```
console.log(a > 3); // true - Mayor que
```

```
console.log(a < 10); // true - Menor que
```

```
console.log(a >= 5); // true - Mayor o igual
```

```
console.log(a <= 5); // true - Menor o igual
```

Operadores Lógicos

javascript

```
let edad = 25;
let tieneLicencia = true;

// AND (&&) - Ambas condiciones deben ser verdaderas
if (edad >= 18 && tieneLicencia) {
  console.log("Puede conducir");
}

// OR (||) - Al menos una condición debe ser verdadera
if (edad < 18 || !tieneLicencia) {
  console.log("No puede conducir");
}

// NOT (!) - Invierte el valor booleano
console.log(!true); // false
console.log(!false); // true
```

Funciones

Las funciones son bloques de código reutilizable que realizan tareas específicas.

Declaración de Funciones

```
javascript
```

// Función básica

```
function saludar() {  
  console.log("¡Hola!");  
}
```

// Llamar la función

```
saludar(); // Imprime: ¡Hola!
```

// Función con parámetros

```
function saludarPersona(nombre) {  
  console.log("¡Hola, " + nombre + "!");  
}
```

```
saludarPersona("Ana"); // Imprime: ¡Hola, Ana!
```

// Función que retorna un valor

```
function sumar(a, b) {  
  return a + b;  
}
```

```
let resultado = sumar(5, 3);  
console.log(resultado); // 8
```

Funciones Arrow (Flecha)

javascript

// Función arrow básica

```
const saludar = () => {  
  console.log("¡Hola!");  
};
```

// Con parámetros

```
const saludarPersona = (nombre) => {  
  console.log(`¡Hola, ${nombre}!`);  
};
```

// Función arrow con return implícito

```
const sumar = (a, b) => a + b;
```

// Si solo hay un parámetro, no necesitas paréntesis

```
const cuadrado = x => x * x;
```

Scope (Alcance) de Variables

javascript

```
let variableGlobal = "Soy global";
```

```
function miFuncion() {  
  let variableLocal = "Soy local";  
  console.log(variableGlobal); // Accede a la variable global  
  console.log(variableLocal); // Accede a la variable local  
}
```

```
miFuncion();
```

```
// console.log(variableLocal); // ERROR: no está definida aquí
```

Estructuras de Control

Condicionales - if, else if, else

javascript

```
let edad = 20;
```

```
if (edad < 13) {  
    console.log("Eres un niño");  
} else if (edad < 18) {  
    console.log("Eres un adolescente");  
} else if (edad < 65) {  
    console.log("Eres un adulto");  
} else {  
    console.log("Eres un adulto mayor");  
}
```

// Condicional ternario (operador ?)

```
let mensaje = edad >= 18 ? "Eres mayor de edad" : "Eres menor de edad";
```

Switch

javascript

```
let dia = 3;
let nombreDia;

switch (dia) {
  case 1:
    nombreDia = "Lunes";
    break;
  case 2:
    nombreDia = "Martes";
    break;
  case 3:
    nombreDia = "Miércoles";
    break;
  case 4:
    nombreDia = "Jueves";
    break;
  case 5:
    nombreDia = "Viernes";
    break;
  default:
    nombreDia = "Fin de semana";
}

console.log(nombreDia); // "Miércoles"
```

Bucles

Bucle for

javascript

// For básico

```
for (let i = 0; i < 5; i++) {  
  console.log("Número: " + i);  
}
```

// Imprime: 0, 1, 2, 3, 4

// For con diferentes incrementos

```
for (let i = 0; i <= 10; i += 2) {  
  console.log(i); // 0, 2, 4, 6, 8, 10  
}
```

// For hacia atrás

```
for (let i = 5; i > 0; i--) {  
  console.log(i); // 5, 4, 3, 2, 1  
}
```

Bucle while

javascript

```
let contador = 0;

while (contador < 3) {
  console.log("Contador: " + contador);
  contador++;
}
// Imprime: 0, 1, 2

// Do-while (se ejecuta al menos una vez)
let numero = 0;
do {
  console.log("Número: " + numero);
  numero++;
} while (numero < 3);
```

Arrays (Arreglos)

Los arrays almacenan múltiples valores en una sola variable.

Crear y Acceder a Arrays

```
javascript
```

// Crear arrays

```
let frutas = ["manzana", "banana", "naranja"];
```

```
let numeros = [1, 2, 3, 4, 5];
```

```
let mixto = ["texto", 42, true, null];
```

// Acceder a elementos (índice comienza en 0)

```
console.log(frutas[0]); // "manzana"
```

```
console.log(frutas[1]); // "banana"
```

```
console.log(frutas[2]); // "naranja"
```

// Longitud del array

```
console.log(frutas.length); // 3
```

// Cambiar un elemento

```
frutas[1] = "pera";
```

```
console.log(frutas); // ["manzana", "pera", "naranja"]
```

Métodos de Arrays

javascript

```
let colores = ["rojo", "verde", "azul"];
```

// Agregar elementos

```
colores.push("amarillo"); // Agrega al final
```

```
console.log(colores); // ["rojo", "verde", "azul", "amarillo"]
```

```
colores.unshift("negro"); // Agrega al inicio
```

```
console.log(colores); // ["negro", "rojo", "verde", "azul", "amarillo"]
```

// Eliminar elementos

```
let ultimo = colores.pop(); // Elimina el último
```

```
console.log(ultimo); // "amarillo"
```

```
let primero = colores.shift(); // Elimina el primero
```

```
console.log(primero); // "negro"
```

// Buscar elementos

```
let indice = colores.indexOf("verde");
```

```
console.log(indice); // 1
```

```
let existe = colores.includes("rojo");
```

```
console.log(existe); // true
```

Recorrer Arrays

javascript

```
let animales = ["perro", "gato", "pájaro"];
```

```
// Con for tradicional
```

```
for (let i = 0; i < animales.length; i++) {  
    console.log(animales[i]);  
}
```

```
// Con for...of (más moderno)
```

```
for (let animal of animales) {  
    console.log(animal);  
}
```

```
// Con forEach
```

```
animales.forEach(function(animal, indice) {  
    console.log(indice + ": " + animal);  
});
```

Objetos

Los objetos almacenan datos en pares clave-valor.

Crear y Usar Objetos

```
javascript
```

// Crear un objeto

```
let persona = {  
  nombre: "Carlos",  
  edad: 30,  
  ciudad: "Madrid",  
  esEstudiante: false  
};
```

// Acceder a propiedades

```
console.log(persona.nombre); // "Carlos"  
console.log(persona["edad"]); // 30
```

// Modificar propiedades

```
persona.edad = 31;  
persona.ciudad = "Barcelona";
```

// Agregar nuevas propiedades

```
persona.profesion = "Programador";  
persona.saludar = function() {  
  console.log("Hola, soy " + this.nombre);  
};
```

// Llamar método del objeto

```
persona.saludar(); // "Hola, soy Carlos"
```

Objetos Anidados

javascript


```
let estudiante = {
  nombre: "Ana",
  edad: 22,
  direccion: {
    calle: "Gran Vía 123",
    ciudad: "Madrid",
    codigoPostal: "28013"
  },
  materias: ["Matemáticas", "Física", "Química"],
  nota: {
    matematicas: 8.5,
    fisica: 7.8,
    quimica: 9.2
  }
};

// Acceder a propiedades anidadas
console.log(estudiante.direccion.ciudad); // "Madrid"
console.log(estudiante.materias[0]);      // "Matemáticas"
console.log(estudiante.nota.matematicas); // 8.5
```

Recorrer Objetos

javascript

```
let producto = {  
  nombre: "Laptop",  
  precio: 899,  
  marca: "TechBrand",  
  disponible: true  
};  
  
// Recorrer propiedades con for...in  
for (let propiedad in producto) {  
  console.log(propiedad + ": " + producto[propiedad]);  
}  
  
// Obtener todas las claves  
let claves = Object.keys(producto);  
console.log(claves); // ["nombre", "precio", "marca", "disponible"]  
  
// Obtener todos los valores  
let valores = Object.values(producto);  
console.log(valores); // ["Laptop", 899, "TechBrand", true]
```

Strings (Cadenas de Texto)

Métodos Básicos de Strings

javascript

```
let texto = "Hola Mundo JavaScript";

// Longitud
console.log(texto.length); // 21

// Mayúsculas y minúsculas
console.log(texto.toUpperCase()); // "HOLA MUNDO JAVASCRIPT"
console.log(texto.toLowerCase()); // "hola mundo javascript"

// Buscar texto
console.log(texto.indexOf("Mundo")); // 5
console.log(texto.includes("Java")); // true
console.log(texto.startsWith("Hola")); // true
console.log(texto.endsWith("Script")); // true

// Extraer partes del texto
console.log(texto.slice(0, 4)); // "Hola"
console.log(texto.substring(5, 10)); // "Mundo"

// Dividir string en array
let palabras = texto.split(" ");
console.log(palabras); // ["Hola", "Mundo", "JavaScript"]

// Reemplazar texto
let nuevoTexto = texto.replace("Mundo", "Universo");
console.log(nuevoTexto); // "Hola Universo JavaScript"
```

Template Strings

```
javascript
```

```
let nombre = "María";
let edad = 28;

// Concatenación tradicional
let mensaje1 = "Hola, soy " + nombre + " y tengo " + edad + " años";

// Template string (más moderno y legible)
let mensaje2 = `Hola, soy ${nombre} y tengo ${edad} años`;

// Template strings multilínea
let html = `
  <div>
    <h1>${nombre}</h1>
    <p>Edad: ${edad}</p>
  </div>
`;
```

La Consola del Navegador

console.log() y Variaciones

javascript

// Mostrar valores

```
console.log("Hola mundo");  
console.log("El resultado es:", 5 + 3);
```

// Mostrar múltiples valores

```
let nombre = "Ana";  
let edad = 25;  
console.log("Nombre:", nombre, "Edad:", edad);
```

// Otras funciones de consola

```
console.warn("Esto es una advertencia");  
console.error("Esto es un error");
```

// Mostrar objetos y arrays de forma legible

```
let persona = {nombre: "Carlos", edad: 30};  
console.log(persona);
```

```
let numeros = [1, 2, 3, 4, 5];  
console.log(numeros);
```

Depuración (Debugging)

javascript

```
function calcular(a, b) {  
  console.log("Entrada:", a, b); // Ver qué valores llegan  
  
  let resultado = a + b;  
  console.log("Resultado:", resultado); // Ver el resultado  
  
  return resultado;  
}  
  
let suma = calcular(5, 3);  
console.log("Final:", suma);
```

Conceptos Útiles

Math Object

javascript

// Números aleatorios

```
console.log(Math.random());           // Entre 0 y 1  
console.log(Math.random() * 10);       // Entre 0 y 10  
console.log(Math.floor(Math.random() * 10)); // Entero entre 0 y 9
```

// Redondeo

```
console.log(Math.round(4.7)); // 5  
console.log(Math.floor(4.7)); // 4  
console.log(Math.ceil(4.1));  // 5
```

// Otros métodos útiles

```
console.log(Math.max(1, 5, 3, 9, 2)); // 9  
console.log(Math.min(1, 5, 3, 9, 2)); // 1  
console.log(Math.abs(-5));             // 5  
console.log(Math.pow(2, 3));           // 8  
console.log(Math.sqrt(16));            // 4
```

Conversión de Tipos

javascript

// String a Number

```
let texto = "123";  
let numero1 = Number(texto);    // 123  
let numero2 = parseInt(texto);  // 123  
let numero3 = parseFloat("12.34"); // 12.34
```

// Number a String

```
let numero = 456;  
let texto1 = String(numero);    // "456"  
let texto2 = numero.toString(); // "456"
```

// Boolean a otros tipos

```
let verdadero = true;  
let falso = false;  
console.log(Number(verdadero)); // 1  
console.log(Number(falso));     // 0  
console.log(String(verdadero)); // "true"
```

Manejo de Errores Básico

javascript


```
function dividir(a, b) {  
  if (b === 0) {  
    console.error("Error: No se puede dividir por cero");  
    return null;  
  }  
  return a / b;  
}
```

```
let resultado = dividir(10, 0);  
if (resultado !== null) {  
  console.log("El resultado es:", resultado);  
}
```

Consejos y Buenas Prácticas

Nomenclatura

javascript

//  Buenas prácticas

let nombreUsuario = "Carlos"; // camelCase

const MAXIMO_INTENTOS = 3; // Constantes en MAYÚSCULAS

let esValido = true; // Nombres descriptivos

//  Evitar

let x = "Carlos"; // Nombre poco descriptivo

let nombre_usuario = "Carlos"; // snake_case (no común en JS)

Comentarios

javascript

// Comentario de una línea

let edad = 25;

/*

Comentario de
múltiples líneas

*/

function calcularImpuesto(precio) {

// Impuesto del 21%

return precio * 0.21;

}

Estructura del Código

javascript

// Agrupar variables relacionadas

```
let nombre = "Ana";
```

```
let apellido = "García";
```

```
let edad = 28;
```

// Separar funciones con líneas vacías

```
function saludar(nombre) {  
    return `Hola, ${nombre}!`;  
}
```

```
function despedir(nombre) {  
    return `Adiós, ${nombre}!`;  
}
```

// Usar indentación consistente

```
if (edad >= 18) {  
    console.log("Mayor de edad");  
    if (edad >= 65) {  
        console.log("Adulto mayor");  
    }  
}
```

Próximos Pasos

Ahora que conoces los fundamentos de JavaScript, estás listo para:

1. **Practicar con los ejercicios** - Aplica estos conceptos resolviendo problemas
2. **Explorar el DOM** - Manipular elementos HTML con JavaScript

3. **Aprender sobre eventos** - Hacer que tu código responda a clicks y otras interacciones
4. **Estudiar programación asíncrona** - Trabajar con APIs y datos externos

¡Continúa practicando y experimentando! La programación se aprende haciendo. 🚀