Introducción a JS

Variables, estructuras de control y manejo de la consola

Javier Ribal del Río

2025-10-11

Table of contents

Contenido

- Funcionamiento de JavaScript
- Variables y operadores
- Condicionales `if` y `switch`
- Bucles `for` e `while`

1 Introducción a JS

1.1 ¿Qué es JavaScript?

- Es el lenguaje de programación del navegador.
- Creado en 1995 por Brendan Eich
- Permite crear comportamientos dinámicos: responder a clics, validar formularios, mostrar u ocultar elementos, generar contenido nuevo, etc.
- Se ejecuta directamente en el navegador, sin necesidad de instalación.
- Junto con HTML y CSS forma el trinomio esencial del desarrollo web

La siguiente tabla recuerda las difrencias entre HTML, CSS y JS:

Capa	Lenguaje	Función principal
Contenido	HTML	Estructura del documento
Presentación	CSS	Apariencia y diseño
Comportamiento	JavaScript	Interactividad y lógica

2 Programación en JS

2.1 Características generales de JS

- Tipado dinámico
- Tipado débil **Coerción Implicita**
- Multiparadigma
- Ámbito léxico (estático)
- Modelo basado en prototipos (polimorfismo de inclusión por composición)
- Funciones de primera clase
- Funciones de orden superior
- Closures / Copia viva
- Reflexión (`Reflect`, `Proxy`, descriptores)
- Metaprogramación
- Sin Polimorfismo Ad-hoc de sobrecarga, ni Polimorfismo universal genérico
- Ejecución single-threaded
- Asincronía no bloqueante
- Promises / async–await
- Garbage collection
- Interoperabilidad con JSON

2.2 Tipos

2.2.1 Primitivos

- **string** — cadenas de texto
- **number** — números (enteros y decimales)
- **boolean** — verdadero o falso
- **null** — ausencia intencional de valor
- **undefined** — valor no inicializado
- **symbol**
- **bigint**

2.2.2 Tipos de referencia

- **Object**
- **Array**

- **Func tion**
- **Map / Set**
- **Date**

2.3 Características del sistema de tipos

- **Tipado dinámico**
- **Tipado débil**
- Los primitivos se copian **por valor**
- Los objetos se copian **por referencia**

2.4 Variables

2.4.1 `let`

- Tipado: dinámico
- Alcance: estático
- Mutables
- Hoisting
- No globales
- Shadowing permitido

2.4.2 `const`

No son verdaderamente constantes

- Tipado: dinámico
- Alcance: estático
- Inmutables en la referencia (no permiten reasignación)
- Hoisting
- No globales
- **Shadowing permitido**

2.4.3 var

No se usa

- Tipado: dinámico
- Alcance: de función (no de bloque)
- Mutables
- Hoisting (inicializadas como undefined)
- Globales si se declaran en el ámbito global (propiedad de window/globalThis)
- Shadowing permitido, pero con reglas inconsistentes (puede causar illegal shadowing en combinación con let/const)