Curso: JavaScript desde Cero

Duración estimada: 6 a 8 semanas

Modalidad: 100% online, clases grabadas + sesiones en vivo (opcional)

Nivel: Principiante

# Módulo 1: Introducción al mundo de la programación y JavaScript

* ¿Qué es JavaScript?
* Historia y evolución del lenguaje
* Cómo se usa JavaScript en el desarrollo web
* Herramientas necesarias (navegador, editor de código, consola)
* Primer programa: console.log("Hola mundo")

# Módulo 2: Fundamentos del lenguaje

* Variables (var, let, const)
* Tipos de datos: string, number, boolean, null, undefined
* Operadores: aritméticos, lógicos, comparación
* Comentarios en el código

# Módulo 3: Estructuras de control

* Sentencias if, else, else if
* Operador ternario
* Sentencia switch
* Ciclos: for, while, do...while

# Módulo 4: Funciones y alcance

* Declaración y uso de funciones
* Parámetros y retorno
* Funciones flecha (arrow functions)
* Scope: global vs local

# Módulo 5: Arreglos y objetos

* Creación y manipulación de arrays
* Métodos comunes (push, pop, shift, unshift, splice, etc.)
* Bucles con arrays: for, forEach, map
* Objetos: creación, propiedades, métodos

# Módulo 6: Programación orientada a eventos y DOM

* ¿Qué es el DOM?
* Selección de elementos del DOM
* Manejo de eventos (click, input, submit, etc.)
* Manipulación del DOM (texto, atributos, clases, estilos)

# Módulo 7: Buenas prácticas y herramientas modernas

* Introducción a la depuración (uso de console, herramientas del navegador)
* Buenas prácticas de nombrado y estructura
* Código limpio y comentarios útiles
* Introducción a ES6+: destructuración, spread/rest, templates literales

# Módulo 8: Proyecto final práctico

* Desarrollar una pequeña aplicación interactiva (ej.: to-do list, calculadora, trivia)
* Aplicación de todo lo aprendido: eventos, funciones, DOM, objetos, arrays
* Revisión y retroalimentación

# Extras (opcional, según duración del curso)

* Introducción a JSON y fetch API
* Guardar datos en el navegador (localStorage)
* Qué es Node.js y cómo se relaciona con JavaScript