



Yeies es una iniciativa educativa que va más allá de un curso de JavaScript. Nuestro objetivo es ofrecer una formación integral en desarrollo web y software, comenzando con JavaScript y expandiéndonos a otros cursos clave. Próximamente, incorporaremos programas dedicados a frameworks del frontend, del backend, bases de datos y mucho más. Con esta evolución, buscamos proporcionar un aprendizaje progresivo y completo, preparándote para dominar cada aspecto del desarrollo moderno.

## Programa completo: Curso de JavaScript con Enfoque en Desarrollo Profesional

Este programa está diseñado para formar desarrolladores altamente capacitados en JavaScript, combinando una sólida base teórica con un fuerte enfoque en la práctica. A lo largo del curso trabajarán en numerosos ejercicios, proyectos innovadores y casos reales, además de desarrollar habilidades en lógica de programación para enfrentar con éxito los desafíos del desarrollo moderno.

**Duración:** 9 meses (36 semanas)

**Modalidad:** Sesiones sincrónicas de 2 horas semanales, complementadas con material didáctico avanzado.

**Enfoque:** Desarrollo de lógica computacional, fundamentos teórico-prácticos robustos y aplicación en proyectos reales.

### Módulo 1: Introducción a JavaScript y Fundamentos de Programación

- **Duración:** 4 semanas
- **Objetivo:** Establecer una base sólida en los principios de JavaScript y la lógica algorítmica.
- **Contenido Técnico:**
  - **Contextualización de JavaScript:**
    - Evolución histórica y relevancia en el ecosistema de desarrollo moderno.
    - Comparativa con lenguajes como Python, Java y TypeScript.
    - Entornos de ejecución: Browser Runtime vs. Node.js Runtime.
    - Configuración de entornos de desarrollo: Visual Studio Code, Chrome DevTools, y debugging básico.
  - **Sintaxis y Semántica:**

- Estructura léxica de un programa en JavaScript.
- Convenciones de escritura: comentarios, estilo de código (linting) y buenas prácticas.
- Uso de console.log para trazabilidad y depuración.
- **Variables y Tipos de Datos:**
  - Declaración y scope: let, const, var.
  - Tipos primitivos: string, number, boolean, null, undefined.
  - Coerción de tipos y conversiones explícitas.
- **Operadores y Expresiones:**
  - Aritméticos: +, -, \*, /, %, \*\*.
  - Comparación: ==, ===, !=, !==, >, <, >=, <=.
  - Lógicos: &&, ||, !.
  - Operador ternario: condición ? expr1 : expr2.
- **Estructuras de Control de Flujo:**
  - Condicionales: if, else, else if.
  - Selección múltiple: switch-case.
  - Bucles: while, do while, for.

## Módulo 2: Funciones y Estructuras de Datos Básicas

- **Duración:** 4 semanas
- **Objetivo:** Dominar el paradigma funcional y la manipulación de estructuras de datos fundamentales.
- **Contenido Técnico:**
  - **Funciones:**
    - Declaración de funciones: function declaration vs. function expression.
    - Parámetros formales y argumentos reales.
    - Valores de retorno y el concepto de return.
    - Scope: ámbito léxico y hoisting.
  - **Funciones Avanzadas:**
    - Funciones anónimas y su aplicación en callbacks.
    - Arrow functions: sintaxis concisa y diferencias en el contexto de this.
    - Callbacks: patrones de programación asíncrona y manejo de flujos no bloqueantes.
  - **Arrays:**
    - Creación y manipulación de arrays.
    - Métodos esenciales: push, pop, shift, unshift, slice, splice.
    - Iteración: for, forEach, map.
  - **Objetos:**
    - Creación de objetos literales.
    - Acceso y manipulación de propiedades: notación de punto y corchetes.
    - Métodos de objetos y el contexto de this.

## Módulo 3: Programación Orientada a Objetos (POO)

- **Duración:** 4 semanas
- **Objetivo:** Implementar los principios de la POO en JavaScript.
- **Contenido Técnico:**
  - **Conceptos Fundamentales de POO:**
    - Clases y objetos: abstracción y encapsulación.
    - Propiedades y métodos: estado y comportamiento.
    - Instanciación: uso de new y constructores.
  - **Herencia y Prototipos:**
    - Modelo de prototipos en JavaScript.
    - Cadena de prototipos (prototype chain).
    - Métodos estáticos y de instancia.
  - **Encapsulación y Abstracción:**
    - Uso de closures para encapsulación.
    - Getters y setters: control de acceso a propiedades.
  - **Polimorfismo:**
    - Sobrescritura de métodos.
    - Implementación de interfaces mediante patrones de diseño.

## Módulo 4: Manipulación del DOM y Eventos

- **Duración:** 4 semanas
- **Objetivo:** Interactuar dinámicamente con el DOM y gestionar eventos.
- **Contenido Técnico:**
  - **Introducción al DOM:**
    - Estructura jerárquica del DOM.
    - Selección de elementos: querySelector, getElementById.
  - **Modificación del DOM:**
    - Manipulación de contenido: innerHTML, textContent.
    - Estilos: style y classList.
    - Creación y eliminación de nodos: createElement, appendChild, removeChild.
  - **Eventos:**
    - Tipos de eventos: click, mouseover, keydown, etc.
    - Manejo de eventos: addEventListener.
    - Propagación y delegación de eventos.

## Módulo 5: Programación Funcional y Métodos Avanzados

- **Duración:** 4 semanas
- **Objetivo:** Dominar técnicas de programación funcional y métodos avanzados de manipulación de datos.
- **Contenido Técnico:**
  - **Funciones de Orden Superior:**
    - Funciones como argumentos y valores de retorno.
  - **Métodos de Arrays:**
    - map, filter, reduce, forEach.
    - Aplicaciones prácticas en transformación de datos.
  - **Closures:**
    - Concepto y aplicaciones en encapsulación y factory functions.
  - **Recursividad:**
    - Concepto y casos de uso.
    - Ejercicios prácticos con recursión.

## Módulo 6: Asincronía y Manejo de Promesas

- **Duración:** 4 semanas
- **Objetivo:** Gestionar operaciones asíncronas y flujos no bloqueantes.
- **Contenido Técnico:**
  - **Introducción a la Asincronía:**
    - Callbacks y el problema del "callback hell".
    - Promesas: creación, encadenamiento y manejo de errores.
  - **Async/Await:**
    - Sintaxis y uso en flujos asíncronos.
    - Manejo de errores con try/catch.
  - **Peticiones HTTP:**
    - Uso de fetch y axios.
    - Manejo de respuestas y errores.

## Módulo 7: Almacenamiento y APIs Externas

- **Duración:** 4 semanas
- **Objetivo:** Almacenar datos en el cliente y consumir APIs externas.
- **Contenido Técnico:**
  - **Almacenamiento en el Navegador:**
    - localStorage y sessionStorage.
    - Limitaciones y buenas prácticas.
  - **Consumo de APIs:**

- Introducción a REST y JSON.
- Autenticación con tokens (JWT).

## Módulo 8: Módulos y Herramientas Modernas

- **Duración:** 4 semanas
- **Objetivo:** Utilizar herramientas modernas para desarrollo profesional.
- **Contenido Técnico:**
  - **Módulos en JavaScript:**
    - import y export.
    - Uso de bundlers: Webpack, Parcel.
  - **Gestión de Dependencias:**
    - Introducción a npm.
    - Uso de paquetes externos.

## Módulo 9: Temas Avanzados y Optimización

- **Duración:** 4 semanas
- **Objetivo:** Profundizar en temas avanzados y optimización de código.
- **Contenido Técnico:**
  - **Web Workers:**
    - Concepto y uso en paralelización.
  - **Optimización de Rendimiento:**
    - Buenas prácticas y herramientas de profiling.
  - **Testing:**
    - Introducción a Jest y Mocha.
    - Pruebas unitarias y de integración.