

Fundamentos de JavaScript

Configuración de VS Code y extensiones recomendadas

Instalar extensiones para mejorar la experiencia de desarrollo en JavaScript.

Algunas extensiones recomendadas son :

- **JavaScript (ES6) code snippets**: Ayuda a escribir código más rápido con atajos.
- **Prettier**: Formatea el código para que sea más legible [1, 2]. Se puede configurar para que formatee automáticamente al guardar o usando Shift + Alt + F.
- **Bracket Pair Colorizer**: Colorea los pares de corchetes para facilitar la identificación de bloques de código.
- **Indent Rainbow**: Añade colores a la indentación para mejorar la legibilidad del código.
- **Quokka**: Proporciona un espacio para experimentar con código JavaScript y ver los resultados en tiempo real.
- **Live Server**: Permite visualizar los cambios en el navegador en tiempo real al guardar el archivo HTML.

¿Qué es JavaScript?

JavaScript es un lenguaje de programación interpretado, lo que significa que no necesita ser compilado. El navegador actúa como intérprete, ejecutando el código línea por línea.

Inicialmente, se creó para añadir interactividad a las páginas web.

Permite crear aplicaciones web interactivas donde el usuario puede interactuar con elementos como botones y formularios.

Hoy en día, JavaScript se utiliza para mucho más que solo interactividad web, incluyendo la creación de servidores web, aplicaciones móviles y de escritorio.

Diferentes navegadores interpretan JavaScript, pero Ecma se encarga de estandarizar el lenguaje, asegurando que las aplicaciones funcionen de manera similar en diferentes navegadores.

Las versiones estandarizadas más utilizadas son ES5 y ES6, siendo ES6 la más extendida actualmente.

¿Qué es Node.js?

Node.js es un entorno de ejecución para JavaScript que permite ejecutarlo fuera del navegador. Fue creado en 2009 por Ryan Dahl.

Permite utilizar JavaScript para desarrollar aplicaciones de servidor, herramientas de línea de comandos y aplicaciones de escritorio, entre otros.

Node.js es multiplataforma, lo que significa que el código puede ejecutarse en diferentes sistemas operativos como Windows, macOS y Linux.

¿Qué es npm (Node Package Manager)?

npm es el gestor de paquetes para Node.js.

Se utiliza para instalar y gestionar dependencias (módulos y librerías) en proyectos de Node.js.

npm facilita la reutilización de código y la gestión de dependencias en proyectos JavaScript.

Declaración de variables

JavaScript utiliza tipado inferido, lo que significa que no es necesario declarar el tipo de una variable. El tipo se determina en tiempo de ejecución.

Hay tres formas de declarar variables:

- **var**: Tiene un alcance de función o global.
- **let**: Tiene un alcance de bloque.
- **const**: Se utiliza para declarar constantes, que son variables cuyo valor no puede ser reasignado después de la inicialización.

Tipos de datos primitivos

- **Number**: Representa números enteros y decimales.
- **String**: Representa cadenas de texto.
- **Boolean**: Representa valores lógicos true o false.
- **Null**: Representa un valor nulo.
- **Undefined**: Representa una variable no definida.

Tipos de objetos

- **Objetos**: Colección de propiedades.

- **Arrays (listas):** Colección ordenada de elementos.
- **Fechas:** Representan un momento en el tiempo.

Notación en JavaScript

- **Punto y coma (;):** Se utiliza para delimitar el final de una línea.
Aunque es opcional, se recomienda usarlo para evitar errores.
- **Punto (.):** Se utiliza para acceder a las propiedades de un objeto.
- **Corchetes cuadrados ([]):** Se utilizan para definir y acceder a elementos de listas (arrays).
- **Paréntesis ():** Se utilizan en funciones, tanto para definir los parámetros como para invocar la función.
- **Llaves ({}):** Se utilizan para definir objetos, funciones y estructuras de control (como if, else, bucles).

Listas (Arrays)

Una lista (o array) es una colección ordenada de elementos.

Los arrays pueden contener elementos de diferentes tipos (números, strings, booleanos, objetos, etc.).

Se pueden declarar de dos maneras:

- Usando corchetes: `const lista = [1, "hola", true];`
- Usando el constructor Array: `const lista = new Array(1, "hola", true);`

Se accede a los elementos de un array mediante su índice, comenzando desde 0.

Objetos

Un objeto es una colección de propiedades, donde cada propiedad tiene un nombre (clave) y un valor.

Los objetos se definen usando llaves {}.

Se accede a las propiedades de un objeto utilizando el punto (.) o los corchetes [].

Los nombres de las propiedades pueden ser strings, pero si no son identificadores válidos (por ejemplo, si contienen espacios o caracteres especiales), deben ir entre comillas.

Fechas

JavaScript tiene un objeto Date para trabajar con fechas y horas.

Se pueden crear objetos Date para representar fechas y horas específicas, o para obtener la fecha y hora actual.

Existen librerías como Moment.js que facilitan el manejo de fechas en JavaScript.