



Universidad CENFOTEC

Curso: Fundamentos de Programación Web

Tarea #1

Nombre: Joshua Ulloa Mora

Profesor: Francisco José Jiménez Bonilla

## 1. Historia de JavaScript

JavaScript fue creado en 1995 por Brendan Eich mientras trabajaba en Netscape Communications. Originalmente se llamó Mocha, luego LiveScript, y finalmente JavaScript. Fue diseñado en solo 10 días para hacer las páginas web más dinámicas e interactivas.

Con el tiempo, se estandarizó bajo el nombre de ECMAScript (a partir de 1997) y ha evolucionado en versiones modernas que soportan programación funcional, orientada a objetos y asincronía.

## 2. ¿Qué es JavaScript?

JavaScript es un lenguaje de programación interpretado, dinámico y de alto nivel, usado principalmente para dar interactividad y dinamismo a las páginas web.

Se ejecuta en el navegador del usuario (aunque hoy también en el servidor con Node.js).

## 3. Relación entre HTML y JavaScript

- **HTML** define la estructura y el contenido de una página web (texto, imágenes, enlaces, formularios).
- **JavaScript** manipula esa estructura, permitiendo interactividad (validar formularios, animaciones, interacciones en tiempo real).

•

#### **4. Beneficios de usar Bootstrap con JavaScript**

Bootstrap es un framework CSS y JS que:

- Acelera el desarrollo de sitios responsivos y modernos.
- Incluye componentes interactivos listos (sliders, modals, menús desplegables) que usan JS.
  - Reduce errores y tiempo de codificación.
- Garantiza compatibilidad con distintos navegadores y dispositivos.

#### **5. Semejanzas y diferencias entre PHP y JavaScript**

Tanto PHP como JavaScript son lenguajes de programación utilizados en el desarrollo web, pero cumplen funciones distintas: JavaScript se ejecuta principalmente en el navegador del cliente para dar dinamismo e interactividad a las páginas, aunque también puede usarse en el servidor con Node.js, mientras que PHP se ejecuta únicamente en el servidor, siendo muy utilizado para manejar bases de datos, formularios y lógica del backend. Ambos permiten generar contenido dinámico, pero la diferencia principal radica en dónde se ejecutan y en el tipo de tareas para las que son más adecuados.

## 6. Formas de agregar JS en una página web

1. En línea (inline): dentro de un atributo HTML.
2. `<button onclick="alert('Hola!')">Click</button>`
3. En la misma página (interno): dentro de etiquetas `<script>`.
  4. `<script>`
  5. `console.log("Hola desde JS interno");`
  6. `</script>`
7. Archivo externo: enlazado con `src`.  
`<script src="app.js"></script>`

## 7. Función principal de la consola en JS

La consola sirve para:

- Depurar el código (ver errores).
- Mostrar mensajes con `console.log()`.
- Probar fragmentos de código en el navegador.

## 8. Diferencia entre var, let y const

En JavaScript, la diferencia entre `var`, `let` y `const` radica principalmente en su alcance y en si pueden ser reasignadas o redefinidas. La palabra clave **var** tiene alcance de función o global, permite reasignar y redefinir variables, aunque su uso ya casi no se recomienda. **let** tiene alcance de bloque, permite

reasignar valores pero no redefinir la variable dentro del mismo bloque, lo que la hace más segura y predecible. Por último, **const** también tiene alcance de bloque, pero a diferencia de **let**, no permite reasignar ni redefinir la variable, siendo útil para valores que deben mantenerse constantes durante la ejecución del programa.

## 9. Función de minificar archivos JavaScript

Minificar significa reducir el tamaño de un archivo JS eliminando espacios, saltos de línea y comentarios.

- Mejora la velocidad de carga.
- Reduce el uso de ancho de banda.
- Se usa en producción para optimizar el rendimiento.

•

## 10. ¿Qué es ECMAScript 6 (ES6)?

ECMAScript 6 (ES6), lanzado en 2015, es la sexta versión del estándar que define cómo funciona JavaScript y representa una de las actualizaciones más importantes del lenguaje. Introdujo características modernas que facilitan la programación, como el uso de **let** y **const** para un mejor control de variables, las funciones flecha que simplifican la sintaxis, las clases para la programación orientada a objetos, las plantillas de texto (template strings) que permiten interpolación de variables, el uso de módulos para organizar mejor el código y el manejo de la asincronía con promesas. Gracias a ES6, JavaScript se volvió más potente, claro y escalable, consolidándose como un lenguaje esencial para el desarrollo web actual.

## Conclusión

Con esta investigación pude entender mucho mejor qué es JavaScript, cómo surgió y por qué es tan importante en el desarrollo web actual. Aprendí que no solo sirve para darle vida e interactividad a las páginas, sino que también ha evolucionado con ECMAScript 6 para ofrecer nuevas herramientas que facilitan la programación. Además, me quedó más clara la diferencia entre PHP y JavaScript, la forma en que se relaciona con HTML, y cómo frameworks como Bootstrap ayudan a trabajar más rápido y con mejores resultados. En general, siento que ahora tengo una base más sólida para seguir aprendiendo y aplicando estos conocimientos en futuros proyectos de programación web.

# Referencias

Mozilla Developer Network. (s. f.). *¿Qué es JavaScript?* Recuperado de [https://developer.mozilla.org/es/docs/Learn\\_web\\_development/Core/Scripting/What\\_is\\_JavaScript](https://developer.mozilla.org/es/docs/Learn_web_development/Core/Scripting/What_is_JavaScript)

EDteam. (s. f.). *La historia completa de JavaScript, el único lenguaje que entienden los navegadores*. Recuperado de <https://ed.team/blog/la-historia-completa-de-javascript-el-unico-lenguaje-que-entienden-los-navegadores>

Digitrix. (2024, 30 de noviembre). *PHP vs. JavaScript: Similarities and Distinctions*. Recuperado de <https://www.digitrix.com/blogs/php-vs-javascript-similarities-and-distinctions>  
[Digitrix Infotech](#)

Arsys. (s. f.). *ECMAScript / JavaScript*. Recuperado de <https://www.arsys.es/blog/ecmascript-javascript>

Arsys. (2024, 14 de junio). *Guía completa sobre Bootstrap*. Recuperado de <https://www.arsys.es/blog/guia-completa-sobre-bootstrap> Arsys

LenguajeJS. (s. f.). *La consola JavaScript*. Recuperado de <https://lenguajejs.com/javascript/introduccion/consola-de-javascript/>