



Tarea #1

Triani Yuan Argüello

Universidad CENFOTEC

Fundamentos de Programación Web

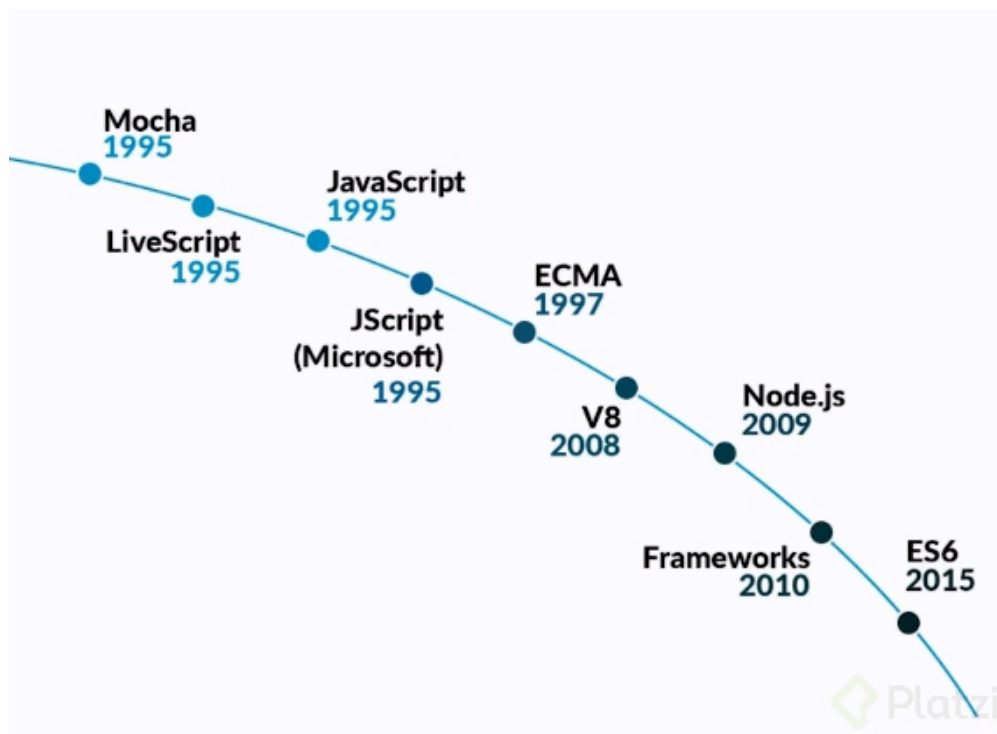
Francisco José Jiménez Bonilla

Fecha: Enero, 2024

1. ¿Escriba la historia del lenguaje JavaScript?

JavaScript se introdujo en 1995 como una forma de agregar programas a páginas web en el navegador Netscape Navigator. Este fue ganando popularidad ya que era una función que permitía a los desarrolladores agregar interactividad a las páginas web. Luego en 1997, JavaScript tenía el nombre de ECMAScript, para que diversos navegadores pudieran entenderlo por igual.

En los años 2000, JavaScript se volvió aún más esencial con la introducción de AJAX, permitiendo actualizar partes de una página web sin tener que recargar completamente. Con el tiempo, surgieron herramientas como jQuery, AngularJS y React, simplificando la programación con JavaScript. Además, en 2009, la llegada de Node.js expandió su uso al servidor, no solo en el navegador. Actualmente, JavaScript es uno de los lenguajes más populares, utilizado en una variedad de aplicaciones, desde sitios web hasta apps móviles y de escritorio. Continúa evolucionando con nuevas características y mejoras.



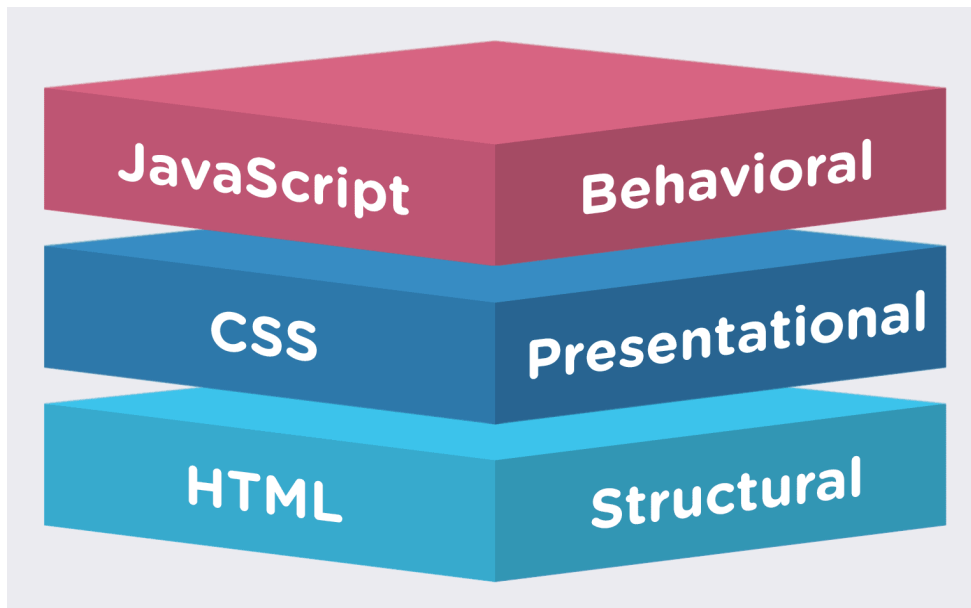
2. ¿Por qué se debe aprender JavaScript?

Aprender JavaScript es esencial para poder crear sitios web que logren ser dinámicos e interactivos. Ya que sin este, las páginas web serían estáticas y aburridas.

3. ¿Cuál es la relación entre HTML y JavaScript?

La relación entre HTML y JavaScript es que HTML define la estructura y contenido básico de una página web, como el esqueleto, mientras que JavaScript proporciona la

interactividad y el dinamismo, actuando como los músculos y tendones que le dan movilidad, permitiendo manipular y modificar el contenido HTML en respuesta a eventos del usuario.



4. ¿En qué beneficia usar Bootstrap para sitios y aplicaciones web en JS?

Usar Bootstrap en tus sitios web y aplicaciones en JavaScript es como tener un conjunto de herramientas útiles que ayudan a hacer un proyecto más bonito y fácil de usar, ya que son interfaces de para el desarrollo de web responsivas. Bootstrap ofrece cosas como diseños que se ajustan automáticamente a diferentes tamaños de pantalla, elementos que se ven bien desde teléfonos hasta computadoras, y estilos que hacen que tu proyecto luzca profesional sin tener que diseñar todo tú mismo. Además, se ahorra bastante tiempo porque ya viene con piezas predefinidas que puedes usar. También hace que un proyecto funcione bien en diferentes navegadores, para que los usuarios no tengan problemas al usarlo.

5. ¿Qué semejanza y diferencia tienen los lenguajes web PHP y JavaScript?

PHP y JavaScript son dos lenguajes populares en el desarrollo web, ambos son interpretados, lo que significa que el código se va ejecutando línea por línea. Ambos se usan para hacer que los sitios web sean más dinámicos. La diferencia principal es que PHP se ejecuta en el servidor y se utiliza para crear contenido web, mientras que JavaScript se ejecuta en el navegador del usuario y se emplea para añadir funciones interactivas a las páginas web.

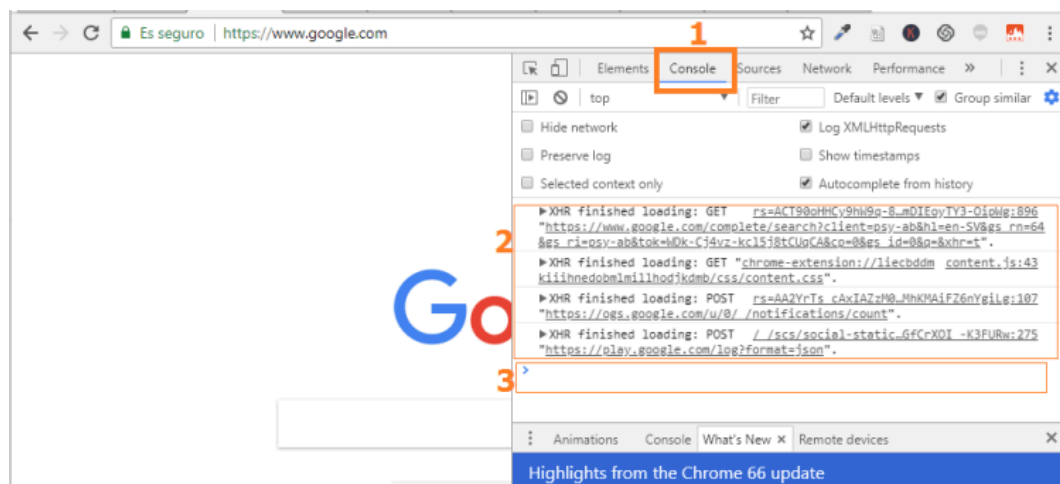


6. ¿Cite 3 formas en que se puede agregar código JS en una página web?

La primera forma es dentro del código de HTML, donde se utiliza la etiqueta `<script></script>`, normalmente dentro del body o head. La segunda forma es hacer un documento.js, en el cual se escribe todo el código normal y luego se agrega al HTML así: `<script src="documento.js"></script>`. Por último, están los eventos en HTML, estos hacen que se pueda conectar el código JavaScript a diferentes acciones que ocurren en index.html, como cuando alguien hace click en un botón o cambia algo de sitio: "onclick" y "onchange".

7. ¿Cuál es la función principal de la consola en JS?

La consola en JavaScript es como una ventana especial donde se pueden ver mensajes importantes del código mientras se está desarrollando una página web. Se puede usar para mostrar información sobre variables, mensajes de error si algo sale mal en el código, y también para probar pequeños pedazos de código para ver cómo funcionan antes de incluirlos en la página web. Es como tener un asistente que ayuda a entender qué está pasando en el código y a corregir cualquier problema que se pueda encontrar.



8. ¿Cuál es la diferencia que existe en las declaraciones var, let y const en JS?

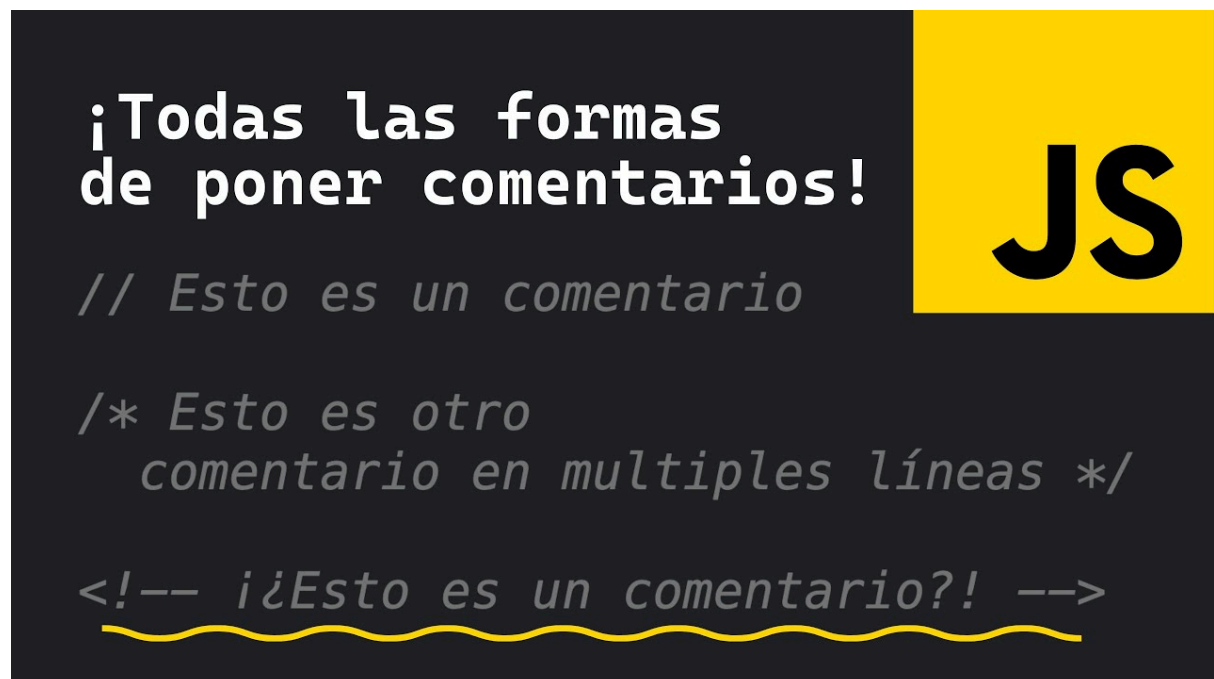
- var: Tiene alcance de función, permite la redeclaración y puede ser reasignada.
- let: Tiene alcance de bloque, no permite la redeclaración en el mismo bloque y puede ser reasignada.
- const: Tiene alcance de bloque, no permite la reasignación después de su declaración.

Alcance de bloque es cuando hay código dentro de llaves “{}”, fuera de este bloque la variable no es visible ni accesible. Alcance de función es que no importa donde fue declarada, la variable estará disponible en cualquier parte de la función en la que fue definida.

9. ¿Explique los 2 tipos de comentarios que se pueden aplicar en JS?

Comentarios en una línea, que con sólo colocar “//”, posteriormente solo se escribe el comentario y listo. Por otro lado, está el de bloque o de varias líneas, donde se debe poner /* */ y listo. Ej:

```
/* comentario
comentario
comentario */
```



10. ¿Qué es ECMAScript 6? Explique claramente.

ECMAScript 6 es el estándar que define cómo debe funcionar JavaScript desde 2015 hasta hoy. Se encarga de establecer las reglas para interpretar y ejecutar JavaScript

en diferentes plataformas, como navegadores web, Node.js y aplicaciones para diversos sistemas operativos. Los creadores de navegadores y de JavaScript se aseguran de que el código JavaScript sea interpretado según lo establecido por ECMAScript. Además de establecer reglas para la interpretación del código, ES6 introduce una serie de nuevas características que mejoran la forma en que se escribe JavaScript:

- Declaración de variables con `let` y `const`, que permiten un mejor control sobre el alcance y la inmutabilidad de las variables.
- Arrow functions (funciones de flecha), que ofrecen una sintaxis más compacta y clara para definir funciones.

```
1 //Antes
2 var saludo = function (nombre) {
3   return 'Hola ' + nombre;
4 };
5 console.log( saludo('Jonathan') ); //Imprime Hola Jonathan
6
7 //Ahora
8 let saludo = nombre => `Hola ${nombre}`;
9 console.log( saludo('Jonathan') ); //Imprime Hola Jonathan
```

- Template literals, que facilitan la creación de cadenas de texto largas e interpolación de variables.

```
let a = 5;
let b = 10;
console.log(`Quince es ${a + b} y
no ${2 * a + b}.`);
// "Quince es 15 y
// no 20."
```

- Destructuring (desestructuración), que simplifica la extracción de valores de arreglos u objetos.

JavaScript Demo: Expressions - Destructuring assignment

```
1 let a, b, rest;
2 [a, b] = [10, 20];
3
4 console.log(a);
5 // Expected output: 10
6
7 console.log(b);
8 // Expected output: 20
9
10 [a, b, ...rest] = [10, 20, 30, 40, 50];
11
12 console.log(rest);
13 // Expected output: Array [30, 40, 50]
14
```

- Clases, que proporcionan una sintaxis más orientada a objetos para definir clases en JavaScript.

```
class Rectangulo {
  constructor(alto, ancho) {
    this.alto = alto;
    this.ancho = ancho;
  }
  // Getter
  get area() {
    return this.calcArea();
  }
  // Método
  calcArea() {
    return this.alto * this.ancho;
  }
}

const cuadrado = new Rectangulo(10, 10);

console.log(cuadrado.area); // 100
```

- Módulos, que introducen un sistema estándar para organizar y reutilizar el código en aplicaciones JavaScript, mejorando la modularidad y mantenibilidad del código.

En resumen, es lo que regula la forma en que se debe escribir el código, para que todos sigan el mismo formato y se vea ordenado.

¿Qué enseñanza le brindó la tarea?

En esta tarea, he aprendido sobre la importancia de ECMAScript 6 (ES6) en JavaScript y cómo afecta la forma en que escribimos y entendemos el código en diferentes plataformas. También he entendido las nuevas características introducidas por ES6, como las variables `let` y `const`, las funciones de flecha, los `template literals`, la desestructuración, las clases y los módulos, y cómo estas hacen que nuestro código sea más claro y fácil de entender. En resumen, esta tarea me ha ayudado a comprender más las diferentes características del lenguaje JavaScript para escribir de manera más efectiva código.