

‘Unidad 3:’ **{ Desarrollo Web con** **Javascript** **}**



CONTENIDOS DE LA CLASE {

Introducción a JavaScript

}

OBJETIVOS {

Conocer de dónde proviene JavaScript y por qué es tan importante.

Tener una noción de todo lo que se puede hacer con JavaScript.

Entender la integración única que posee JavaScript con la programación web.

Conocer los recursos más utilizados para documentarse sobre JavaScript.

}

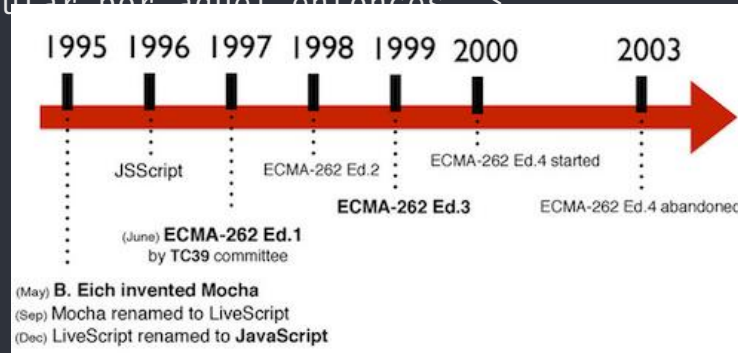
```
1  {  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14 }
```

[Introducción a JavaScript]



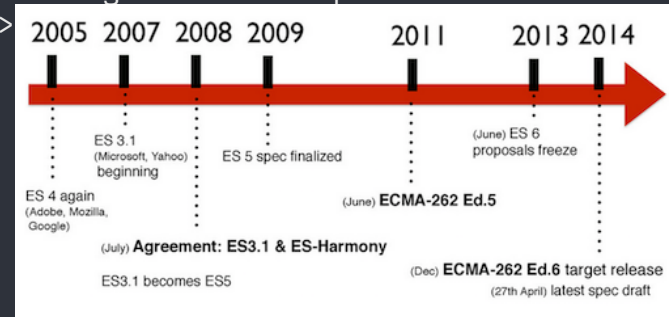
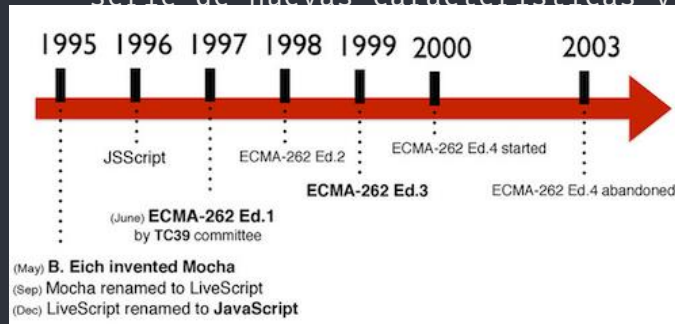
¿Cuál es la historia de JavaScript?

< JavaScript fue creado por Brendan Eich en 1995 mientras trabajaba en Netscape Communications Corporation. La idea original era crear un lenguaje de programación que pudiera ejecutarse en el navegador del usuario, lo que permitiría a los desarrolladores crear páginas web interactivas y dinámicas. El primer nombre del lenguaje fue Mocha, En Septiembre de ese año lo renombra a LiveScript hasta que le cambiaron el nombre a JavaScript debido a una estrategia de marketing, ya que Netscape fue adquirida por Sun Microsystems, propietaria del lenguaje Java, muy popular por aquel entonces. >



¿Cuál es la historia de JavaScript?

< En febrero de 1996, JavaScript fue adoptado por Microsoft en Internet Explorer, lo que llevó a una rápida expansión de su uso. La especificación de JavaScript fue estandarizada en 1997 por la Ecma International, una organización dedicada a la estandarización de tecnologías de la información y la comunicación. En 1999 aparece la 3a versión del estándar ECMAScript. Hubo pequeños intentos de escribir la versión 4, pero hasta 2011 no se aprobó y se estandarizó la versión 5 (ES5). En junio de 2013 quedó parado el borrador de la versión 6, pero en Diciembre de 2014 se aprobó al fin y a partir de Junio de 2015 fue la sexta versión importante del estándar ECMAScript y trajo consigo una serie de nuevas características y mejoras significativas para el



¿Con qué objetivo fue creado? {

< JavaScript fue creado con el objetivo de agregar interactividad y dinamismo a las páginas web. Cuando se desarrolló y se llamó "LiveScript", se diseñó para ser una extensión de Java para agregar interactividad a las páginas web. Más tarde cuando se renombró a JavaScript, se desarrolló para ser un lenguaje de programación independiente para la web.

Antes de JavaScript, las páginas web eran principalmente estáticas y no tenían la capacidad de responder a la entrada del usuario o de actualizar dinámicamente la página sin tener que cargarla completamente de nuevo. Con JavaScript, los desarrolladores pueden crear aplicaciones web interactivas que responden a la entrada del usuario en tiempo real y actualizan dinámicamente el contenido de la página sin tener que volver a cargar la página completa. >

}

¿Cómo y cuándo se agregan nuevas funcionalidades en javascript? {

< Las nuevas funcionalidades de JavaScript se agregan a través de actualizaciones de las especificaciones llamadas ECMAScript. ECMAScript es un estándar que define la sintaxis y el comportamiento del lenguaje de programación JavaScript. Cada nueva versión de ECMAScript incluye nuevas funcionalidades y mejoras en el lenguaje.

Antes de JavaScript, las páginas web eran principalmente estáticas y no tenían la capacidad de responder a la entrada del usuario o de actualizar dinámicamente la página sin tener que cargarla completamente de nuevo. Con JavaScript, los desarrolladores pueden crear aplicaciones web interactivas que responden a la entrada del usuario en tiempo real y actualizan dinámicamente el contenido de la página sin tener que volver a cargar la página completa. >

}

¿Qué es y en qué consiste la especificación ECMAScript?

< La especificación ECMAScript es un estándar que define el lenguaje de programación JavaScript. ECMAScript establece la sintaxis, la semántica y las reglas para la implementación del lenguaje de programación JavaScript. Es desarrollado y mantenido por el comité técnico TC39 de la organización internacional de normalización ECMA, de ahí su nombre. La especificación ECMAScript define la estructura básica del lenguaje, incluyendo la sintaxis y las características clave como las variables, los operadores, las estructuras de control de flujo y las funciones. También define las nuevas funcionalidades y mejoras que se agregan a JavaScript en cada versión.

La especificación ECMAScript es importante para los desarrolladores porque les permite escribir código JavaScript que funcionará de manera consistente en diferentes plataformas y navegadores, y les permite utilizar las últimas características y funcionalidades del lenguaje. Además, la especificación ECMAScript proporciona un marco de referencia común para los desarrolladores, lo que les permite colaborar y compartir código de manera más efectiva.>

¿Cuál es el alcance y las ramas en las que JavaScript está involucrado? {

<JavaScript se utiliza en una amplia variedad de aplicaciones y ramas de la tecnología.>

Desarrollo web: {

< JavaScript se utiliza ampliamente en el desarrollo web para crear páginas web interactivas y dinámicas. Con la ayuda de bibliotecas y frameworks como React, Vue.js, Angular, entre otros, JavaScript permite crear aplicaciones web complejas y escalables. >

}

Desarrollo móvil: {

<JavaScript se utiliza en el desarrollo móvil para crear aplicaciones móviles híbridas y nativas. Frameworks como React Native y Ionic permiten a los desarrolladores utilizar JavaScript para crear aplicaciones móviles que funcionan en múltiples plataformas. >

}

}

¿Cuál es el alcance y las ramas en las que JavaScript está involucrado? {

<JavaScript se utiliza en una amplia variedad de aplicaciones y ramas de la tecnología.>

Desarrollo de juegos: {

< JavaScript se utiliza en el desarrollo de juegos en línea y móviles. Con la ayuda de bibliotecas y frameworks como Phaser, Three.js y Babylon.js, los desarrolladores pueden crear juegos 2D y 3D interactivos y de alta calidad. >

}

Desarrollo de aplicaciones de escritorio {

<JavaScript se utiliza en el desarrollo de aplicaciones de escritorio, especialmente en el ámbito de la automatización de tareas. Frameworks como Electron y NW.js permiten a los desarrolladores utilizar JavaScript para crear aplicaciones de escritorio multiplataforma. >

}

}

¿Cuáles son las capacidades de uso en multiplataforma que ofrece el lenguaje javascript? {

< Para comprender: >

- 1) < JavaScript es un lenguaje de programación interpretado: Esto significa que el código JavaScript se ejecuta en tiempo de ejecución, lo que permite que se adapte a diferentes plataformas. El intérprete de JavaScript está presente en muchos navegadores web y en entornos de servidor como Node.js. >
- 2) < JavaScript es un lenguaje de programación con tipado dinámico: Esto significa que las variables no tienen un tipo de datos fijo y pueden cambiar de tipo durante la ejecución de la aplicación. Esto permite una mayor flexibilidad en el uso de las variables y una adaptabilidad a diferentes plataformas. >

}

¿Cuáles son las capacidades de uso en multiplataforma que ofrece el lenguaje javascript? {

< Para comprender: >

< JavaScript tiene una amplia variedad de bibliotecas y frameworks:

Hay una gran cantidad de bibliotecas y frameworks de JavaScript

3) disponibles que facilitan el desarrollo de aplicaciones multiplataforma. Estos frameworks y bibliotecas permiten a los desarrolladores crear aplicaciones que funcionan en diferentes plataformas, desde navegadores web hasta dispositivos móviles. >

< JavaScript se puede utilizar tanto en el lado del cliente como en

4) el lado del servidor: Esto significa que JavaScript se puede utilizar para crear tanto aplicaciones web como aplicaciones de escritorio. Node.js es un entorno de servidor de JavaScript que permite la creación de aplicaciones web y de escritorio utilizando JavaScript. >

}

Ejemplos de aplicaciones creadas con una base en JavaScript {

<Gmail> { es uno de los servicios de correo electrónico más populares del mundo y utiliza una gran cantidad de JavaScript para proporcionar una experiencia de usuario dinámica e interactiva. Gmail utiliza AJAX (Asynchronous JavaScript and XML) para actualizar el contenido de la página sin necesidad de recargarla. }



<Facebook-Meta> { es una de las redes sociales más populares del mundo y utiliza una gran cantidad de JavaScript para proporcionar una experiencia de usuario fluida e interactiva. Facebook utiliza una gran cantidad de bibliotecas y frameworks de JavaScript, como React y GraphQL, para crear una aplicación web escalable y robusta. }



}

Ejemplos de aplicaciones muy utilizadas creadas con una base en JavaScript {

<Google Maps> { es un servicio de mapas en línea muy popular que utiliza una gran cantidad de JavaScript para proporcionar una experiencia de usuario fluida e interactiva. Google Maps utiliza una técnica llamada "carga perezosa" para cargar los mapas y los marcadores solo cuando son necesarios, lo que mejora la velocidad de carga de la aplicación. }



<Spotify> { es un servicio de música en línea muy popular que utiliza una gran cantidad de JavaScript para proporcionar una experiencia de usuario dinámica e interactiva. Spotify utiliza una técnica llamada "carga perezosa" para cargar las canciones y las listas de reproducción solo cuando son necesarias, lo que mejora la velocidad de carga de la aplicación. }



¿Qué hace especial a JavaScript? {

- 1) < es el único lenguaje de programación que funciona directamente en el navegador web, lo que permite a los desarrolladores crear aplicaciones web interactivas y dinámicas que responden a la interacción del usuario. Esto ha hecho que JavaScript sea muy popular entre los desarrolladores de aplicaciones web y ha llevado a la creación de muchas bibliotecas y frameworks de JavaScript para facilitar el desarrollo web. >
- 2) < es relativamente fácil de aprender, en comparación con otros lenguajes de programación, debido a su sintaxis simple y a que no requiere una configuración complicada para comenzar a trabajar con él. Además, hay una gran cantidad de recursos en línea disponibles para aprender JavaScript, incluidos tutoriales, videos y documentación oficial. >
- 3) < es un lenguaje de programación dinámico, lo que significa que puede ejecutarse en tiempo real y reaccionar a eventos del usuario. Esto lo convierte en un lenguaje ideal para aplicaciones web interactivas y dinámicas, como juegos en línea, aplicaciones de chat en vivo y otras aplicaciones que requieren una respuesta en tiempo real. >
- 4) < tiene una gran y activa comunidad de desarrolladores que contribuyen a bibliotecas, frameworks y herramientas para el lenguaje. Esto hace que sea fácil encontrar soluciones a problemas específicos y mejorar la calidad de la programación en general. >

}

Ventajas y desventajas de JavaScript {

<VENTAJAS>

< **Lenguaje de programación de alto nivel:** es un lenguaje de programación de alto nivel que permite a los desarrolladores escribir código más rápido y de manera más eficiente. >

< **Fácil de aprender:** es relativamente fácil de aprender y tiene una curva de aprendizaje suave para aquellos que ya tienen experiencia en programación.>

< **Interactivo y dinámico:** permite crear aplicaciones web interactivas y dinámicas que responden a la entrada del usuario en tiempo real y actualizan dinámicamente el contenido de la página sin tener que volver a cargar la página completa.>

< **Amplia comunidad de desarrolladores:** cuenta con una gran comunidad de desarrolladores que comparten su conocimiento y recursos en línea.>

<**Utilizado en una amplia variedad de aplicaciones:** se utiliza en una amplia variedad de aplicaciones web, móviles y de escritorio, lo que lo convierte en un lenguaje de programación muy versátil.>

}

Ventajas y desventajas de JavaScript {

<DESVENTAJAS>

< **Falta de seguridad:** JavaScript puede ser vulnerable a ataques de seguridad, como inyección de código malicioso y cross-site scripting (XSS).>

< **Dificultades en la depuración de errores:** A menudo, puede ser difícil depurar los errores en JavaScript, especialmente en aplicaciones web complejas.>

< **Dependencia de los navegadores:** JavaScript está estrechamente relacionado con los navegadores web, y algunas funcionalidades pueden no ser compatibles con ciertos navegadores, lo que puede requerir el uso de bibliotecas o herramientas adicionales para garantizar la compatibilidad.>

< **Rendimiento:** Aunque el rendimiento de JavaScript ha mejorado con el tiempo, todavía puede ser un poco más lento que otros lenguajes de programación en algunas aplicaciones.>

< **Limitaciones en la programación de aplicaciones de escritorio:** Aunque JavaScript se utiliza para desarrollar aplicaciones de escritorio, todavía tiene algunas limitaciones en términos de acceso a recursos del sistema y funcionalidad avanzada.>

}

```
1      ;Cuales son los usos principales de JavaScript?  {
```

```
2  
3      Interactividad {
```

```
4          <Una de las razones principales por las que se utiliza JavaScript  
5          en el desarrollo web es para crear interactividad en una página.  
6          Con JavaScript, se puede crear botones, animaciones y otros  
7          elementos interactivos que hacen que el usuario se involucre más  
8          en la experiencia del sitio web. >
```

```
9      }
```

```
10     Validación de formularios {
```

```
11         < JavaScript se utiliza comúnmente para validar la información  
12         que un usuario introduce en un formulario, como asegurarse de que  
13         se introduzca un correo electrónico válido o un número de  
14         teléfono. Esto ayuda a mejorar la experiencia del usuario y  
15         garantiza que los datos se recopilen correctamente. >
```

```
16     }
```

```
17 }
```

```
1      ;Cuales son los usos principales de JavaScript?  {
```

```
2  
3      Manipulación del DOM {
```

```
4          <El Document Object Model (DOM) es la estructura que se utiliza  
5          para representar un documento HTML en un navegador web.  
6          JavaScript se puede utilizar para manipular el DOM y actualizar  
          dinámicamente la página web sin tener que recargarla. >
```

```
7      }
```

```
8      Aplicaciones web {
```

```
9          < Con el auge de las aplicaciones web, JavaScript se ha  
10         convertido en una parte integral del desarrollo de aplicaciones.  
11         Los frameworks de JavaScript como Angular, React y Vue.js se  
12         utilizan comúnmente para construir aplicaciones web que ofrecen  
           una experiencia de usuario similar a la de una aplicación nativa.  
           >
```

```
13     }
```

```
14 }
```

¿Qué es una documentación en javascript? {

< Es un conjunto de información escrita que describe el código y la funcionalidad de un programa, biblioteca o framework de JavaScript. La documentación es esencial para cualquier proyecto de programación, ya que proporciona información sobre cómo utilizar el código, cómo se relacionan las diferentes partes del código, las entradas y salidas esperadas, y cualquier otra información relevante para los usuarios.

Los comentarios en el código son la forma más básica de documentación. Se utilizan para describir lo que hace una sección específica del código o para explicar por qué se escribió una línea de código de cierta manera. Los comentarios pueden ayudar a otros desarrolladores a entender el código y hacer modificaciones con mayor facilidad.

```
// Este es un ejemplo de un comentario en una sola línea

/*
Este es otro ejemplo de un comentario
que abarca varias líneas de texto.
En este caso se utiliza un bloque de comentarios
que se cierra con el símbolo de asterisco y la barra.
*/

function suma(a, b) {
  // En este caso, el comentario explica que la función suma dos números
  return a + b;
}
```

¿Qué es una documentación en javascript? {

< La documentación en JavaScript es una colección de información y recursos que explican cómo usar y trabajar con el lenguaje de programación JavaScript. Esta documentación incluye descripciones detalladas de las características y funcionalidades del lenguaje, así como ejemplos y tutoriales para ayudar a los desarrolladores a entender y aplicar el código.

La documentación puede tomar muchas formas diferentes, desde comentarios en el propio código, hasta guías y tutoriales en línea, documentación de referencia en formato de libro, y otros recursos en línea, como sitios web de programación y foros de discusión.

La documentación en JavaScript es útil para los desarrolladores que están aprendiendo a utilizar el lenguaje, o para aquellos que necesitan ayuda para implementar una funcionalidad específica en su proyecto. La documentación puede ayudar a los desarrolladores a entender cómo funciona el lenguaje, a utilizarlo de manera efectiva y a resolver problemas que puedan surgir durante el desarrollo de un proyecto. >

}

```
1  ¿Cuáles son las cuestiones a tener en cuenta a la hora de  
2  leer una documentación en javascript?  {
```

```
3      < Versión del lenguaje: Asegúrase que la documentación que se está leyendo  
4      se corresponde con la versión de JavaScript que se está utilizando. Cada  
5      versión puede tener diferencias significativas en cuanto a la sintaxis y  
6      funcionalidades. >
```

```
7      < Objetivo y contexto: Entender el objetivo de la documentación y el  
8      contexto en el que se utiliza el código. Esto ayudará a entender cómo se  
9      pueden aplicar las funcionalidades y a qué tipo de proyectos están  
10     destinados. >
```

```
11     < Formato: Asegurarse de entender el formato de la documentación. Algunas  
12     documentaciones pueden estar en formato de guía, tutorial, referencia o  
13     ejemplos, y cada uno tiene su propio enfoque y nivel de detalle. >
```

```
14     < Vocabulario técnico: Aprender el vocabulario técnico específico del  
15     lenguaje. Esto permitirá entender los términos técnicos utilizados en la  
16     documentación, lo que a su vez facilitará la comprensión de las  
17     funcionalidades y su implementación. >
```

```
18 }
```

```
1  ¿Cuáles son las cuestiones a tener en cuenta a la hora de  
2  leer una documentación en javascript?  {
```

```
3      < Ejemplos: Prestar atención a los ejemplos proporcionados en la  
4      documentación. Estos pueden ayudar a entender cómo se utiliza una  
5      funcionalidad en particular y a visualizar su implementación en código. >
```

```
6      < Actualización: Tener en cuenta que la documentación puede actualizarse con  
7      el tiempo, y que puede haber cambios en las funcionalidades o nuevas  
8      adiciones. Asegurarse de estar consultando la documentación más actualizada  
9      para evitar posibles errores o confusiones. >
```

```
10     < Fuentes confiables: Asegurarse de que la documentación que se está leyendo  
11     proviene de una fuente confiable, como la documentación oficial del lenguaje  
12     o sitios web reconocidos en la comunidad de desarrollo de JavaScript. >
```

```
13  
14 }
```


Ejemplos de aplicaciones creadas con una base en JavaScript {

<Documentación oficial de JavaScript> { proporciona información completa y detallada sobre el lenguaje. Puedes encontrar información sobre la sintaxis, las funciones, las estructuras de control de flujo, los objetos y mucho más. }



<MDN Web Docs> { Mozilla Developer Network (MDN) Web Docs es una excelente fuente de información sobre JavaScript. Ofrece documentación detallada sobre el lenguaje, junto con tutoriales, guías, demos y ejemplos de código. }



}

Ejemplos de aplicaciones creadas con una base en JavaScript {

<Stack Overflow> { es una comunidad en línea muy activa de desarrolladores que pueden ayudarte a resolver problemas y responder preguntas sobre JavaScript. Puedes buscar preguntas y respuestas sobre temas específicos en JavaScript, o publicar una pregunta tú mismo. }



<W3Schools> { es un sitio web popular que ofrece tutoriales y recursos para desarrolladores web, incluyendo tutoriales sobre JavaScript. Los tutoriales son concisos y fáciles de seguir, y también hay ejemplos de código para ayudarte a entender los conceptos. }



}

¿Cómo integrar JavaScript en un documento html? {

[`<script>`] < Usando la etiqueta script, se tiene que agregar la etiqueta script dentro de la sección head o body del documento HTML y dentro de ella escribir el código JavaScript.>

[`.js`] < Usando un archivo externo con extensión .js , también es posible incluir JavaScript en un archivo externo y vincularlo al documento HTML. Esto se hace mediante la etiqueta `<script>` con el atributo "src" que apunta al archivo JavaScript externo. >

}

¿Cómo los navegadores interpretan javascript dentro de un documento html? {

- 1) **<Descarga el documento HTML:** El navegador descarga el documento HTML desde el servidor web y lo analiza para construir el árbol de nodos del documento.>
- 2) **<Encuentra las etiquetas <script>:** El navegador busca las etiquetas <script> en el documento HTML y extrae el código JavaScript que se encuentra dentro de ellas.>
- 3) **<Analiza el código JavaScript:** El motor de JavaScript analiza el código JavaScript para detectar cualquier error de sintaxis y compila el código en un formato que pueda ser ejecutado por el navegador.>
- 4) **<Ejecuta el código JavaScript:** El motor de JavaScript ejecuta el código JavaScript en el orden en que se encuentra en el documento HTML. Si se encuentra algún código que hace referencia a elementos HTML en el documento, el motor de JavaScript espera a que esos elementos sean contruidos antes de ejecutar ese código.>
- 5) **<Modifica el contenido de la página:** Si el código JavaScript modifica el contenido de la página, el motor de JavaScript actualiza el DOM (Document Object Model) del documento HTML para reflejar los cambios.>