



<Tarea 1>

<Christopher Corrales Forbes>

Universidad CENFOTEC

<Fundamentos de Programación Web>

<FRANCISCO JOSE JIMENEZ BONILLA>

Fecha: <Mayo, 2024

1. ¿Escriba la historia del lenguaje Java Script?

JavaScript surgió como una tecnología del lado del navegador para hacer que las aplicaciones web fueran más dinámicas. Por medio de JavaScript, los navegadores eran capaces de responder a la interacción de los usuarios y cambiar la distribución del contenido en la página web.

JavaScript se introdujo en 1995 como una forma de agregar programas a páginas web en el navegador Netscape Navigator. En su momento fue una idea novedosa. En los primeros días de la World Wide Web, HTML era bastante simple, y bastante fácil de aprender casi todo lo que se necesitaba saber para agrupar páginas web.

Aunque JavaScript surgió como un lenguaje de script para mejorar las capacidades de la web de la época allá por 1995 por la extinta Netscape, JavaScript no ha dejado de evolucionar desde entonces.

Brendan Eich, un programador que trabajaba en Netscape, pensó que podría solucionar las limitaciones de la web de entonces, adaptando otras tecnologías existentes (como ScriptEase) al navegador Netscape Navigator 2.0, que iba a lanzarse en aquel año. Inicialmente, Eich denominó a su lenguaje LiveScript y fue un éxito.

Fue entonces cuando, justo antes del lanzamiento, Netscape decidió cambiar el nombre por el de JavaScript y firmó una alianza con Sun Microsystems para continuar el desarrollo del nuevo lenguaje de programación.

Microsoft, al ver el movimiento de uno de sus principales competidores, también decidió incorporar su propia implementación de este lenguaje, llamada JScript, en la versión 3 de su navegador Internet Explorer.

Esto contribuyó todavía más al empuje y popularización del lenguaje, pero comenzaron a presentarse pequeños problemas por las diferencias entre implementaciones.

Por lo que se decidió estandarizar ambas mediante la versión JavaScript 1.1 como propuesta a ECMA, que culminó con el estándar ECMA-262. Este estándar dicta la base del lenguaje ECMAScript a través de su sintaxis, tipos, sentencias, palabras clave y reservadas, operadores y objetos, y sobre la cual se pueden construir distintas implementaciones. La versión JavaScript 1.3 fue la primera implementación completa del estándar ECMAScript.

ECMAScript es una especificación de lenguaje de programación publicada por Ecma International. El desarrollo empezó en 1996 y estuvo basado en el popular lenguaje JavaScript propuesto como estándar por Netscape Communications Corporation. Actualmente está aceptado como el estándar ISO/IEC 22275:2018.

2. ¿Por qué se debe aprender Java Script?

JavaScript es un lenguaje de programación que los desarrolladores utilizan para hacer páginas web interactivas. Desde actualizar fuentes de redes sociales a mostrar animaciones y mapas interactivos, las funciones de JavaScript pueden mejorar la experiencia del usuario de un sitio web. Como lenguaje de scripting del lado del servidor, se trata de una de las principales tecnologías de la World Wide Web. Por ejemplo, al navegar por Internet, en cualquier momento en el que vea un carrusel de imágenes, un menú desplegable “click-to-show” (clic para mostrar), o cambien de manera dinámica los elementos de color en una página web, estará viendo los efectos de JavaScript.

Actualmente es uno de los lenguajes de programación más demandado y se encuentra presente en prácticamente cualquier aplicación web. Empresas como Google o Facebook han desarrollado incluso sus propias implementaciones basadas en este lenguaje para lograr por ejemplo una experiencia visual del usuario más atractiva.

3. ¿Cuál es la relación entre HTML y Java Script?

HTML proporciona el contenido y la estructura, CSS añade el diseño y el estilo, y JavaScript aporta elementos interactivos.

Los tres lenguajes son parte integrante de la creación de una página web funcional y visualmente atractiva.

Trabajan juntos para dar vida a un sitio web.

Sin embargo, es importante tener en cuenta que mientras el HTML y el CSS son necesarios para que un sitio web funcione, el JavaScript no siempre es necesario. Hay ciertos elementos que se pueden conseguir solo con HTML y CSS, como los efectos hover o el diseño responsivo.

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

Bootstrap es un conjunto de herramientas de código abierto muy popular entre los expertos en desarrollo web, ya que está ideado para un desarrollo responsive.

Es decir, gracias a Bootstrap, los desarrolladores web pueden crear páginas web visibles en diferentes formatos y pantallas.

Esta herramienta permite crear una interfaz web, a través de lenguajes de programación como HTML, CSS o JavaScript; los cuales permiten adaptar el contenido de la web a los diferentes tipos de pantallas (diseño responsive) y mejorando, por tanto, la experiencia de usuario.

Su framework es muy flexible e incluye diferentes elementos para maquetar tu sitio web. Son plantillas de diseño web muy intuitivas y con excelentes resultados.

Ventajas de usar Bootstrap en tu web:

- Uso fácil
- Compatible con distintos navegadores
- Diseño Responsive
- Personalización

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

Son varias las diferencias con las que cuentan estos lenguajes de programación:

PHP (Hypertext Preprocessor) es un lenguaje que utiliza secuencias de comandos del lado del servidor, mientras que Javascript lo hace del lado del cliente.

PHP no se ejecuta en el navegador, JavaScript, sí lo hace.

PHP admite bases de datos, por el contrario, JavaScript no lo hace.

PHP acepta variables en minúscula y mayúscula, JavaScript no.

PHP no admite el intercambio de objetos y matrices. Por su parte, JavaScript sí admite intercambiar objetos y matrices.

PHP es un lenguaje de programación que está orientado a objetos. Su código es realmente fácil de usar y requiere HTTP para poder ejecutarse. Su lenguaje de secuencias de comandos está del lado del servidor.

De la misma forma, JavaScript también está orientado a objetos. Cuenta con un código sencillo de utilizar y su lenguaje de secuencias de comandos está del lado del cliente.

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

La integración de JavaScript en una página web es muy flexible.

Su instrucción principal es la siguiente:

```
<script>  
//aquí van las instrucciones  
</script>
```

```
<!DOCTYPE html>  
<html>  
<head>  
<title>Hello World!</title>  
</head>  
<body>  
  <script>  
    alert("Hello world!");  
  </script>  
</body>  
</html>
```

Las 3 formas en que se puede agregar código JS en una página web son las siguientes:

Forma #1 Se incluye en la parte superior de la página web en la zona </head>.

```
<head>
  <meta charset="UTF-8">
  <title>JavaScript Hello World Example - Estrada Web Group</title>
  <script>
    alert('Hello, World Estrada Web Group!');
  </script>
</head>
```

Forma #2

Se incluye dentro de una instrucción HTML utilizando eventos interactivos.

```
<p class="cursor" onclick="contenido1()">
  JavaScript es un lenguaje de programación
  que una página web hace algo más que ser
  contenido, mapas interactivos, animación
  probablemente JavaScript está involucrada
  hemos cubierto con mucho más detalle en
</p>
```

Forma #3

Se incluye en la parte inferior de la página web antes del cierre </body>

Es la más utilizada ya que la programación JS debe ir en un archivo externo.

Se guardan en la carpeta JS de la estructura de proyectos web.

```
<!-- end of page content here-->
<script src="js/service.js"></script>
<script src="js/app.js"></script>
</body>
</html>
```

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

Puede escribir cualquier expresión, instrucción o fragmento de código de JavaScript en la consola y se ejecuta de forma inmediata e interactiva a medida que escribe. Esto es posible porque la herramienta Consola de DevTools es un entorno REPL . REPL significa Lectura, Evaluación, Impresión y Bucle.

La consola:

Lee el Código JavaScript que escribe en él.
Evalúa el código.
Imprime el resultado de la expresión.
Vuelve al primer paso.

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

Al declarar variables en JavaScript, hay tres opciones: var, let y const. Aunque las tres palabras clave permiten declarar variables, tienen diferentes alcances y comportamientos.

Var

Var es la palabra clave original utilizada para declarar variables en JavaScript, y ha existido desde que se introdujo el lenguaje por primera vez.

Una de las desventajas de var es que tiene ámbito de función. Esto significa que una variable declarada con var está disponible en toda la función en la que se declara. Si declaras una variable dentro de un bloque o bucle, sigue estando disponible fuera de ese bloque o bucle, lo que puede dar lugar a resultados inesperados al intentar reutilizar nombres de variables o al trabajar con funciones anidadas.

Let

Let es una palabra clave relativamente nueva introducida en ES6 (ECMAScript 6). Se introdujo para resolver algunos de los problemas asociados con var. La principal diferencia entre let y var es que let tiene ámbito de bloque. Esto significa que una variable declarada con let sólo está disponible dentro del bloque en el que se declara. Si se declara una variable dentro de un bloque o bucle, no estará disponible fuera de ese bloque o bucle.

Otra diferencia entre let y var es que no puedes declarar la misma variable varias veces utilizando let. Esto puede ayudar a prevenir conflictos de nombres y hacer tu código más legible.

Const

Const es otra palabra clave introducida en ES6, y se utiliza para declarar constantes. Una constante es un valor que no puede ser reasignado una vez que ha sido declarado. Esto puede ser útil cuando se trabaja con valores que nunca deben cambiar, como las constantes matemáticas o las opciones de configuración.

En resumen, var es la palabra clave más flexible, pero también la más propensa a errores para declarar variables en JavaScript. let es una palabra clave más reciente que proporciona ámbito

de bloque y evita conflictos de nombres. const se utiliza para declarar constantes que no deben reasignarse.

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

1. De una línea

```
<script>
// Este un comentario de una única línea
alert("Escribiendo comentarios en javascript!");
//Aquí puedo poner una nota de lo que hace esta línea
// alert("Esto no se ejecuta");
</script>
```

2. De varias líneas (multilínea)

```
<script>
alert("Escribiendo comentarios multi-línea en javascript");
/*
alert("Esto no se ejecuta");
alert("Esto no se ejecuta");
alert("Esto no se ejecuta");
alert("Esto no se ejecuta");
*/
</script>
```

10. ¿Qué es ECMAScript6? Explique claramente.

ES6 o ECMAScript 6 es un estándar de lenguaje de scripting desarrollado de acuerdo con las indicaciones contenidas en el documento ECMA-262 proporcionado por Ecma International (European Computer Manufacturers Association). Ecma international es una asociación fundada en 1961 que se ocupa de la estandarización en el sector de las tecnologías de la información y los sistemas de comunicación.

ECMAScript es, por lo tanto, un estándar creado con el objetivo de permitir la interoperabilidad de diferentes páginas web entre distintos navegadores. El estándar de lenguaje especifica las características que debe tener un lenguaje de scripting y cómo deben implementarse.

Conclusión del aprendizaje obtenido.

Evidentemente el lenguaje de programación JavaScript ha evolucionado con el tiempo y ha traído muchas facilidades y mejoras al campo del desarrollo web, por lo cual, para nosotros como estudiantes, es importante su entendimiento. Al conocer mas sobre este lenguaje de programación que funciona del lado del cliente, será más fácil adquirir mejores conocimientos durante el curso de Fundamentos de Programación Web, los cuales son la base de la experiencia que vamos a llevar al campo laboral, a través de las tareas, prácticas y proyectos que estaremos realizando.