

Actividad 4: Cuestionario

Responde a las siguientes cuestiones:

1. ¿Qué es el motor de renderizado de un navegador? Pon ejemplos:

El motor de renderizado es un software que toma contenido en lenguaje de marcas tipo HTML, XML, archivos de imágenes, etc. e información de formato tipo CSS, XSL, etc. y luego muestra el contenido ya formateado en la pantalla de aplicaciones. El motor "pinta" en el área de contenido de una ventana, la cual es mostrada en una pantalla. Los motores de renderizado se usan típicamente en navegadores web, pero también en clientes de correo electrónico, u otras aplicaciones que deban mostrar y editar contenidos web.

Algunos de los motores de renderizado más notables son:

- Gecko, utilizado en Mozilla Suite, y otros navegadores como Galeon.
- Trident, el motor de Internet Explorer para Windows.
- KHTML/WebCore, el motor de Konqueror. Antecesor del WebKit.
- Presto, el antiguo motor de Opera.
- Tasman, el motor de Internet Explorer para Mac.
- gzilla, el motor de Dillo.
- GtkHTML, el motor de Links.
- WebKit, el motor de Epiphany, Safari.
- Blink, el nuevo motor de Google Chrome, Microsoft Edge (desde 2019), Opera y Maxthon (se trata de una bifurcación de WebKit).
- Servo, nuevo motor en desarrollo por parte de Mozilla (con el apoyo de Samsung), está siendo optimizado para la arquitectura ARM y la plataforma Android. En desarrollo.

2. Indica la manera de acceder a la consola de depuración de diferentes navegadores:

Chromium y navegadores basados en él tipo Google Chrome o Brave Browser:

Atajo con el teclado:

- Windows y GNU/Linux: Ctrl, Shift y J al mismo tiempo.
- MacOS: Comando, Opción y J al mismo tiempo.

A través del Menú:

1. Selecciona los tres puntos de la esquina superior derecha de tu pantalla.
2. Selecciona "Más herramientas", después "Herramientas para desarrolladores".
3. Se abrirá una nueva ventana, selecciona la pestaña "Consola" que se encuentra arriba.

Mozilla Firefox:

Atajo con el teclado:

- Windows y GNU/Linux: presiona Control, Shift y K al mismo tiempo
- MacOS: presiona Comando, Opción y K al mismo tiempo

Método del menú:

1. Desde el botón en la esquina superior derecha de tu navegador (3 líneas horizontales), selecciona 'Desarrollador'
2. En el menú emergente, selecciona 'Consola Web'

3. Diferencias entre W3C y WHATWG:

Primero hablemos de un poco sobre cada una de las organizaciones de estándares dentro de la WWW.

W3C:

En español, Consorcio World Wide Web, se formó en 1994, y es la comunidad donde las organizaciones, los desarrolladores y el público en general trabajan en conjunto desarrollando recomendaciones y estándares que aseguran el crecimiento de la World Wide Web a largo plazo.

WHATWG:

El Web Hypertext Application Technology Working Group es un consorcio formado en 2004 a raíz de un desacuerdo en un meeting del W3C, después que se anunciara la decisión de pasar a centrarse en XHTML como evolución de HTML. Dicho consorcio está formado por empresas como Apple, la Fundación Mozilla y Opera Software.

La diferencia principal nace por un aparente desprecio, por parte de W3C y según la WHATWG, por las necesidades de los desarrolladores web del mundo real. Esto ha generado a que tengamos a dos rivales luchando por la estandarización web.

4. Muestra con ejemplos algunas diferencias entre HTML y XHTML:

HTML y XHTML son las dos variantes que existen en el lenguaje HTML 5. Las diferencias entre esas dos variantes son básicamente sintácticas: la sintaxis HTML está inspirada en la norma SGML (aunque no la cumple estrictamente), mientras que la sintaxis XHTML está basada en la recomendación XML (aunque tampoco la cumple estrictamente).

En general, la sintaxis XHTML es más "estricta", en el sentido de imponer restricciones en la forma de escribir etiquetas, atributos o valores, mientras que la sintaxis HTML es más "flexible".

Si un documento contiene errores de sintaxis (HTML o XHTML) se dice que es un documento inválido. Si las páginas web se sirven al navegador como application/xhtml+xml, el navegador debe rechazar las páginas inválidas, pero si se sirven al navegador como text/html, los navegadores intentan mostrar el documento aunque contenga errores. Aunque los navegadores a menudo

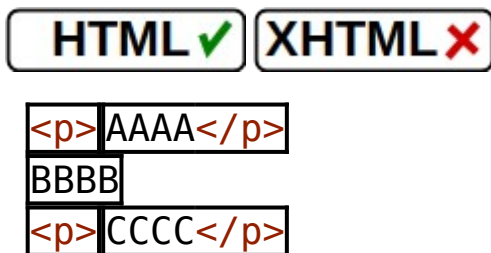
consiguen mostrar documentos inválidos, se aconseja validar y corregir los documentos para asegurar que los navegadores puedan mostrarlos correctamente.

En esta lección se comentan algunas de las diferencias entre las sintaxis HTML y XHTML. En casi todos los casos se puede escribir código válido para ambas sintaxis.

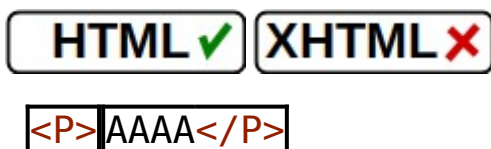
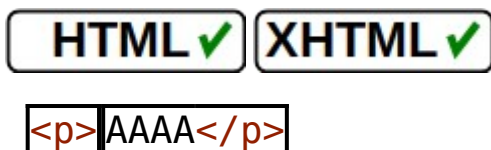
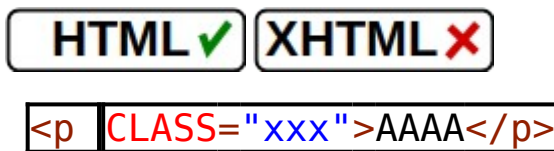
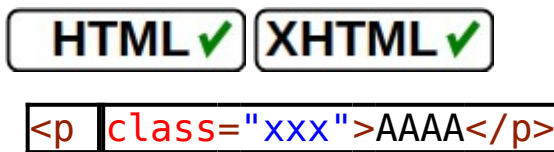
Ejemplos:

En XHTML para que un documento se muestre en el navegador tiene que ser "bien formado", es decir, tiene que cumplir todas las reglas sintácticas. En HTML, los documentos se muestran en el navegador aunque contengan errores sintácticos (la visualización puede ser correcta o incorrecta, dependiendo del tipo de errores de la página).

En XHTML no puede haber texto no incluido en alguna etiqueta, pero en HTML puede haberlo:



En XHTML, los nombres de elementos y atributos deben escribirse en minúsculas. En HTML no es necesario.



En XHTML, las referencias a entidades en hexadecimal deben escribirse en minúsculas. En HTML se pueden escribir en minúsculas o mayúsculas.



```
<p>&#x3b1; &#x3b2; &#x3c0;</p>
```




HTML ✓ **XHTML** ✗

```
<p>&#X3b1; &#X3b2; &#X3c0;</p>
```

Para evitar problemas con los caracteres "<" y "&", en XHTML el código javascript (y de estilo) se suele incluir en secciones CDATA. En HTML no es necesario:

```
<script type="text/javascript">
<![CDATA[
... unescaped script content ...
]]>
</script>
```

Tabla

	Servicio	Protocolo	Protocolo	Ejemplos Servidor	Ejemplos Cliente
	Correo Electronico	POP3 IMAP SMTP		POP3	POP3.gmail.com
	Archivos	FTP		MEGA.nz	Archivos locales
	Web	HTTP y HTTPS		Apache	Chromium