



Carmelo José Jaén Díaz



# Prefijos CSS de los navegadores

- 
- C.F.G.S. DAW
  - 6 horas semanales
  - Mes aprox. de impartición: Ene - Feb
  - iPasen - [cjaedia071@g.educaand.es](mailto:cjaedia071@g.educaand.es)

# Índice



Objetivo

Glosario

Interacción persona - ordenador

Objetivos

Características. Usable.

Características. Visual.

Características. Educativo y actualizado.

# OBJETIVO



- **Analizar y seleccionar los colores y las tipografías adecuados para la visualización en la pantalla.**
- **Utilizar marcos y tablas para presentar la información de manera ordenada.**
- **Identificar nuevos elementos y etiquetas en HTML5.**
- **Reconocer las posibilidades de modificar etiquetas HTML.**
- **Valorar la utilidad de las hojas de estilo para conseguir un diseño uniforme en todo el sitio web.**

# GLOSARIO

---



**Formularios.** Documentos interactivos utilizados para recoger información en un sitio web. Esta información es enviada al servidor, donde es procesada. Cada formulario contiene uno o varios tipos de controles que permiten recolectar la información de varias formas diferentes.

**Fuentes seguras.** Fuentes tipográficas que los usuarios tenían instaladas por defecto en su dispositivo. En la actualidad, gracias a que la mayoría de los navegadores soportan la directiva @font-face, es posible utilizar casi cualquier tipografía a través de Google Fonts.

**Guías de estilo.** Documentos con directrices que permiten la normalización de estilos. En estas guías se recogen los criterios y normas que debe seguir un proyecto; de esta forma se ofrece una apariencia más uniforme y atractiva para el usuario.

**HTML.** Lenguaje de marcado de hipertexto utilizado en las páginas web. Este tipo de lenguaje presenta una forma estructurada y agradable, con hipervínculos que conducen a otros documentos y con inserciones multimedia (sonido, imágenes, vídeos...).

# GLOSARIO

---



**HTML5.** Última versión del lenguaje para la programación de páginas web HTML. Los sitios implementados con este lenguaje solo pueden visualizarse correctamente en los navegadores más actuales.

**Legibilidad.** Cualidad deseable en una familia tipográfica. Se trata de la facilidad de la lectura de una letra. Esta cualidad puede venir determinada por varios parámetros como el interletrado, el interpalabrado o el interlineado.

**Marcos.** Son las ventanas independientes incorporadas dentro de la página general. Gracias a ellos, cada página quedará dividida en varias subpáginas, permitiendo realizar un diseño más organizado y limpio a la vista. Con HTML5 ha quedado obsoleto.

**Tipografía.** Se trata del tipo de letra que se escoge para un determinado diseño. Según la RAE, significa "modo o estilo en que está impreso un texto" o "clase de tipos de imprenta".

# INTRODUCCIÓN

---



El uso de prefijos CSS específicos para navegadores ha disminuido considerablemente en los últimos años. Anteriormente, los prefijos como `-webkit-`, `-moz-`, `-ms-`, y `-o-` eran necesarios para habilitar funciones experimentales o nuevas de CSS en ciertos navegadores. Sin embargo, con la evolución de los estándares web y la mejor compatibilidad entre navegadores, hoy en día la mayoría de las propiedades CSS no requieren estos prefijos.



# INTRODUCCIÓN

---



Aun así, en casos específicos donde una característica aún no está completamente estandarizada o no es compatible uniformemente entre navegadores, algunos prefijos podrían ser necesarios. Sin embargo, es recomendable consultarlos solo cuando sea estrictamente necesario, y usar herramientas como Can I use para verificar la compatibilidad de las propiedades CSS antes de añadir prefijos manualmente.

Además, existen herramientas como autoprefixer, que se encargan de añadir los prefijos automáticamente según las necesidades de compatibilidad de cada proyecto, lo que facilita mucho este proceso.

Veamos qué son estos prefijos y cómo implementarlos:

# PREFIJOS CSS DE LOS NAVEGADORES

---



Los navegadores tienen implementadas algunas de las nuevas características de CSS utilizando sus propias versiones de cada propiedad mediante prefijos. Esto se hace así para evitar los posibles errores ocasionados por las primeras implementaciones que aún no son estables. Por ello, los navegadores proporcionan valores utilizando sus prefijos propios y una declaración sin prefijo.

Después de un tiempo, cuando las especificaciones son estables, se eliminarán las propiedades con prefijo. Los prefijos para los navegadores más comunes son los siguientes:



# PREFIJOS CSS DE LOS NAVEGADORES

---



Prefijo	Navegador
<b>-moz-</b>	Firefox
<b>-webkit-</b>	Safari y Chrome
<b>-o-</b>	Opera
<b>-khtml-</b>	Konqueror
<b>-ms-</b>	Internet Explorer y Microsoft Edge

# PREFIJOS CSS DE LOS NAVEGADORES

---



Así por ejemplo para transformar un elemento en Firefox, es necesario utilizar la propiedad `-moz-transform`; en los navegadores basados en *WebKit*, como Safari y Google Chrome, se utiliza la propiedad `-webkit-transform`.

Por este motivo, no es de extrañar que en algunos casos, tengamos que añadir hasta cuatro líneas de código para una única propiedad CSS.

```
transform: rotate(20deg);  
-moz-transform: rotate(20deg); /* Firefox */  
-webkit-transform: rotate(20deg); /* Safari y Chrome */  
-o-transform: rotate(20deg); /* Opera */  
-ms-transform: rotate(20deg); /* Internet Explorer y Microsoft Edge */
```

# PREFIJOS CSS DE LOS NAVEGADORES

---



Es importante destacar que con el tiempo, muchos navegadores han adoptado estándares y ya no requieren prefijos para las propiedades CSS más comunes. Sin embargo, en algunos casos, especialmente cuando se trabaja con características experimentales, los prefijos todavía pueden ser útiles para garantizar la compatibilidad entre navegadores. Se pueden ver los navegadores que soportan una determinada propiedad CSS o un elemento HTML5 en la página [caniuse.com](http://caniuse.com).

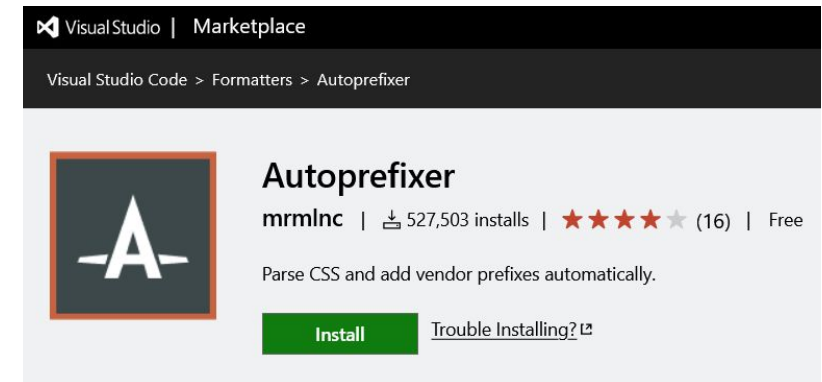
# EXTENSIÓN *Autoprefixer* PARA Visual Studio Code



Para ahorrar tiempo y facilitarnos la tarea de incluir los prefijos de las propiedades CSS que todavía no son estables podemos hacer uso de la extensión *Autoprefixer* en Visual Studio Code.

```
test.scss
1 @mixin flexboxMe() {
2   display: flex;
3 }
4
5 .box {
6   @include flexboxMe();
7 }
8
```

- Add Cursor Above
- Add Cursor Below
- Add Line Comment
- Add Selection To Next Find Match
- Autoprefix CSS**
- Change All Occurrences
- Change End of Line Sequence
- Change File Encoding
- Change Language Mode
- Clear Editor History
- Close Notification Messages
- Close Window
- Configure Language
- Convert Indentation to Spaces
- Convert Indentation to Tabs



# EXTENSIÓN *Autoprefixer* PARA Visual Studio Code



Como se estudió anteriormente, para instalar una extensión en Visual Studio Code tenemos que acceder a *view /extensions*. Una vez instalada la extensión, en *view /command palette* (o *Ctrl+shift+p*) escribimos *autoprefixer run* para añadir todos los prefijos necesarios en una hoja de estilos.

Puedes probar que funciona añadiendo esta propiedad en un estilo CSS y viendo cómo se insertan sus correspondientes prefijos:

```
user-select: none; /* controla si el usuario puede seleccionar el texto */
```

```
-moz-user-select: none; /* Firefox */
```

```
-webkit-user-select: none; /* Safari y Chrome */
```

```
-ms-user-select: none; /* Internet Explorer y Microsoft Edge */
```

*Nota: Desde los ajustes de la propia extensión se puede indicar que se busque si faltan prefijos cada vez que se guarda el documento.*

# EJERCICIO PROPUESTO

---



Realiza la instalación de la extensión y añade rápidamente todos los prefijos necesarios sobre una hoja de estilos que disponga de propiedades CSS que aún no sean estables.