



APRENDERAPROGRAMAR.COM

PREFIJOS CSS DE  
NAVEGADOR: CHROME,  
FIREFOX, INTERNET  
EXPLORER, OPERA,  
SAFARI. -WEBKIT- -MOZ-  
-MS- -O- EJEMPLOS  
(CU01056D)

Sección: Cursos

Categoría: Tutorial básico del programador web: CSS desde cero

Fecha revisión: 2029

**Resumen:** Entrega nº56 del Tutorial básico: "CSS desde cero".

Autor: César Krall

## PREFIJOS CSS DE NAVEGADORES

Se llaman prefijos de navegador o prefijos comerciales (vendedor prefixes) a un prefijo que se antepone a una regla CSS destinado a que dicha regla sea leída y aplicada exclusivamente por un navegador concreto (por ejemplo Chrome) pero no por el resto de navegadores. El uso de prefijos suele aplicarse a propiedades que se encuentran en fase experimental o que aún no se han convertido en un estándar.



Al igual que los comentarios condicionales que se han venido usando específicamente para Microsoft Internet Explorer, los prefijos son un tipo de filtro que permite que una instrucción CSS se aplique específicamente a un navegador o familia de navegadores pero no a los demás. Sin embargo, a diferencia de los comentarios condicionales, existen prefijos específicos para todos los tipos de navegador. A continuación se indican los prefijos más habituales y seguidamente pasaremos a ver un ejemplo.

Prefijo	Familia de navegadores a los que aplica
-webkit-	Chrome, Safari, Android, iOS  
-moz-	Firefox 
-o-	Opera 
-ms-	Microsoft Internet Explorer 

Veamos un ejemplo de uso que es quizás lo que mejor nos haga comprender para qué se usan los prefijos. Escribe este código HTML y CSS (estableciendo el nombre de archivo adecuado) que procedemos a explicar a continuación:

```
<html>
<head>
<title>Portal web - aprenderaprogramar.com</title> <meta charset="utf-8">
<link rel="stylesheet" type="text/css" href="CU01056D_A.css">
</head>
<body>
<div class="fondoGradient"><h1>aprenderaprogramar.com</h1></div>
</body>
</html>
```

```
/* Curso CSS estilos aprenderaprogramar.com*/
*{font-family: arial;}

.fondoGradient {
text-align:center; width: 500px; height: 9em;
line-height: 4.5em; margin: 0 auto;
border: solid 3px;
background-color: #F4FA58; /*Navegadores que no aceptan el gradiente de fondo*/
background-image: -webkit-gradient(linear, left top, left bottom, from(#F4FA58), to(#FF0000)); /* Chrome, Safari versiones
más antiguas */
background-image: -webkit-linear-gradient(top, #F4FA58, #FF0000); /* Chrome, Safari versiones relativamente modernas */
background-image: -moz-linear-gradient(top, #F4FA58, #FF0000); /* Firefox versiones relativamente modernas*/
background-image: -o-linear-gradient(top, #F4FA58, #FF0000); /* Opera versiones relativamente modernas*/
background-image: linear-gradient(to bottom, #F4FA58, #FF0000); /* Chrome, Firefox, IE, Opera versiones actuales */
}
```

Vamos a comentar el código CSS que hemos utilizado y que a primera vista podremos calificar como “poco elegante”, ya que como vemos hay una aparente repetición de instrucciones con distintos prefijos o variantes de sintaxis. La repetición de código nunca es deseable porque dificulta la comprensión y mantenimiento de nuestros desarrollos web. No obstante, tenemos que estar preparados para interpretar este tipo de situaciones cuando estemos analizando una página web. También podemos usar código de este tipo cuando consideremos que es razonadamente necesario, pero debemos evitar su uso continuado y sin reflexión.

El código CSS define una clase donde el texto del elemento se alinea al centro mediante la propiedad `text-align`, la anchura del elemento se establece en 500 píxeles, y su anchura en 9 veces la altura de letra de aplicación. Para centrar verticalmente el texto se ha usado `line-height`. Se ha establecido un borde de 3 píxeles y un color de fondo con código `#F4FA58` que es un tono de amarillo mediante la propiedad `background-color`.

A continuación se escribe una propiedad `background-image` en distintas ocasiones. Recordar que la propiedad `background-image` coloca una imagen encima del `background-color`. Por tanto si la imagen ocupa todo el contenedor, el color de fondo quedará oculto. La primera línea que encontramos es:

`background-image: -webkit-gradient(linear, left top, left bottom, from(#F4FA58), to(#FF0000));` Esta línea aplica el prefijo `-webkit-` a una propiedad denominada `gradient` que crea una imagen de fondo tipo “degradado de colores” a partir de la definición de los colores que deben intervenir y de otros parámetros. El prefijo `webkit` significa que esta línea está dirigida a navegadores Chrome, Safari,

Android, iOS y que el resto de navegadores ignorarán lo que aquí se indique. Sin embargo en la siguiente línea volvemos a encontrar el prefijo webkit de esta manera:

```
background-image: -webkit-linear-gradient(top, #F4FA58, #FF0000);
```

 ¿Por qué se repite dos veces el prefijo webkit?

Como hemos dicho, los prefijos se usan normalmente para propiedades experimentales, que están en fase de diseño o prueba. La sintaxis para la propiedad con el prefijo -webkit- la definen los desarrolladores de estos navegadores y a lo largo del tiempo es posible que cambien la forma en que debe escribirse una regla experimental. Ello da lugar a situaciones como esta: existe una sintaxis que se venía usando en algunas versiones de estos navegadores (que es la definida en la primera línea). Luego la sintaxis fue cambiada y se empezó a usar otra diferente (que es la definida en la segunda línea). Para que la página se visualice como esperamos tanto en navegadores Chrome más antiguos como en navegadores Chrome más modernos, se incluyen ambas formas de la sintaxis. El navegador ignorará aquella que no es adecuada y utilizará la que puede interpretar correctamente.

A continuación nos encontramos con líneas que son exclusivamente de aplicación a navegadores Firefox (prefijo -moz-) y Opera (prefijo -o-).

Finalmente nos encontramos con esta línea:

```
background-image: linear-gradient(to bottom, #F4FA58, #FF0000);
```

 Aquí podemos comprobar que no aparece ningún prefijo. ¿Por qué? Porque a partir de una determinada versión esta propiedad ha dejado de considerarse experimental y se ha convertido en una propiedad "estándar" (al menos para algunos navegadores). Ahora para que los usuarios que usen navegadores en los que esta propiedad se ha normalizado apliquemos esta línea.

Con estas repeticiones de la misma regla con diferentes prefijos o sintaxis buscamos lograr que la visualización sea correcta en:

- Las versiones de navegadores que introdujeron esta propiedad como experimental por primera vez.
- Las versiones de navegadores que mantuvieron la propiedad como experimental pero fueron introduciendo cambios en la sintaxis.
- Las versiones de navegadores que dejaron de considerar a esta propiedad como experimental y la empezaron a emplear como una propiedad normalizada.

De este modo tratamos de cubrir el mayor rango de navegadores posibles para que el mayor número de usuarios posible vean la página web como nosotros esperamos. Ahora bien, ¿qué ocurre con usuarios que empleen versiones de navegadores en los que esta propiedad no existía ni siquiera como experimental? Estos navegadores ignorarán la sintaxis incluso aunque exista un prefijo dirigido a ellos porque no están preparados para reconocer esas instrucciones. Al ignorar background-image aplicarán la propiedad background-color, que sí es reconocida por todos los navegadores. El efecto no será el mismo, pero es la alternativa que se ha dado aquí.

En los navegadores que sí reconozcan el gradiente el resultado será similar a este:



## BUEN USO Y MAL USO DE PREFIJOS

A la hora de usar propiedades experimentales y aplicar prefijos conviene plantearse cuáles son las ventajas y los inconvenientes existentes para cada caso particular. Supongamos que el ejemplo anterior simplemente trataba de aplicar una imagen con gradiente de color al fondo de una caja en la página web. ¿Qué ventajas e inconvenientes presentaría el uso de la propiedad `linear-gradient`?

Ventajas:

- Posiblemente el tiempo de carga de la página sea más rápido usando esta propiedad que usando una imagen. Con esta propiedad el navegador simplemente tiene que renderizar (dibujar) a partir de una instrucción, mientras que con una imagen es necesario descargar el archivo y cada descarga de archivo implica un pequeño consumo de tiempo.

Inconvenientes:

- Al ser una propiedad con poca trayectoria histórica las diferentes versiones de navegadores requieren de distintas sintaxis, lo que obliga a la repetición de varias líneas de código con el mismo fin.
- Podemos tener dudas de que la visualización vaya a ser buena en la mayor parte de dispositivos y navegadores, ya que algunos de ellos (en especial los más antiguos) no reconocerán esta propiedad.

¿Qué será mejor entonces, aplicar o no aplicar este tipo de propiedades? En principio te recomendaríamos que siguieras estos criterios:

- Si dispones de una alternativa altamente estandarizada, simple, que permite alcanzar el objetivo planteado y que evita la repetición de código, úsala y prescinde de propiedades experimentales.
- Si consideras que te aporta más ventajas que inconvenientes y te decides a usarla, trata de cubrir el mayor rango posible de navegadores y versiones y establece una regla "de salvaguarda", es decir, una regla que permita que la página web se vea razonablemente bien si la propiedad que estás intentando aplicar no es admitida por un navegador concreto. En nuestro ejemplo anterior la regla de salvaguarda era `background-color`, alternativa en caso de que fallara la aplicación de `background-image`.

Respecto al uso de prefijos, úsalos sólo cuando sean realmente necesarios. Aquí hemos visto cómo se aplica el prefijo a una propiedad, pero también se aplican en ocasiones a selectores o valores. Antes de aplicar prefijos infórmate en páginas que den información de calidad como Mozilla Developer Network.

Otra cuestión a valorar es si merece escribir las distintas sintaxis para dar soporte a versiones anteriores de navegadores. Cuando la propiedad se ha convertido en un estándar, algunos programadores y diseñadores optan por escribir simplemente la sintaxis estándar junto a la salvaguarda y se olvidan de la repetición de distintas sintaxis para versiones anteriores de navegadores. Es decir, en el ejemplo anterior en lugar de las diversas líneas escribiríamos simplemente:

```
background-color: #F4FA58;
```

```
background-image: linear-gradient(to bottom, #F4FA58, #FF0000);
```

Es decir, se sintetiza en dos líneas: la salvaguarda y la sintaxis estandarizada. En algunos casos ni siquiera será necesario la salvaguarda. Por ejemplo imagínate que pretendes aplicar un borde redondeado y usas una propiedad que ha sido experimental hasta hace poco sin aplicar salvaguarda. Los navegadores que reconozcan la regla mostrarán el borde redondeado, mientras que los que no la reconozcan lo mostrarán rectangular. Si este no es un aspecto clave y no te preocupa, puedes utilizar una sola línea y dejar que se muestre de una forma u otra según el navegador que esté empleando el usuario.

Como ves existen distintas alternativas y recursos para resolver problemas o conseguir objetivos, no hay un criterio único. Algunos equipos de trabajo siguen prefiriendo usar imágenes antes que propiedades avanzadas porque piensan que así se garantizan la misma representación en todos los navegadores, mientras que otros equipos optan por el uso de nuevas propiedades considerando que aportan más ventajas que inconvenientes. Con el tiempo irás formándote tus propios criterios y formas de actuar y razonar en lo relativo al uso de CSS. No pretendas hacer las cosas perfectas, simplemente ocúpate de hacer que las cosas funcionen bien aunque la codificación no sea la mejor posible.

## MÁS FILTROS CSS

Hemos comentado ya dos técnicas de filtrado, una basada en comentarios condicionales y otra en prefijos, pero hay otras técnicas. Algunas de esas otras técnicas las veremos más adelante dentro del curso.

## EJERCICIO

Busca en alguna página web o blog de internet una técnica de filtrado basada en prefijos para el navegador que estés utilizando. Aplícala y explica su utilidad. Para comprobar si tu código y respuestas son correctas puedes consultar en los foros [aprenderaprogramar.com](http://aprenderaprogramar.com).

**Próxima entrega:** CU01057D

**Acceso al curso completo** en [aprenderaprogramar.com](http://aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente:  
[http://aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=75&Itemid=203](http://aprenderaprogramar.com/index.php?option=com_content&view=category&id=75&Itemid=203)