# Hojas de estilo (CSS)

1. Hojas de estilo CSS

1.1. Qué son las Hojas de Estilo (CSS)

1.2. Métodos de uso

1.2.1. Hojas de estilo en línea

1.2.1.1. Para modificar una etiqueta concreta (atributo STYLE)

1.2.1.2. Para modificar un grupo de etiquetas o sección de la página (<DIV>, <SPAN><HEADER><NAV><SECTION><ARTICLE>…)

1.2.2. Hojas de estilo internas o incrustadas en el documento (<STYLE>)

1.2.3. Hojas de estilo externas (<LINK>)

1.3. Mezclando los métodos de uso.

1.4 Ventajas y desventajas de las Hojas de Estilo.

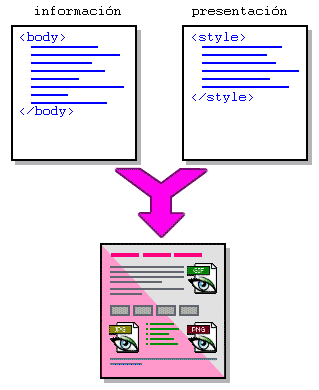
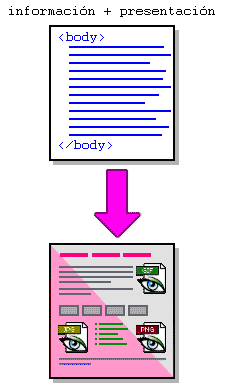
## 1.1. Qué son las Hojas de Estilo (CSS)

Las **Hojas de Estilo** (o **CSS**, por Cascading Style Sheets) son un mecanismo que permiten aplicar formato a los documentos escritos en HTML (y en otros lenguajes estructurados, como XML) **separa**ndo **el contenido de las páginas de su apariencia**.

Fue desarrollado por el Consorcio WWW (W3C) con el objetivo de simplificar el método de presentación de la información, además de ampliar las posibilidades con nuevos efectos para los distintos elementos.

Esta tecnología nace para solventar los problemas que ocasionaron la tendencia de los diseñadores Web, cada vez más exigentes en cuanto al atractivo de sus presentaciones. **El incremento de efectos visuales suponía también añadir más y más etiquetas HTML**, necesarias **para obtener el resultado deseado**. Además **existía** la **limitación** propia de dicho conjunto de etiquetas, **que no conseguían todo el potencial deseado** para los diseños finales.

**CSS** **aporta**, por un lado, **un gran número de efectos antes inalcanzables con el uso de HTML** **y** por el otro, permite realizar una **clara separación entre información y formato de presentación**.



|  |  |
| --- | --- |
| **En** el **código HTML** encontraremos tanto las **etiquetas** **para** la **información** **como para** la **presentación** | **Con CSS** nos valemos de **HTML** **para** definir la **estructura** de la **información**, **y** el **aspecto** se proporciona **mediante** **CSS** |

**Para** el **diseñador**, esto **significa** que la **información** estará **contenida** **en la página HTML**, pero este archivo **no debe definir CÓMO será visualizada** esa información. Las indicaciones acerca de **la composición visual** del documento **estará**n **especificada**s **en el archivo de la CSS**.

A partir de la separación que nos proporciona CSS entre contenido y presentación se deduce que la información puede recopilarse de forma coherente y ordenada, sin preocuparse del aspecto final de la misma. De este modo existe la posibilidad de realizar dos tareas diferenciadas a la hora de desarrollar documentos Web.

* Recopilación y estructuración de la información
* Diseño de la presentación que tendrá dicha información

Otra de las **ventaja**s que nos ofrece **CSS** consiste en la **reutilización** **de** los **estilos** que podremos aplicar **en varios documentos simultáneamente**. Esto supondrá, además de **comodidad** en cuanto a **mantener** una **"imagen de marca"** en varios documentos, unos **códigos fuente más cortos, y** por tanto, documentos **que** **tardarán menos tiempo en descargarse**.

Las posibilidades que ofrecen las hojas de estilo afectarán a los elementos HTML de diversos modos:

• Modificando el formato y las fuentes de los textos

• Dotando a los elementos de colores en primer plano, o colores y/o imágenes para el fondo.

• Proporcionando alineaciones, separaciones entre líneas, distancia entre palabras o letras.

• Agregando márgenes, rellenos de separación, bordes, dimensiones de anchura y altura

• Situando los elementos en diversas posiciones dentro del contexto del documento

• Permitiendo o deshabilitando su visibilidad

Además de poder dotar de estilo a los elementos HTML, **mediante CSS podemos acceder a ciertas partes de los mismos de un modo que antes no era posible.** Así nuestros estilos podrán afectar a:

• Los elementos que se encuentren en una determinada situación dentro del contexto del documento, como por ejemplo los textos en negrita dentro de una lista de elementos.

• Los elementos que definan un determinado atributo HTML.

• Partes concretas del elemento, como la primera línea o la primera letra de un texto.

• Elementos que puedan verse afectados por la interacción con el usuario, como por ejemplo situar el puntero del ratón sobre dicho elemento.

• El fondo del elemento, o del documento completo.

• Algunos aspectos referidos a la información del sistema.

## 1.2. Métodos de uso

Las **hojas de estilo en cascada** nos **permiten** **redefinir** las **reglas** que **utiliza** el **navegador** para **presentar** una **página** del **Web**.

Todos **los navegadores tienen sus reglas**. Como vimos anteriormente, HTML define que las cabeceras del nivel uno (el texto que está entre las etiquetas <H1> y </H1>) se muestre como un texto alineado a la izquierda, un par de puntos mayor que el resto, en negrita y en la fuente “Times New Roman”.

**Cada etiqueta define sus propias reglas** para mostrar el texto que engloba. Con las hojas de estilo podemos cambiar cada una de estas reglas si lo deseamos. Es decir, podemos modificar a nuestro antojo el comportamiento de cada etiqueta.

**Hay tres maneras de añadir estilos a nuestras páginas Web:**

1. Añadiendo instrucciones de estilo sólo a etiquetas concretas, o a un grupo de ellas (**ESTILOS EN LÍNEA**)

2. Incluyendo las instrucciones de estilo en el documento HTML de una página concreta, de una manera análoga a cómo se incluye un script de Javascript. Esto permite cambiar la apariencia de una página entera, cambiando unas pocas líneas (**ESTILOS INCRUSTADOS EN DOCUMENTO**)

1. Enlazando todos los documentos HTML de todas las páginas que componen un sitio del Web con un fichero de definición del estilo. De esta manera, se puede cambiar la apariencia de múltiples páginas retocando un único fichero (**ESTILOS MEDIANTE FICHERO EXTERNO DE HOJA DE ESTILOS**).

### 1.2.1. Hojas de estilo en línea

Este método es el apropiado si sólo se desea cambiar el estilo en unas secciones determinadas de una página. Con él podemos manipular las propiedades de alguna etiqueta o de algún grupo de etiquetas.

#### 1.2.1.1. Para modificar una etiqueta concreta (atributo STYLE)

Se hace añadiendo el **atributo STYLE**, que contendrá una serie de valores dentro de la etiqueta correspondiente.

Los valores que contiene estarán formados por un conjunto de pares

***nombre\_de\_propiedad****:* ***valor***separados unos de otros por ( ; )

<**etiqueta** **style**="propiedad1: valor1; propiedad2: valor2; ...">

...

<**/etiqueta**>

Veamos el siguiente ejemplo:

<P>

"Este es un párrafo normal, sin márgenes a la izquierda ni a la derecha, con el color de su texto en negro".

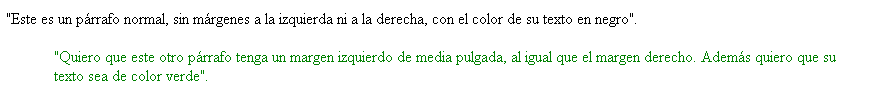
</P>

<**P** **STYLE**="margin-left: 0.5in; margin-right: 0.5in; color:green">

"Quiero que este otro párrafo tenga un margen izquierdo de media pulgada, al igual que el margen derecho. Además quiero que su texto sea de color verde".

</P>

Que se vería de la siguiente manera:



Como se ve en este ejemplo, hemos variado la manera de cómo se comporta habitualmente la etiqueta <P>. Podíamos haber conseguido otras muchas más cosas, como se verá más adelante.

Obsérvese **la sintaxis** **de**l atributo **STYLE**. **Se ponen entre comillas una serie de propiedades** (por ejemplo, margin-left), **y después de dos puntos se pone el** **valor** de esa propiedad (en este caso, 0.5in o sea, 0.5pulgadas). **Las distintas propiedades deberán estar separadas por punto y coma**.

Ejemplo: ...

<p style=”color: red;">Párrafo de texto en color rojo</p>

<p>Párrafo de texto sin color</p>

EJEMPLO\_1:

<html>

<head>

</head>

<body>

<h1 style="color:#ff0000;background-color:#ffff00">

Este mensaje es de color rojo sobre fondo amarillo.

</h1>

</body>

</html>

Por defecto, todo navegador tiene un estilo definido para cada marca HTML, lo que hacemos con la propiedad style es redefinir el estilo por defecto. En este problema definimos que la marca h1 defina como color de texto el rojo y como color de fondo el amarillo:

<h1 style="color:#ff0000;background-color:#ffff00">

Este mensaje es de color rojo sobre fondo amarillo.

</h1>

Veremos más adelante que hay muchas propiedades en CSS. En este primer ejemplo inicializamos las propiedades color (define el color del texto) y background-color (define el color de fondo del texto).  
Cada vez que inicializamos una propiedad debemos separarla de la siguiente por punto y coma.  
Cuando definimos estilos directamente en las marcas HTML, tenemos que tener en cuenta que el estilo se aplica únicamente a la marca donde inicializamos la propiedad style, es decir, si tenemos dos secciones h1, deberemos definir la propiedad style para cada marca:

<h1 style="color:#ff0000;background-color:#ffff00">

Primer título

</h1>

<h1 style="color:#ff0000;background-color:#ffff00">

Segundo título

</h1>

Como podemos observar, más allá que los dos estilos son exactamente iguales, estamos obligados a definirlos para cada marca HTML.

EJEMPLO\_2:

<!..El objetivo de esta pequeña página es ver cómo afecta el estilo definido a la marca p (párrafo) >

<html>

<head>

</head>

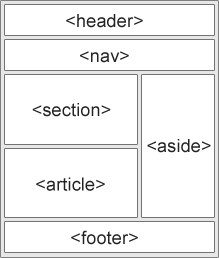
<body>

<p **style="color:#000000;background-color:yellow;font-family:verdana;font-size:25px;text-align:center"**>Esto es un ejemplo</p>

</body>

</html>

#### 1.2.1.2. Para modificar un grupo de etiquetas o sección de la página <div><span><header><nav><section><article><aside><footer>)



Si queremos cambiar la apariencia de una sección entera (que agrupe un conjunto de etiquetas), se puede utilizar las **etiquetas <DIV> <header>, <section>, <nav>…..**, con la que definiremos el estilo globalmente para esa sección.

Por supuesto, la definición del estilo dentro de la propia etiqueta no es la manera más eficaz de utilizar las hojas de estilo, pero pueden existir casos que lo justifiquen.

Esta forma de aplicar la hoja de estilo directamente en la etiqueta es similar al modo usado en el código HTML, pero con dos importantes diferencias:

1. El conjunto de **propiedades** que se pueden **aplicar** es mucho **mayor**
2. Los estilos especificados en una hoja de estilo (cualquiera sea su origen) **tienen preferencia** por **sobre los formatos aplicados con HTML**.

Esto significa que si en nuestra hoja de estilo hemos definido: **P {text-align: left}**

Todos los párrafos quedarán alineados por la izquierda y, si en un párrafo determinado, usamos: **<P align="center">**

El párrafo seguirá estando alineado por la izquierda porque las reglas de las CSS tienen más fuerza que el formato aplicado con HTML.

EJEMPLO:

<html>

<head>

<title>Ejemplo </title>

<style type="text/css">

p{color:red; text-align: right}

</style>

</head>

<body>

<h1>Titular de la página</h1>

<p align="center">Un párrafo de texto no muy largo.</p>

</body>

</html>

Realizar un ejercicio con las propiedades del ejemplo anterior utilizando<DIV style =……..>. Qué diferencia hay con el anterior?.

### 1.2.2. Hojas de estilo internas o incrustadas en el documento (<STYLE>)

También existe la posibilidad de poner la hoja de estilo **dentro** de una página HTML. Se hace introduciendo la **etiqueta <STYLE> .. </STYLE>**, que deberá estar colocada dentro de la cabecera del documento, entre las etiquetas </TITLE> y </HEAD>.

Este método permite aplicar la hoja de estilo solamente a la página que la contiene. Resulta útil cuando en nuestro sitio tenemos algunas pocas páginas que usan un formato distinto al resto.

En muchos casos convendrá usar ambos métodos simultáneamente: poner un link a una hoja externa para aplicar un estilo general y luego crear una hoja incrustada en la que solamente deberemos definir las reglas específicas para esa página. Por las leyes de cascada de las CSS, en caso de existir una misma regla (una en la hoja externa, otra en la hoja incrustada) con distintos valores, tiene prioridad la definida en la hoja incrustada.

La etiqueta **<STYLE>** tiene parámetros:

* **TYPE**

Especifica el tipo de contenido de la hoja de estilos, por ejemplo "**text/css**". Esto indica que el contenido del archivo consiste en definiciones de reglas de estilo CSS que permitirá a los navegadores que no soporten este tipo el ignorar la hoja de estilo.

* **MEDIA**

Especifica el medio sobre el que se verá la información. Es muy importante en el sentido de que dependiendo del lugar donde se represente la información se verá mejor o peor. Los distintos valores soportados son:

* + - **Screen**

Pensado para ordenadores personales.

* + - **Tty**

Pensado para teletipos, terminales, portátiles, etc.

* + - **Tv**

Pensado para televisores.

* + - **Projection**

Proyectores.

* + - **Handheld**

Dispositivos de pequeñas dimensiones como teléfonos móviles. Agendas, etc.

* + - **Print**

Pensado para impresión.

* + - **Braille**

Pensado para dispositivos táctiles braille.

* + - **Aural**

Para sintetizadores de voz.

* + - **All**

Adecuado para todo tipo de dispositivos.

La estructura será la siguiente:

<HTML>

<HEAD>

<TITLE> </TITLE>

<**STYLE** **TYPE**="**text/css**">

[Bloque de instrucciones de estilo]

etiqueta1 HTML { propiedad1: valor1; propiedad2: valor2; }

etiqueta2 HTML { propiedad3: valor3; propiedad4: valor4; }

<**/STYLE**>

</HEAD>

<BODY>

[conjunto de todas las etiquetas que componen la página]

</BODY>

</HTML>

Cada vez que se use una de las etiquetas que tienen una definición de estilo, no se comportará de la forma habitual, sino que lo hará tal como lo hemos definido. El resto de las etiquetas, que no están incluidas en el bloque de definición del estilo, se comportarán de la forma habitual.

Se pueden definir tantos estilos como se deseen en la cabecera de la página. Por ejemplo una página no se verá igual en un monitor color que en uno monocromo. Por ello podemos hacer por ejemplo:

***<HTML>***

***<HEAD>***

***<TITLE> P****ágina variada* ***</TITLE>***

***<STYLE TYPE****="****text/css****"* ***MEDIA="Projection">***

***font.color1*** *{* ***color****:black;* ***background****:white; }*

***font.color2*** *{* ***color****:red;* ***background****:black; }*

***font.color3*** *{* ***color****:yellow;* ***background****:black; }*

***</STYLE>***

***<STYLE TYPE****="****text/css****"* ***MEDIA="Printer">***

***font.color1*** *{* ***color****:black;* ***background****:white; }*

***font.color2*** *{* ***color****:black;* ***background****:white; }*

***font.color3*** *{* ***color****:black;* ***background****:white; }*

***</STYLE>***

***</HEAD>***

***<BODY****>*

*.... bla, bla, bla...*

***</BODY>***

***</HTML>***

EJEMPLO:

<html>

<head>

<title>Medios</title>

<style type="text/css">

**@media print {**

**p { font-family: "Times New Roman"; color: black; }**

**}**

**@media screen {**

**p { font-family: Verdana; color: red; }**

**}**

</style>

</head>

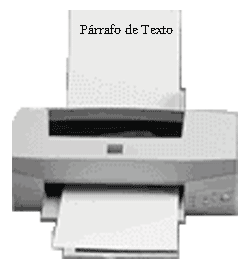
<body>

<p>Párrafo de texto</p>

</body>

</html>

El resultado es el siguiente:



A la hora de imprimir el contenido tendrá el aspecto definido en la regla-arroba, es decir, el párrafo aparecerá en color negro y utilizando la fuente "Times New Roman".

Al representar el documento en pantalla seguirá los estilos definidos en esta regla-arroba, es decir, el párrafo en la pantalla será de color rojo y empleando la fuente "Verdana".

EJEMPLO\_1

Problema: Mostrar dos títulos con texto de color rojo sobre fondo amarillo.

<html> <head> <title>Problema</title> <style type="text/css"> h1 {color:#ff0000;background-color:#ffff00} </style> </head> <body> <h1>Primer título</h1> <h1>Segundo título</h1> </body> </html> Podemos observar que en la cabecera de la página definimos la sección: <style type="text/css"> h1 {color:#ff0000;background-color:#ffff00} </style> Debe estar encerrada por la marca style. En este ejemplo indicamos al navegador que en todos los lugares de esta página donde se utilice la marca h1 debe aplicar como estilo de color de texto el rojo y fondo el amarillo. Podemos observar que es mucho más eficiente que definir los estilos directamente sobre las marcas HTML.

* Este método solo permite aplicar la hoja de estilo sólo a la página que la contiene.

### 1.2.3. Hojas de estilo externas (<LINK>)

Si nuestro sitio del Web está compuesto por muchas páginas, y queremos darles un estilo uniforme a unas cuantas, en vez de incluir un bloque de definición de estilo repetido en cada una de ellas, se puede establecer un enlace a un fichero de texto que contiene el bloque de definición del estilo.

Para utilizar este método debemos escribir las reglas de estilo en un fichero externo. Éste no tendrá ninguna etiqueta, pues no es un documento normal HTML, sino que es un fichero de texto que **sólo contiene** **las reglas de estilos**. Debe tener necesariamente la **extensión .CSS**

Para hacer uso de este fichero de definición de estilos en todas las páginas que queramos, solamente deberemos añadir entre </TITLE> y </HEAD>, la **etiqueta <LINK>**, que proporciona un método para definir relaciones con otros documentos y recursos.

Para acceder desde un documento HTML a la página de estilos externa se hace mediante la etiqueta <link> situada en la cabecera del documento entre la etiqueta <head> y </head>. del siguiente modo:

**<head>**

**<link rel="stylesheet" type="text/css" href="fichero\_de\_estilos">**

**</head>**

Esta etiqueta posee los atributos:

* **REL (relation)**

Indica el tipo de enlace entre el documento y el recurso especificado en el atributo HREF.

Puede tomar distintos valores, pero para el caso que nos ocupa usaremos el valor "**stylesheet**", que especifica que el archivo incorporado es una hoja de estilos.

* **TYPE**

Indica que el contenido del archivo consiste en definiciones de reglas de estilo CSS, y tomará el valor "**text/css**"

* **HREF**

Indica la URL donde se encuentra el fichero externo que define las reglas de estilo que se emplearán en el documento HTML.

Un ejemplo sería:

<**LINK HREF**="**miestilo.css**" **REL**="**stylesheet**" **TYPE**="**text/css**">

Aquí se ha supuesto que el fichero de estilos se ha dejado en el mismo directorio donde están los documentos HTML. Si no es así, es decir, si está en otro subdirectorio en el servidor, habría que indicar la ruta (por, ejemplo: HREF="subdirectorio/miestilo.css").

Para utilizar este método debemos escribir las reglas de estilo en un fichero externo con la extensión **.css**

Este archivo no contendrá la estructura <style> . . . .</style> tan sólo las reglas de estilos.

EJEMPLO:

Ahora veremos la primera página HTML que tiene asociada una hoja de estilo en un archivo externo. El archivo HTML es (pagina1.html):

<html>

<head>

<title>Problema</title>

<link rel="StyleSheet" href="estilos.css" type="text/css">

</head>

<body>

<h1>Definición de hojas de estilo en un archivo externo.</h1>

<p>

Hasta ahora hemos visto la definición de estilos a nivel de marca HTML y la definición de estilos a nivel de página. Dijimos que la primera forma es muy poco recomendada y la segunda es utilizada cuando queremos definir estilos que serán empleados solo por esa página.

Ahora veremos que la metodología más empleada es la definición de una hoja de estilo en un archivo separado que deberá tener la extensión css.

</p>

</body>

</html>

El archivo que tiene las reglas de estilo es (estilos.css):

body {

background-color:#eafadd;

}

h1 {

color:#0000cc;

font-family:times new roman;

font-size:25px;

text-align:center;

text-decoration:underline;

}

p {

color:#555555;

font-family:verdana;

text-align:justify;

}

Como podemos observar, para indicar el archivo de estilos externo, debemos agregar en la cabecera (head) del archivo HTML la siguiente marca:

**<link rel="StyleSheet" href="estilos.css" type="text/css">**

**La propiedad href hace referencia al archivo externo que afectará la presentación de esta página. En la propiedad type, indica al navegador cual es el formato de archivo. El atributo rel se usa para definir la relación entre el archivo enlazado y el documento HTML.**

**1.3. Mezclando los métodos de uso**

Como hemos visto, hay tres métodos de aplicar las hojas de estilo en cascada.

***¿Se pueden aplicar más de uno a la vez en una misma página?***

**SÍ**

Para evitar conflictos entre los distintos métodos usados simultáneamente, existe un **orden de precedencia**, es decir, cuál prevalecerá sobre el otro si dan órdenes contradictorias sobre un aspecto concreto.

El orden de precedencia es el siguiente (de mayor a menor):

1. **Estilo dentro** de una **etiqueta**
2. **Bloque** de **estilo** **en** la **cabecera** del documento
3. **Enlace** **a** un **fichero** que contiene la **hoja** **de** **estilo**.

Esto quiere decir que una orden de estilo, por ejemplo el color del texto, puesta dentro de una etiqueta, prevalecerá sobre la que esté indicada en los otros dos métodos, si es que se usan en esa página.

Lo más práctico es utilizar el tercer método (enlace a una hoja de estilo), para dar una apariencia consistente a todas las páginas, y si es necesario modificar un aspecto concreto en alguna de ellas, utilizando el primer o segundo método.

**1.4. Ventajas (y desventajas) de las Hojas de Estilo**

Si consideramos lo visto anteriormente, resultan evidentes algunas **ventajas**:

1. Con una Hoja de Estilo podemos alterar la presentación de cada elemento sin tocar el código HTML, ahorrando esfuerzo y tiempo de edición. Si quisiéramos alinear a la izquierda los encabezados H1 de nuestras páginas, bastaría con cambiar en la CSS la declaración "text-align: center" por "text-align: left" e inmediatamente cada H1 se alinearía a la izquierda en *todas* las páginas vinculadas a la Hoja de Estilo. De este modo no sólo **simplificamos el mantenimiento** del sitio sino que además **reducimos** las **posibilidad**es de **cometer errores**.
2. CSS ofrece herramientas de **composición más potentes que HTML**. Se puede especificar una fuente alternativa genérica (en HTML no existen estas fuentes genéricas). Con HTML, el tamaño de la fuente se especifica con un sistema de medidas predeterminadas por el navegador (en el ejemplo, SIZE=5), con las CSS se puede especificar el tamaño en puntos tipográficos (y podemos hacerlo en cm, pixeles, cuadratines, altura de la x, etc.). Más aún, las CSS permiten aplicar prácticamente todas las propiedades a cualquier elemento de la página, mientras que HTML sólo permite un número limitado de propiedades para cada elemento.
3. Se **evita** tener que **recurrir** a **trucos** para **conseguir** algunos **efectos**. Con CSS no es necesario usar imágenes invisibles para hacer una sangría (la propiedad text-indent se encarga de eso) o usar una tabla para ubicar un elemento en determinado lugar de la pantalla (las CSS permiten posicionar con precisión cualquier elemento).
4. El **lenguaje** de las Hojas de Estilo, aunque **muy potente**, es relativamente **sencillo** y **fácil** de **aprender**.
5. Los documentos que usan CSS generalmente resultan más compactos.
6. Las Hojas de Estilo pueden **aplicarse** de varias maneras y combinarse **formando** una **cascada de estilos** con la información de cada una.
7. Pueden **usarse con** otros **lenguajes** de **programación** (como JavaScript) **para conseguir efectos dinámicos** en las páginas.
8. Se pueden especificar **Hojas de Estilo para diferentes navegadores y** tipos de **medios** (impresos, braille, auditivos, etc.).

En cuanto a las **desventajas** en el uso de las Hojas de Estilo, la única de importancia es el **soporte irregular** que tienen las CSS por parte de los **navegadores**. Ciertas propiedades que funcionan en un navegador no funcionan en otros, o existen diferencias en un mismo navegador según sea para Windows o Mac. También existen diferencias entre distintas versiones de un mismo navegador.

Debe entenderse que las Hojas de Estilo fueron diseñadas para permitir que los autores *influyan* en la composición de la página, pero no para que la controlen. Una **CSS *sugiere* al navegador un estilo** de composición para el documento, **pero no puede forzarlo** a aplicar un formato determinado.

* Las hojas de estilo CSS pueden validarse a través de la web de W3C: <http://jigsaw.w3.org/css-validator>. Si no es correcto nos muestra información de cómo validarlo.