

Hojas de estilo

Nota: Este documento es parte de una traducción al castellano de la Recomendación del W3C "HTML 4.01 Specification" ([más información](#)). Puede consultar la [versión original](#) del mismo.

Contenidos

1. [Introducción a las hojas de estilo](#)
2. [Añadir estilos al HTML](#)
 1. [Especificación del lenguaje de hojas de estilo por defecto](#)
 2. [Información de estilo en línea](#)
 3. [Información de estilo en cabecera: el elemento `STYLE`](#)
 4. [Tipos de medios](#)
3. [Hojas de estilo externas](#)
 1. [Hojas de estilo preferentes y alternativas](#)
 2. [Especificación de hojas de estilo externas](#)
4. [Hojas de Estilo en Cascada](#)
 1. [Cascadas dependientes del medio](#)
 2. [Herencia y cascada](#)
5. [Ocultar datos de estilo a los agentes de usuario](#)
6. [Vinculación de hojas de estilo mediante encabezados HTTP](#)

14.1 Introducción a las hojas de estilo

Las hojas de estilo representan un avance importante para los diseñadores de páginas web, al darles un mayor rango de posibilidades para mejorar la apariencia de sus páginas. En los entornos científicos en que la Web fue concebida, la gente estaba más preocupada por el contenido de sus páginas que por su presentación. A medida que la Web era descubierta por un espectro mayor de personas de distintas procedencias, las limitaciones del HTML se convirtieron en fuente de continua frustración, y los autores se vieron forzados a superar las limitaciones estilísticas del HTML. Aunque las intenciones han sido buenas -- mejorar la presentación de las páginas web --, las técnicas para conseguirlo han tenido efectos secundarios negativos. Entre estas técnicas, que dan buenos resultados para algunas personas, algunas veces, pero no siempre ni para todas las personas, se incluyen:

- La utilización de extensiones propietarias del HTML
- Conversión del texto en imágenes
- Utilización de imágenes para controlar el espacio en blanco
- La utilización de tablas para la organización de las páginas
- Escribir programas en lugar de usar HTML

Estas técnicas incrementan considerablemente la complejidad de las páginas web, ofrecen una flexibilidad limitada, sufren de problemas de interoperabilidad, y crean dificultades para las personas con discapacidades.

Las hojas de estilo resuelven estos problemas al mismo tiempo que reemplazan al limitado rango de mecanismos de presentación del HTML. Con las hojas de estilo es más fácil especificar la cantidad de espacio entre líneas, el sangrado de las líneas, los colores a utilizar para el texto y el fondo, el tamaño y estilo de las fuentes, y otros muchos detalles.

Por ejemplo, la siguiente hoja de estilo CSS (almacenada en el fichero "especial.css") hace que el color del texto de un párrafo sea verde ("green") y lo rodea de un borde rojo ("red") continuo ("solid"):

```
P.especial {
color : green;
border: solid red;
}
```

Los autores pueden vincular esta hoja de estilo a su documento fuente HTML con el elemento [LINK](#):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<HTML>
  <HEAD>
    <TITLE>Un documento con una hoja de estilo externa</TITLE>
    <LINK href="especial.css" rel="stylesheet" type="text/css">
  </HEAD>
  <BODY>
    <P class="especial">Este párrafo debería tener texto especial verde.
  </BODY>
</HTML>
```

HTML 4 soporta las siguientes características de hojas de estilo:

Colocación flexible de la información de estilo

Al colocar las hojas de estilo en ficheros separados es más fácil reutilizarlas. Algunas veces es útil incluir instrucciones de representación dentro del documento al que se aplican, ya sea agrupadas al comienzo del documento, o en atributos de los elementos a lo largo del cuerpo del documento. Para facilitar la gestión de estilos a nivel de sitio, esta especificación describe cómo utilizar los encabezados HTTP para especificar qué hojas de estilo se aplican a un documento.

Independencia de lenguajes de hojas de estilo específicos

Esta especificación no liga el HTML con ningún lenguaje de hojas de estilo en particular. Esto permite que se puedan usar otros lenguajes diferentes, desde los lenguajes más simples válidos para la mayoría de los usuarios, hasta los más complejos, útiles para una minoría de usuarios con necesidades muy especializadas. Todos los ejemplos que se incluyen en lo que sigue se sirven del lenguaje CSS (Hojas de Estilo en Cascada, Cascading Style Sheets, [\[CSS1\]](#)), pero también son posibles otros lenguajes de hojas de estilo.

Cascada

Se llama así a la capacidad que proporcionan algunos lenguajes de hojas de estilo tales como CSS que permiten que las informaciones de estilo provenientes de varias fuentes puedan combinarse. Éstas podrían ser, por ejemplo, las guías de estilo de una empresa, los estilos comunes a un grupo de documentos, y los estilos específicos de un documento en particular. Al almacenarse independientemente, las hojas de estilo pueden reutilizarse, lo cual simplifica las tareas de diseño y hace más efectiva la utilización de la memoria caché de la red. La cascada define una secuencia ordenada de hojas de estilo en la que las reglas de las últimas hojas tienen una prioridad mayor que las de las primeras. No todos los lenguajes de hojas de estilo soportan la cascada.

Dependencias de los medios

HTML permite a los autores especificar sus documentos de una manera independiente del medio. Esto permite a los usuarios acceder a las páginas web usando una amplia gama de dispositivos y medios, p.ej., pantallas gráficas para ordenadores bajo Windows, Macintosh OS y X11, dispositivos para aparatos de televisión, teléfonos adaptados y dispositivos portátiles PDA, navegadores basados en voz, y dispositivos táctiles Braille.

Las hojas de estilo, en contraste, se aplican a medios o grupos de medios específicos. Una hoja de estilo diseñada para una pantalla puede ser aplicable para una salida impresa, pero es de poca utilidad para los navegadores basados en voz. Esta especificación le permite definir categorías generales de medios a los que es aplicable una hoja de estilo dada. Esto permite a los agentes de usuario evitar la descarga de hojas de estilo que no sean apropiadas. Los lenguajes de hojas de estilo pueden incluir características para describir dependencias del medio dentro de una misma hoja de estilo.

Estilos alternativos

Los autores pueden querer ofrecer a los lectores varias maneras diferentes de ver un documento. Por ejemplo, una hoja de estilo para representar documentos compactos con fuentes pequeñas, o una que especifique fuentes más grandes para una lectura más fácil. Esta especificación permite a los autores especificar una hoja de estilo preferente así como hojas alternativas que se dirijan a medios o usuarios específicos. Los agentes de usuario deberían dar a los usuarios la oportunidad de seleccionar una de las hojas de estilo alternativas o incluso de desactivar las hojas de estilo completamente.

Consideraciones de rendimiento

Algunas personas han expresado su preocupación acerca de los posibles problemas de rendimiento relacionados con las hojas de estilo. Por ejemplo, la obtención de una hoja de estilo externa puede retrasar la presentación del documento al usuario. Algo parecido sucede si la cabecera del documento contiene un conjunto muy grande de reglas de estilo.

La propuesta actual resuelve estos problemas, pues permite a los autores incluir instrucciones de representación dentro de cada elemento HTML. Así, la información de representación siempre estará disponible en el momento en que el agente de usuario quiera representar cada elemento.

En muchos casos los autores sacarán partido de una hoja de estilo común a un grupo de documentos. En este caso, la distribución de reglas de estilo a lo largo del documento conducirá a un rendimiento peor que si se usara una hoja de estilo vinculada, ya que para la mayoría de los documentos la hoja de estilo ya estará almacenada en la caché local. La disponibilidad pública de hojas de estilo de calidad potenciará este efecto.

14.2 Añadir estilos al HTML

Nota. La hoja de estilo por defecto del HTML 4 que se incluye en [\[CSS2\]](#) expresa la información de estilo por defecto generalmente aceptada para cada elemento. Los autores, así como los implementadores, pueden encontrar en ella un recurso útil.

Los documentos HTML pueden contener reglas de hojas de estilo directamente, o bien pueden importar hojas de estilo.

Se puede usar cualquier lenguaje de hojas de estilo con HTML. Un lenguaje simple de hojas de estilo puede ser suficiente para la mayoría de los usuarios, pero para otras necesidades más

especializadas pueden ser apropiados otros lenguajes. Esta especificación utiliza para los ejemplos el lenguaje de estilo llamado "Hojas de Estilo en Cascada" ("Cascading Style Sheets", [\[CSS1\]](#)), abreviado CSS.

La [sintaxis de los datos de estilo](#) depende del lenguaje de hojas de estilo.

14.2.1 Especificación del lenguaje de hojas de estilo por defecto

Los autores deben especificar el lenguaje de hojas de estilo de la información de estilo asociada con un documento HTML.

Los autores deberían usar el elemento [META](#) para especificar el lenguaje de hojas de estilo por defecto de un documento. Por ejemplo, para especificar que el valor por defecto es CSS, los autores deberían poner la siguiente declaración en la sección [HEAD](#) de sus documentos:

```
<META http-equiv="Content-Style-Type" content="text/css">
```

El lenguaje de hojas de estilo por defecto también puede ser establecido con encabezados HTTP. La declaración [META](#) de arriba es equivalente al encabezado HTTP siguiente:

```
Content-Style-Type: text/css
```

Los agentes de usuario deberían determinar el lenguaje de hojas de estilo por defecto de acuerdo con los siguientes pasos (ordenados de mayor a menor prioridad):

1. Si alguna declaración [META](#) especifica "Content-Style-Type", la última de ellas en el flujo de caracteres determina el lenguaje de hojas de estilo por defecto.
2. De otro modo, si algún encabezado HTTP especifica "Content-Style-Type", el último de ellos en el flujo de caracteres determina el lenguaje de hojas de estilo por defecto.
3. De otro modo, el lenguaje de hojas de estilo por defecto es "text/css".

Los documentos que incluyan elementos que especifiquen el atributo [style](#) pero que no definan un lenguaje de hojas de estilo por defecto son incorrectos. Las herramientas de creación deberían generar información de lenguaje de hojas de estilo por defecto (normalmente una declaración [META](#)) de modo que los agentes de usuario no tengan que atenerse al valor por defecto "text/css".

14.2.2 Información de estilo en línea

Definiciones de atributos

`style = estilo [CN]`

Este atributo especifica información de estilo para el elemento actual.

La [sintaxis](#) del valor del atributo [style](#) viene determinada por el [lenguaje de hojas de estilo por defecto](#). Por ejemplo, para los estilos `[[CSS2]]` en línea, se usa la sintaxis de bloque de declaraciones descrita en la sección 4.1.8 (sin las llaves).

Este ejemplo CSS establece información sobre el color y el tamaño de la fuente del texto de un párrafo específico.

```
<P style="font-size: 12pt; color: fuchsia">¿No son maravillosas las hojas de
estilo?
```

En CSS, las declaraciones de propiedades son de la forma "nombre : valor" y están separadas por un punto y coma.

Para especificar información de estilo en línea para más de un elemento, los autores deberían usar el elemento `STYLE`. Para lograr una flexibilidad óptima, los autores deberían definir los estilos en hojas de estilo externas.

14.2.3 Información de estilo en cabecera: el elemento `STYLE`

```
<!ELEMENT STYLE - - %StyleSheet -- información de estilo -->
<!ATTLIST STYLE
  %i18n; -- lang, dir, para usar con title --
  type %ContentType; #REQUIRED -- tipo de contenido del lenguaje de
estilos --
  media %MediaDesc; #IMPLIED -- diseñado para usar con estos medios -
-
  title %Text; #IMPLIED -- título consultivo --
>
```

*Etiqueta inicial: **obligatoria**, Etiqueta final: **obligatoria***

Definiciones de atributos

`type` = [tipo de contenido \[CI\]](#)

Este atributo especifica el lenguaje de hojas de estilo de los contenidos del elemento y prevalece sobre el lenguaje de hojas de estilo por defecto. El lenguaje de hojas de estilo se especifica como un tipo de contenido (p.ej., "text/css"). Los autores deben proporcionar un valor para este atributo; no hay valor por defecto para este atributo.

`media` = [descriptores de medios \[CI\]](#)

Este atributo especifica el medio destino al que se dirige la información de estilo. Puede ser un solo descriptor de medios, o una lista de ellos separados por comas. El valor por defecto de este atributo es "screen" (pantalla).

Atributos definidos en otros lugares

- [lang](#) ([información sobre el idioma](#)), [dir](#) ([dirección del texto](#))
- [title](#) ([título del elemento](#))

El elemento `STYLE` permite a los autores poner reglas de hojas de estilo en la cabecera del documento. HTML permite cualquier número de elementos `STYLE` en la sección `HEAD` de un documento.

Los agentes de usuario que no soporten hojas de estilo, o que no soporten el lenguaje de hojas de estilo utilizado por un elemento `STYLE` en particular, deben ocultar los contenidos del elemento `STYLE`. Es un error representar el contenido como parte del texto del documento. Algunos lenguajes de hojas de estilo soportan sintaxis para [ocultar el contenido](#) a agentes de usuarios no conformes.

La [sintaxis de los datos de estilo](#) depende del lenguaje de hojas de estilo.

Algunas implementaciones de hojas de estilo pueden permitir una gama más amplia de reglas en el elemento [STYLE](#) que en el atributo [style](#). Por ejemplo, con CSS, pueden declararse reglas dentro de un elemento [STYLE](#) para:

- Todas las apariciones de un elemento de HTML específico (p.ej., todos los elementos [P](#), todos los elementos [H1](#), etc.)
- Todas las apariciones de un elemento de HTML que pertenezcan a una clase específica (es decir, cuyo atributo [class](#) tome cierto valor).
- Las apariciones únicas de un elemento de HTML (es decir, cuyo atributo [id](#) tome cierto valor).

Las reglas de precedencia y herencia de las reglas de estilo dependen del lenguaje de hojas de estilo.

La siguiente declaración [STYLE](#) CSS pone un borde alrededor de todos los elementos [H1](#) del documento y los centra en la página.

```
<HEAD>
  <STYLE type="text/css">
    H1 {border-width: 1px; border: solid; text-align: center}
  </STYLE>
</HEAD>
```

Para especificar que esta información de estilo sólo debería aplicarse a los elementos [H1](#) de una clase específica, la modificamos como sigue:

```
<HEAD>
  <STYLE type="text/css">
    H1.miclasse {border-width: 1px; border: solid; text-align: center}
  </STYLE>
</HEAD>
<BODY>
  <H1 class="miclasse"> Este H1 está afectado por nuestro estilo </H1>
  <H1> Este no está afectado por nuestro estilo </H1>
</BODY>
```

Finalmente, para limitar el alcance de la información de estilo a una sola aparición de [H1](#), establecemos el atributo [id](#):

```
<HEAD>
  <STYLE type="text/css">
    #miid {border-width: 1px; border: solid; text-align: center}
  </STYLE>
</HEAD>
<BODY>
  <H1 class="miclasse"> Este H1 no está afectado </H1>
  <H1 id="miid"> Este H1 está afectado por el estilo </H1>
  <H1> Este H1 no está afectado </H1>
</BODY>
```

Aunque se puede especificar información de estilo prácticamente para cualquier elemento, hay dos elementos, [DIV](#) y [SPAN](#), que son particularmente útiles en el sentido de que no imponen ningún

significado presentacional (aparte de la distinción [en bloque/en línea](#)). Combinados con hojas de estilo, estos elementos permiten a los usuarios extender el HTML indefinidamente, en especial si se utilizan junto con los atributos [class](#) e [id](#).

En el siguiente ejemplo, utilizamos el elemento [SPAN](#) para especificar que el estilo de fuente de las primeras palabras de un párrafo sea versalitas ("small-caps").

```
<HEAD>
  <STYLE type="text/css">
    SPAN.ver-ex { font-variant: small-caps }
  </STYLE>
</HEAD>
<BODY>
  <P><SPAN class="ver-ex">Las primeras</SPAN> palabras
    de este párrafo van en versalitas.
</BODY>
```

En el siguiente ejemplo, utilizamos el elemento [DIV](#) y el atributo [class](#) para establecer la justificación del texto de una serie de párrafos que constituyen el abstracto de un artículo científico. Esta información de estilo podría reutilizarse para otras secciones de abstracto especificando el valor correspondiente del atributo [class](#) en cualquier otro lugar del documento.

```
<HEAD>
  <STYLE type="text/css">
    DIV.Abstracto { text-align: justify }
  </STYLE>
</HEAD>
<BODY>
  <DIV class="Abstracto">
    <P>La gama de productos Gran Jefe es nuestra apuesta
      para el año que viene. Este informe sienta las bases
      para confrontar a Gran Jefe con los productos de la
      competencia.

    <P>Gran Jefe reemplaza a nuestra gama "Commander",
      si bien ésta seguirá de momento en catálogo.
    </DIV>
</BODY>
```

14.2.4 Tipos de medios

HTML permite a los autores diseñar documentos que saquen partido de las características de los medios en los que el documento vaya a ser representado (p.ej., pantallas gráficas, pantallas de televisión, dispositivos de mano, navegadores basados en voz, dispositivos táctiles Braille, etc.). Al especificar el atributo [media](#), los autores permiten a los agentes de usuario cargar y aplicar las hojas de estilo de manera selectiva. Consulte la lista de [descriptores de medios](#) reconocidos.

Las siguientes declaraciones de ejemplo se aplican a elementos [H1](#). Cuando se proyecten en una reunión de negocios, todos ellos aparecerán en azul. Cuando se impriman, aparecerán centrados.

```
<HEAD>
  <STYLE type="text/css" media="projection">
    H1 { color: blue }
  </STYLE>
```

```
<STYLE type="text/css" media="print">
  H1 { text-align: center }
</STYLE>
```

Este ejemplo añade efectos de sonido a los vínculos para su uso en salidas por voz:

```
<STYLE type="text/css" media="aural">
  A { cue-before: uri(bell.aiff); cue-after: uri(dong.wav) }
</STYLE>
</HEAD>
```

El control de medios es particularmente interesante cuando se aplica a hojas de estilo externas, ya que los agentes de usuario pueden ahorrar tiempo si obtienen de la red únicamente aquellas hojas de estilo que se apliquen el dispositivo actual. Por ejemplo, los navegadores basados en voz pueden evitar la descarga de hojas de estilo diseñadas para la representación visual. Véase la sección sobre [cascadas dependientes del medio](#) para más información.

14.3 Hojas de estilo externas

Los autores pueden separar las hojas de estilo de los documentos HTML. Esto ofrece varias ventajas:

- Los autores y los administradores de sitios web pueden compartir hojas de estilo entre varios documentos (y sitios).
- Los autores pueden cambiar la hoja de estilo sin necesidad de hacer modificaciones en el documento.
- Los agentes de usuario pueden cargar hojas de estilo selectivamente (en función de los descriptores de medios).

14.3.1 Hojas de estilo preferentes y alternativas

HTML permite a los autores asociar cualquier número de hojas de estilo externas a un documento. El lenguaje de hojas de estilo define el modo en que interaccionan varias hojas de estilo (por ejemplo, las reglas de "cascada" de CSS).

Los autores pueden especificar cualquier número de hojas de estilo mutuamente excluyentes llamadas hojas de estilo *alternativas*. Los usuarios pueden seleccionar su hoja favorita según sus preferencias. Por ejemplo, un autor puede especificar una hoja de estilo diseñada para pantallas pequeñas y otra para usuarios con poca visión (p.ej., con fuentes grandes). Los agentes de usuario deberían permitir a los usuarios seleccionar las hojas de estilo alternativas.

El autor puede especificar que una de las alternativas es una hoja de estilo *preferente*. Los agentes de usuario deberían aplicar la hoja de estilo preferente del autor a menos que el usuario haya seleccionado una alternativa diferente.

Los autores pueden agrupar varias hojas de estilo alternativas (incluyendo las hojas de estilo preferentes del autor) bajo un *nombre de estilo* común. Cuando un usuario selecciona un nombre de estilo, el agente de usuario debe aplicar todas las hojas de estilo con ese nombre. Los agentes de usuario no deben aplicar hojas de estilo alternativas con un nombre de estilo diferente. La sección sobre [especificación de hojas de estilo externas](#) explica cómo dar un nombre a un grupo de hojas de estilo.

Los autores también pueden especificar hojas de estilo *persistentes* que los agentes de usuario deben aplicar además de cualquier hoja de estilo alternativa.

Los agentes de usuario deben respetar los [descriptores de medios](#) al aplicar cualquier hoja de estilo.

Los agentes de usuario también deberían permitir a los usuarios deshabilitar completamente las hojas de estilo del autor, en cuyo caso el agente de usuario no debería aplicar ninguna hoja de estilo persistente ni alternativa.

14.3.2 Especificación de hojas de estilo externas

Los autores especifican hojas de estilo externas mediante los siguientes atributos del elemento [LINK](#):

- Haciendo el valor de [href](#) igual a la localización del fichero de la hoja de estilo. El valor de [href](#) es un [URI](#).
- Haciendo que el valor del atributo [type](#) indique el lenguaje del recurso (hoja de estilo) vinculado. Esto permite al agente de usuario evitar la descarga de una hoja de estilo cuyo lenguaje no soporte.
- Especificando si la hoja de estilo es persistente, preferente, o alternativa:
 - Para hacer una hoja de estilo persistente, se hace el atributo [rel](#) igual a "stylesheet" y no se establece el atributo [title](#).
 - Para hacer una hoja de estilo preferente, se hace el atributo [rel](#) igual a "stylesheet" y se da un nombre a la hoja de estilo mediante el atributo [title](#).
 - Para especificar una hoja de estilo alternativa, se hace el atributo [rel](#) igual a "alternate stylesheet" y se da un nombre a la hoja de estilo mediante el atributo [title](#).

Los agentes de usuario deberían posibilitar a los usuarios ver la lista de estilos alternativos y escoger uno de ellos. Se recomienda que el nombre de cada opción sea el valor del atributo [title](#).

En este ejemplo, especificamos en primer lugar una hoja de estilo persistente localizada en el fichero `miestilo.css`:

```
<LINK href="miestilo.css" rel="stylesheet" type="text/css">
```

Si establecemos el atributo [title](#) la convertimos en la hoja de estilo preferente del autor:

```
<LINK href="miestilo.css" title="compacto" rel="stylesheet" type="text/css">
```

Si añadimos la palabra clave "alternate" al atributo [rel](#) la convertimos en una hoja de estilo alternativa:

```
<LINK href="miestilo.css" title="Medium" rel="alternate stylesheet" type="text/css">
```

Para más información sobre hojas de estilo externas, consulte la sección sobre [vínculos y hojas de estilo externas](#).

Los autores también pueden usar el elemento [META](#) para establecer la hoja de estilo preferente del documento. Por ejemplo, para especificar que la hoja de estilo preferente sea "compacto" (ver el ejemplo precedente), los autores pueden incluir la siguiente línea en el [HEAD](#):

```
<META http-equiv="Default-Style" content="compacto">
```

La hoja de estilo preferente también se puede especificar mediante encabezados HTTP. La declaración [META](#) precedente es equivalente a este encabezado HTTP:

```
Default-Style: "compacto"
```

Si hay dos o más declaraciones [META](#) o encabezados HTTP que especifiquen la hoja de estilo preferente, la que prevalece es la última. A estos efectos, se considera que los encabezados HTTP aparecen antes que la sección [HEAD](#) del documento.

Si hay dos o más elementos [LINK](#) que especifiquen una hoja de estilo preferente, el que prevalece es el primero.

Las hojas de estilo preferentes especificadas con un elemento [META](#) o con encabezados HTTP prevalecen sobre las especificadas con el elemento [LINK](#).

14.4 Hojas de Estilo en Cascada

Los lenguajes de hojas de estilo en *cascada*, tales como CSS, permiten que se pueda combinar la información de varias fuentes. Sin embargo, no todos los lenguajes de hojas de estilo soportan la cascada. Para definir una cascada, los autores especifican una secuencia de elementos [LINK](#) y/o [STYLE](#). La información de estilo se combina en cascada según el orden en que aparecen los elementos en la sección [HEAD](#).

Nota. Esta especificación no especifica cómo se combinan en cascada hojas de estilo de lenguajes de hojas de estilo distintos. Los autores deberían evitar mezclar lenguajes de hojas de estilo.

En el siguiente ejemplo, especificamos dos hojas de estilo alternativas llamadas "compacto". Si el usuario selecciona el estilo "compacto", el agente de usuario debe aplicar ambas hojas de estilo externas, así como la hoja de estilo persistente "comun.css". Si el usuario selecciona el estilo "tipos grandes", sólo se aplicarán la hoja de estilo alternativa "tgrandes.css" y "comun.css".

```
<LINK rel="alternate stylesheet" title="compacto"
      href="peq-base.css" type="text/css">
<LINK rel="alternate stylesheet" title="compacto"
      href="peq-extras.css" type="text/css">
<LINK rel="alternate stylesheet" title="tipos grandes"
      href="tgrandes.css" type="text/css">
<LINK rel="stylesheet" href="comun.css" type="text/css">
```

Aquí tenemos un ejemplo de cascada que incluye tanto al elemento [LINK](#) como al [STYLE](#).

```
<LINK rel="stylesheet" href="empresa.css" type="text/css">
<LINK rel="stylesheet" href="informe.css" type="text/css">
<STYLE type="text/css">
  p.especial { color: rgb(230, 100, 180) }
```

</STYLE>

14.4.1 Cascadas dependientes del medio

Una cascada puede incluir hojas de estilo aplicables a medios diferentes. Tanto [LINK](#) como [STYLE](#) pueden usarse con el atributo [media](#). El agente de usuario es por tanto responsable de filtrar aquellas hojas de estilo que no se apliquen al medio actual.

En el siguiente ejemplo, definimos un cascada en la cual se incluyen varias versiones de la hoja de estilo "empresa": una diseñada para su impresión, otra para la pantalla y otra para navegadores basados en voz (útil, por ejemplo, para leer correo electrónico en el coche). La hoja de estilo "informe" se aplica a todos los medios. La regla de color definida por el elemento [STYLE](#) se usa para impresoras y pantallas, pero no para la representación auditiva.

```
<LINK rel="stylesheet" media="aural" href="empresa-aural.css" type="text/css">
<LINK rel="stylesheet" media="screen" href="empresa-screen.css"
type="text/css">
<LINK rel="stylesheet" media="print" href="empresa-print.css" type="text/css">
<LINK rel="stylesheet" href="informe.css" type="text/css">
<STYLE media="screen, print" type="text/css">
    p.especial { color: rgb(230, 100, 180) }
</STYLE>
```

14.4.2 Herencia y cascada

Cuando el agente de usuario quiere representar un documento, necesita encontrar valores para las propiedades de estilo, p.ej., la familia tipográfica, el tamaño de fuente, la altura de línea, el color del texto, etc. El mecanismo exacto depende del lenguaje de la hoja de estilo, pero en general es válida la descripción siguiente:

Se utiliza el mecanismo de cascada cuando varias reglas de estilo se aplican directamente a un mismo elemento. El mecanismo permite al agente de usuario ordenar las reglas según su especificidad para determinar la regla a aplicar. Si no encuentra ninguna regla, el siguiente paso depende de si la propiedad de estilo puede ser heredada o no. No todas las propiedades se heredan. Para estas propiedades el lenguaje de la hoja de estilo proporciona unos valores por defecto que son los que se usan cuando no hay reglas explícitas para un elemento en particular.

Si la propiedad puede ser heredada, el agente de usuario examina el elemento contenedor inmediato para ver si hay alguna regla que se aplique a él. Este proceso continúa hasta que se encuentra alguna regla aplicable. Este mecanismo permite que las hojas de estilo puedan definirse de una manera compacta. Por ejemplo, los autores pueden especificar la familia tipográfica de todos los elementos contenidos en [BODY](#) con una sola regla que se aplique al elemento [BODY](#).

14.5 Ocultar datos de estilo a los agentes de usuario

Algunos lenguajes de hojas de estilo soportan sintaxis pensada para permitir a los autores ocultar el contenido de los elementos [STYLE](#) a los agentes de usuario no conformes.

Este ejemplo ilustra cómo comentar en CSS el contenido de los elementos [STYLE](#) para asegurarse de que los agentes de usuario antiguos y no conformes no lo representarán como texto.

```
<STYLE type="text/css">
<!--
  H1 { color: red }
  P  { color: blue}
-->
</STYLE>
```

14.6 Vinculación de hojas de estilo mediante encabezados HTTP

Esta sección sólo se aplica a los agentes de usuario conformes con las versiones de HTTP que definan un campo de encabezado Link. Obsérvese que HTTP 1.1, tal y como se define en [\[RFC2616\]](#), no incluye un campo de encabezado Link (ver sección 19.6.3).

Los administradores de servidores web pueden crear conveniente configurar un servidor de modo que una hoja de estilo se aplique a un grupo de páginas. El encabezado HTTP Link tiene el mismo efecto que un elemento [LINK](#) con los mismos atributos y valores. Varios encabezados Link corresponderían a varios elementos [LINK](#) que aparecieran en el mismo orden. Por ejemplo,

```
Link: <http://www.acme.com/empresa.css>; REL=stylesheet
```

corresponde a:

```
<LINK rel="stylesheet" href="http://www.acme.com/empresa.css">
```

Es posible especificar varias hojas de estilo alternativas especificando varios encabezados Link, y a continuación utilizar el atributo [rel](#) para determinar el estilo por defecto.

En el ejemplo siguiente, se aplica "compacto" por defecto, ya que omite la palabra clave "alternate" del atributo [rel](#).

```
Link: <compacto.css>; rel="stylesheet"; title="compacto"
Link: <tgrandes.css>; rel="alternate stylesheet"; title="tipos grandes"
```

Esto debería funcionar cuando se envían documentos HTML por correo electrónico. Algunos agentes de correo pueden alterar el orden de los encabezados de [\[RFC822\]](#). Para evitar que esto afecte al orden de cascada de las hojas de estilo especificadas con encabezados Link, los autores pueden usar la concatenación de encabezados para combinar varias apariciones del mismo campo de encabezado. Las comillas sólo se necesitan si los valores de los atributos contienen algún espacio en blanco. Utilice entidades SGML para hacer referencia a caracteres que no se permitan en encabezados de HTML o de correo electrónico, o que sean susceptibles de resultar afectados por el tránsito a lo largo de la red.

Los elementos [LINK](#) y [META](#) declarados implícitamente por encabezados HTTP aparecen antes de cualquier elemento explícito [LINK](#) o [META](#) en el [HEAD](#) del documento.

Alineación, estilos de fuente y separadores horizontales

Nota: Este documento es parte de una traducción al castellano de la Recomendación del W3C "HTML 4.01 Specification" ([más información](#)). Puede consultar la [versión original](#) del mismo.

Contenidos

1. [Formato](#)
 1. [Color de fondo](#)
 2. [Alineación](#)
 3. [Objetos flotantes](#)
 - [Hacer flotar a un objeto](#)
 - [Texto flotante alrededor de un objeto](#)
2. [Fuentes](#)
 1. [Elementos de estilo de fuente: los elementos TT, I, B, BIG, SMALL, STRIKE, S y U](#)
 2. [Elementos de modificación de fuentes: FONT y BASEFONT](#)
3. [Separadores: el elemento HR](#)

Esta sección de la especificación trata sobre algunos elementos y atributos HTML que pueden utilizarse para el formato visual de elementos. Muchos de ellos están [desaprobados](#).

15.1 Formato

15.1.1 Color de fondo

Definiciones de atributos

`bgcolor` = [color](#) [[CI](#)]

Desaprobado. Este atributo establece el color de fondo del cuerpo del documento o de las celdas de una tabla.

Este atributo establece el color de fondo del lienzo del cuerpo del documento (el elemento [BODY](#)) o de las tablas (los elementos [TABLE](#), [TR](#), [TH](#) y [TD](#)). Con el elemento [BODY](#) pueden utilizarse atributos adicionales para especificar el color del texto.

Este atributo ha sido [desaprobado](#) en favor de las hojas de estilo para la especificación de información sobre el color de fondo.

15.1.2 Alineación

Es posible alinear elementos en bloque (tablas, imágenes, objetos, párrafos, etc.) en el lienzo con el atributo `align`. Aunque este atributo se puede establecer en muchos elementos HTML, su rango de valores posibles difiere en ocasiones de unos elementos a otros. Aquí sólo se considera el significado del atributo `align` para texto.

Definiciones de atributos

`align = left | center | right | justify` [\[CI\]](#)

Desaprobado. Este atributo especifica la alineación horizontal de su elemento con respecto al contexto circundante. Valores posibles:

- `left`: las líneas de texto se representan con los bordes izquierdos alineados.
- `center`: las líneas de texto son centradas.
- `right`: las líneas de texto se representan con los bordes derechos alineados.
- `justify`: las líneas de texto se justifican en ambos márgenes.

El valor por defecto depende de la dirección base del texto. Para un texto de izquierda a derecha el valor por defecto es `align=left`, mientras que para un texto de derecha a izquierda el valor por defecto es `align=right`.

EJEMPLO DESAPROBADO:

Este ejemplo centra un encabezado en el lienzo.

```
<H1 align="center"> Cómo Tallar la Madera </H1>
```

Usando CSS, por ejemplo, podríamos conseguir el mismo efecto de la manera siguiente:

```
<HEAD>
  <TITLE>Cómo Tallar la Madera</TITLE>
  <STYLE type="text/css">
    H1 { text-align: center }
  </STYLE>
<BODY>
  <H1> Cómo Tallar la Madera </H1>
```

Obsérvese que esto centrará todas las declaraciones [H1](#). Podríamos reducir el alcance del estilo estableciendo el atributo [class](#) del elemento:

```
<HEAD>
  <TITLE>Cómo Tallar la Madera</TITLE>
  <STYLE type="text/css">
    H1.madera {text-align: center}
  </STYLE>
<BODY>
  <H1 class="madera"> Cómo Tallar la Madera </H1>
```

EJEMPLO DESAPROBADO:

Análogamente, para alinear a la derecha un párrafo en el lienzo con el atributo de HTML [align](#) podríamos tener:

```
<P align="right">...Montones de párrafos de texto...
```

lo cual, con CSS, se convertiría en:

```
<HEAD>
  <TITLE>Cómo Tallar la Madera</TITLE>
  <STYLE type="text/css">
    P.mipar {text-align: right}
```

```
</STYLE>
<BODY>
  <P class="mipar">...Montones de párrafos de texto...
```

EJEMPLO DESAPROBADO:

Para alinear a la derecha una serie de párrafos, los agrupamos con el elemento [DIV](#):

```
<DIV align="right">
  <P>...texto del primer párrafo...
  <P>...texto del segundo párrafo...
  <P>...texto del tercer párrafo...
</DIV>
```

Con CSS, la propiedad text-align se hereda del elemento padre a los hijos, y por tanto podemos decir:

```
<HEAD>
  <TITLE>Cómo Tallar la Madera</TITLE>
  <STYLE type="text/css">
    DIV.parrafos {text-align: right}
  </STYLE>
<BODY>
  <DIV class="parrafos">
    <P>...texto del primer párrafo...
    <P>...texto del segundo párrafo...
    <P>...texto del tercer párrafo...
  </DIV>
```

Para centrar el documento completo con CSS:

```
<HEAD>
  <TITLE>Como Tallar la Madera</TITLE>
  <STYLE type="text/css">
    BODY {text-align: center}
  </STYLE>
<BODY>
  ...el cuerpo está centrado...
</BODY>
```

El elemento CENTER es exactamente equivalente a especificar el elemento [DIV](#) con el atributo [align](#) igual a "center". **El elemento [CENTER](#) está [desaprobado](#).**

15.1.3 Objetos flotantes

Las imágenes y los objetos pueden aparecer directamente "en línea" o pueden hacerse "flotar" a un lado de la página, alterando temporalmente los márgenes del texto, que puede fluir por los lados del objeto.

Hacer flotar a un objeto

El atributo [align](#) aplicado a objetos, imágenes, tablas, marcos, etc., hace que el objeto flote hasta el margen izquierdo o derecho. Los objetos flotantes en general comienzan una nueva línea. Este atributo acepta los siguiente valores:

- `left`: Hace flotar al objeto hasta el margen izquierdo actual. El texto subsiguiente fluye alrededor del borde derecho del objeto.
- `right`: Hace flotar al objeto hasta el margen derecho actual. El texto subsiguiente fluye alrededor del borde izquierdo del objeto.

EJEMPLO DESAPROBADO:

El siguiente ejemplo muestra cómo hacer flotar un elemento [IMG](#) hasta el margen izquierdo actual del lienzo.

```
<IMG align="left" src="http://blabla.com/unaimagen.gif" alt="mi barca">
```

Algunos atributos de alineación también permiten el valor "center", que no hace que el objeto flote, sino que lo centra con respecto a los márgenes actuales. Sin embargo, para [P](#) y [DIV](#), el valor "center" hace que los contenidos del elemento estén centrados.

Texto flotante alrededor de un objeto

Otro atributo, definido para el elemento [BR](#), controla el flujo del texto alrededor de objetos flotantes.

Definiciones de atributos

`clear = none | left | right | all` [\[CI\]](#)

Desaprobado. Especifica dónde debería aparecer la línea siguiente en un navegador visual después del salto de línea provocado por este elemento. Este atributo tiene en cuenta los objetos flotantes (imágenes, tablas, etc.). Valores posibles:

- `none`: La siguiente línea empezará normalmente. Este es el valor por defecto.
- `left`: La siguiente línea comenzará en la primera línea que quede por debajo de los objetos que estén flotando en el margen izquierdo.
- `right`: La siguiente línea comenzará en la primera línea que quede por debajo de los objetos que estén flotando en el margen derecho.
- `all`: La siguiente línea comenzará en la primera línea que quede por debajo de los objetos que estén flotando en cualquiera de los dos márgenes.

Consideremos el siguiente escenario visual, en el cual el texto fluye por la derecha de una imagen hasta que la línea se rompe con un [BR](#):

```
*****  -----
|          |  -----
| imagen   |  --<BR>
|          |
*****  -----
```

Si el atributo `clear` se hace igual a `none`, la línea que siga al [BR](#) comenzará inmediatamente debajo de él en el margen derecho de la tabla:

```
*****  -----
|          |  -----
| imagen   |  --<BR>
|          |  -----
```

EJEMPLO DESAPROBADO:

Si el atributo [clear](#) se hace igual a `left` o `all`, la siguiente línea aparecerá como se muestra a continuación:

```
***** -----
| imagen | -----
|         | --<BR clear="left">
|         |
*****
-----
```

Usando hojas de estilo, podríamos especificar que todos los saltos de línea deberían comportarse de esta manera para los objetos (imágenes, tablas, etc.) que estuvieran flotando en el margen izquierdo. Con CSS podríamos conseguirlo de esta manera:

```
<STYLE type="text/css">
BR { clear: left }
</STYLE>
```

Para especificar este comportamiento para un elemento [BR](#) específico, podríamos combinar la información de estilo y el atributo [id](#):

```
<HEAD>
...
<STYLE type="text/css">
BR#mibr { clear: left }
</STYLE>
</HEAD>
<BODY>
<P>...
***** -----
| tabla | -----
|         | --<BR id="mibr">
|         |
*****
-----
...
</BODY>
```

```

<html>
<head>
<title>Ejemplo CSS 2</title>
<style type="text/css">
    .menutopbg {
        background-image: url("http://www.transexpress.com.sv/upload/pwb1/prac230309/menubg.jpg");
        background-repeat: repeat-x;
        background-position: bottom;
    }
    a.menutop {
        background-image: url("http://www.transexpress.com.sv/upload/pwb1/prac230309/menutem.jpg");
        background-repeat: repeat-x;
        background-position: bottom;
        display: block;
        height: 26px;
        font-family: "Trebuchet MS", Tahoma, Arial;
        font-size: 10px;
        font-weight: bold;
        color: #14148c;
        text-decoration: none;
    }
    a.menutop:hover {
        background-image: url("http://www.transexpress.com.sv/upload/pwb1/prac230309/menutemon.jpg");
        background-repeat: repeat-x;
        background-position: bottom;
        color: #916dc8;
        text-decoration: none;
    }
    #menutable {
        border: 1px solid #999999;
        font-size: 10px;
    }
</style>
</head>
<body>
<table width="50%" height="45" cellpadding="0" cellspacing="0" id="menutable" align="center">
    <tr>
        <td width="50" align="center" valign="bottom" class="menutopbg">
            <a href="pag1.htm" class="menutop">Pag 1 </a></td>
            <td width="2" align="center" valign="bottom" class="menutopbg">&nbsp;</td>
            <td width="75" align="center" valign="bottom" class="menutopbg">
                <a href="pag2.htm" class="menutop">Pag 2</a></td>
                <td width="2" align="center" valign="bottom" class="menutopbg">&nbsp;</td>
                <td width="83" align="center" valign="bottom" class="menutopbg">
                    <a href="pag3.htm" name="link3" class="menutop" id="link1">Pag 3 </a></td>
                    <td width="2" align="center" valign="bottom" class="menutopbg">&nbsp;</td>
                    <td width="100" align="center" valign="bottom" class="menutopbg">
                        <a href="Pag4.htm" class="menutop">Pag 4 </a></td>
                        <td align="left" valign="top" class="menutopbg">&nbsp;</td>
        </tr>
        <tr>
            <td colspan="8">
                <table>
                    <tr>
                        <td>
                            <td>
                                <h1>Registrese!</h1>
                            </td>
                        </tr>
                        <tr>
                            <td>
                                Comience a gozar de los beneficios de su casillero internacional.
                            </td>
                        </tr>
                    </tr>
                </table>
            </td>
        </tr>
    </table>
</body>
</html>

```