

**UF1303: Elaboración de hojas de estilo**

Elaborado por: Lorena Rodríguez Cortés

Edición: 5.0

**EDITORIAL ELEARNING S.L.**

ISBN: 978-84-16424-26-9 • Depósito legal: MA 508-2015

No está permitida la reproducción total o parcial de esta obra bajo cualquiera de sus formas gráficas o audiovisuales sin la autorización previa y por escrito de los titulares del depósito legal.

Impreso en España - Printed in Spain

# Presentación

## Identificación de la Unidad Formativa

Bienvenido a la Unidad Formativa **UF1303: Elaboración de hojas de estilo**. Esta Unidad Formativa pertenece al Módulo Formativo **MF0950\_2: Construcción de páginas web** que forma parte del Certificado de Profesionalidad **IFCD0110: Confección y publicación de páginas web**, de la familia de Informática y comunicaciones.

## Presentación de los contenidos

La finalidad de esta Unidad Formativa es enseñar al alumno a crear hojas de estilo para homogeneizar el aspecto de las páginas y hacerlas más atractivas, según especificaciones de diseño y desarrollo recibidas. Para ello, se estudiará en primer lugar la función de las hojas de estilo en la construcción de páginas web y posteriormente se analizará el diseño, ubicación y optimización de los contenidos de una página web.

## Objetivos de la Unidad Formativa

Al finalizar esta Unidad Formativa aprenderás a:

- Describir las características de las hojas de estilo para dar formato a las páginas web y crear ficheros de estilo de acuerdo a un diseño especificado.
- Diseñar, ubicar y optimizar los contenidos de una página web, para adecuarla al formato de la misma, facilitar su manejo a los usuarios y optimizarla de acuerdo a un diseño especificado.

# Índice

UD1. Hojas de estilo en la construcción de páginas web .....	9
1.1. Funciones y características .....	11
1.1.1. Descripción de estilos .....	12
1.1.2. Utilización de estilos .....	13
1.1.3. Los estilos en el lenguaje de marcas.....	16
1.1.4. Los estilos con herramientas de edición web .....	18
1.2. Hojas de estilo y accesibilidad.....	23
1.2.1. Adecuación de las hojas de estilos .....	26
1.3. Tipos de estilo: incrustados, enlazados,importados, en línea..	31
1.3.1. Descripción de los tipos de estilo .....	33
1.3.2. Enlazar una hoja de estilo externa a un documento HTML .....	34
1.3.3. Incrustar un estilo dentro de un documento HTML.....	39
1.3.4. Importar una hoja de estilo desde un documento HTML .....	42
1.3.5. Importar una hoja de estilo a través de un archivo con estilos.....	43
1.3.6. Utilización y optimización de los tipos de estilos.....	44
1.4. Selectores y reglas de estilo .....	45

1.4.1. Estructura de los estilos.....	46
1.4.2. Sintaxis básica de estilos.....	48
1.4.3. Utilización de elementos y seudoelementos .....	54
1.4.4. Utilización de clases y seudoclases .....	56
1.5. Atributos de estilo para fuentes, color y fondo, texto y bloques (párrafos).....	60
1.5.1. Descripción de los atributos de estilo.....	62
1.5.2. Utilización de los atributos de estilo .....	70
1.6. Creación de ficheros de estilo .....	247
1.6.1. Definición de los ficheros de estilo .....	248
1.6.2. Creación de ficheros de estilo genéricos .....	250
1.6.3. Adaptación de los ficheros de estilo para distintas páginas web .....	257
 UD2. Diseño, ubicación y optimización de los contenidos de una página web .....	269
2.1. Creación de un documento funcional .....	271
2.1.1. Descripción de los elementos funcionales de la página web.....	274
2.1.2. Definición de los elementos funcionales de la página web.....	279
2.1.3. Descripción de cada elemento .....	286
2.2. Diseño de los contenidos.....	289
2.3. Identificación de la información a ubicar en la página web....	297
2.4. Selección de contenidos para cada elemento de la página..	312
2.5. Utilización del documento funcional para las especificaciones del diseño .....	318
2.6. Tipos de página para la ubicación de contenidos .....	341
2.7. Definición de los tipos de página en base a los contenidos y funcionalidades.....	346
2.8. Selección de los tipos de página para la página web.....	350

# Índice

2.9. Utilización del documento funcional para las especificaciones del tipo de página .....	351
2.10. Especificaciones de navegación .....	353
2.11. Creación de un mapa de navegación de páginas .....	357
2.12. Utilización del documento funcional para integrar el mapa de navegación .....	360
2.13. Elementos utilizados para la navegación .....	368
2.13.1. Definición de los elementos utilizados para navegar ..	368
2.13.2. Utilización del documento funcional para especificar los elementos de navegación .....	370
2.14. Elaboración de una guía de usuario .....	372
 Glosario .....	385
 Soluciones .....	389
 Anexo .....	391

Área: informática y comunicaciones

# UD1

Hojas de estilo en  
la construcción de  
páginas web

- 1.1. Funciones y características
  - 1.1.1. Descripción de estilos
  - 1.1.2. Utilización de estilos
  - 1.1.3. Los estilos en el lenguaje de marcas
  - 1.1.4. Los estilos con herramientas de edición web
- 1.2. Hojas de estilo y accesibilidad
  - 1.2.1. Adecuación de las hojas de estilos
- 1.3. Tipos de estilo: incrustados, enlazados, importados, en línea
  - 1.3.1. Descripción de los tipos de estilo
  - 1.3.2. Enlazar una hoja de estilo externa a un documento HTML
  - 1.3.3. Incrustar un estilo dentro de un documento HTML.
  - 1.3.4. Importar una hoja de estilo desde un documento HTML
  - 1.3.5. Importar una hoja de estilo a través de un archivo con estilos
  - 1.3.6. Utilización y optimización de los tipos de estilos
- 1.4. Selectores y reglas de estilo
  - 1.4.1. Estructura de los estilos
  - 1.4.2. Sintaxis básica de estilos
  - 1.4.3. Utilización de elementos y seudoelementos
  - 1.4.4. Utilización de clases y seudoclases
- 1.5. Atributos de estilo para fuentes, color y fondo, texto y bloques (párrafos)
  - 1.5.1. Descripción de los atributos de estilo
  - 1.5.2. Utilización de los atributos de estilo
- 1.6. Creación de ficheros de estilo
  - 1.6.1. Definición de los ficheros de estilo
  - 1.6.2. Creación de ficheros de estilo genéricos
  - 1.6.3. Adaptación de los ficheros de estilo para distintas páginas web

## 1.1. Funciones y características

Las hojas de estilo o CSS (Cascading Style Sheets) es la tecnología que nos permite controlar las propiedades visuales de los elementos que componen una página Web. Mientras HTML se preocupa por la estructura de los contenidos, CSS se hace cargo de la apariencia y el estilo.

CSS trabaja en un documento HTML modificando sus propiedades, tales como el color, el borde, el posicionamiento y el tamaño, de una forma muy precisa y exacta, los que da al desarrollador Web un control máximo sobre dónde y cómo se visualizará esa página.

A este gran control sobre las propiedades visuales de la página, debemos sumar otra gran contribución de CSS al desarrollo web: este nos permite separar el estilo del contenido. La forma del elemento. Es decir, que cuando realicemos un cambio en el estilo de nuestra web éste se aplicará en todos los elementos asignados a dicho estilo, lo que hará que tengamos mucho más ahorro de tiempo y consigamos mayor facilidad a la hora de añadir algún cambio. También nos permite crear colecciones de estilos (hojas de estilos) que podremos aplicar después a un documento HTML, permitiendo usar los mismo estilos para muchas páginas a la vez.

Cuando no existía el CSS, teníamos que repetir constantemente cierta información como el tamaño y el color de los textos. Además, HTML por si sólo nos ofrecía muy pocas posibilidades de cambiar la apariencia de una página más allá de un cambio de color o tipografía de una fuente.

Es por esto que el uso de hojas de estilos nos permite ir donde HTML no puede llegar. Al usar CSS lo que hacemos es definir un mayor número de características para un elemento y dichas propiedades quedan ya establecidas sin necesidad de repetirlas.

Veamos un ejemplo:

```
DIV { font-family: Verdana;
```

```
font-size: 12px;  
color: #000000  
}
```

Con CSS no es necesario repetir este código por cada marca <DIV>. Al definirlo como una regla CSS, el navegador lo interpretará cambiando las características del texto de los elementos <DIV>, con el consiguiente ahorro de trabajo y tiempo. No hay que olvidar que la mayoría de propiedades que nos ofrece CSS no sería posible definirlas con HTML.



### 1.1.1. Descripción de estilos

En resumen, CSS nació como un estándar para complementar HTML, lo que permitió un control mucho mayor sobre la visualización de los elementos. La misión de CSS en principio era sencilla: dar estilo y mejorar la apariencia de HTML, ya que éste dio mucha importancia a la estructura de la página web, pero dejó de lado un aspecto muy importante para los diseñadores como es la presentación.

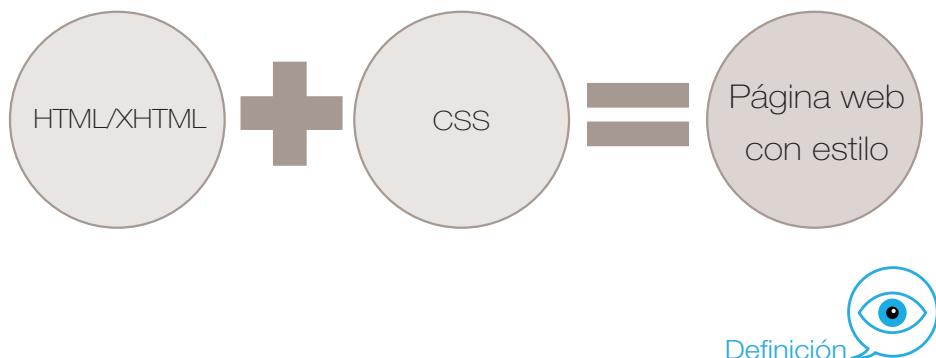
Con CSS podremos hacer más atractiva la visualización de un documento HTML sin modificar el contenido que se utiliza.

Las ventajas que existen al realizar una separación entre los contenidos y la definición de su aspecto son numerosas, ya que esto hace que se creen documentos HTML/XHTML muy bien definidos y con un significado mucho más completo. A este tipo de documentos se les conoce como “documentos semánticos”.

Por otro lado, la accesibilidad del documentos es mucho mejor, el mantenimiento de la página es mucho más fácil de llevar a cabo y podremos visualizar el documento web en numerosos dispositivos diferentes.

Cuando creamos una página web, lo que hacemos primero es utilizar el lenguaje HTML/XHTML para marcar los contenidos, es decir, a cada elemento se le designa una función dentro de la página, como por ejemplo un párrafo, un titular, una tabla, una lista de otros elementos, un texto destacado, etc.

Cuando el desarrollador web ya ha creado los contenidos en HTML/XHTML, utilizará el lenguaje CSS para poder dar definición a cada uno de los elementos creados, tales como color, tamaño y tipo de letra del texto, la separación tanto horizontal como vertical que tendrán los diferentes elementos, la posición de cada uno de ellos dentro de la página, etc.



[Las hojas de estilo](#) (Cascading Style Sheets) es el lenguaje de hojas de estilo que se usa para conformar el aspecto y el formato de un documento web escrito en un lenguaje de marcas, lo que incluye varios lenguajes basados en XML como son XHTML, HTML o SVG.

### 1.1.2. Utilización de estilos

Antes de que entre los desarrolladores web se normalizara el uso de CSS, se utilizaban etiquetas especiales de HTML para poder cambiar el aspecto de los elementos que constituían una página. En el siguiente ejemplo podemos ver una página HTML con algunos estilos definidos, donde no se ha utilizado CSS:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.  
dtd">  
  
<html>  
  
<head>  
  
  <meta http-equiv="Content-Type" content="text/html;  
charset=iso-8859-1" />  
  
  <title>Ejemplo de estilos sin CSS</title>  
  
</head>  
  
<body>  
  
  <h1><font color="red" face="Arial" size="5">Titular de  
la página</font></h1>  
  
  <p><font color="gray" face="Verdana" size="2">Un párrafo  
de texto no muy largo.</font></p>  
  
</body>  
  
</html>
```

Como puedes observar en el ejemplo, se ha utilizado la etiqueta `<Font>` con sus atributos `color`, `face` y `size`, lo que definirá el color, tipo y tamaño de la letra de cada elemento de la página.

La problemática a la que nos enfrentamos al hacer uso de este método para definir el aspecto de los distintos elementos de nuestra página es la economía de los elementos. Imaginemos que tuviéramos 50 elementos diferentes, donde tuviéramos que insertar 50 etiquetas `<font>`. Si la página se compusiera de 10.000 páginas diferentes, tendríamos que definir 500.000 etiquetas `<font>`, y ésta, al tener tres atributos diferentes, tendríamos que definirla con 1.5 millones de atributos.

Como el diseño de los sitios web está en constante evolución, es habitual modificar cada cierto tiempo el aspecto de las páginas del sitio. Siguiendo con el ejemplo anterior, cambiar el aspecto del sitio requeriría modificar 500.000 etiquetas y 1.5 millones de atributos.

También tendríamos tener en cuenta que el diseño de una página web siempre está en constante cambio, por lo que es algo habitual modificar el aspecto

de los documentos web del sitio de vez en cuando. Es por esto que usaremos CSS, ya que si tuviéramos que cambiar miles de etiquetas y atributos, nos sería un trabajo demasiado largo en el tiempo. Veamos en el siguiente ejemplo cómo hemos hecho uso del CSS para mejorar este aspecto en el documento web:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
    <title>Ejemplo de estilos con CSS</title>
    <style type="text/css">
      h1 { color: red; font-family: Arial; font-size: large; }
      p { color: gray; font-family: Verdana; font-size: medium; }
    </style>
  </head>
  <body>
    <h1>Titular de la página</h1>
    <p>Un párrafo de texto no muy largo.</p>
  </body>
</html>
```

Como podemos ver en el ejemplo, CSS lo que hace es separar los contenidos de la página por un lado, y la información sobre el aspecto que esta tendrá por otro. Dentro del documento HTML, se ha creado una zona en concreto donde se incluye toda la información relacionada en concreto con los estilos que tendrá la página.

Concretamente, podemos observar en el ejemplo que, dentro de la zona delimitada para el CSS se han indicado que todas las etiquetas `<h1>` de la página se deben ver de color rojo, en letra Arial y en un tamaño grande. Por otro lado, se indica que las etiquetas `<p>` del documento irán en color gris, en letra Verdana y con un tamaño medio.

Utilizando CSS, se pueden establecer los mismos estilos con menos esfuerzo y sin ensuciar el código HTML de los contenidos con etiquetas `<font>`. Como se verá más adelante, la etiqueta `<style>` crea una zona especial donde se incluyen todas las reglas CSS que se aplican en la página.

Las ventajas que obtendremos utilizando CSS en un documento web son múltiples:

Podremos establecer los mismos estilos con mucho menos esfuerzo.

- El código HTML se ensuciará menos con etiquetas `<font>`.
- En el código se visualizará claramente donde se encuentra la zona en la que se incluyen las reglas CSS que se aplicarán posteriormente en la página.

Si comparamos los dos ejemplos, nos damos cuenta que haciendo uso de las hojas de estilo, podremos ahorrarnos miles de etiquetas, millones de atributos, pero aún así sigue siendo una solución no del todo cómoda.

Más adelante sabremos que los estilos CSS sólo se aplican en la página que los incluye, por lo que veremos algunas soluciones que nos dan las hojas de estilo para no tener que copiar las reglas CSS en cada documento web.

### 1.1.3. Los estilos en el lenguaje de marcas

Los estilos sólo hacen una cosa: dar formato al contenido de una página web. Son archivos de texto, como los archivos HTML, pero cuya extensión es `".css"`. Por lo tanto, podemos redactarlos en el mismo editor de texto que usaremos para una página web.

CSS es un lenguaje que permite aplicar definiciones de estilos a los objetos (o bloques) de los que se compone la página web: el texto, una imagen, un párrafo, un título, una tabla, etc. Todo esto se puede modificar gracias a CSS.

Para poner una analogía, el HTML sería el esqueleto de la página y las hojas CSS su silueta, que puede ser más o menos seductora, más o menos sofisticada. En función de nuestros gustos y necesidades.

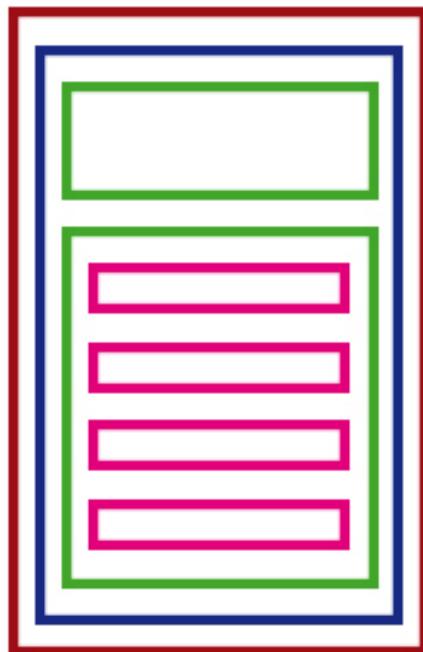
CSS son las siglas de Cascading Style Sheets en inglés, u hojas de estilo en cascada. “Cascada” se refiere al hecho de que los estilos se aplican en cascada, unos detrás de otros.

Por otro lado, el contenido de la página está organizado en bloques. La página en sí misma está contenida en un solo bloque, el body, que a su vez contiene los demás bloques: los párrafos, los títulos, las imágenes dentro de los párrafos, las tablas, etc.

Imaginemos que un bloque es como una caja rectangular que actúa como contenedor. Las hojas de estilos CSS permiten justamente aplicar efectos al contenido de las distintas cajas.

Por tanto, cada caja puede contener otras cajas, como si encajasen una dentro de otra. Para caracterizar esta relación entre los bloques se suele hablar de padres e hijos. Un parent o parente es un bloque que contiene otros bloques, y un hijo es el que está contenido dentro de otro. Todos los bloques, o casi todos, pueden ser padres de otros bloques y al mismo tiempo hijos de otros bloques.

Veamos a continuación una imagen donde podemos observar gráficamente las diferentes partes de una página web dividida en bloques y la relación entre éstos como padres e hijos:



El bloque azul es hijo del bloque rojo, que a su vez es padre de los dos bloques verdes e, indirectamente, también de los bloques rosas contenidos dentro del segundo bloque verde, etc.

El bloque rojo es el body. Él no es hijos de nadie, pero es padre de todos los demás. Si le aplicamos un estilo, dicho estilo se aplicará en cascada de un bloque a otro más pequeño. Esto se conoce como la herencia: un estilo siempre se aplicará de padres a hijos, pero no al revés.

Sin embargo, si el estilo se aplica automáticamente a los hijos, ¿cómo podemos aplicarles un estilo distinto al de los padres? Lo veremos con calma más adelante, Sólo decir que la cascada de los estilos también respeta una regla de precedencia: una definición heredada de un parente siempre será ignorada por el navegador si ya existe directamente a nivel del hijo. Por tanto, es posible definir un tipo de fuente para toda la página, luego definir otra específica para los títulos, o incluso una para un solo título, por ejemplo.

#### 1.1.4. Los estilos con herramientas de edición web

Existen numerosas formas de diseñar el estilo de una página web. Ya sabemos que podemos realizarlo directamente desde un archivo de texto, sin embargo, también podemos aplicar estilos a través de otras herramientas de edición web. Estas herramientas facilitan bastante el trabajo, en el caso concreto de CSS, algunas nos ayudan a escribir el código y otras a utilizarlo.

A continuación nombraremos algunas de estas herramientas y cómo comenzar a utilizarlas.

##### **Firebug**

Es una de las herramientas imprescindibles para cualquier creador Web. Se trata de una extensión gratuita de navegador (también gratuito) Firefox, aunque también está disponible en el caso de que utilicemos otro navegador.

Podemos descargar una copia en la página siguiente: <http://www.getfirebug.com>. Basta con hacer clic en el botón Install Firebug, que se encuentra en la parte superior derecha, e instalarlo. Una vez que lo hayamos instalado, bastará con abrir el navegador Firefox.

Para instalar Firebug en el caso de no utilizar Firefox, tenemos la posibilidad de habilitarlo en Internet Explorer, Opera o Safari visitando el siguiente enlace: <http://www.getfirebug.com/lite.html>. Sólo deberemos seguir las instrucciones para habilitarlo en nuestro navegador a una página que estemos probando o

añadirlo como bookmarklet a la barra del navegador, que es la opción que se recomienda. Esta versión no es tan completa, aún así es muy útil y potente.

Nos sería imposible hablar de todas las posibilidades de Firebug, ya que son de una gran extensión. Sin embargo, destacaremos algunos aspectos importantes.

Una vez que abrimos una página web en Firefox con Firebug instalado, ésta se muestra de forma diferente. Primero mostrará la página web, justo debajo a la izquierda de la página, pero nos mostrará la ficha HTML mostrándonos la estructura del documento web que estamos visualizando, con flechas que nos permiten expandir o contraer el subárbol del documento.

Podemos observar que, al pasar el ratón sobre el nombre de un elemento en la ficha HTML, este quedará resaltado en la página, mostrando el área del contenido, los márgenes y el relleno mediante regiones de color.

En la parte derecha de la ficha HTML, podremos ver el código CSS correspondiente al elemento inspeccionado haciendo clic en la ficha Estilo.

Deberemos tener varias cosas en cuenta cuando naveguemos por la ficha Estilo:

Mostrará en ocasiones propiedades que no hemos especificado, como por ejemplo `-moz-background-clip`. En principio podemos ignorarlas, a menos que las hayamos declarado explícitamente.

Si hemos utilizado una propiedad abreviada, éstas aparecerán de forma individual.

Las reglas se enumerarán en orden inverso de especificidad, es decir, la primera es la regla más específica que se aplica al elemento inspeccionado, la segunda es la siguiente más específica, y así sucesivamente.

Por otro lado, para inspeccionar cualquier elemento simplemente bastará con hacer clic con el botón derecho sobre éste y seleccionar Inspeccionar elemento, en el menú contextual.

Si hacemos clic en un pequeño ícono con un puntero señalando una caja (que se encuentra situado junto al ícono de Firebug), podremos cambiar al modo Inspección. Esto permitirá que los elementos se marquen a medida que nos movemos por la página y al hacer clic sobre ellos, comienza su inspección.

En el caso de que queramos desactivar cualquier declaración, simplemente deberemos hacer clic a nuestra izquierda, lo cual nos resultará muy útil cuando

estemos comprobando cómo interactúan las propiedades entre sí, observando los efectos al desactivar una a una.

También podemos comprobar cómo, al hacer clic sobre un valor en concreto en el panel de la izquierda, aparece una pequeña caja en la ficha Estilo que nos mostrará el color definido en la declaración de color correspondiente.

En el caso de que lo que queramos es conocer las dimensiones y tamaños exactos de los componentes del modelo de cajas de un elemento, como el alto, el ancho, relleno o márgenes, seleccionaremos la ficha Maquetación. Ésta nos mostrará los valores en píxeles, y al pasar el puntero sobre el cuadro que aparece en el panel, aparecerán unas reglas en píxeles en la propia página, en los lados superior e izquierdo del borde exterior del elemento.

También podremos modificar los valores o atributos de elementos (como class), o incluso el propio contenido del elemento, hasta añadir o editar valores y propiedades CSS.

## Web Developer Toolbar

El WDT o Web Developer Toolbar (Barra de herramientas para desarrollo Web), es considerado otro elemento esencial entre los desarrolladores Web. Se trata de otra extensión gratuita de Firefox. Podremos conseguirla visitando el enlace siguiente: <http://www.addons.mozilla.org> y buscar Web Developer e instalarla desde la página de WDT, tal y como puedes ver en la siguiente imagen:



Al igual que con Firebug, no podemos hablar de la multitud de opciones que nos ofrece WDT, sin embargo hablaremos sobre algunos menús importantes. Es recomendable dedicarle tiempo para explorar todos los menús que nos ofrece, para conocer mejor todas las opciones de las que dispondremos para así aprovechar todas sus posibilidades.