

**Formas de aplicar CSS:
interno y externo. link
rel y archivos de
extensión css.
Comentarios CSS
(CU01010D)**

FORMAS DE APLICAR CSS

Ya hemos visto que existen estilos por defecto que aplican automáticamente los navegadores web cuando no hay estilos especificados, así como que podemos embeber código CSS en las propias etiquetas de HTML, “en línea”. Vamos a ver ahora dos formas adicionales de aplicar estilos CSS: en la parte inicial del documento HTML (aplicación de estilos interna o CSS interno) o en un archivo de extensión .css independiente del archivo HTML (aplicación de estilos externa o CSS externo).



Volvemos a lo que hemos denominado código HTML base para el desarrollo del curso, código HTML donde no se aplican estilos.

En el epígrafe anterior vimos cómo aplicábamos estilos CSS en línea con este ejemplo:

```
<ul style = "color: red;">
<li><a href="#" style = "color: green; text-decoration: none;">Inicio</a></li>
<li> <a href="libros.html" style = "color: green; text-decoration: none;">Libros de
programación</a> </li>
<li> <a href="cursos.html" style = "color: green; text-decoration: none;">Cursos de
programación</a> </li>
<li> <a href="humor.html" style = "color: green; text-decoration: none;">Humor
informático</a> </li>
</ul>
```

CSS INTERNO

Vamos a ver ahora cómo podemos definir esos estilos en la cabecera del documento HTML usando la siguiente sintaxis:

```
<head>
...
<style type="text/css">
elementoAfectadoPorElEstilo {
propiedad1ParaEseTipoDeElementos:valor;
propiedad2ParaEseTipoDeElementos:valor;
propiedad3ParaEseTipoDeElementos:valor;
...
propiedadnParaEseTipoDeElementos:valor;
}
</style>
</head>
```

Las diferentes propiedades y valores se pueden poner en una misma línea o en distintas líneas según se prefiera (siempre separados mediante punto y coma).

Dentro de las etiquetas `<head> ... </head>` incluiremos una etiqueta de apertura de declaración de estilos `<style type="text/css">`, a continuación colocaremos la definición de tantos estilos como deseemos y cerraremos la declaración con `</style>`. En HTML 5 no es necesario especificar `type="text/css"` pero de momento seguimos recomendando que se use esta sintaxis.

En nuestro código de ejemplo el cambio de los elementos del menú para que tengan el texto color verde y el icono o símbolo de viñeta de color rojo se haría de esta forma:

```
<!DOCTYPE html>
<!-- Código ejemplo para el curso CSS -->
<html>
<head>
<title>Portal web - aprenderaprogramar.com</title>
<meta charset="utf-8">
<meta name="description" content="Portal web aprenderaprogramar.com">
<meta name="keywords" content="aprender, programar, cursos, libros">
<style type="text/css">
ul {color:red;}
a {color:green; text-decoration: none;}
</style>
</head>
<!-- Contenido de la página web -->
<body>
<!-- Cabecera de la página web -->
<div>
<h1>Portal web aprenderaprogramar.com</h1>
<h2>Didáctica y divulgación de la programación</h2>
</div>
<!-- Fin de la cabecera de la página web -->
<br />
<!-- Contenedor para la parte central -->
<div>
<!-- menu -->
<div>
<div>Menú</div>
<hr/>
<ul>
<li><a href="#">Inicio</a></li>
<li><a href="libros.html">Libros de programación</a> </li>
<li><a href="cursos.html">Cursos de programación</a> </li>
<li><a href="humor.html">Humor informático</a> </li>
</ul>
</div>
<!-- fin menu -->
<!-- Aquí el resto del código del ejemplo -->
</html>
```

Puedes comprobar que hemos usado las mismas propiedades CSS y los mismos valores que usábamos en línea.

Visualiza la página en tu navegador. El resultado para el menú será el mismo que cuando aplicamos CSS en línea:



No obstante podrás comprobar que hay otros elementos de la página web que también se ven afectados, por ejemplo los links presentes en el texto. Esto se debe a que hay una diferencia importante entre aplicar estilos CSS en línea y aplicarlos como CSS interno en la cabecera `<head> ... </head>` del documento HTML. Al aplicar estilos en línea, tenemos que repetir la aplicación de estilos en cada una de las líneas que queramos modificar y en cada ocasión afectamos a **únicamente una línea**. Con la aplicación de estilos interna nos basta con declarar una vez el estilo y el tipo de elemento al que se aplica, y automáticamente se aplicará a **todos** los elementos de ese tipo existentes dentro de la página web. Esto permite que en una página web de gran extensión nos ahorremos tener que escribir múltiples veces la definición de estilos (una en cada línea), ya que una sola declaración inicial nos bastará, lo cual es una ventaja bastante evidente.

Sin embargo ahora surge una cuestión adicional: es posible que tengamos dos o más listas de tipo `` y en algunos casos, por ejemplo menús, queramos aplicar un estilo, y en otros casos, por ejemplo listas de elementos dentro de un artículo periodístico, queramos aplicar otro estilo. También es posible que deseemos que los elementos del menú sean links con texto verde y sin subrayado, pero queramos mantener el resto de links como texto azul con subrayado. Tal y como hemos hecho la definición de estilos no conseguimos hacer esto, ya que estamos modificando **todas** las listas y **todos** los links presentes en el documento. CSS permite resolver satisfactoriamente esta problemática para aplicar estilos específicamente allí donde queremos. Veremos cómo próximamente.

CSS EXTERNO

Aunque el CSS interno nos permite unificar en una declaración todos los estilos para un archivo html, seguimos teniendo el problema de tener que repetir la definición de estilos en la cabecera de cada uno de los archivos html de nuestro desarrollo web. Si el desarrollo tiene pocos archivos quizás sea menos problemático, pero cuando el desarrollo tiene cientos o miles de archivos sí se convierte en un verdadero problema, ya que cada vez que introdujéramos cambios habría que hacerlo en los cientos o miles de archivos de que constara el desarrollo.

Para solventar esta solución se ideó la declaración externa de CSS, basada en declarar los estilos CSS en un archivo independiente que puede servir como referente para dotar de estilos a decenas, cientos o miles de archivos html, que únicamente deberán invocar el nombre de archivo utilizando una sintaxis específica. De este modo un cambio en los estilos habrá que hacerlo únicamente en el archivo de estilos correspondiente, lo cual supone una gran ventaja. Incluso un cambio completo de los estilos de una página web se puede conseguir simplemente sustituyendo el archivo correspondiente.

Vamos a generar un archivo de estilos independiente. Para ello abre Notepad++ (o el editor de texto que estés usando) y crea un archivo con el siguiente contenido:

```
/* Comentario en CSS estilos  
aprenderaprogramar.com*/  
ul {color:red;}  
a {color:green; text-decoration: none;}
```

Seguimos usando las mismas propiedades CSS y los mismos valores que hemos usado en la forma CSS en línea y en la forma CSS interna.

Los contenidos que se encuentren entre los símbolos `/* */` serán considerados comentarios y se pueden usar para introducir información del autor del archivo, versión, etc. así como para escribir aquellas aclaraciones sobre los estilos que se consideren necesarias. Un comentario puede comprender una o varias líneas según se desee.

Elige Guardar Como... y guarda el archivo con un nombre y extensión css, por ejemplo estilos.css. Asegúrate que la extensión del archivo sea css. No es válido si no tiene esta extensión.

En nuestro archivo HTML eliminaremos la definición de estilos interna y dejaremos sólo la invocación al archivo escrita con la siguiente sintaxis:

```
<head>
...
<link rel="stylesheet" type="text/css"
href="rutaDelArchivo.css">
</head>
```

link es una etiqueta que se usa en HTML para establecer vínculos entre un documento HTML y otros recursos como una imagen de icono o una hoja de estilos CSS. En este caso se indica con la propiedad rel (relación o relationship) que el documento HTML debe usar los estilos definidos como texto/css en un archivo con el nombre indicado. Ten en cuenta que si como ruta de archivo indicas simplemente el nombre del archivo, éste debe encontrarse en la misma carpeta en que se encuentre el documento HTML. Si el archivo CSS se encuentra en otra carpeta deberás indicarlo escribiendo una URL completa u bien una URL relativa que determine la ruta.

Nosotros ubicaremos de momento el archivo CSS en la misma carpeta que el archivo HTML. En el archivo HTML tendremos lo siguiente:

```
<!DOCTYPE html>
<!-- Código ejemplo para el curso CSS -->
<html>
<head>
<title>Portal web - aprenderaprogramar.com</title>
<meta charset="utf-8">
<meta name="description" content="Portal web aprenderaprogramar.com">
<meta name="keywords" content="aprender, programar, cursos, libros">
<link rel="stylesheet" type="text/css" href="estilos.css">
</head>
<!-- Aquí el resto del código del ejemplo -->
</html>
```

Ahora procedemos a visualizar el archivo HTML en nuestro navegador. Si se está cargando la hoja de estilos correctamente deberemos obtener el mismo resultado que habíamos obtenido con la aplicación de estilos interna. Prueba a hacer distintos cambios en el archivo css y visualiza los resultados.

¿QUÉ TIPO DE CSS USAR?

En la siguiente tabla hacemos un resumen de las características de los distintos tipos de CSS que hemos visto hasta el momento:

DECLARACIÓN CSS	ARCHIVOS AFECTADOS	CSS SE APLICA A
En línea	Uno, aquel donde se realiza la declaración	Una línea de código.
Interna	Uno, aquel en cuya cabecera se realiza la declaración	Todos los elementos del archivo en los que resulte de aplicación el estilo
Externa	Todos los archivos que invoquen el archivo css	Todos los elementos de los archivos afectados en que resulte de aplicación el estilo

¿Cuál de estas formas de aplicar CSS es mejor? ¿Cuál usar? En primer lugar cabe hacer una reflexión sobre qué forma de aplicar CSS hará nuestros desarrollos web más fácilmente mantenibles y aptos para ser modificados con poco esfuerzo. Parece claro que es la declaración externa la que mejor independiza los estilos del contenido y la que menor número de modificaciones implicará en general. Por ello es la forma de trabajar con CSS a la que debemos acostumbrarnos.

No obstante, cuando trabajes con desarrollos web comprobarás que también son de uso frecuente la aplicación de estilos en línea o interna. En algunos casos esto es debido a desconocimiento de la persona que realiza el desarrollo o a la forma de funcionamiento del programa con el que se ha creado la página web, pero en otros casos obedece a que se ha querido hacer así.

En ocasiones se opta por hacer modificaciones en línea porque se buscan efectos puntuales que sólo se quieren aplicar en un punto concreto y en ningún otro. En ocasiones se opta por hacer definiciones CSS internas porque se quiere afectar muy puntualmente a un archivo y a ningún otro. Y en otros casos, se usan los estilos en línea o internos simplemente “por las prisas” o “por ser lo más rápido”.

A efectos del navegador, una declaración en línea tiene precedencia sobre una declaración interna, y a su vez una declaración interna tiene precedencia sobre una declaración externa. Por tanto podríamos tener CSS sobreescrito varias veces: la declaración externa puede ser sobreescrita por la interna, y ésta a su vez por la en línea. El navegador aplicará el orden:

Declaración en línea > Declaración interna > Declaración externa

Podría ocurrir que los estilos que visualicemos en el navegador estén definidos “a trozos” de modo que parte de lo que se visualiza se debe a las declaraciones CSS externas, parte a las declaraciones CSS internas y parte a declaraciones en línea. Esto en general será indeseable, ya que hará difícil de mantener el desarrollo web, no se sabrá con certeza cómo se generan los estilos y el desarrollo puede terminar convirtiéndose en un caos difícil de mantener. También puede dar lugar a visualizaciones incorrectas o erróneas.

A modo de resumen y como recomendación: mantén los estilos de forma externa, lo más ordenadamente posible y sin utilizar redefiniciones internas ni en línea. Usa este tipo de definiciones (interna o en línea) exclusivamente cuando resulte necesario y comenta adecuadamente el código. Trabajar ordenadamente será algo que a la larga te resultará ventajoso, tanto a ti mismo como a otras personas que tengan que trabajar en desarrollos donde tú hayas participado.