

UF1303

Elaboración de hojas de estilo



TEMA 1. Hojas de estilo en la construcción de páginas web

TEMA 2. Diseño, ubicación y optimización de los contenidos de una página web

OBJETIVOS:

- 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

Los lenguajes de marcas

1. FUNCIONES Y CARACTERÍSTICAS

En este tema vamos a definir y a aprender a usar lo que se conoce como CSS, que son las siglas en inglés de *Cascading Style Sheet*, o en español, Hojas de Estilo en Cascada.

1.1 Descripción de estilos

Para empezar digamos lo que no es CSS: no es un lenguaje de programación como javascript, ni es un lenguaje de marcas como HTML.

- Funciones y características
- Hojas de estilo y accesibilidad
- Tipos de estilo: incrustados, enlazados, importados, en línea
- Selectores y reglas de estilo
- Atributos de estilo para fuentes, color y fondo, texto y bloques (párrafos)
- Creación de ficheros de estilo

OBJETIVOS:

- Describir las características de las hojas de estilo para dar formato a las páginas web, crear ficheros de estilo de acuerdo a un diseño especificado
- Identificar las características y ventajas que proporcionan las hojas de estilo
- Describir las etiquetas y los atributos que se utilizan para definir estilos



- Aprender la sintaxis y atributos que se utilizan para crear estilos
- Enlazar las páginas de un sitio web con un fichero de estilo
- Realizar un fichero de estilo para homogeneizar el formato de un sitio web



La principal función y su razón de ser puede resumirse en la siguiente frase: **Con las Hojas de Estilo separamos el contenido de su formato**

Entonces ¿qué son las *Hojas de Estilo*? Pueden ser archivos o estar integradas en el propio código HTML, incluso en las mismas etiquetas. Con CSS podemos establecer unas directivas para definir colores, tamaños, posiciones y todo aquello que tiene que ver con el formato y la posición de los objetos HTML de la página (en CSS3 se incorporan también animaciones, pero todavía no es un estándar)

CSS fue incorporado por la w3c desde HTML 4.0; a partir de esta versión del lenguaje de marcas para el desarrollo de páginas web se podía utilizar esta nueva manera de afrontar el problema de los estilos.

Actualmente y desde el 7 de junio de 2011 se encuentra en vigor la versión 2.1, siendo soportada enteramente por todos los navegadores. De la versión 3.0 hay liberados numerosos módulos, pero al no ser todavía una versión totalmente estandarizada por la w3c podemos encontrar problemas de compatibilidad según el navegador que se utilice.

Otro aspecto importante y que aparece en el mismo nombre de la tecnología que nos ocupa es el de “en Cascada”. El sentido de esta denominación viene dado por el carácter hereditario de las propiedades, de manera que si defino una

característica para un objeto HTML, los objetos que a su vez contenga adquirirán esa misma característica, ese mismo estilo; si a su vez hay otro objeto dentro de este segundo ocurrirá lo mismo, de ahí la denominación de en cascada.

1.2 Utilización de estilos

Un ejemplo típico de definición de estilos podría ser

```
Selector {  
  
    Propiedad_1:valor;  
    Propiedad_2:valor;  
    ...  
    Propiedad_N:valor;  
  
}
```

La palabra *Selector* hace referencia a cualquier elemento HTML sobre el que queramos aplicar el estilo. Podemos estar hablando de una etiqueta simple, como por ejemplo *body* o *p*. También podemos utilizar atributos HTML como clases (*class*), identificadores (*id*) o lo que más adelante definiremos como seudoelementos y pseudoclases.

Entre las llaves tendremos las propiedades que van a ser definidas, estas son propiedades aplicables al elemento o elementos HTML sobre el que queremos definir el estilo, sobre un párrafo, *p*, tendrá sentido definir un estilo de tipo de letra, pero no será así sobre una imagen, *img*.

A continuación y separado por dos puntos tendremos el valor que se aplica. Si es un tipo de letra: *arial*, o si es un color: *red*, valores hexadecimales (**#FF4000**) o incluso el valor del color en RGB, *rgb* (255, 64, 0), para poder jugar con una mayor gama de colores más allá de los predefinidos. Por último, pero importante, toda sentencia de declaración CSS se termina con un punto y coma. A lo largo del tema dedicaremos mucho tiempo a describir todo lo que podemos hacer sobre los elementos de nuestra página.

ATENCIÓN

¿Qué sabe de los códigos de colores y HTML? En el ejemplo vamos a utilizar tres formas de nombrar los posibles colores: la primera con el propio nombre del color (este es el sistema más sencillo), pero también puede ser el menos eficiente porque hay que recordar el nombre en inglés. Los básicos son fáciles, por ejemplo: *red*, *blue*, *yellow*. Pero también hay colores como *seagreen* o *darkgoldenrod* menos previsibles. Por ello y para utilizar toda la gama de colores se utilizan los sistemas hexadecimal y *rgb*.

Pongamos un ejemplo concreto:

```
p {  
    color:red;  
    Font-family: arial  
}
```

Con este ejemplo estamos especificando que todos aquellos párrafos HTML, `<p>`, que contenga nuestra página tendrán el texto en color *rojo* y un tipo de letra *arial*. Así de sencillo es utilizar CSS.

1.3 Los estilos en el lenguaje de marcas

Los estilos dentro del lenguaje de marcas son un complemento cuya misión es mantener separados por un lado el contenido y por otro el diseño, consiguiendo así una mayor eficacia a la hora de crear y mantener la web. Es conveniente tener claro desde el principio que CSS no es un lenguaje de programación, ni siquiera es un lenguaje de marcas, sino una especificación de estilos, una sintaxis por medio de la cual se declara todo aquello que tiene que ver con el aspecto visual de una página web.

1.4 Los estilos con herramientas de edición web

En este apartado vamos a tratar de decidir con que herramienta crearemos nuestras páginas de estilos. Como veremos más detenidamente en secciones posteriores, los estilos pueden estar insertados en la misma página que el código HTML o en un archivo externo. La diferencia para el tema que ahora mismo nos interesa es que si vamos a escribirlo con el HTML entonces obviamente utilizaremos el mismo editor.

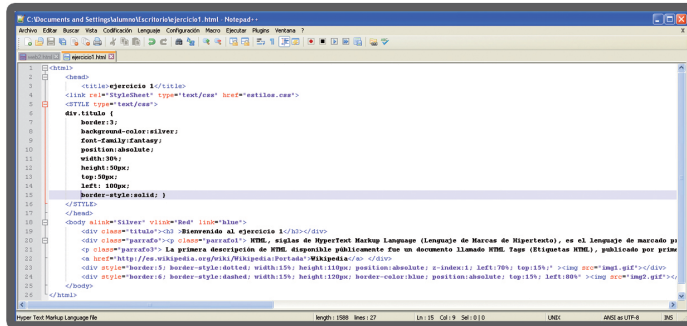
Inicialmente lo único que necesitamos un editor de texto plano, tipo *block de notas*, pero si bien para hacer pequeños retoques no sería suficiente debemos pedir algo más a nuestro editor de CSS. Las dos cosas que debemos pedir a esta herramienta son:

1. Que utilice una gama de colores para diferenciar distintas etiquetas, selectores y propiedades.
2. Que cuando esté escribiendo el código me sugiera posibilidades: de esta manera evitamos tener que recordar al pie de la letra la sintaxis y de sobre todo evitamos errores en el código. Esto es **ayuda contextual**.
3. Como último deseo podemos exigir que nos pueda ser útil además para desarrollar páginas más avanzadas, por ejemplo con javascript, php, etc. Esta tercera exigencia se retroalimenta de las dos anteriores, ya que podemos pedir, que cada lenguaje adopte un color y además que nos sugiera el código tanto si estamos escribiendo HTML, como javascript e incluso PHP o java. De esta manera no tenemos que estar cambiando de editor según trabajemos con uno u otro.

Hagamos un repaso de herramientas partiendo de las más sencillas a las más complejas.

En primer lugar tendremos notepad++, según su propia página web: es un editor de código “libre” que reemplaza al Bloc de notas, compatible con varios lenguajes. Se ejecuta en el entorno MS Windows y su uso se rige por la licencia GPL.

Es un editor sencillo que cumple el primer requisito:



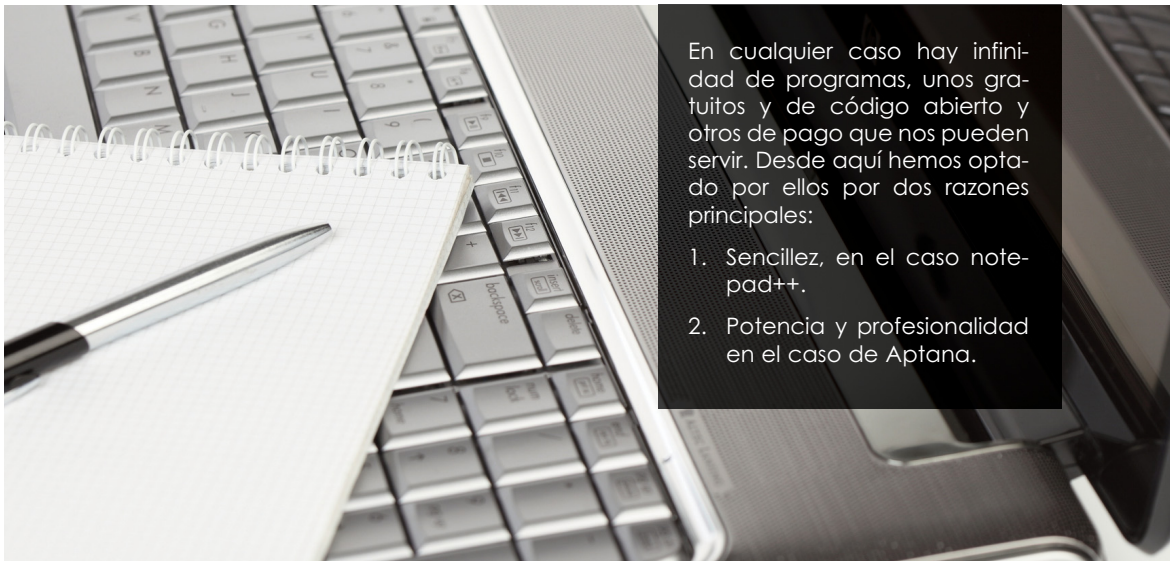
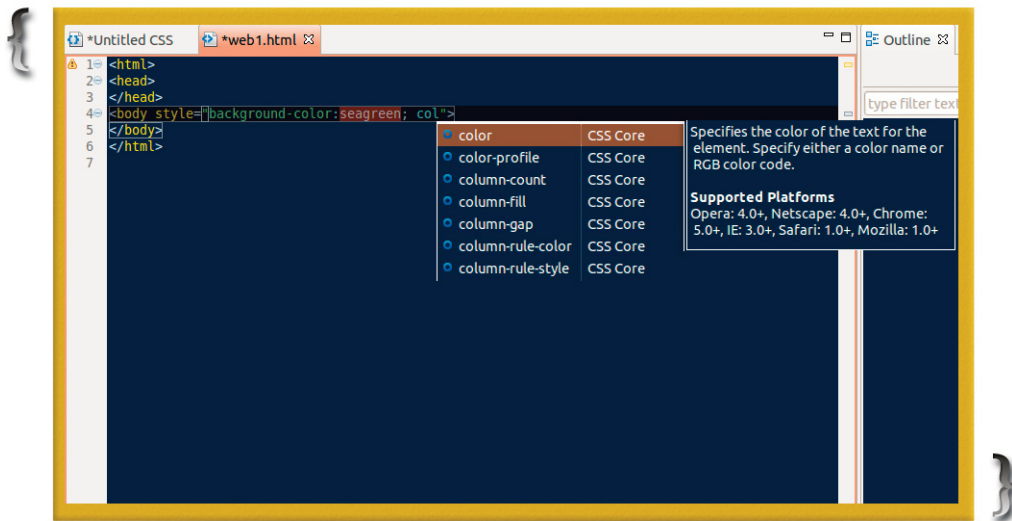
Como podemos ver asigna un color distinto en función del tipo de código. También, y por medio de un sistema de plugins, podemos hacer que nos sugiera la sintaxis. Es un programa muy completo para lo sencillo que es.

Si queremos un editor más completo y que nos sirva para todo el proceso de creación de una página web podemos optar por soluciones como Eclipse.

Este software es un IDE (Entorno de Desarrollo Integrado) de código abierto y multiplataforma, es decir, puede descargarse y utilizarse libremente y funciona tanto en entorno Windows, como Linux o IOS de Apple. Pensado inicialmente para Java y desarrollado originalmente por IBM, es utilizado ampliamente en el mundo de la programación con diferentes lenguajes gracias a su sistema de plugins. Mediante ellos podemos extender la funcionalidad del programa y generar código en múltiples lenguajes. En concreto y en el ámbito que nos interesa existe una adaptación de Eclipse pensada para el desarrollo web llamada Aptana Studio. Este IDE nos ofrece un entorno integrado y eficiente que cumple los tres requisitos que nos propusimos anteriormente.

En la siguiente imagen podemos ver en acción como cuando escribimos nuestro código CSS, el programa nos sugiere posibilidades además de indicarnos de la compatibilidad de la propiedad con los diferentes navegadores, algo muy útil.





Ambos programas proporcionan un entorno desde el que desarrollar de principio a fin nuestras páginas web (HTML, CSS, javascript,...) sin necesidad de recurrir a ningún otro de programa.

2. HOJAS DE ESTILO Y ACCESIBILIDAD

Podemos decir que la accesibilidad es la facilidad y/o eficiencia con que podemos utilizar un sitio web independientemente del nivel técnico, capacidades físicas o la tecnología elegida por el usuario.

2.1 Adecuación de las hojas de estilos

Debemos desechar la idea de que la accesibilidad es solo algo para aquellas personas con minusvalía o problemas físicos (usuarios con visibilidad o movilidad reducida). Pero hay otro aspecto que debemos tener en cuenta: hay muchas maneras distintas de navegar por la red. No todo el mundo lo hace de igual forma, ni con el mismo navegador, ni utilizando los mismos dispositivos. Hoy día cada vez más estamos acostumbrados a navegar utilizando un teléfono móvil, una tableta, un ordenador portátil o uno de sobremesa. Cada dispositivo tiene un tamaño de pantalla, un navegador predeterminado, incluso su morfología nos induce una forma de uso que nos condiciona cuando estamos en la red.

Los beneficios de una web accesible son:

- **Económicos:** se consigue un aumento de las visitas y por lo tanto un incremento de la cuota de mercado.
- **Técnicos:** las páginas son más fáciles de mantener y actualizar. El tiempo de carga se optimiza.
- **Calidad:** se siguen los estándares y de forma indirecta mejora la visión que tendrán los usuarios de nuestra web.
- **Usabilidad:** se gana en facilidad de uso y en la experiencia de usuario.
- **Posicionamiento:** se mejora el comportamiento frente a los robots de los buscadores.

Por ello debemos seguir las pautas que el w3c nos hace sobre accesibilidad. Estas pautas serían:

1. Hay que usar el menor número posible de hojas de estilo y siempre que se pueda descartar los estilos incrustados.



Nuestro objetivo, al menos teórico, es que nuestra página tenga un acceso universal y para ello es importante tener presente unas premisas básicas que lo posibiliten.