

INTRODUCCIÓN A CSS

1. Conceptos básicos

CONTENIDO

1. Introducción	1
2. Cómo incluir CSS en un documento (X)HTML	1
3. Reglas CSS.....	3
4. Medios.....	4
5. Selectores	5
a. Selectores básicos	5
b. Selectores avanzados	8
6. Herencia	9
7. Colisiones de estilos	9

1. Introducción

CSS es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos de su presentación y es imprescindible para crear páginas web complejas.

Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento de manera homogénea en infinidad de dispositivos diferentes.

Al crear una página web, se utiliza en primer lugar el lenguaje HTML/XHTML para marcar los contenidos. Una vez creados los contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, etc.

La W3C es responsable del mantenimiento del estándar CSS cuya versión estable más reciente es CSS 2.1, que data del año 2007. Paralelamente, desde el año 1998 se viene trabajando en la versión CSS 3 que, por el momento, todavía es un borrador.

El trabajo del diseñador web siempre está limitado por las posibilidades de los navegadores que utilizan los usuarios para acceder a sus páginas. Por este motivo es imprescindible conocer el soporte de CSS y (X)HTML en cada uno de los navegadores más utilizados del mercado. Debido a la lentitud con la que el W3C administra los estándares, la mayoría de navegadores han comenzado a dar soporte a CSS 3 y HTML 5 basándose en el borrador publicado. Asimismo, ante la lentitud con la que el W3C gestiona los estándares, algunas empresas han optado por formar un grupo de presión (WHATWG) y han elaborado borradores para CSS 3 y HTML5 lo que complica aún más el desarrollo de documentos “cross-browser”.

2. Cómo incluir CSS en un documento (X)HTML

Existen tres alternativas para incluir CSS en un documento:

- Definir el estilo a aplicar en cada etiqueta mediante el **atributo “style”** de la etiqueta. Esta opción se conoce como **estilos in-line**. Resulta una opción poco recomendable puesto que el código de la página se vuelve ilegible y además, si se quiere aplicar el mismo estilo a varias etiquetas distintas, será necesario volver a escribir el mismo estilo en todas las etiquetas:

```
<p style="color: black; font-family: Verdana;">Un párrafo de texto.</p>
<p style="color: green; font-family: Arial;">Otro párrafo de texto.</p>
```

1. Conceptos básicos

- Utilizar la **etiqueta <style>** para definir un bloque de estilos. Esto permite reutilizar los estilos en distintas partes del documento, sin necesidad de volver a escribirlos en cada etiqueta. Habitualmente la etiqueta <style> se incluye dentro de la etiqueta <head> para que los estilos estén disponibles para todo el documento desde el momento de la carga. El inconveniente de utilizar la etiqueta <style> reside en que no permite compartir los estilos definidos con otras páginas web:

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

<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Ejemplo de estilos CSS en el propio documento</title>

    <style type="text/css">
      p { color: black; font-family: Verdana; }
    </style>
  </head>

  <body>
    <p>Un párrafo de texto.</p>
  </body>
</html>
```

- Definir los estilos en un documento CSS independiente y enlazar dicho documento a la página web mediante la etiqueta **<link>**. Esto permite el máximo nivel de separación y reutilización, puesto que se centralizan los cambios en un único documento y además los estilos que se definen en él son utilizables por todas las páginas web que enlacen con ese documento CSS:

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

<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
    <title>Ejemplo de estilos CSS en un archivo externo</title>
    <link rel="stylesheet" type="text/css" href="/css/estilos.css" media="screen"/>
  </head>

  <body>
    <p>Un párrafo de texto.</p>
  </body>
</html>
```

De acuerdo con en este último ejemplo, deberemos tener un documento llamado “estilos.css”, en una carpeta llamada “css” bajo el directorio raíz del sitio web. En el fichero “estilos.css” se encuentra definido el estilo:

```
p { color: black; font-family: Verdana; }
```

La sintaxis y los atributos de la etiqueta <link> los hemos visto en el documento “12 - UNIDAD 01 – ENLACES”, en el apartado 6.2 .

En vez de utilizar la etiqueta <link>, también es posible enlazar un documento CSS externo utilizando la etiqueta <style>, mediante la siguiente construcción:

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

<html xmlns="http://www.w3.org/1999/xhtml">

  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
    <title>Ejemplo de estilos CSS en un archivo externo</title>

    <style type="text/css" media="screen">
      @import '/css/estilos.css';
    </style>
  </head>

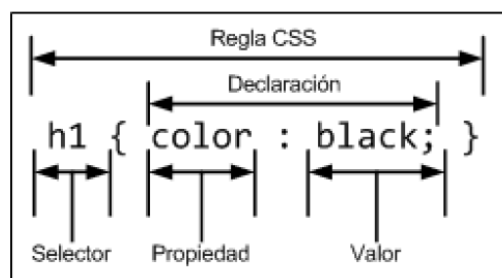
  <body>
    <p>Un párrafo de texto.</p>
  </body>
</html>
```

La URL del archivo CSS externo se indica mediante una cadena de texto encerrada con comillas simples o dobles, o mediante la palabra reservada **url()**. De esta forma, las siguientes sentencias son equivalentes:

```
@import '/css/estilos.css';
@import "/css/estilos.css";
@import url('/css/estilos.css');
@import url("/css/estilos.css");
```

3. Reglas CSS

CSS define una serie de términos que permiten describir cada una de las partes que componen los estilos CSS:



- **Hoja de estilos:** documento que contiene los estilos a aplicar en un documento.
- **Regla:** cada uno de los estilos que componen la hoja de estilos.
- **Selector:** indica el elemento o elementos a los que se aplica la regla.
- **Declaración:** indica los estilos que se definen en la regla.

4. Medios

CSS permite definir distintos estilos en función del dispositivo o medio utilizado para acceder al documento (monitor, impresora, smartphone, proyector, etc.). Además, es posible definir características específicas para cada medio, como los saltos de página para los medios impresos o el volumen y tipo de voz para los medios de audio. Estos son los tipos de medios soportados por CSS:

Medio	Descripción
all	Todos los medios definidos
braille	Dispositivos táctiles que emplean el sistema braille
embosed	Impresoras braille
handheld	Dispositivos de mano: móviles, PDA, etc.
print	Impresoras y navegadores en el modo <i>"Vista Previa para Imprimir"</i>
projection	Proyectores y dispositivos para presentaciones
screen	Pantallas de ordenador
speech	Sintetizadores para navegadores de voz utilizados por personas discapacitadas
tty	Dispositivos textuales limitados como teletipos y terminales de texto
tv	Televisores y dispositivos con resolución baja

La forma de vincular los estilos CSS a un medio determinado depende del modo en que se incluyan los estilos en la página web. Sólo es posible especificar un medio cuando se incluyen los estilos mediante las etiquetas `<link>` o `<style>`. No es posible especificar medios cuando se especifican estilos CSS in-line.

Como ya vimos al estudiar la etiqueta `link`, al utilizar esta etiqueta para vincular el documento CSS externo, la especificación de un medio se realiza mediante el atributo ***"media"***. Si, por el contrario, se utiliza la etiqueta `<style>` para enlazar la hoja de estilos o definir un bloque de estilos, es posible indicar el medio en la propia hoja de estilos o dentro del bloque de estilos con las reglas ***@media*** o ***@import*** (indistintamente) tal y como se recoge en los ejemplos siguientes:

- En este ejemplo se utiliza la etiqueta `<link>` con el atributo ***"media"***. Se aplicarán los estilos de la hoja de estilos ***"basico.css"*** cuando se visualice el documento en pantalla. Si, por el contrario, se imprime el documento o se visualiza desde una tablet o Smartphone, se aplicarán los estilos contenidos en la hoja ***"especial.css"***:

```
[...]
<link rel="stylesheet" type="text/css" media="screen" href="basico.css" />

<link rel="stylesheet" type="text/css" media="print, handheld"
href="especial.css" />
[...]
```

1. Conceptos básicos

- b) El código de los dos siguientes ejemplos puede encontrarse en una hoja de estilos o dentro de un bloque de estilos delimitado por la etiqueta `<style>`:

```
[...]
@media print
{
    body { font-size: 10pt }
}

@media screen
{
    body { font-size: 13px }
}

@media screen, print
{
    body { line-height: 1.2 }
}
[...]
```

```
[...]
@import url("estilos_basicos.css") screen;
@import url("estilos_impresora.css") print;
[...]
```

Si no se especifica un medio, el navegador asume que el medio es **all**, es decir, que los estilos se aplican en todos los medios.

5. Selectores

En una regla CSS, la declaración indica “*qué estilo mostrar*” y el selector indica “*quién debe mostrarlo*”. CSS incluye numerosos tipos de selectores que permiten seleccionar de forma muy precisa etiquetas individuales o conjuntos de etiquetas.

a. Selectores básicos

- **Selector universal:** se representa mediante el carácter asterisco (`'*`') y se utiliza para seleccionar a todas las etiquetas de la página. Por ejemplo, la siguiente regla selecciona todas las etiquetas del documento y les aplica un margen y espaciado interno de cero píxeles.

```
*
{
    margin: 0;
    padding: 0;
}
```

- **Selector de tipo de etiqueta:** selecciona todas las etiquetas de la página cuyo nombre coincide con el nombre del selector.

```
h1, h2
{
  color:red;
}

h1    {font-size: 2em;}
h2    {font-size: 1.5em;}

p
{
  color: black;
}
```

En el ejemplo anterior, se ha definido una regla común que afecta a todos los titulares <h1> y <h2> aplicándoles un mismo estilo (color rojo). A continuación se definen estilos específicos para las etiquetas <h1> y <h2>, aplicándoles distintos tamaños de fuente según sea una etiqueta <h1> o <h2>. Finalmente, se ha incluido una regla que selecciona todas las etiquetas <p> del documento y las muestra en color negro.

- **Selector de clase:** Suele ser el selector más utilizado. Permite seleccionar aquellas etiquetas cuyo atributo “**class**” tiene el mismo valor que el nombre del selector. Para aplicar estilos CSS utilizando un selector de clase añadimos el atributo “class” a la etiqueta o etiquetas a las que queremos aplicar el estilo:

```
<body>
  <p class="destacado">Lorem ipsum dolor sit amet...</p>
  <p>
    Nunc sed lacus et <a href="#" class="destacado">est adipiscing</a>
    accumsan...
  </p>
  <p>Class aptent taciti <em class="destacado">sociosqu ad</em> litora...</p>
</body>
```

Y a continuación, definimos en la hoja de estilos o en la etiqueta de estilos la regla CSS a aplicar mediante selectores de clase:

```
.destacado {color: red;}
```

Observa que el nombre del selector de clase va precedido de un punto. En el ejemplo anterior, el selector “.destacado” se interpreta como “*cualquier etiqueta de la página cuyo atributo class sea igual a ‘destacado’*”. La ventaja de este tipo de selectores es que pueden reutilizarse en distintas etiquetas con sólo añadir el mismo valor del atributo “class” a cada etiqueta. Asimismo, también es posible especificar múltiples valores separados por espacios para el atributo “class”:

1. Conceptos básicos

```
[...]
<style type="text/css">
    .error          {color: red;}
    .destacado      {font-size: 15px}
    .especial       {font-weight: bold;}
</style>
[...]
```

```
<p class="especial destacado error">Lorem ipsum dolor sit amet...</p>

[...]
```

- **Selector de ID:** funciona igual que el selector de clase pero en vez de utilizar el atributo “class”, utiliza el atributo “id” de las etiquetas. Este tipo de selector se utiliza cuando se quiere seleccionar una única etiqueta del documento ya que el valor del atributo “id” debe ser único. A diferencia de los selectores de clase, los selectores de id van precedidos por el carácter ‘#’ en vez de ir precedidos por un punto.

```
[...]
<head>
[...]
```

```
    <style type="text/css">
        #destacado    { color: red; }
    </style>
[...]
```

```
</head>

<body>
    <p id="destacado">Lorem ipsum dolor sit amet...</p>
</body>
```

- **Selector descendente:** Selecciona etiquetas que se encuentran anidadas dentro de otras etiquetas. Por ejemplo, en la siguiente regla:

```
p span {color: red;}
```

el selector “p span” selecciona a todas las etiquetas que se encuentren dentro de una etiqueta <p>. Si aplicamos esta regla al siguiente fragmento:

```
[...]
<body>
    <p>
        <span>texto1</span>
        <a href="#">
            texto2
            <span>texto3</span>
        </a>
    </p>
</body>
[...]
```

el selector “p span” seleccionará tanto texto1 como texto3. El motivo es que en el selector descendente, una etiqueta no tiene porqué ser “hija directa” de otra. La única condición es que una etiqueta debe estar dentro de otra etiqueta, sin importar lo profundo que se encuentre.

Los selectores descendentes siempre están formados por dos o más selectores básicos separados entre sí por un espacio en blanco.

- **Combinación de selectores:** es posible combinar los selectores anteriores para crear reglas de selección más complejas. Veamos algunos ejemplos:
 - **p span em** → Selecciona todas las etiquetas que se encuentren dentro de una etiqueta , que a su vez se encuentren contenidas dentro de una etiqueta <p>.
 - **p * a** → Selecciona todas las etiquetas <a> que se encuentren contenidas dentro de una etiqueta cualquiera (*), que a su vez esté contenida dentro de una etiqueta <p>.
 - **p.destacado** → Selecciona las etiquetas <p> que dispongan de un atributo *class* cuyo valor sea “destacado”.
 - **.error.destacado** → Selecciona las etiquetas que dispongan de un atributo *class* con, al menos, los valores “error” y “destacado”.
 - **.error .destacado** → Selecciona las etiquetas que dispongan de un atributo *class*=“destacado” que se encuentren contenidos dentro de cualquier etiqueta que tenga atributo *class*=“error”.
 - **p #aviso** → Selecciona todas las etiquetas con atributo *id*=“aviso” que estén contenidas dentro de cualquier elemento de tipo “p”

b. Selectores avanzados

Utilizando solamente los selectores básicos de la sección anterior, es posible diseñar prácticamente cualquier página web. No obstante, CSS define otros selectores más avanzados que permiten realizar selecciones más específicas. Estos selectores son:

- **Selector de hijos (>)**
- **Selector adyacente (+)**
- **Selector de atributos ([])**

6. Herencia

Uno de los conceptos más característicos de CSS es la herencia de los estilos definidos para las etiquetas. Cuando se aplica un estilo a una etiqueta, todos sus descendientes heredan inicialmente ese mismo valor.

Si, por ejemplo, se indica un tipo de letra al elemento `<body>` de una página, todos los elementos de la página mostrarán ese tipo de letra, salvo que se indique lo contrario:

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

<html xmlns="http://www.w3.org/1999/xhtml">

    <head>

        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
        <title>Ejemplo de herencia de estilos</title>

        <style type="text/css">
            body { font-family: Arial; color: black; }
            h1 { font-family: Verdana; }
            p { color: red; }
        </style>
    </head>

    <body>
        <h1>Titular de la página</h1>
        <p>Un párrafo de texto no muy largo.</p>
    </body>
</html>
```

En el ejemplo anterior, se ha indicado que la etiqueta `<body>` tiene asignado un tipo de letra Arial y un color de letra negro. Así, todos los elementos de la página (salvo que se indique lo contrario) se muestran de color negro y con la fuente Arial.

La segunda regla indica que los elementos `<h1>` se muestran con otra tipografía diferente a la heredada. La tercera regla indica que los elementos `<p>` varían su color respecto del color que han heredado.

La herencia de estilos no funciona en todas las propiedades CSS, por lo que se debe estudiar cada propiedad de forma individual.

7. Colisiones de estilos

En un sitio web resulta habitual disponer de múltiples hojas de estilo y bloques de estilo a nivel de página, cada uno de los cuales introduce cientos de reglas CSS. Esto supone que en ocasiones muchas reglas definidas en distintos lugares afectan a un mismo selector. En la mayoría de casos esto no es un problema puesto que el navegador aplica todos los estilos, combinándolos. Por ejemplo, si tenemos las siguientes reglas definidas, sin importar su procedencia:

```
p {color: red;}  
p {text-decoration: underline;}
```

el navegador aplicará ambas reglas a las etiquetas <p> (color rojo y subrayado).

Sin embargo, puede ocurrir que varias reglas sean contrapuestas y traten de aplicar estilos incoherentes. Por ejemplo, si el navegador encuentra las siguientes reglas definidas:

```
p {color: red;}  
p {color: green;}
```

En este caso las reglas son excluyentes (o se muestra el texto en rojo o bien se muestra en verde). En estos casos, se dice que se ha producido una **colisión de estilos** y el navegador debe aplicar un sistema de desempate que le permita determinar cuál de las dos reglas a aplicar.

Las características de este sistema de resolución de colisiones son complejas pero, en general, se aplican las siguientes reglas:

1. Se aplica la regla que tenga un selector **más específico** (el que menos etiquetas abarque y más restrictivo sea a la hora de seleccionar etiquetas).
2. A igualdad de especificidad, se aplica la **última regla definida**.

En el ejemplo anterior, existe igualdad de especificidad entre los selectores por lo que se aplica la última regla y el texto se muestra en verde.