

Unidad 3.1: Hojas de Estilo CSS

Cascading Style Sheets

Hojas de Estilo

- ¿Qué son las hojas de estilo?
- Métodos de uso
 - Hojas de estilo en línea
 - Hojas de estilo internas o incrustadas
 - Hojas de estilo externas
 - Mezclando métodos de uso
- Estructura básica de una regla CSS
- Selectores

¿Qué son las Hojas de Estilo (CSS)?

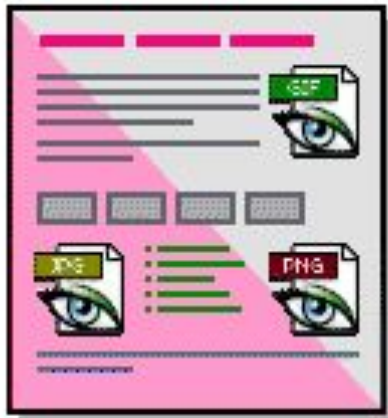
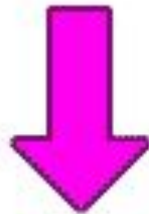
- Las **Hojas de Estilo** (o **CSS**, por Cascading Style Sheets) son un mecanismo que permiten aplicar formato a los documentos escritos en HTML (y en otros lenguajes estructurados, como XML) separando **el contenido de las páginas de su apariencia**.
- Fue desarrollado por el Consorcio WWW (W3C) con el objetivo de simplificar el método de presentación de la información, además de ampliar las posibilidades con nuevos efectos para los distintos elementos.

¿Qué son las Hojas de Estilo (CSS)?

información + presentación

< body >

</body>



información

< body >

</body>

presentación

<style>

</style>



¿Qué son las Hojas de Estilo (CSS)?

- **CSS aporta, por un lado, un gran número de efectos antes inalcanzables con el uso de HTML y por el otro, permite realizar una clara separación entre información y formato de presentación.**

En el código HTML encontraremos tanto las etiquetas para la información como para la presentación	Con CSS nos valemos de HTML para definir la estructura de la información, y el aspecto se proporciona mediante CSS
---	---

¿Qué son las Hojas de Estilo (CSS)?

Las posibilidades que ofrecen las hojas de estilo afectarán a los elementos HTML de diversos modos:

- Modificando el formato y las fuentes de los textos
- Dotando a los elementos de colores en primer plano, o colores y/o imágenes para el fondo
- Proporcionando alineaciones, separaciones entre líneas, distancia entre palabras o letras
- Agregando márgenes, rellenos de separación, bordes, dimensiones de anchura y altura
- Situando los elementos en diversas posiciones dentro del contexto del documento
- Permitiendo o deshabilitando su visibilidad

¿Qué son las Hojas de Estilo (CSS)?

- Podemos consultar la documentación de CSS establecida por el Consorcio WWW (W3C):

<http://www.w3.org/TR/1998/REC-CSS2-19980512/>

- También tenemos disponible una traducción de dicho estándar en la dirección:

<http://www.sidar.org/recur/desdi/traduc/es/css/cover.html>

Métodos de uso

- El código que compone la hoja de estilo está formado por una o más **reglas de estilo**.
- Mediante la hoja de estilo se **identifica** el elemento HTML que se desea seleccionar y la **apariencia** que se le quiere dar.
- Hay tres maneras de añadir estilos a nuestras páginas Web:
 - De forma local
 - Con hojas internas
 - Con hojas externas

Métodos de uso

- **De forma local:** Añadiendo instrucciones de estilo sólo a etiquetas concretas, o a un grupo de ellas (**ESTILOS EN LÍNEA**)
- **Con hojas internas:** Incluyendo las instrucciones de estilo en el documento HTML de una página concreta. Esto permite cambiar la apariencia de una página entera, cambiando unas pocas líneas (**ESTILOS INCRUSTADOS EN DOCUMENTO**)

Métodos de uso

- **Con hojas externas:** Enlazando todos los documentos HTML de todas las páginas que componen un sitio del Web con un fichero de definición del estilo. De esta manera, se puede cambiar la apariencia de múltiples páginas retocando un único fichero

Métodos de uso: **hojas de estilo en línea**

Para modificar una etiqueta concreta (atributo **STYLE**)

- Se hace añadiendo el **atributo STYLE**, que contendrá una serie de valores dentro de la etiqueta correspondiente.
- Los valores que contiene estarán formados por un conjunto de pares

nombre_de_propiedad:valor separados unos de otros por (;)

<etiqueta style="propiedad1: valor1; propiedad2: valor2; ...">

...

</etiqueta>

Métodos de uso: **hojas de estilo en línea**

Ejemplo:

<P>

"Este es un párrafo normal, sin márgenes a la izquierda ni a la derecha, con el color de su texto en negro".

</P>

<P STYLE="margin-left: 0.5in; margin-right: 0.5in; color:green">

"Quiero que este otro párrafo tenga un margen izquierdo de media pulgada, al igual que el margen derecho. Además quiero que su texto sea de color verde".

</P>

Métodos de uso: **hojas de estilo en línea**

- Obsérvese la sintaxis del atributo **STYLE**. Se ponen **entre comillas una serie de propiedades** (por ejemplo, margin-left), **y después de dos puntos se pone el valor** de esa propiedad (en este caso, 0.5in o sea, 0.5pulgadas).
- Las distintas propiedades deberán estar separadas por punto y coma.

Métodos de uso: **hojas de estilo en línea**

- Para modificar un grupo de etiquetas o sección de la página (<DIV>,)
- Si queremos cambiar la **apariencia de una sección entera** (que agrupe un conjunto de etiquetas), se puede utilizar la **etiqueta <DIV>**, con la que definiremos el estilo globalmente para esa sección.
- En el siguiente ejemplo cambiamos el color (a rojo) y el tamaño de la fuente (a 16 puntos de altura) de un bloque de etiquetas por medio de la etiqueta <DIV>, lo que tiene el mismo efecto que si hubiéramos asignado este estilo separadamente a cada una de las etiquetas englobadas (<P>, y)

Métodos de uso: **hojas de estilo en línea**

```
<DIV STYLE="font-size: 16pt; color: red">
```

```
<P>La definición del estilo (rojo y 16 puntos), afecta a todo  
el bloque de etiquetas.</P>
```

```
<P>Este texto también es rojo y tiene 16 puntos de altura.  
</P>
```

```
<P>Al igual que esta línea. </P>
```

```
</DIV>
```

Métodos de uso: **hojas de estilo en línea**

- ¿Qué ocurre si a una de las etiquetas englobadas por DIV le introducimos un estilo distinto?
- Comprobémoslo introduciendo un estilo propio a la segunda etiqueta <P>:

```
<DIV STYLE="font-size: 16pt; color: red">
```

```
<P>
```

Veamos si la definición del estilo (rojo y 16 puntos), afecta a todo el bloque de etiquetas, hasta la etiqueta de cierre de DIV. </P>

```
<P STYLE="color: blue">Esta línea es azul, a pesar de estar englobada dentro de DIV, porque tiene su propio estilo. </P>
```

```
<P>Esta otra cumple con el estilo de DIV, porque no tiene estilo propio. </P>
```

```
</DIV>
```


Métodos de uso: **hojas de estilo en línea**

- Como se comprueba en este caso, **el estilo dentro de una etiqueta concreta tiene prioridad sobre el estilo del bloque**, impuesto por DIV.
- Aquí **comenzamos a ver la noción de precedencia de unos métodos sobre otros**, que son muy variados. Este es el **motivo de que se llamen Hojas de Estilo en Cascada** (hay una cascada de métodos para aplicar los estilos, teniendo un orden de precedencia riguroso de unos sobre otros).

Métodos de uso: **hojas de estilo en línea**

- Existe también la **etiqueta ** que tiene una misión parecida a **<DIV>**, pero que se usa para cambiar elementos más pequeños (por ejemplo, palabras o incluso letras), en lugar de bloques enteros.

<P>

En este párrafo solamente queremos atribuir un estilo a esta

Palabra,

que queremos que su fuente sea courier, de 15 puntos de altura y de color fucsia.

</P>

Métodos de uso: **hojas de estilo en línea**

- Por supuesto, la definición del estilo dentro de la propia etiqueta no es la manera más eficaz de utilizar las hojas de estilo, pero pueden existir casos que lo justifiquen.
- Esta forma de aplicar la hoja de estilo directamente en la etiqueta es similar al modo usado en el código HTML, pero con dos importantes diferencias:
- El conjunto de **propiedades** que se pueden **aplicar** es mucho **mayor**
- Los estilos especificados en una hoja de estilo (cualquiera sea su origen) **tienen preferencia sobre los formatos aplicados con HTML.**
- Esto significa que si en nuestra hoja de estilo hemos definido:

```
P {text-align: left}
```
- Todos los párrafos quedarán alineados por la izquierda y, si en un párrafo determinado, usamos:

```
<P align="center">
```
- El párrafo seguirá estando alineado por la izquierda porque **las reglas de las CSS tienen prioridad sobre el formato aplicado con HTML.**

Métodos de uso: **hojas de estilo internas**

- Este otro método también se conoce con el nombre de hojas de estilo incrustadas, debido a que se pone la hoja de estilo dentro de una página HTML.
- Se utiliza la **etiqueta** `<STYLE> .. </STYLE>`, dentro de la cabecera del documento, entre las etiquetas `</TITLE>` y `</HEAD>`.
- Este método permite aplicar la hoja de estilo ***solamente a la página que la contiene***. Resulta útil cuando en nuestro sitio tenemos algunas páginas que usan un formato distinto al resto.

Métodos de uso: **hojas de estilo internas**

- En muchos casos convendrá usar ambos métodos simultáneamente: poner un link a una hoja externa para aplicar un estilo general y luego crear una hoja incrustada en la que solamente deberemos definir las reglas específicas para esa página.
- Por las leyes de cascada de las CSS, en caso de existir una misma regla (una en la hoja externa, otra en la hoja incrustada) con distintos valores, **tiene prioridad la definida en la hoja incrustada.**

Métodos de uso: **hojas de estilo internas**

La etiqueta **<STYLE>** tiene parámetros:

- **TYPE**: Especifica el tipo de contenido de la hoja de estilos, por ejemplo "**text/css**". Esto indica que el **contenido** del archivo consiste en **definiciones de reglas de estilo CSS** que permitirá a los navegadores que no soporten este tipo el ignorar la hoja de estilo.
- **MEDIA**: Especifica el medio sobre el que se verá la información.

Métodos de uso: **hojas de estilo internas**

- Los distintos valores soportados por MEDIA son:
 - **Screen:** Pensado para ordenadores personales.
 - **Tty:** Pensado para teletipos, terminales, portátiles, etc.
 - **Tv:** Pensado para televisores.
 - **Projection:** Proyector.
 - **Handheld:** Dispositivos de pequeñas dimensiones como teléfonos móviles. Agendas, etc.
 - **Print:** Pensado para impresión.
 - **Braille:** Pensado para dispositivos táctiles braille.
 - **Aural:** Para sintetizadores de voz.
 - **All:** Adecuado para todo tipo de dispositivos.

Métodos de uso: **hojas de estilo internas**

- La estructura será la siguiente:

<HTML>

<HEAD>

<TITLE> </TITLE>

<STYLE TYPE="text/css">

[Bloque de instrucciones de estilo]

etiqueta1 HTML { propiedad1: valor1; propiedad2: valor2;... }

etiqueta2 HTML { propiedad3: valor3; propiedad4: valor4;... }

</STYLE>

</HEAD>

<BODY>

[conjunto de todas las etiquetas que componen la página]

</BODY> </HTML>

Métodos de uso: **hojas de estilo externas**

- Si nuestro sitio del Web está compuesto por **muchas páginas**, y queremos darles un **estilo uniforme** a unas cuantas, en vez de incluir un bloque de definición de estilo repetido en cada una de ellas, se puede establecer un **enlace a un fichero de texto** que contiene el **bloque de definición del estilo**.
- Para utilizar este método debemos escribir las reglas de estilo en un fichero externo. Éste no tendrá ninguna etiqueta, pues no es un documento normal HTML, sino que es un fichero de texto que **sólo contiene las reglas de estilos**.
- Debe tener necesariamente la **extensión .CSS**

Métodos de uso: **hojas de estilo externas**

- Para hacer uso de este fichero de definición de estilos en todas las páginas que queramos, solamente deberemos añadir entre `</TITLE>` y `</HEAD>`, la **etiqueta `<LINK>`**, que proporciona un método para definir relaciones con otros documentos y recursos.

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

Métodos de uso: **hojas de estilo externas**

La etiqueta LINK posee los atributos:

- **REL (relation)**: Indica el tipo de enlace entre el documento y el recurso especificado en el atributo HREF.

Puede tomar distintos valores, pero como ejemplo usaremos el valor "**stylesheet**", que especifica que el archivo incorporado es una hoja de estilos.

Métodos de uso: **hojas de estilo externas**

La etiqueta LINK posee los atributos:

- **TYPE:** Indica que el contenido del archivo consiste en **definiciones de reglas de estilo CSS**, y tomará el valor **"text/css"**
- **HREF:** Indica la URL donde se encuentra el fichero externo que define las reglas de estilo que se emplearán en el documento HTML.

Métodos de uso: **hojas de estilo externas**

Ejemplo:

```
<LINK HREF="miestilo.css" REL="stylesheet"  
      TYPE="text/css">
```

- Aquí se ha supuesto que el fichero de estilos se ha dejado en el mismo directorio donde están los documentos HTML. Si no es así, es decir, si está en **otro subdirectorio** en el servidor, habría que **indicar la ruta** (por, ejemplo: HREF="subdirectorio/miestilo.css").

EJERCICIO:HOJAS DE ESTILO

- Realizar una página web que nos muestre HOLA MUNDO utilizando:
 - 1- Hoja de estilo de etiqueta para que subraye el texto y lo muestre en verde.
 - 2- Hoja de estilo interna.
 - 3- Hoja de estilo externa.

Mezclando métodos de uso

- Como hemos visto, hay tres métodos de aplicar las hojas de estilo en cascada.

*¿Se pueden aplicar más de uno a la vez en una misma página? **SÍ***

- Para evitar conflictos entre los distintos métodos usados simultáneamente, existe un **orden de precedencia**, es decir, cuál prevalecerá sobre el otro si dan órdenes contradictorias sobre un aspecto concreto.

Mezclando métodos de uso

El orden de precedencia es el siguiente (de mayor a menor):

- **Estilo dentro de una etiqueta**
- **Bloque de estilo en la cabecera** del documento
- **Enlace a un fichero** que contiene la **hoja de estilo**.

Mezclando métodos de uso

- Esto quiere decir que una orden de estilo, por ejemplo el color del texto, puesta dentro de una etiqueta, prevalecerá sobre la que esté indicada en los otros dos métodos, si es que se usan en esa página.
- Lo **más práctico** es utilizar el tercer método (enlace a una hoja de estilo), para dar una **apariencia consistente a todas las páginas**, y si es necesario modificar un aspecto concreto en alguna de ellas, utilizando el primer o segundo método.

Estructura básica de una regla CSS

- Una **hoja de estilos** está compuesta por un **conjunto de reglas** que especifican el aspecto que tendrán los elementos de un documento HTML.
- Cada una de estas reglas contiene una lista de propiedades acompañadas de uno o más valores. Será el valor, aplicado a una propiedad concreta, el que define el aspecto final del elemento.
- Cada regla está formada por dos partes diferenciadas:

Selector { **Declaración** }

Estructura básica de una regla CSS

- **Selector:** Es un identificador que especifica sobre qué parte del documento se aplicará el estilo definido.
- **Declaración:** Es una lista de definiciones (propiedades y valores) entre dos llaves que establecen el estilo a presentar. Esta lista debe contener al menos un elemento. Si existe más de un elemento deben separarse por (;).

P { **color**: **blue**; }

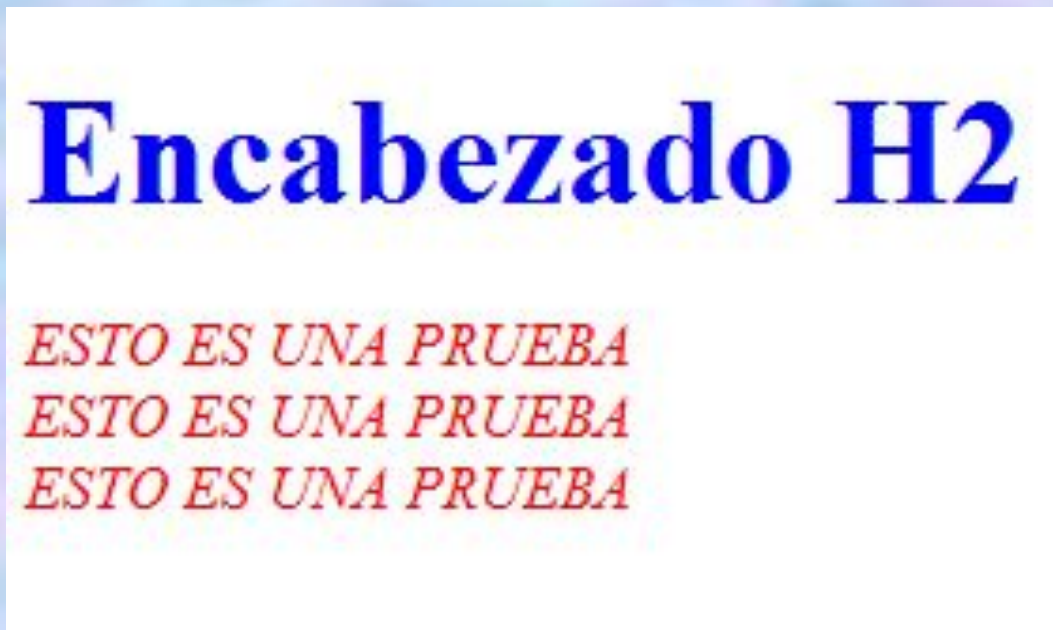
Estructura básica de una regla CSS

El selector

- Está formado en principio por un **identificador** que sirve para determinar **qué parte del documento utilizará el estilo definido en la Declaración**. Este identificador sigue las siguientes reglas sintácticas:
 - Comenzará por un carácter alfabético (A..Z o a..z).
 - A continuación se incluirán más caracteres alfabéticos, guiones (-) y números (0-9).
- Según utilicemos los selectores se pueden distinguir varios tipos:
 - Selectores de etiquetas
 - Selector universal
 - Selectores de Identificadores
 - Selectores de Clases
 - Selectores de pseudo-clases
 - Selectores de pseudo-elementos

Selectores de etiquetas HTML

- Definirán el tipo de elemento HTML que presentará el estilo definido en la regla (por ejemplo p, h2, li, etc.).
- Para obtener esto



sería:

Selectores de etiquetas HTML

<html>

<head>

<style type="text/css">

h2 { color:blue; font-size:30pt; font-family: times; }

p { color:red; font-family: Comics Sans MS; text-align: justify;
font-style: italic}

</style>...

</head>

<body>

<h2>Encabezado H2</h2>

<p>

ESTO ES UNA PRUEBA
ESTO ES UNA PRUEBA

ESTO ES UNA PRUEBA
 </p>

</body></html>

Selector universal

- Se indica por medio de un carácter asterisco “*”. Su función es la de representar a cualquier elemento de la estructura de un documento HTML.
- Ejemplo:

```
<style type="text/css">
```

```
* { color: blue; } /*Se aplica a cualquier etiqueta HTML*/
```

```
...
```

```
</style>
```

Selectores de identificadores

- Las etiquetas HTML disponen de un atributo llamado **“identificador”**, cuya finalidad es la de **asignar una identificación al elemento**. De esta forma, CSS u otro lenguaje podrá seleccionarlo y distinguirlo de los demás.
- Para definirlo se utiliza el atributo **“id”**, y con ello se **asigna a esa etiqueta un nombre** o expresión para distinguirlo. El uso de este **atributo es opcional** por lo que no tiene sentido definirlo si después no se prevé que CSS u otro lenguaje vaya a usarlo.

Selectores de identificadores

- Un ejemplo sería:

`<p id="despedida">`

- Mediante este tipo de selector dotamos de estilo a un único elemento de todo el documento.
- El método para **declarar** un Selector de Identificador es el siguiente:
 - Primero debemos **definir una regla de estilo** que utilice un selector de id, es decir, un identificador **precedido** del **carácter almohadilla (#)**.
 - Posteriormente **incluimos el atributo id a un único elemento de todo el documento** en el cuál indicaremos el identificador (sin almohadilla #) que hemos usado en el selector.

Selectores de identificadores

Ejemplo:

```
<html>
```

```
<head>
```

```
  <style type="text/css">
```

```
    #elemento1{color:blue;
```

```
    font-size:30pt; font-family: times;}
```

```
  </style>
```

```
</head>
```

```
<body>
```

```
  <p id="elemento1"> Párrafo que aplica el estilo. </p>
```

```
  <p> Párrafo que no aplica el estilo. </p>
```

```
</body>
```

```
</html>
```

Selectores de identificadores

El resultado es.

Párrafo que aplica el estilo.

Párrafo que no aplica el estilo.

Selectores de clases

- Las etiquetas HTML disponen de un atributo llamado “**clase**”, cuya finalidad es la de agrupar los elementos por clases o grupos. De esta forma, CSS u otro lenguaje podrá seleccionarlo y distinguirlo de los demás.
- Para definirlo se utiliza el atributo “**class**”, y con ello se asigna un nombre o expresión para distinguirlo. El uso de este **atributo es opcional** por lo que no tiene sentido definirlo si después no se prevé que CSS u otro lenguaje vaya a usarlo.

Selectores de clases

- Una etiqueta puede pertenecer a varias clases a la vez ya que éstas no son excluyentes.
- Un ejemplo sería:

```

```

```

```

Selectores de clases

- Las clases son definiciones de estilo comunes a varios elementos, aunque éstos sean de distintos tipos. El método para utilizar las clases consiste en:
 - Primero debemos **declarar** una **regla de estilo** en la cuál el selector será el identificador que deseemos para la clase precedido de un punto ' . '.
 - Después añadiremos el nombre de la clase (sin punto ' . ') que hemos decidido para la regla, en el atributo HTML **class** de todos aquellos elementos que queramos que apliquen ese estilo.

Selectores de clases

Ejemplo:

```
<html>
```

```
<head>
```

```
  <style type="text/css">
```

```
    .GREENBOLD {color: green; font-weight: bold;} </style>
```

```
</head>
```

```
<body>
```

```
<h3 class="GREENBOLD">Este título es muy verde</h3>
```

```
<p class="GREENBOLD">
```

Este párrafo usa la clase de estilo GREENBOLD. Se puede utilizar el atributo class para especificar una clase de estilo para cualquier elemento HTML.

```
</p>
```

Selectores de clases

Continúa:

```
<blockquote class="GREENBOLD">
```

Este bloque usa la clase de estilo GREENBOLD. En consecuencia, es

verde y está en negrita. Puede ser útil para hacer que los bloques

resalten del resto de la página.

```
</blockquote>
```

```
</body>
```

```
</html>
```


Selector de pseudo-clases

- **Las pseudos-clases clasifican a las etiquetas según el estado de las mismas.** Es decir, son agrupadas en función de los eventos que les hayan ocurrido (que el enlace haya sido visitado, que el cursor del ratón se encuentre encima o que deje de estar sobre él, etc.). Así pues, **seleccionan los diferentes estados por los que puede pasar.**
- De las distintas etiquetas con pseudo-clases existentes hasta la fecha, las **más utilizadas** son las referentes a los hipervínculos (**etiqueta <a>**). Existen otros referentes a:
 - El idioma (:lang)
 - Al primer elemento después de otro (:first-child)

Selector de pseudo-clases

- La sintaxis de las pseudo-clases es:

etiqueta:pseudo-clase

- Las pseudo-clases pueden ser dinámicas en el sentido de que un elemento puede **adquirirla o perderla** a medida que el **usuario interactúa** con el documento. Estas pueden aplicarse a elementos distintos de hipervínculos.

Selector de pseudo-clases **a:link**

- Para definir el estilo de los enlaces según se cargan en el documento. Es decir, todavía no se ha visitado.

```
<html>
```

```
<head>
```

```
  <style type="text/css">
```

```
    a:link{ text-decoration: none; font-weight: bold;  
            background-color: #dddddd; }
```

```
  </style></head>
```

```
<body>
```

```
<p>Este <a href="enlace.htm">enlace</a> no aparece como suele  
  ser habitual,
```

```
  porque<br> se muestra sin subrayar, en azul y con color de fondo.
```

```
</p> </body></html>
```

Selector de pseudo-clases **a:link**

- Se vería:

Este **enlace** no aparece como suele ser habitual, porque se muestra sin subrayar, en negrita y con color de fondo.

Selector de pseudo-clases **a:visited**

- Para definir el estilo de los enlaces ya visitados

```
<style type="text/css">
```

```
  a:link{ text-decoration: none; font-weight: bold;  
    background-color: #dddddd; }
```

```
  a:visited {background-color: yellow; color: #999999;}
```

```
</style>
```

```
</head>
```

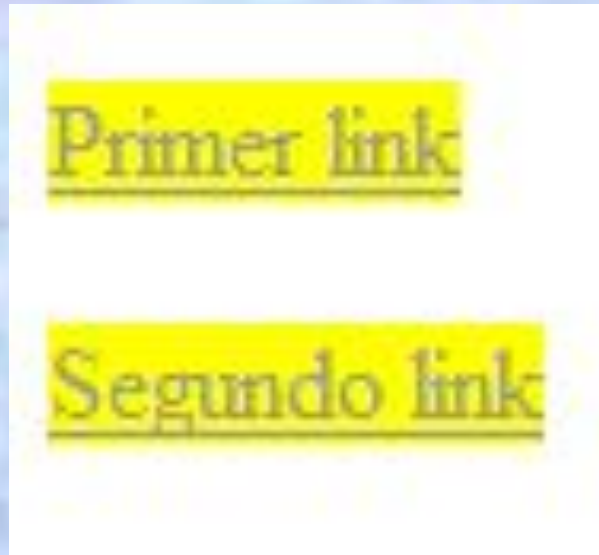
```
<body>
```

```
<p><a href="enlaces.htm">Primer link</a></p>
```

```
<p><a href="enlaces2.htm">Segundo link</a></p>
```

Selector de pseudo-clases **a:visited**

- Se vería:



Selector de pseudo-clases **a:hover**

- Para definir el estilo de los enlaces sobre los que está el puntero del ratón.

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
a:link {font-family: serif; text-decoration: none}
```

```
a:hover {cursor: help; background-color: yellow;  
font-family: sans-serif}
```

```
</style>
```

```
</head>
```

Selector de pseudo-clases **a:hover**

Continua:

<body>

<p>

`Link 2`

[illegible]

</p>

</body>

</html>

Selector de pseudo-clases **a:hover**

- Se vería:

Link 2

Link 3

Link de control?

Selector de pseudo-clases **a:active**

- Para definir el estilo de los enlaces sobre los que se **está haciendo clic**, durante el momento que transcurre entre presionar y soltar el botón del ratón sobre el elemento.

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
A:link {color: red}
```

```
A:visited {color: gray}
```

```
A:active {color: fuchsia}
```

```
</style>
```

Selector de pseudo-clases **a:active**

- Continua:

```
</head>
```

```
<body>
```

```
<p>
```

```
<a href="ps_din2.html">Enlace 1</a> <br>
```

```
<a href="ps_din3.html"> Enlace 2 </a> <br>
```

```
<a href="link_falso.html"> Enlace 3</a>
```

```
</p>
```

```
</body>
```

```
</html>
```

Selector de pseudo-clases **a:focus**

- Para definir el estilo de los enlaces que están siendo seleccionados mediante el teclado.

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
  A:link {color: red}
```

```
  A:visited {color: gray}
```

```
  A:hover {color: blue}
```

```
  A:active {color: fuchsia}
```

```
  A:focus {background: #cc0000;color: #fff;  
            border: 1px dotted #ccc;}
```

```
</style>
```


Selector de pseudo-clases **a:focus**

```
</head>
```

```
<body>
```

```
<p>
```

```
<a href="ps_din2.html">Enlace 1</a> <br>
```

```
<a href="ps_din3.html">Enlace 2</a> <br>
```

```
<a href="link_falso.html">Enlace 3</a>
```

```
</p>
```

```
</body>
```

```
</html>
```



Selector de pseudo-elementos

- Los pseudo-elementos son “partes” de elementos de una página a los que **HTML no otorga identidad** propia y, por tanto, no pueden ser seleccionados por dicho lenguaje. Sin embargo, **sí que pueden ser identificados por las reglas de estilo** y ser formateados de manera distinta a la de los elementos a los que pertenecen.
- Por ejemplo, en HTML no existe ninguna etiqueta para diferenciar la primera letra o la primera línea de cada párrafo (subelementos del elemento párrafo). En cambio mediante los pseudo-elementos es posible acceder a ellas y dotarles de un estilo distinto al del párrafo.

Selector de pseudo-elementos

- La sintaxis de los pseudo-elementos es:

etiqueta:pseudo-elemento

Ejemplo:

P: first-letter {color:red}

La primera letra de cada párrafo será roja

Selector de pseudo-elementos **:first-line**

- Para definir el estilo de la primera línea visual del texto de un párrafo. Esta primera línea va a referirse a la **primera línea que produce el navegador** al presentar el texto, de forma que acaba cuando se produce la ruptura de la línea para pasar a la siguiente. Debemos tener presente que **no se trata de un elemento HTML** real, es decir, no se define mediante una etiqueta, tan sólo es interpretada por el navegador.
- Para especificar las propiedades de este pseudo-elemento el selector estará compuesto del **identificador del elemento** (etiqueta HTML) y a continuación los caracteres **":first-line"**.

Selector de pseudo-elementos **:first-line**

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
  body { font: 12pt Arial; }
```

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

```
  p:first-line { color: blue; }
```

```
</style>
```

```
<body>
```

```
<p>Este es el primer párrafo del texto sobre el que  
hemos definido un estilo para la primera línea  
mediante un pseudo-elemento.</p>
```

Selector de pseudo-elementos :first-line

Continúa:

```
<p>Este es el segundo párrafo <br>del texto sobre el que  
hemos  
definido un estilo para la primera línea mediante un  
pseudo-elemento.</p></body>  
</html>
```

Selector de pseudo-elementos :first-line

El resultado:

Este es el primer párrafo del texto sobre el que hemos definido un e

Este es el segundo párrafo

del texto sobre el que hemos definido un estilo para la primera línea

Selector de pseudo-elementos **:first-line**

Sólo las siguientes propiedades se pueden aplicar al pseudo-elemento :first-line:

- la fuente, el color y el fondo
- word-spacing
- letter-spacing
- text-decoration
- vertical-align
- text-transform
- line-height
- text-shadow
- clear

Selector de pseudo-elementos **:first-letter**

- Para definir el estilo de la primera letra de todo un bloque, por ejemplo, un párrafo de texto.

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
body { font-family: Arial; }
```

```
p { font-size: 12pt; line-height: 12pt; color: red; }
```

```
p:first-letter { font-size: 250%; font-weight: bold;  
color: maroon; background: salmon;float: left; }
```

```
</style>
```

Selector de pseudo-elementos :first-letter

```
<body>
```

```
<p>Este párrafo de texto tiene definido un estilo para la  
primera letra para que parezca una capitular caída.
```

```
<br>El tamaño es del 250%, con color marrón y fondo  
color salmón.</p>
```

```
</body></html>
```

Selector de pseudo-elementos **:first-letter**

Sólo las siguientes propiedades se pueden aplicar al pseudo-elemento :first-letter:

- la fuente, el color y el fondo
- los márgenes
- el relleno o padding
- los bordes
- text-decoration
- vertical-align (sólo si tiene definida la propiedad float con valor none)
- text-transform
- line-height
- text-shadow
- float
- clear

Selector de pseudo-elementos **:before** y **:after**

- Para definir el estilo de un contenido que se puede insertar antes o después de un elemento determinado. La **propiedad 'content'**, junto con estos pseudo-elementos, especifican lo que se inserta.
- Podemos lograr que antes de cada elemento <H3> aparezca el texto "Tema:" sin necesidad de tener que escribirlo en cada título. También podemos hacer que cada párrafo termine con un pequeño ícono o poner "Fin" al pie de cada página usando las siguientes reglas.
- En la última regla hemos especificado también "display: block" para que la palabra "Fin" comience en una nueva línea (como si fuese un nuevo párrafo).

Selector de pseudo-elementos **:before y :after**

```
<html>
```

```
<head>
```

```
<title>Untitled</title>
```

```
<style type="text/css">
```

```
  H3:before {content: "Tema: "}
```

```
  P:after {content: url("cursorhelp.gif")}
```

```
  BODY:after {content: "Fin"; display: block;}
```

```
</style>
```

```
</head>
```

Selector de pseudo-elementos **:before** y **:after**

```
<body>
```

```
<h3>1. Internet</h3>
```

```
<h3>2. Páginas y Sitios Web</h3>
```

```
<p>
```

Ahora vamos a escribir algo

```
</p>
```

```
</body>
```

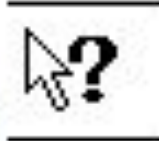
```
</html>
```

Selector de pseudo-elementos **:before** y **:after**

El resultado sería:

Tema: 1. Internet

Tema: 2. Páginas y Sitios Web

Ahora vamos a escribir algo 

Fin

Selector de atributos

- Para definir el estilo de aquellos elementos que tienen un determinado atributo o un valor de atributo dado. Aunque no son muy corrientes, pueden ser de utilidad en determinadas situaciones.
- La forma en que se declaran es:

Elemento[atributo] { estilo }

Selector de atributos

Se pueden utilizar distintos patrones:

- **x[atributo]**

Para hacer referencia a cualquier elemento del tipo x que cuente con un determinado atributo (independientemente del valor que tenga el atributo)

- **x[atributo="valor"]**

Para hacer referencia a cualquier elemento del tipo x que cuente con un determinado atributo cuyo valor sea el especificado

Selector de atributos

Se pueden utilizar distintos patrones:

- **x[atributo~="valor"]**

Algunos elementos pueden admitir varios valores separados por **espacios** para un atributo.

Para hacer referencia a cualquier elemento del tipo x que cuente con un determinado atributo, y alguno de cuyos **valores sea el especificado**.

- **x[atributo|="valor"]**

Algunos elementos pueden admitir varios valores **separados por guiones** para un atributo.

Para hacer referencia a cualquier elemento del tipo x que cuente con un determinado atributo, y cuyos valores **comiencen** por el especificado.

Selector de atributos

Ejemplos:

- el siguiente selector de atributo equivale a todos los elementos H1 que especifican el atributo "title", cualquiera sea su valor:

```
H1[title] { color: blue; }
```

- El selector equivale a todos los elementos SPAN cuyo atributo "class" tiene exactamente el valor "ejemplo":

```
SPAN[class=ejemplo] { color: blue; }
```

Selector de atributos

Ejemplo:

```
<html>
```

```
<head>
```

```
<style>
```

```
    h2[title] {color: red}
```

```
    h2[title="azul"] {color: blue}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
    <h2 title="algo">En color rojo</h2>
```

```
    <h2>En el color por defecto</h2>
```


Selector de atributos

Ejemplo:

```
<h2 title="azul">En color azul</h2>
```

```
<p><br></p>
```

<p>En el primer caso se aplica el estilo a todos los elementos h2 con el atributo especificado (title). En el segundo caso, sólo a los que contengan el atributo especificado con el valor dado.</p>

```
</body>
```

```
</html>
```

Selector por contexto del elemento

Para el uso correcto de este tipo de selectores debemos explicar previamente el concepto de **HERENCIA**.

- Cada página HTML está compuesta por una serie de elementos (títulos, párrafos, listas, tablas, etc.) organizados en una estructura donde cada elemento está contenido por otro elemento, que a su vez puede estar contenido por otro.

Selector por contexto del elemento

```
<html>  
  <head>  
    <title>                </title>  
  </head>  
  
  <body>  
    <h1>                  </h1>  
  
    <div>  
      <p><img>              </p>  
      <p><strong> </strong> </p>  
    </div>  
  
    <p>                    </p>  
  
    <ul>  
      <li> </li>  
      <li> </li>  
    </ul>  
  
  </body>  
</html>
```

Selector por contexto del elemento

- En esta estructura existe un **elemento raíz** que es el que actúa de contenedor de todos los demás elementos. En HTML se puede considerar como elemento raíz al elemento **<BODY>** o al elemento **<HTML>**.
- La importancia de este hecho es que **cada elemento hereda las propiedades del elemento que lo contiene** (llamado el **elemento padre**). Quiere decir que si especificamos la propiedad color: red para **<BODY>**, todos los elementos de la página heredarán esta característica y no será necesario especificar nuevamente la propiedad color en cada uno de ellos.

Selector por contexto del elemento

- Ejemplo:

```
<html>
```

```
<head>
```

```
<title>Untitled</title>
```

```
    <style type="text/css">
```

```
        body { color:red}
```

```
    </style>
```

```
</head>
```

```
<body>
```

```
<p>Parrafo1</p><p>Parrafo2</p>
```

```
Cursor de ayuda de Windows</body></html>
```

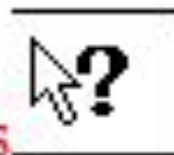
Selector por contexto del elemento

Ejemplo:

Parrafo1

Parrafo2

Cursor de ayuda de Windows



Selector por contexto del elemento

- Esta situación genera unas **relaciones** que pueden calificarse de **familiares** (de hecho, la nomenclatura para indicarlas utilizará los términos padre, hijo, ascendiente y descendiente entre otros).
- Vamos a ver a continuación algunos ejemplos de lo que significa esta jerarquía en la organización de los elementos de una página html para lo que podremos guiarnos por el siguiente esquema de relaciones:

Selector por contexto del elemento

Un elemento es **padre** de otro si...

- El **segundo elemento** se encuentra directamente entre la apertura y el cierre de las etiquetas del elemento padre. Ejemplo

```
<p>Los <strong>rumiantes</strong> son  
<em>mamíferos</em></p>
```

El elemento **<p>** es **padre** de **** y de ****

Selector por contexto del elemento

Un elemento es **hijo** de otro si...

- La **relación anterior es recíproca**, por lo que los elementos `` y `` del ejemplo son hijos del elemento `<p>`

Un elemento es **ascendiente** de otro si...

- Si el **segundo** se encuentra **entre las etiquetas de apertura y cierre del primero** (aunque sea de forma indirecta). Ejemplo:

Selector por contexto del elemento

Un elemento es **ascendiente** de otro si...

- Ejemplo:

```
<div>
```

```
  <p>Los <strong>rumiantes</strong> son  
  <em>mamíferos</em></p>
```

```
</div>
```

El elemento **<div>** es **ascendiente** de **<p>** de ****
y de ****

Selector por contexto del elemento

Un elemento es **descendiente** de otro si...

- Está incluido, **directa o indirectamente**, dentro de otro elemento.
- En el ejemplo anterior `` es descendiente de `<p>` y también de `<div>`

Un elemento es **adyacente o hermano** a otro si ...

- Comparte el **mismo padre** y, además se encuentra **inmediatamente consecutivo al anterior**.

Selector por contexto del elemento

Un elemento es **adyacente o hermano** a otro si ...

- Ejemplo:

```
<p>Los <strong>ungulados</strong> son  
  <em>mamíferos</em> con <em>pezuñas</em></p>
```

En este ejemplo **sólo el primer sería adyacente a ** ya que el segundo, aunque tiene el mismo padre ya no es inmediatamente posterior.

Selector por contexto del elemento

Los patrones para indicar estas relaciones para elaborar los selectores serían los siguientes:

Patrón

Utilización

X Y

Para apuntar hacia cualquier elemento del tipo **Y** que sea **descendiente** de un elemento del tipo **X**

X > Y

Para referirnos a cualquier elemento del tipo **Y** que sea **hijo** de un elemento del tipo **X**

X + Y

Para identificar cualquier elemento del tipo **Y** que sea **adyacente** a un elemento del tipo **X**

Selector por contexto del elemento

- Teniendo en cuenta estas relaciones podemos realizar una selección muy específica de elementos del documento para aplicarles un determinado estilo.

Ejemplo: Podríamos querer que los fragmentos que estuvieran enfatizados mediante la etiqueta `` aparecieran en color verde. Pero podríamos ser incluso más específicos indicando que **sólo** queremos que **aparezcan en color verde los fragmentos enfatizados que formen parte de una lista**. Esto lo haríamos colocando primero el **selector** `li` y, a continuación separado por un espacio, el **selector** `em` para luego especificar las propiedades. Añadir la regla al siguiente código.

Selector por contexto del elemento

```
<html>
```

```
<head>
```

```
<title>Untitled</title>
```

```
<body>
```

```
<p>La palabra <em>"atención"</em> está enfatizada  
pero se presenta en su color normal porque:</p>
```

Selector por contexto del elemento

```
<ol>
```

```
<li>Está enfatizada pero... </li>
```

```
<li><em>No pertenece a una lista</em></li>
```

```
</ol>
```

```
<p>La segunda condición también está enfatizada,  
pero en este caso pertenece a una lista y se ve en  
verde </p>
```

```
</body></html>
```


Selector por contexto del elemento

La palabra "*atención*" está enfatizada pero se presenta en su color normal porque:

1. Está enfatizada pero...
2. *No pertenece a una lista*

La segunda condición también está enfatizada, pero en este caso pertenece a una lista

Selector por contexto del elemento

- Se pueden utilizar los criterios de selección contextual para buscar etiquetas, clases, ids o combinaciones de estos.

Ejemplo:

- Se crea la **clase MAGENTA** que lo colorea todo de magenta. Todos los párrafos MAGENTA que estén dentro de un `<div>` estarán en cursiva. Además los textos dentro de `` anidados dentro de párrafos dentro de un `<div>` en MAGENTA usaran una fuente grande. Añadir las reglas al siguiente código.

Selector por contexto del elemento

<html>

<head>

<title>Untitled</title>

</head>

<body>

<div >

<h4> Titulo h4 en MAGENTA</h4>

Selector por contexto del elemento

Titulo h4 en MAGENTA

Este párrafo debería ser magenta y cursivo. Ahora viene un **texto grande**. Conseguimos esto

Este párrafo todavía es MAGENTA, pero como no está dentro de un bloque `<div>`, no es c

Agrupación de selectores

- Cuando tenemos **varios selectores** que **comparten** las mismas **declaraciones** pueden agruparse en una **lista separada por comas (,)**. Se trata de una fórmula abreviada para aplicar las mismas propiedades a distintos selectores en un solo paso.

- Por ejemplo, si tenemos:

H1 {font-family: Arial, Sans-serif}

P {font-family: Arial, Sans-serif}

TABLE {font-family: Arial, Sans-serif}

#volanta {font-family: Arial, Sans-serif}

- Podemos simplificarlo de este modo:

H1, P, TABLE, #volanta {font-family: Arial, Sans-serif}

Combinación de selectores de elementos, con s. de clases y con s. de identificador

- Los tipos de selectores: de elemento, de clase y de identificador pueden combinarse obteniendo selectores más complejos:
- **elemento.clase**
Con este selector, que combina un Selector de Elemento delante de un Selector de Clase, **distinguimos** a unos determinados **elementos** (aquellos que definen el **atributo class** con el **nombre de la clase**) de entre todos de su mismo tipo.

Combinación de selectores de elementos, con s. de clases y con s. de identificador

- **elemento#id_elemento**

Con este otro selector, que combina un Selector de Elemento con un Selector de Id se diferencia un elemento concreto (aquel que defina el atributo id con el nombre del identificador) del resto de elementos de su mismo tipo.

Ejemplo:

Combinación de selectores de elementos, con s. de clases y con s. de identificador

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
p.clase1 { font-family:serif;color:red; }
```

```
h2.clase2 { font-family:Courier;color:blue;  
font-weight:italic; }
```

```
h1#ident1 { font-family:Arial;color:green;  
font-weight:bold;}
```

```
</style>
```

```
</head>
```

Combinación de selectores de elementos, con s. de clases y con s. de identificador

```
<body>
```

```
<h1>Cabecera de tipo 1 sin estilo </h1>
```

```
<h1 id="ident1">Cabecera de tipo 1 con el estilo único  
para este elemento</h1>
```

```
<h2 class="clase2">Cabecera de tipo 2 con el estilo de  
clase2</h2>
```

```
<p>Párrafo sin ningún estilo</p>
```

```
<p class="clase1">Párrafo con el estilo de la clase1</p>
```

```
</body></html>
```

Combinación de selectores de elementos, con s. de clases y con s. de identificador

Cabecera de tipo 1 sin estilo

Cabecera de tipo 1 con el estilo único

Cabecera de tipo 2 con el estilo de clase2

Párrafo sin ningún estilo

Párrafo con el estilo de la clase1

Comentarios

- Cuando se crea una hoja de estilo, es **conveniente añadir comentarios**, especialmente si es extensa y con numerosas líneas de código. De este modo es más fácil recordar, cuando se vuelva a editar el código, qué es lo que hace cada parte.
- Para añadir comentarios a las hojas estilo, tanto internas como externas, se usan el carácter (`/*`) como inicio del comentario, y finalizaremos añadiendo (`*/`). Todo lo que se encuentre entre ambas marcas no será evaluado por el navegador.

Comentarios

- Ejemplo:

```
<style type="text/css">  
  /* p { color: red; }  
  div { color: red; } */  
  h1 { color: red; }  
</style>
```

Uso de mayúsculas y minúsculas

- Al escribir el código de las hojas de estilos no hay problema en que los nombres de las propiedades o de los valores aparezcan en mayúsculas o en minúsculas.
- Los nombres de selectores referentes a elementos HTML (body, div, ol, ul, etc.) tampoco tienen distinción entre un tipo de carácter u otro, pero los **nombres de clases** y los **identificadores** deben aparecer **exactamente igual** a como se hayan definido en el código HTML.