



# Conceptos básicos

---

- **World Wide Web (WWW):** Sistema de información.
- **Navegador:** Es el programa que nos ofrece acceso a Internet.
- **Servidor:** Se encarga de proporcionar al navegador los documentos y medios que este solicita.
- **HTTP:** Es el protocolo de transferencia de hipertexto, o sea, el protocolo que los servidores de World Wide Web utilizan para mandar documentos HTML a través de Internet.
- **URL:** Es el Localizador Uniforme de Recursos.
- **HTML:** "Lenguaje para marcado de hipertexto".

# El lenguaje HTML

---

- Lenguaje que estructura el documento.
- Proporciona el contenido (*utilizaremos CSS para el formato...*)
- Compatibilidad.
- Etiquetas:
  - `<tag>` texto afectado `</tag>`

## Estructura básica de un doc. HTML: Cabecera y cuerpo del documento

---

- **<HTML>**: Limitan el documento e indica que se encuentra escrito en este lenguaje.
- **<HEAD>**: Especifica el prólogo del resto del archivo. Son pocas las tags que van dentro de ella, destacando la del título **<TITLE>**.
- **<BODY>**: Encierra el resto del documento, el contenido.

# Primer ejemplo HTML

---

```
<HTML>  
<HEAD>  
<TITLE>Ejemplo 1</TITLE>  
</HEAD>  
<BODY>  
Hola mundo  
</BODY>  
</HTML>
```



## Primeros pasos: <H1> <P> <BR>

- <H1>, <H2>, <H3>...: Titulares. Sirven para dividir el texto en secciones. Se pueden definir seis niveles de titulares.
- <P>: Párrafos.
- <BR>: Saltos de línea.
- <!-- -->: Comentarios.

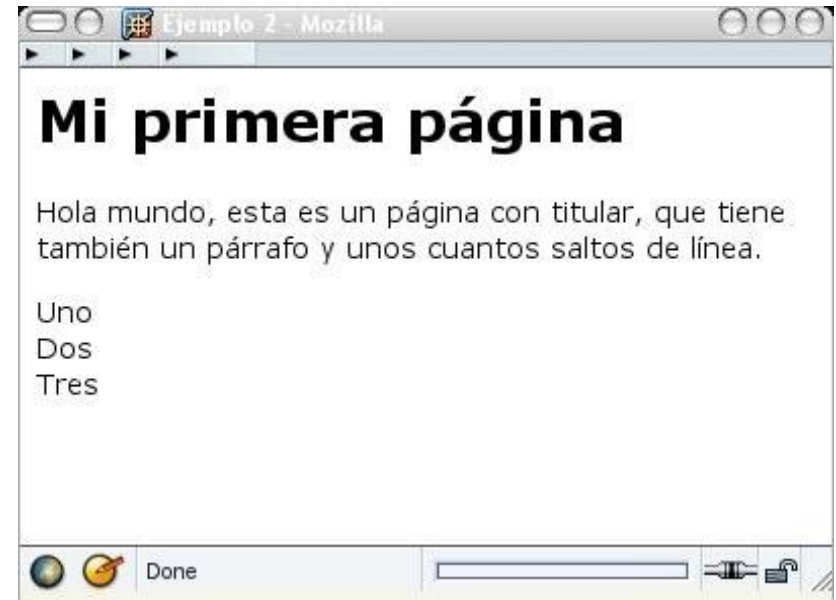
# Ejemplo

```
<HTML>
<HEAD>
<TITLE>Ejemplo 2</TITLE>
</HEAD>
<BODY>
  <H1>Mi primera página</H1>

  <!-- Aqui va un comentario que no es
        interpretado por el navegador -->

  <P>Hola mundo, esta es un página con titular,
    que tiene también un párrafo y unos cuantos
    saltos de línea.</P>

  Uno<br>
  Dos<br>
  Tres<br>
</BODY>
</HTML>
```



## Creación de enlaces <A>

---

- Lo característico del lenguaje HTML es el poder generar vínculos de hipertexto para enlazar con ellos todos sus documentos en web:
  - `<A HREF="URL">.....</A>`: Sirve para saltar entre diferentes URLs.
  - `<A NAME="parte1">Primera parte</A>`: Utilizamos el atributo name para dar nombre a una sección de nuestro documento.



# Ejemplo enlaces

```

<HTML>
<HEAD>
<TITLE>Ejemplo 3</TITLE>
</HEAD>
<BODY>
  <A NAME="arriba"><H1>Página de enlaces</H1></A>
  <A HREF="#abajo">Ir abajo</A><br>

  <A HREF="ej4.html">Ir a ejemplo 4</A><br><br>
  <A HREF="http://www.webestilo.com/">Ir a WebEstilo</A><br>

  <br>.<br>.<br>.<br>.<br>.<br>.<br>.<br><br>
  <br><br><br><br><br><br><br><br><br><br><br><br>
  <br><br><br><br><br><br><br><br><br><br><br><br>
  <br><br><br>.<br>.<br>.<br>.<br>.<br>

  <A NAME="abajo"><br></A>
  <A HREF="#arriba">Ir arriba</A>

</BODY>
</HTML>

```



# URL: Localizador Universal de Recursos

- Los URL son las direcciones de las informaciones que buscamos en Internet.
- Constan de:
  - **Protocolo**: Programa que utilizará el navegador para obtener el archivo elegido. (HTTP, FTP...)
  - **Nombre del host**: Se trata del sistema donde se encuentra almacenada la información que buscamos.
  - **Ruta del fichero**: Se trata de la ubicación del archivo dentro del host.
- Ej:
  - <http://www.bienvenidos.es/publico/saludos.html>

## Tipos de URL

---

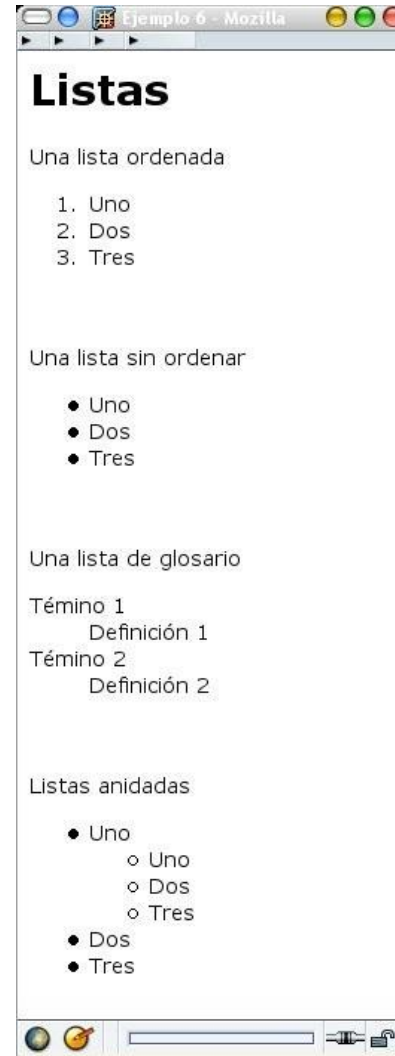
- **HTTP**: Los utilizados por los servidores de WWW.
- **FTP**: Apuntar archivos que estén en servidores que usan el protocolo FTP (File Transfer Protocol). Utilizado para enviar y recibir ficheros.
- **File**: Apuntan hacia archivos contenidos en el mismo disco que se encuentra el navegador
- **Mailto**: Se usa para mandar correos electrónicos.  
(webmaster@bienvenidos.es)
- **News**: Son URL de grupos de noticias.

# Listas <UL> <OL> <LI>

```

<HTML>
<HEAD>
<TITLE>Ejemplo 6</TITLE>
</HEAD>
<BODY>
<H1>Listas</H1>
Una lista ordenada
<ol>
<li>Uno</li>
<li>Dos</li>
<li>Tres</li>
</ol>
<br><br>
Una lista sin ordenar
<ul>
<li>Uno</li>
<li>Dos</li>
<li>Tres</li>
</ul>
<br><br>
Una lista de glosario
<dl>
<dt>Término 1</dt>
<dd>Definición 1</dd>
<dt>Término 2</dt>
<dd>Definición 2</dd>
</dl>
<br><br>
Listas anidadas
<ul>
<li>Uno
  <ul>
<li>Uno</li>
<li>Dos</li>
<li>Tres</li>
</ul>
</li>
<li>Dos</li>
<li>Tres</li>
</ul>
</BODY>
</HTML>

```



# Estilos de caracter <B> <U> <I>

```

<HTML>
<HEAD>
<TITLE>Ejemplo 7</TITLE>
</HEAD>
<BODY>

<H1>Estilos de caracter</H1>
Un ejemplo de texto de <EM>realzado</EM>.<br>
Un ejemplo de texto de <STRONG>realzado</STRONG>.<br>
Un ejemplo de texto de <CODE>realzado</CODE>.<br>
Un ejemplo de texto de <SAMP>realzado</SAMP>.<br>
Un ejemplo de texto de <KDB>realzado</KDB>.<br>
Un ejemplo de texto de <VAR>realzado</VAR>.<br>
Un ejemplo de texto de <DFN>realzado</DFN>.<br>
Un ejemplo de texto de <CITE>realzado</CITE>.<br>
<br>
Un ejemplo de texto de <B>realzado</B>.<br>
Un ejemplo de texto de <I>realzado</I>.<br>
Un ejemplo de texto de <U>realzado</U>.<br>
Un ejemplo de texto de <TT>realzado</TT>.<br>

</BODY>
</HTML>

```



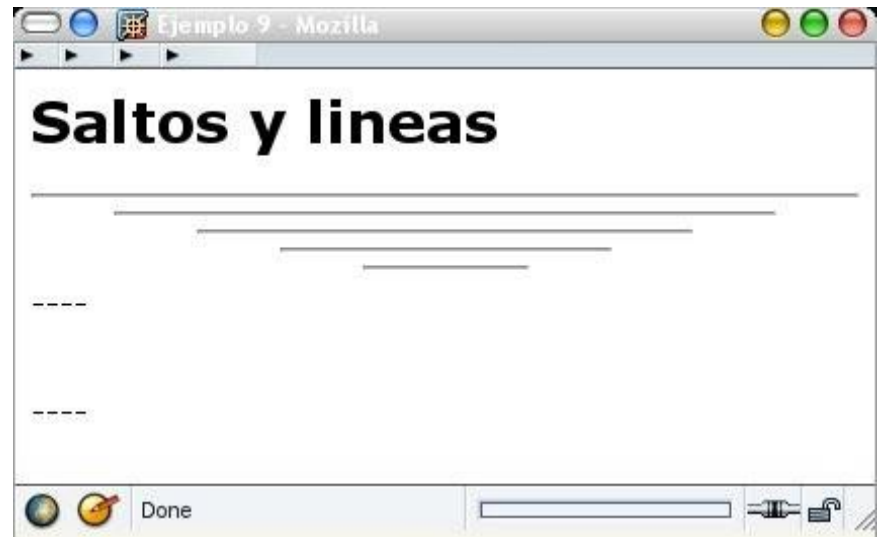
# Saltos y líneas <HR> <BR>

```
<HTML>
<HEAD>
<TITLE>Ejemplo 9</TITLE>
</HEAD>
<BODY>

<H1>Saltos y lineas</H1>

<HR>
<HR width="80%">
<HR width="60%">
<HR width="40%">
<HR width="20%">

----
<br><br><br>
----
</BODY>
</HTML>
```



## Carácteres especiales

---

- Insertar un carácter que no sea ASCII en nuestro documento HTML.
- Estas entidades pueden ser:
  - **Nombradas**: Se suelen construir con el signo &, la abreviatura del carácter y acabado con el signo punto y coma. El símbolo de registrado se construiría: &reg;
  - **Numeradas**: Se construyen con los signos &#, el número correspondiente a la posición del carácter en el conjunto ISO-Latin-1 (ISO-8859-1). El mismo símbolo de registrado, sería ahora: &#174;
- Existen unas entidades reservadas:
  - signo <, entidad &lt;
  - signo >, entidad &gt;
  - signo &, entidad &amp;
  - signo ", entidad &quot;

—

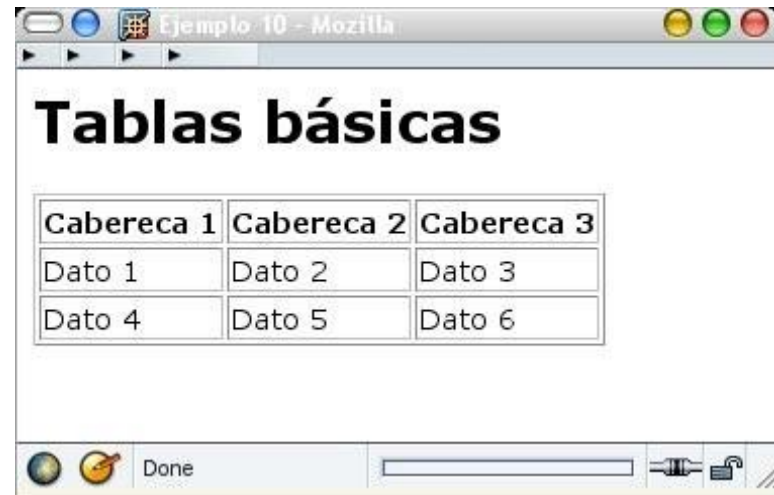
# Tablas

```
<HTML>
<HEAD>
<TITLE>Ejemplo 10</TITLE>
</HEAD>
<BODY>

<H1>Tablas básicas</H1>

<TABLE BORDER="1">
<TR>
  <TH>Cabecera 1</TH>
  <TH>Cabecera 2</TH>
  <TH>Cabecera 3</TH>
</TR>
<TR>
  <TD>Dato 1</TD>
  <TD>Dato 2</TD>
  <TD>Dato 3</TD>
</TR>
<TR>
  <TD>Dato 4</TD>
  <TD>Dato 5</TD>
  <TD>Dato 6</TD>
</TR>
</TABLE>

</BODY>
</HTML>
```





# Tablas avanzadas

---

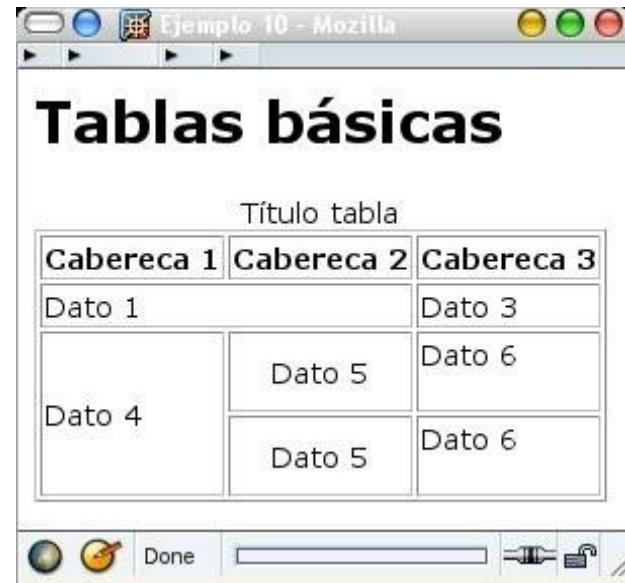
- Título
  - Cuando queramos titular una tabla usamos los tags `<caption>.....</caption>`. El tag de apertura puede llevar el atributo `align` que indicará si el título va encima o debajo de la tabla. `align="top"` indicaría encima de la tabla y `align="bottom"` indicaría en la parte de abajo.
- Alineación de celdas
  - Atributo `align=` define horizontalmente los datos al margen izquierdo (**left**), al derecho (**right**) o centrado (**center**).
  - Atributo `valign=` define verticalmente los datos en la parte superior (**top**), en la parte inferior (**bottom**) o centrado (**middle**).
- Celdas extendidas
  - Para crear una celda que abarque varias filas o columnas, debemos colocar en las tags `<th>` o `<td>` los atributos:
    - `rowspan=` más un valor para indicar el número de filas que se quiere abarcar.
    - `colspan=` más un valor para indicar el número de columnas que se quiere abarcar.

# Ejemplo tabla avanzada

```

<html><head>
<title>Ejemplo 10</title>
</head>
<body>
<h1>Tablas básicas</h1>
<table border="1">
<caption>Título tabla</caption>
<tr>
<th>Cabereca 1</th>
<th>Cabereca 2</th>
<th>Cabereca 3</th>
</tr>
<tr>
<td colspan="2">Dato 1</td>
<td>Dato 3</td>
</tr>
<tr>
<td rowspan="2">Dato 4</td>
<td align="center" valign="middle">Dato 5</td>
<td>Dato 6<br><br></td>
</tr>
<tr>
<td align="center" valign="middle">Dato 5</td>
<td>Dato 6<br><br></td>
</tr>
</table>
</body></html>

```



## Tablas (espaciado)

---

- Podemos modificar el aspecto de la tabla cambiando el ancho de los bordes, el espaciado entre celdas y el ancho de las mismas.
  - **width=** Acompaña a `<table>` y especifica el ancho de la tabla, tanto en número de píxeles como en porcentaje respecto al ancho de la pantalla. También puede acompañar a las tags `<th>` o `<td>` para especificar el ancho de las columnas.
  - **cellspacing=** Suele acompañar a la tag `<table>`. Indica el número de píxeles que separan cada celda. El valor predeterminado suele ser 2.
  - **cellpadding=** También acompaña a la tag `<table>`. Indica el espacio en píxeles entre el borde de la celda y su contenido. El valor predeterminado suele ser 1.

# Ejemplo tabla avanzada (espaciado)

```
<HTML>
<HEAD>
<TITLE>Ejemplo 11</TITLE>
</HEAD>
<BODY>
```

```
<H1>Tablas avanzadas</H1>
```

```
<TABLE WIDTH="50%" BORDER="1" CELSPACING="3" CELLPADDING="2">
```

```
<TR>
```

```
<TD COLSPAN="2" ALIGN="right">Dato 1</TD>
```

```
<TD>Un texto cualquiera</TD>
```

```
<TD ROWSPAN="2">Dato 3</TD>
```

```
</TR>
```

```
<TR>
```

```
<TD>Dato 4</TD>
```

```
<TD>Dato 4</TD>
```

```
<TD ALIGN="center">Dato 5</TD>
```

```
</TR>
```

```
</TABLE>
```

```
</BODY>
```

```
</HTML>
```



# Imágenes <IMG>

- Incluir imágenes en una presentación web es muy sencillo, solo debe de tener en cuenta que las imágenes tienen que tener los formatos GIF, JPEG o PNG. Las imágenes en línea, se especifican a partir de la tag `<img>` que no tiene una tag correspondiente de cierre pero que puede acompañarse de los siguientes atributos:
  - `src`= Este atributo es obligatorio e indica el nombre del archivo de imagen (entre comillas) o la URL que se va a representar.
  - `align`= Permite controlar la alineación de una imagen con respecto a una línea de texto adyacente o a otras imágenes en esa línea. Los tres valores posibles son los ya conocidos left, right, top, middle y bottom.
  - `alt`= Es la alternativa que se estableció cuando todavía existían visualizadores de solo texto. Entre comillas podremos escribir un texto que suplantara a esta imagen si no se carga o mientras se carga o cuando, visualizando ya la imagen, pasamos el ratón por encima.
  - `width`= Este atributo es opcional pero es recomendable ponerlo para ayudar al navegador a representar la imagen, significa el ancho de la imagen que vamos a representar.
  - `height`= Al igual que el atributo WIDTH, es opcional y recomendable ponerlo, este significa el alto de la imagen.
  - `border`= Con BORDER especificamos el ancho de un borde que rodea la imagen.

# Ejemplo imagen

```
<HTML>
<HEAD>
<TITLE>Ejemplo 12</TITLE>
</HEAD>
<BODY>
```

```
<H1>Imágenes</H1>
```

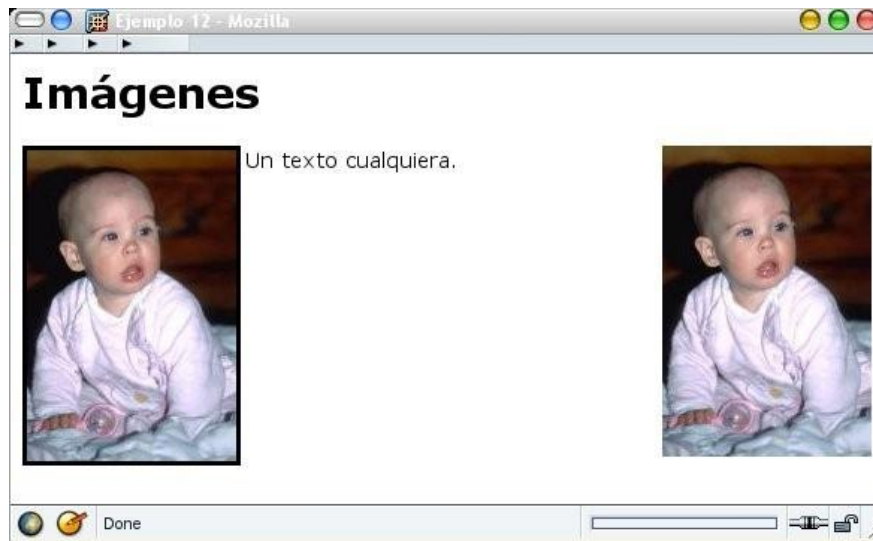
```
<IMG SRC="/graficos/bebe1.jpg" WIDTH=140 HEIGHT=210 BORDER=0 ALT="Un bebé;" ALIGN="RIGHT">
```

```
<IMG SRC="/graficos/bebe1.jpg" WIDTH=140 HEIGHT=210 BORDER=3 ALT="Un bebé;" ALIGN="LEFT">
```

Un texto cualquiera.

```
</BODY>
```

```
</HTML>
```



# Formularios

---

- Los formularios permiten solicitar información al visitante.
- Tags `<form>`.....`</form>` y dentro de ellas se recogerán todas las variables de entrada.
  - `action=""` Indica el programa que va a tratar las variables enviadas con el formulario.
  - `method=""` Indica el método de transferencia de las variables.  
`Post`, si se envía a través del *STDIO*. `Get`, si se envía a través de la URL.

# Ejemplo formulario

```
<HTML>
<HEAD>
<TITLE>Ejemplo 14</TITLE>
</HEAD>
<BODY>
```

```
<H1>Formularios</H1>
```

```
<FORM ACTION="prueba.php" METHOD="POST">
<INPUT TYPE="text" NAME="nombre"><BR>
<INPUT TYPE="submit">
<INPUT TYPE="Reset">
</FORM>
```

```
</BODY>
</HTML>
```





# Campos de entrada <INPUT>

- Atributos:
  - `type=""` tipo de variable a introducir.
    - `text` Campo texto. Sus atributos:
      - `maxlength=""` Número máximo de caracteres a introducir.
      - `size=""` Limitará el numero de caracteres a mostrar en pantalla.
      - `value=""` Valor inicial del campo.
    - `password` Indica que el campo a introducir será una palabra de paso. Mostrará asteriscos. Atributos igual que para text.
    - `checkbox` Opcion casilla cuadrada.
      - `value=""` Valor de la casilla.
      - `checked` Marcada por defecto.
    - `radio` Varias opciones en casilla circular.
      - `value=""` Valor de la casilla.
    - `image` El campo contendrá el valor en coordenadas del punto de la imagen que haya pinchado. Atributo obligatorio:
      - `src=""` Nombre del archivo de imagen.
    - `hidden` Invisible. Se debe añadir el atributo `value=""`
  - `name=""` Nombre del campo.

# Ejemplo INPUT

```
<HTML>
<HEAD>
<TITLE>Ejemplo 15</TITLE>
</HEAD>
<BODY>
```

```
<H1>Formularios</H1>
```

```
<FORM ACTION="prueba.php" METHOD="POST">
```

```
  Texto: <INPUT TYPE="text" NAME="nombre"><BR>
```

```
  Password: <INPUT TYPE="password" NAME="contra"><BR>
```

```
  Sexo: <INPUT TYPE="radio" NAME="boton1" VALUE="1"> Hombre
```

```
<INPUT TYPE="radio" NAME="boton1" VALUE="2">Mujer<BR>
```

```
  Vehiculo: <INPUT TYPE="checkbox" NAME="Moto" VALUE="Si">Moto
```

```
<INPUT TYPE="checkbox" NAME="Coche" VALUE="" CHECKED>Coche
```

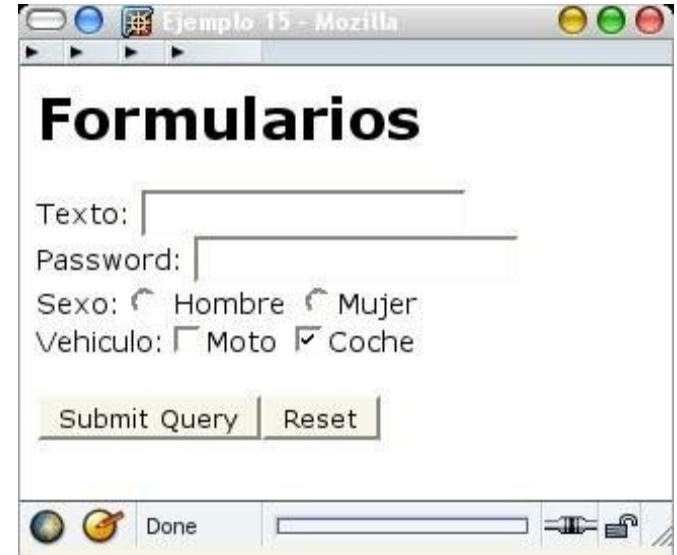
```
<BR><BR>
```

```
<INPUT TYPE="submit"><INPUT TYPE="Reset">
```

```
</FORM>
```

```
</BODY>
```

```
</HTML>
```



## Campos de selección <SELECT>

- Las tags `<select>`.....`</select>` encierran los valores que podremos elegir a partir de una lista.
- Atributos:
  - `name=""` Nombre del campo.
  - `size=""` Número de opciones visibles. 1, menú desplegable. Valor mayor, lista con barra de desplazamiento.
  - `multiple` Si se pueden realizar multiples selecciones.
- Las diferentes opciones de la lista se indicarán mediante la tag `<option>` que puede acompañarse del atributo `selected`.

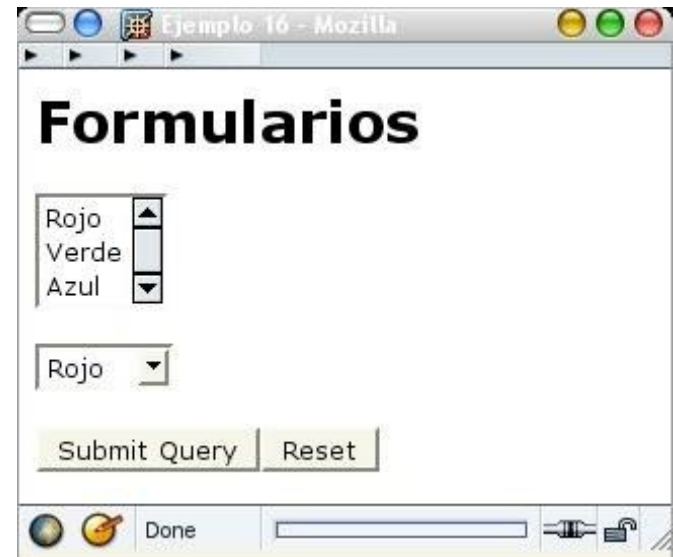
# Ejemplo <SELECT>

```
<HTML>
<HEAD>
<TITLE>Ejemplo 16</TITLE>
</HEAD>
<BODY>

<H1>Formularios</H1>

<FORM ACTION="prueba.php" METHOD="POST">
<SELECT NAME="Colores" MULTIPLE>
  <OPTION VALUE="r">Rojo</OPTION>
  <OPTION VALUE="g">Verde</OPTION>
  <OPTION VALUE="b">Azul</OPTION>
</SELECT>
<BR><BR>
<SELECT NAME="Colores" SIZE="1">
  <OPTION VALUE="r">Rojo</OPTION>
  <OPTION VALUE="g">Verde</OPTION>
  <OPTION VALUE="b">Azul</OPTION>
</SELECT>
<BR><BR>
<INPUT TYPE="submit"><INPUT TYPE="Reset">
</FORM>

</BODY>
</HTML>
```



## Areas de texto <TEXTAREA> y botones <INPUT TYPE="submit">

```
<HTML>
<HEAD>
<TITLE>Ejemplo 16</TITLE>
</HEAD>
<BODY>

<H1>Formularios</H1>

<FORM ACTION="prueba.php" METHOD="POST">
<TEXTAREA COLS=20 ROWS=10 NAME="Texto">
</TEXTAREA>
<BR><BR>
<INPUT TYPE="submit">
<INPUT TYPE="Reset">
</FORM>

</BODY>
</HTML>
```



## Capas <div>

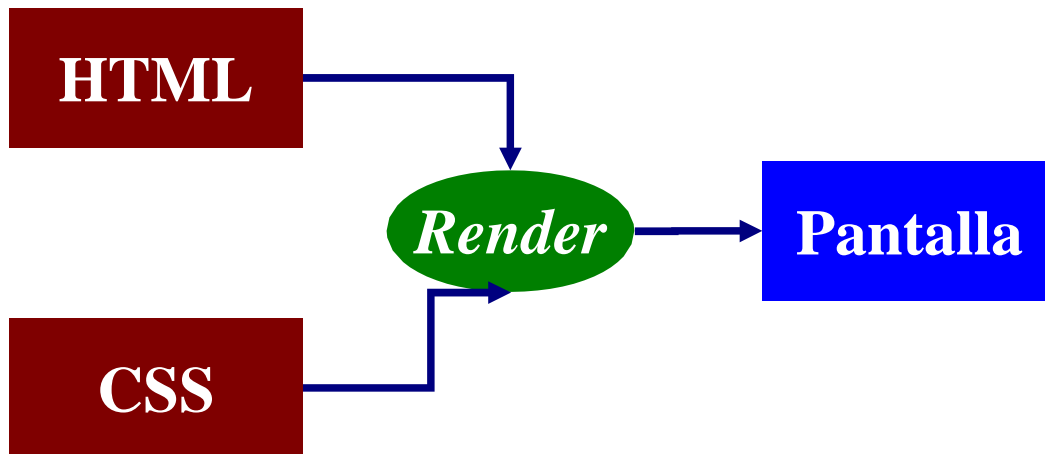
---

- Las capas se pueden definir como páginas que se pueden incrustar dentro de otras.
- Propiedades de estilo:
  - **left** y **top**: Sitúan la esquina superior izquierda de la capa respecto a la esquina superior izquierda de la capa donde esté.
  - **width** y **height**: Determinan la anchura y altura de la capa.
  - **clip**: área que se podrá ver dentro de la capa
    - ej: *clip:rect(20px 30px 40px 10px);* recortará la capa creando un cuadro visible cuya esquina superior izquierda está a 10 pixels por la izquierda y 20 por arriba de la de la capa y cuyo tamaño sería de 30-10 de ancho y 40-20 de alto.  
*rect(top right bottom left)*
  - **z-index**: Las capas con un mayor z-index se colocarán más arriba
  - **visibility**: visible, hidden (oculta) o inherit (hereda la visibilidad de la capa padre)
  - **background-image**: Gráfico de fondo de la capa

# CSS

---

- Separar el formato visual de la página del contenido.
- **CSS** son las siglas de "**Cascade StyleSheet**". Se trata de una especificación sobre los estilos físicos aplicables a un documento HTML, y trata de dar la separación definitiva de la lógica (estructura) y el físico (presentación) del documento.



# Introducción

---

- Objetivos
  - Dar formato páginas web
  - Separar diseño – contenido
  - Centralizar propiedades aspecto
  - Facilitar mantenimiento diseño



## Características

---

- Fichero de texto
- Formato de páginas HTML, PHP, ...
- Herencia de propiedades entre elementos (jerarquía)
- Clases definibles por el usuario

# Propiedades

- Longitud (por ej., márgenes):

- [-]NNtipo

- NN:

<ul style="list-style-type: none"> <li>– <b>em</b> (el alto de la M mayúscula)</li> <li>– <b>ex</b> (la mitad de la altura de la M mayúscula)</li> <li>– <b>px</b> (píxel)</li> </ul>	<i>relativas</i>
---	------------------

<ul style="list-style-type: none"> <li>– <b>pt</b> (puntos)</li> <li>– <b>pc</b> (picas)</li> <li>– <b>in</b> (inches)</li> <li>– <b>mm</b> (milímetros)</li> <li>– <b>cm</b> (centímetros).</li> </ul>	<i>absolutas</i>
---	------------------

- Color:

- Notación **#RRGGBB**, siendo RR, GG, BB los valores en hexadecimal de las componentes roja, verde y azul del color.
  - Nombre predefinido (**white**, **yellow**...).
  - Usando la función **rgb(R,G,B)**, donde R, G, B son los valores en decimal.

# Propiedades – Fuente y Texto

## Fuente <Font>

Atributo	Valores		
font-family	:	arial   sans-serif   helvetica   verdana...	;
font-style	:	normal   italic   oblique	;
font-variant	:	normal   small-caps	;
font-weight	:	normal   bold   bolder   lighter   100..500..900	;
font-size	:	xx-small   x-small   small   medium   large   x-large   xx-large	;

## Texto <Text>

Atributo	Valores		
word-spacing	:	normal y unidades de medida CSS	;
letter-spacing	:	normal y unidades de medida CSS	;
text-decoration	:	none   [ underline    overline    line-through ]	;
vertical-align	:	baseline   sub   super   top   text-top   middle	;
text-transform	:	capitalize   uppercase   lowercase   none	;
text-align	:	left   right   center   justify	;
line-height	:	normal y unidades de medida CSS	;

# Propiedades – Fondo y Clasificación

Fondo <Background>			
Atributo		Valores	
color	:	color, nombre o valor RGB	;
background-color	:	color, nombre o valor RGB	;
background-image	:	URL - nombre de la imagen con path relativo o absoluto	;
background-repeat	:	repeat-x   repeat-y   no-repeat   repeat	;
background-attachment	:	scroll   fixed	;
background-position	:	top   center   bottom   left   right   unidades de medida CSS	;

Clasificación			
Atributo		Valores	
display	:	block   inline   list-item   none	;
white-space	:	normal   pre   nowrap	;
list-style-type	:	disc   circle   square   decimal   lower-roman   upper-roman   lower-alpha	;
list-style-image	:	URL - nombre de la imagen con path relativo o absoluto	;
list-style-position	:	inside   outside	;

# Propiedades - Marco

Marco <Box>			
Atributo		Valores	
margin-top	:	unidades de medida CSS	;
margin-right	:	unidades de medida CSS	;
margin-bottom	:	unidades de medida CSS	;
margin-left	:	unidades de medida CSS	;
margin	:	unidades de medida CSS	;
padding-top	:	unidades de medida CSS	;
padding-right	:	unidades de medida CSS	;
padding-bottom	:	unidades de medida CSS	;
padding-left	:	unidades de medida CSS	;
padding	:	unidades de medida CSS	;
border-top-width	:	thin   medium   thick   unidades de medida CSS	;
border-right-width	:	thin   medium   thick   unidades de medida CSS	;
border-bottom-width	:	thin   medium   thick   unidades de medida CSS	;
border-left-width	:	thin   medium   thick   unidades de medida CSS	;
border-width	:	thin   medium   thick   unidades de medida CSS	;

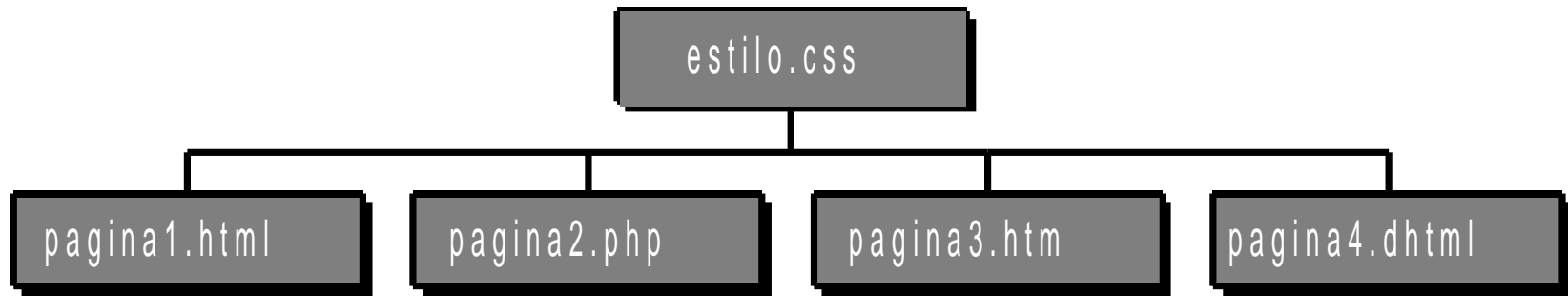
# Propiedades - Marco

border-color	:	color, nombre o valor RGB	;
border-style	:	none   dotted   dashed   solid   double   groove   ridge	;
border-top	:	none   dotted   dashed   solid   double   groove   ridge	;
border-right	:	none   dotted   dashed   solid   double   groove   ridge	;
border-bottom	:	none   dotted   dashed   solid   double   groove   ridge	;
border-left	:	none   dotted   dashed   solid   double   groove   ridge	;
border	:	none   dotted   dashed   solid   double   groove   ridge	;
width	:	auto   unidades de medida CSS	;
height	:	auto   unidades de medida CSS	;
float	:	left   right   none	;
clear	:	left   right   none   both	;

- Referencia: <http://www.w3.org/TR/REC-CSS1>
  - margin, margin-left, border, background, font-family...*

# Página de estilos

---



```
<html>
```

```
<head>
```

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

```
</head>
```

```
<body>
```

```
...
```

```
</body>
```

```
</html>
```

## Aplicación

---

- Aplicación directa en etiquetas.
- Redefinición de etiqueta.
- Separar HTML de CSS.



## Aplicación directa en etiquetas

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Ejemplo 16</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

La negrita que vemos **es más grande y está en rojo**

```
</BODY>
```

```
</HTML>
```

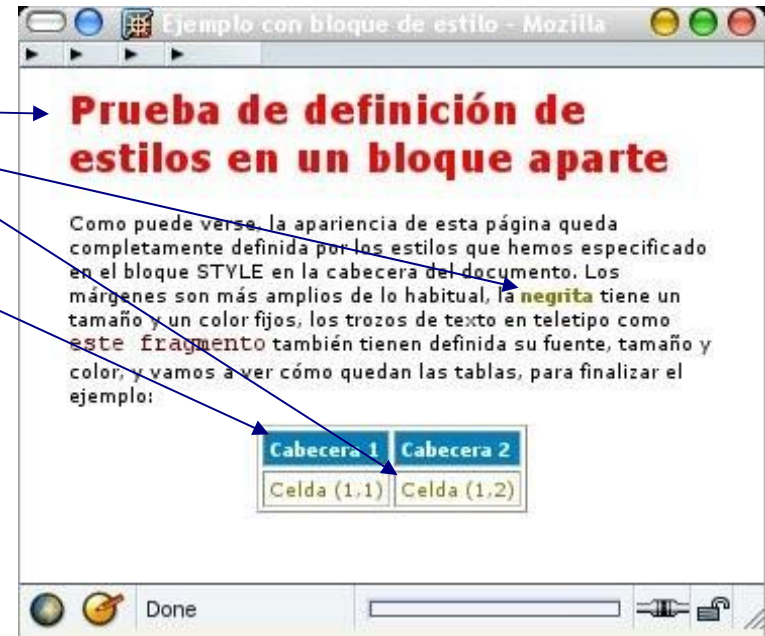
La negrita que vemos **es más grande y está en rojo**

# Redefinición de etiqueta

```

<HTML>
<HEAD>
<TITLE> Ejemplo con bloque de estilo </TITLE>
<STYLE TYPE="text/css">
<!--
BODY {font-family:Verdana,sans-serif;font-size:x-small;
margin-left:0.25in; margin-right:0.25in}
H2 {font-family:Verdana,sans-serif;font-size:14pt;color:red}
B, TD {font-family:Verdana,sans-serif;font-size:x-small;color:olive}
TH {font-family:Verdana,sans-serif;font-size:x-small;
color:white;background-color:#0080C0}
PRE, TT, CODE {font-family:Courier New,Courier;font-size:9pt;color:maroon}
-->
</STYLE>
</HEAD>
<BODY BGCOLOR=white>
<H2>Prueba de definición de estilos en un bloque aparte</H2>
Como puede verse, la apariencia de esta página queda
completamente definida por los estilos que hemos
especificado en el bloque STYLE en la cabecera del
documento. Los márgenes son más amplios de lo habitual,
la <B>negrita</B> tiene un tamaño y un color fijos, los
trozos de texto en teletipo como <TT>este fragmento</TT>
también tienen definida su fuente, tamaño y color,
y vamos a ver cómo quedan las tablas, para finalizar
el ejemplo: <P>
<CENTER>
<TABLE BORDER=1 CELLSPACING=2 CELLPADDING=2>
<TR> <TH>Cabecera 1</TH> <TH>Cabecera 2</TH> </TR>
<TR> <TD>Celda (1,1)</TD> <TD>Celda (1,2)</TD> </TR>
</TABLE>
</CENTER>
</BODY>
</HTML>

```



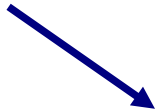
# Separar HTML de CSS

```
<HTML>
<HEAD>
<TITLE> Ejemplo con hoja de estilo externa </TITLE>
<LINK REL="stylesheet" TYPE="text/css" HREF="ejemplo.css">
</HEAD>
<BODY BGCOLOR=white>
```

```
<H2>Prueba de definición de estilos en una hoja de
estilo externa</H2>
```

Como puede verse, la apariencia de esta página queda completamente definida por los estilos que hemos especificado en el bloque STYLE en la cabecera del documento. Los márgenes son más amplios de lo habitual, la <B>negrita</B> tiene un tamaño y un color fijos, los trozos de texto en teletipo como <TT>este fragmento</TT> también tienen definida su fuente, tamaño y color, y vamos a ver cómo quedan las tablas, para finalizar el ejemplo: <P>

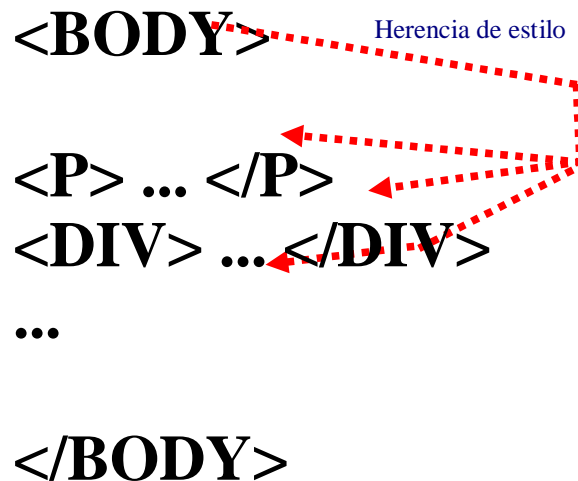
```
<CENTER>
<TABLE BORDER=1 CELLSPACING=2 CELLPADDING=2>
<TR> <TH>Cabecera 1</TH> <TH>Cabecera 2</TH> </TR>
<TR> <TD>Celda (1,1)</TD> <TD>Celda (1,2)</TD> </TR>
</TABLE>
</CENTER>
</BODY>
</HTML>
```



```
BODY {font-family:Verdana,sans-serif;font-size:x-small;
margin-left:0.25in; margin-right:0.25in}
H2 {font-family:Verdana,sans-serif;font-size:14pt;color:red}
B, TD {font-family:Verdana,sans-serif;font-size:x-small;color:olive}
TH {font-family:Verdana,sans-serif;font-size:x-small;
color:white;background-color:#0080C0}
PRE, TT, CODE {font-family:Courier New,Courier;font-size:9pt;
```

# Herencia de estilos

- Existe una jerarquía de etiquetas que contienen a otras, para darnos una relación de herencia:



# Ejemplo herencia

```
<HTML>
<HEAD>
<TITLE> Ejemplo 1 de herencia </TITLE>
<LINK REL="stylesheet" TYPE="text/css" HREF="ejemplo.css">
</HEAD>
<BODY BGCOLOR=white>
```

Todo el texto tiene definido el estilo Fuente: Verdana, Tamaño: x-small, Margen izquierdo: 0.25in, Margen derecho: 0.25in, **<SPAN STYLE="color:red">pero este trozo de línea es de un color distinto, conservando el resto de propiedades</SPAN>**, y eso hace interesante la herencia y la posibilidad de cambiar en partes concretas los estilos heredados.

```
</BODY>
</HTML>
```

*/\* Definición de estilos en un archivo aparte \*/*

*/\* Estilo para el documento \*/*

```
BODY {font-family:Verdana,sans-serif;font-size:x-small;
margin-left:0.25in; margin-right:0.25in}
```

*/\* Estilo para la cabecera de nivel 2 \*/*

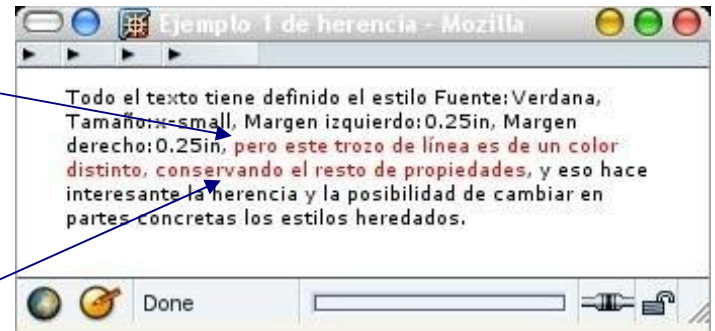
```
H2 {font-family:Verdana,sans-serif; font-size:14pt;color:red}
```

*/\* Estilos para otras etiquetas \*/*

```
B, TD {font-family:Verdana,sans-serif;font-size:x-small;
color:olive}
```

```
TH {font-family:Verdana,sans-serif;font-size:x-small;
color:white;background-color:#0080C0}
```

```
PRE, TT, CODE {font-family:Courier New,Courier;
font-size:9pt;color:maroon}
```



## Estilo en función del contexto

- Si queremos que el texto en negrita sea de color verde cuando se encuentre en una celda de una tabla, o que sea de color púrpura cuando forme parte de una lista.
- Esto lo definiríamos como sigue:

```
TD B {color:green}  
UL B {color:purple}
```

# Clases

---

- Una clase es una definición de estilo que no está asociado a ninguna etiqueta HTML, pero que podemos asociar, en el documento, a etiquetas concretas.
- Definimos la clase:
  - `.Nombre_de_la_Clase {propiedad1:valor;...;propiedadN:valor}`
- Utilizaremos el parámetro CLASS como sigue:
  - `<etiqueta CLASS="Nombre_de_la_Clase"> ... </etiqueta>`

# Ejemplo clases

```

<HTML>
<HEAD>
<TITLE> Ejemplo de uso de clases </TITLE>
<STYLE TYPE="text/css">
<!--
BODY { font-family:Verdana,sans-serif;font-size:x-small}
P,A,B {color:red}
.BAzul {color:blue}
-->
</STYLE>
</HEAD>

```

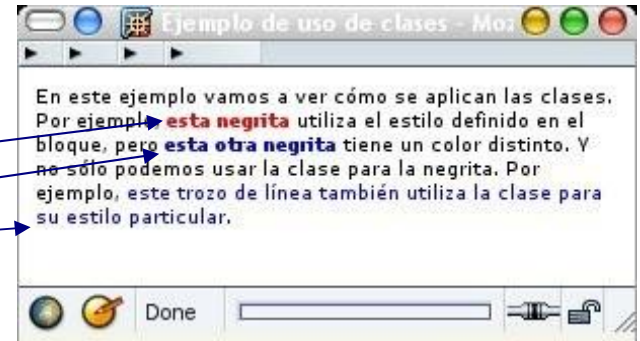
```
<BODY BGCOLOR=white>
```

En este ejemplo vamos a ver cómo se aplican las clases.  
 Por ejemplo, `<B>esta negrita</B>` utiliza el estilo definido en el bloque, pero `<B CLASS="BAzul">esta otra negrita</B>` tiene un color distinto. Y no sólo podemos usar la clase para la negrita. Por ejemplo, `<SPAN CLASS="BAzul">este trozo de línea también utiliza la clase para su estilo particular</SPAN>`.

```

</BODY>
</HTML>

```





# Etiquetas

---

- Cualquier etiqueta HTML puede tener como parámetro la etiqueta ID seguida de un nombre:
  - `<etiqueta ID="NombreReferencia"> ... </etiqueta>`
- Este "NombreReferencia" debe ser único en el documento HTML.
- Para definir un estilo mediante un ID:
  - `#Nombre_del_ID {propiedad1:valor;...;propiedadN:valor}`

# Pseudoclases

---

- Hay ocasiones en las que HTML da a algunas etiquetas un estilo propio: por ejemplo, los enlaces aparecen por defecto de otro color y subrayados. Estos elementos son las pseudoclases. Por ahora, sólo están definidas para la etiqueta `<A>`.
- Definición:
  - `etiqueta:pseudoclase {propiedad1:valor;...;propiedadN:valor}`
- y las pseudoclases de que disponemos son:
  - `link` Nos dice el estilo de un enlace que no ha sido visitado.
  - `visited` Nos dice el estilo de un enlace que ha sido visitado.
  - `active` Nos dice el estilo de un enlace que está siendo pulsado.
  - `hover` Nos dice el estilo de un enlace sobre el que está pasando el ratón.

## Ejemplo pseudoclasas

- Por ejemplo, si deseásemos que apareciesen todos los enlaces sin subrayar, definiríamos los siguientes estilos:
  - **`A:link,A:visited,A:active {text-decoration:none}`**
- Las pseudoclasas pueden usarse de forma conjunta con las clases:
  - **`A.NombreClase:pseudoclase`**
- y también se pueden usar en función del contexto.

# Enlaces

---

- Referencia HTML 4: <http://www.w3.org/TR/REC-html40/>
- Referencia CSS1: <http://www.w3.org/TR/REC-CSS1>
- Cursos:
  - HTML:
    - <http://www.programacion.com/html/tutorial/curso/>
    - <http://www.webestilo.com/html/>
  - CSS:
    - <http://www.programacion.com/html/tutorial/css1/>
    - <http://www.webestilo.com/css/>
- Ejemplos: <http://www.domedia.org/oveklykken/articles.php>
- Validadores: <http://validator.w3.org/>