

---

# Tema 2. Lenguaje de Marcas

# Índice.

---

## 1. Introducción

1. Definición lenguaje de marcas
2. Evolución de los lenguajes de marcas: SGML,HTML,XHTML,XML y características de estos.

## 2. Lenguaje de marcas: HTML-XHTML

## 3. Hojas de Estilo CSS

# Parte I: Introducción

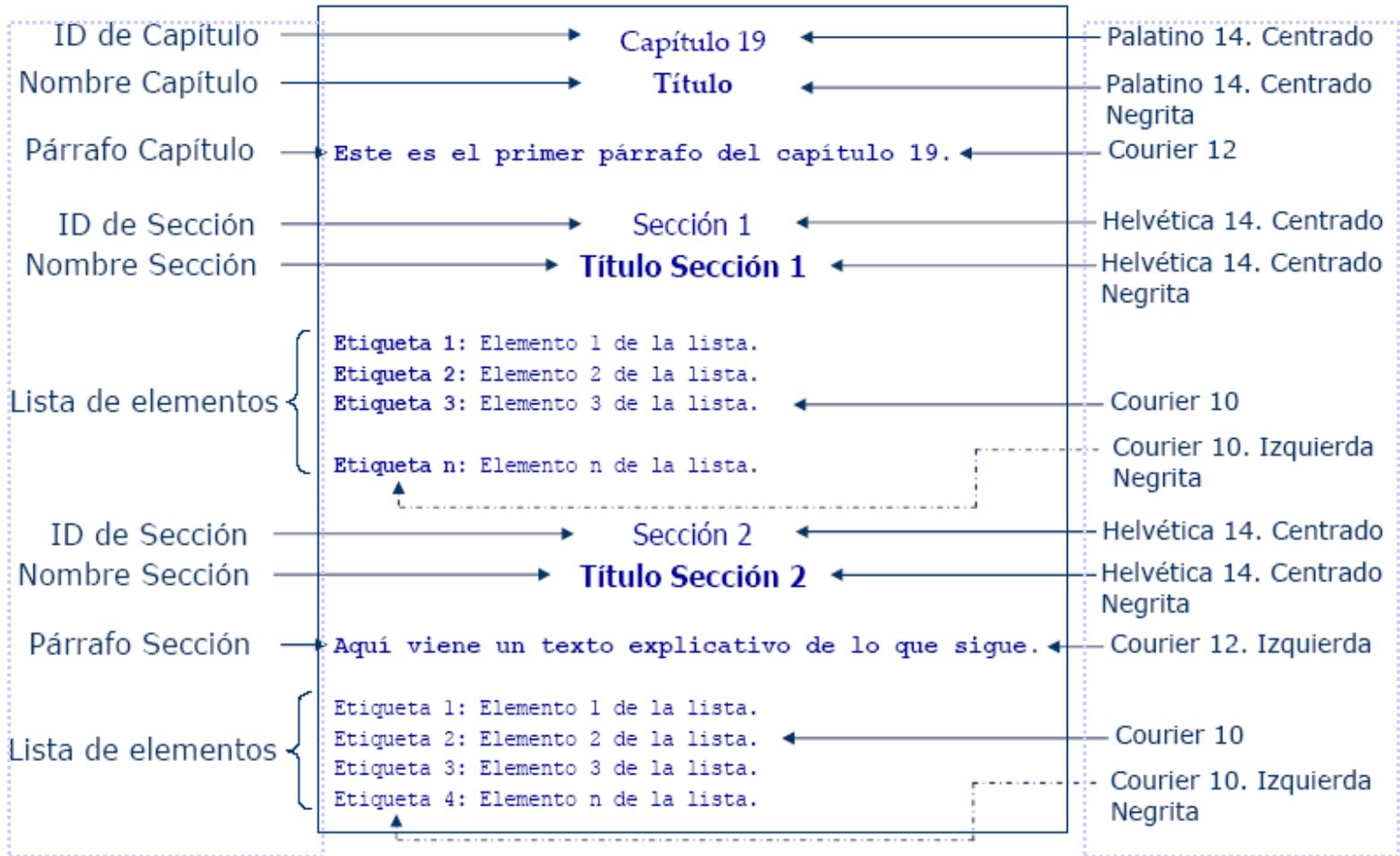
---

# Lenguajes de marcas

---

- Marcas: Término empleado para describir los códigos, también llamados **etiquetas**, añadidos al texto electrónico y que definen la estructura y/o el formato en el que tienen que aparecer
- Puede ser utilizado con múltiples propósitos: escritura, impresión, intercambio de información, presentación de pantallas, etc
- Lenguajes de marcado:
  - **Estructura lógica:** formada por las partes que lo componen y sus relaciones
  - **Estructura física:** indica la apariencia del documento incluyendo sus componentes físicos, posicionamiento de elementos y tipografía empleada

# Lenguajes de marcas



Estructura lógica

Informática II

Estructura física

# Lenguajes de marcas

---

- Lenguajes de marcas genéricos:
  - Son aquellos que sirven para especificar la estructura de cualquier documento *sin tener en cuenta* los aspectos relativos a *la presentación*.
  - Posteriormente, el mismo documento se podrá presentar de diversas formas, según las *normas de estilo* que se le apliquen

# Lenguajes de marcas. SGML.

---

## Standard Generalized Markup Language

- Norma ISO-8879 originada por la comunidad editorial para flexibilizar el diseño de documentos.
- Proporciona una forma *normalizada* de transmitir los documentos en formato adecuado para los procesos de edición e impresión
- Es apropiado para describir texto altamente estructurado.
- Es un **meta-lenguaje** para especificar lenguajes de marcado:
  - Contiene reglas para crear lenguajes de marcado, pero *no describe el formato* de los documentos marcados
- Mediante la utilización de una definición de tipo de documento (**DTD: Document Type Definition**) se puede especificar la estructura lógica de una clase de escrito.

# Documentos estructurados

---

- SGML( Standard Generalized Markup Language):
  - Sistema para definir tipos de estructuras de documentos.
  - Uso de etiquetas para marcar partes de un texto.
  - Deriva de GML, creado por IBM para que un mismo fichero pudiera ser editado por diferentes aplicaciones con distintos formatos de impresión (libros, informes, ediciones electrónicas).
- Ejemplo: para definir un capítulo tendríamos un elemento título, un autor, y uno o más párrafos.

# Lenguajes de marcas. SGML.

```
<!ELEMENT antologia      - - (poema+) >
<!ELEMENT poema         - - (titulo?, estrofa+) >
<!ELEMENT titulo        - - (#PCDATA) >
<!ELEMENT estrofa       - - (linea+) >
<!ELEMENT linea         - - (#PCDATA) >
```

**Ejemplo de DTD**

```
<antologia>
  <poema>
    <titulo>A un panal de rica miel</titulo>
    <estrofa>
      <linea> A un panal de rica miel</linea>
      <linea>cien mil moscas acudieron,</linea>
      <linea>que por golosas murieron,</linea>
      <linea>presas de patas en él.</linea>
    </estrofa>
    <estrofa>
      <linea>Así, si bien se examina,</linea>
      <linea>los humanos corazones</linea>
      <linea>perecen en las prisiones</linea>
      <linea>del vicio que los domina. </linea>
    </estrofa>
  </poema>
  <!-- Aquí irían más poemas de la antología -->
</antologia>
```

**Ejemplo de documento  
basado en el DTD anterior**

# Lenguajes de marcas.HTML

---

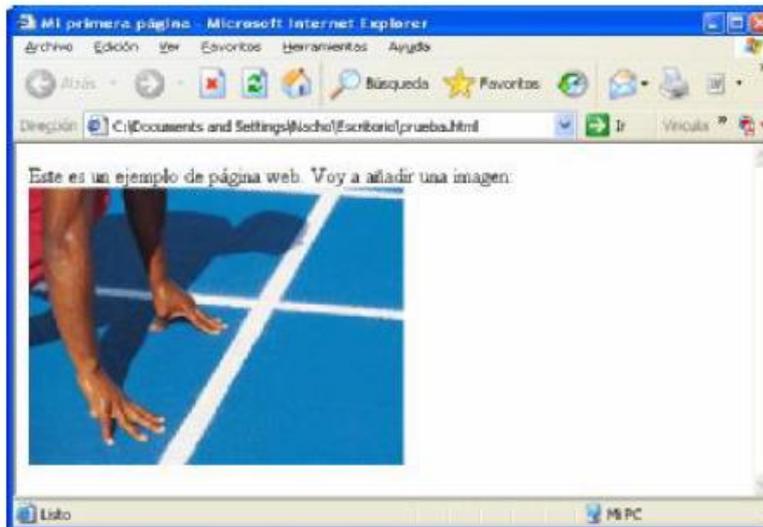
## HyperTextual Markup Language

- HTML aparece como una simplificación de SGML para codificar documentos y publicarlos en la WWW, con una semántica específica
- HTML es una DTD, no un meta-lenguaje como SGML.
- Está orientado en gran medida a la presentación de los documentos más que a su estructura.
- Contiene un **número limitado** y no extensible **de marcas** que se intercalan en el documento y permiten describir el aspecto con el que ha de visualizarse.
- Permite publicar documentos en la red con:
  - Cabeceras, textos, tablas, listas, fotografías...
  - Encontrar información en línea mediante enlaces hipertextuales.
  - Diseñar formularios para realizar transacciones con servicios remotos, buscar información, hacer pedidos, etc.
  - Incluir elementos multimedia: vídeos, sonido...

# HTML.Ejemplos.

```
<HTML>
  <HEAD>
    <TITLE>Mi primera página</TITLE>
    <META name="autor" content="una pruebecilla">
  </HEAD>
  <BODY>
    Este es un ejemplo de página web. Voy a añadir una imagen:<BR>
    <IMG src="prueba.jpg"> <BR>
  </BODY>
</HTML>
```

Ejemplo de HTML



Representación en el navegador

# HTML.Ejemplos.

```
<p> <dt> <b> <a href="/exec/obidos/ASIN/0764531999/qid=919015337">
  Xml : Extensible Markup Language</a></b> ~
<font color=#990033>Usually ships in 24 hours</font>
<dd> Elliotte Rusty Harold / Paperback / Published 1998
<br> Our Price: $31.99 ~ <font color=#990033>You Save: $8.00 (20%)</font>
<br>
<a href="/exec/obidos/ASIN/0764531999/qid=919015337">
  <i>Read more about this title...</i></a>
```

*Ejemplo de HTML. Ficha  
de un libro:*

Estructura de la página.

Visualización.

[Xml : Extensible Markup Language](#) ~ Usually ships in 24 hours  
Elliotte Rusty Harold / Paperback / Published 1998  
Our Price: \$31.99 ~ You Save: \$8.00 (20%)  
[Read more about this title...](#)

# Lenguajes de marcas: HTML

---

- HTML es fácil de aprender y está muy extendido, pero:
  - Es superficial: sólo describe cómo se deben mostrar los documentos, y además describe vagamente su contenido. Esto hace que los servidores web sirvan muchas páginas que los usuarios piden para poder ver su contenido y averiguar si les son útiles o no.
  - Las reglas de HTML son muy relajadas, y además los navegadores suelen ser muy generosos con las páginas que no cumplen correctamente la sintaxis del lenguaje.
  - No es extensible: el conjunto de etiquetas está limitado y no se pueden definir otras nuevas, lo que supone que no se puede adaptar a las diferentes necesidades.

La solución pasa por buscar un lenguaje que permita describir los documentos desde un punto de vista de estructura y no solo de aspecto:  
XML

# La solución: XML

## XML: eXtensible Markup Language

- XML es un derivado del SGML que vendrá a simplificarlo, pero no será tan restringido como HTML.
- XML es un **meta-lenguaje** de marca: lenguaje genérico que podremos particularizar para definir lenguajes de marca para aplicaciones específicas.
- Permite describir la estructura del documento. Antes **‘el qué’** que **‘el cómo’**.

### Ejemplo XML:

```
<LIBRO>
  <TITULO> La Regenta </TITULO>
  <CAPITULO NUM="1" >
    <TITULOC>Uno </TITULOC>
    <PARRAFO NUM="1">
      La heroica ciudad dormía la siesta ...
    </PARRAFO>
  </CAPITULO>
</LIBRO>
```

# XML

---

- XML está orientado a la **estructura de los documentos**, no a su presentación.

## Ejemplo. Tenemos un correo electrónico:

**De:** Antonia Lluesma [antonia.lluesma@uv.es]

**Enviado:** lunes, 03 de mayo de 2004 9:36

**Para:** SI@listserv.uv.es

**Asunto:** Parada Aplicación SIUV en MVS

Mañana martes, día 4-mayo-2004, la Aplicación SIUV (Matrícula, Registro, Nóminas, Actas...) de MVS no estará operativa desde las 15:00 hasta las 16:00 horas, por actualización y mantenimiento.

Rogamos disculpen las molestias.

Un saludo,

-----  
Antonia Lluesma (e-mail: antonia.lluesma@uv.es)

Teléfono 963544312 - Ext.44312

Fax 96-35-44200

Operación de Sistemas

Servicio de Informática

Universidad de Valencia

# XML

**Podríamos identificar sus partes de la siguiente forma:**

```
<?xml version="1.0" ?>
<!doctype email system "http://www.uv.es/DTDs/email.dtd">
<email id="E1X108">
  <head>
    <from><name>De: Antonia Lluesma</name><address> [antonia.lluesma@uv.es]</address></from>
<date>Enviado: lunes, 03 de mayo de 2004 9:36</date>
    <to>Para: SI@listserv.uv.es</to>
    <subject>Asunto: Parada Aplicación SIUV en MVS</subject>
  </head>

  <body>
Mañana martes, día 4-mayo-2004, la Aplicación SIUV (Matrícula, Registro, Nóminas, Actas...)de MVS no
estará operativa desde las 15:00 hasta las 16:00 horas, por actualización y mantenimiento.
Rogamos disculpen las molestias.

Un saludo,
-----
Antonia Lluesma (e-mail: antonia.lluesma@uv.es)
Teléfono 963544312 - Ext.44312
Fax 96-35-44200
Operación de Sistemas
Servicio de Informática
Universidad de Valencia
</body>
</email>
```

# Características de XML

- XML es un subconjunto simplificado pero estricto de SGML (*Standard Generalized Markup Language*):
  - **Extensible:** se pueden definir nuevas etiquetas.
  - **Sintaxis estricta:** lo que hace posible el desarrollo de aplicaciones que evalúen los documentos.
  - **Estructurado:** se puede modelar datos a cualquier nivel de complejidad, y su definición está en una DTD, *Document Type Definition*.
  - **Validable:** cada documento se puede validar frente a una DTD, o en su defecto, se puede declarar bien formado.
  - **Independiente de medio:** para publicar contenidos en múltiples formatos.
  - **Independiente de fabricante y de plataforma:** para poder utilizar cualquier herramienta estándar.
  - **Estándar:** es un lenguaje basado en un estándar público, de modo que no pertenece a ninguna organización comercial.
- Los documentos XML son fácilmente procesables y compartidos en Internet.

# Ejemplo de Documento HTML

```
<p> <dt> <b> <a href="/exec/obidos/ASIN/0764531999/qid=919015337">
  Xml : Extensible Markup Language</a></b> ~
<font color=#990033>Usually ships in 24 hours</font>
<dd> Elliotte Rusty Harold / Paperback / Published 1998
<br/> Our Price: $31.99 ~ <font color=#990033>You Save: $8.00 (20%)</font>
<br/>
<a href="/exec/obidos/ASIN/0764531999/qid=919015337">
  <i>Read more about this title...</i></a>
```

*Ejemplo de HTML. Ficha  
de un libro:*

Estructura de la página.

Visualización.

[Xml : Extensible Markup Language](#) ~ Usually ships in 24 hours  
Elliotte Rusty Harold / Paperback / Published 1998  
Our Price: \$31.99 ~ You Save: \$8.00 (20%)  
[Read more about this title...](#)

# Ejemplo de Documento XML

```
<?xml version="1.0"?>
<libro>
  <titulo>Xml: Extensible Markup Language</titulo>
  <disponible tiempo="24" unidad="hours"/>
  <autor>Elliote Rusty Harold</autor>
  <formato>Paperback</formato>
  <publicacion>1998</publicacion>
  <precio cantidad="31.99" moneda="dolar"/>
  <descuento cantidad="20"/>
  <enlacelibro href="/exec/obidos/ASIN/0764531999/qid=919015337"/>
</libro>
```

*Ejemplo de XML: ficha de un libro.*

[Xml : Extensible Markup Language](#) ~ Usually ships in 24 hours

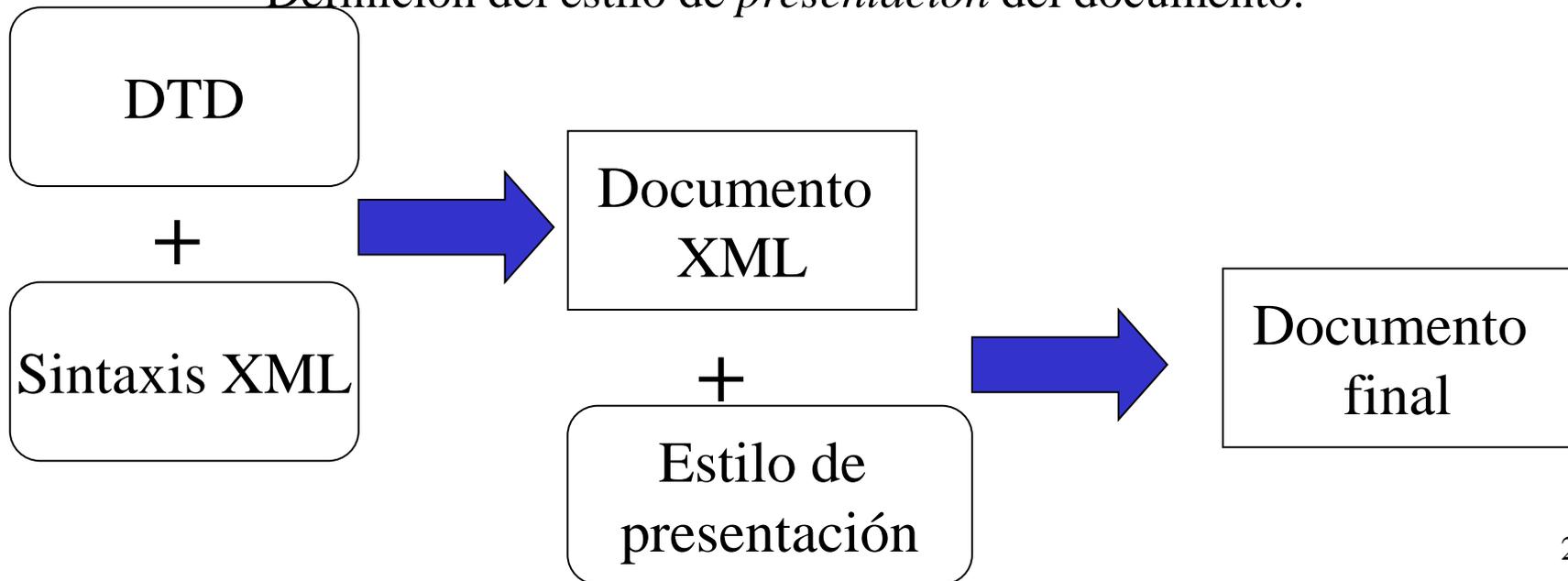
Elliote Rusty Harold / Paperback / Published 1998

Our Price: \$31.99 ~ You Save: \$8.00 (20%)

[Read more about this title...](#)

# Características de XML

- Al tratarse de un meta-lenguaje hemos de distinguir en él los siguientes componentes fundamentales:
  - *Sintaxis XML* (derivada de la de SGML)
  - Especificación de las definiciones de *tipo de documento* (DTD's) que son las que definen cada lenguaje particular.
  - Escritura de Documentos XML
  - Definición del estilo de *presentación* del documento.

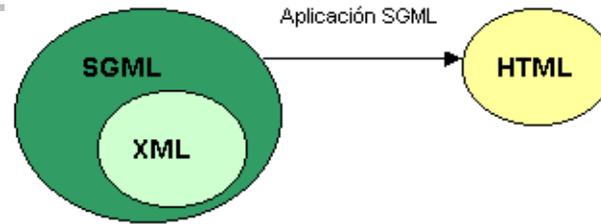


# Alternativa a XML, HTML: XHTML

- XHTML( eXtensible Hypertext Markup Language):
  - Lenguaje de marcas que combina las fortalezas de XML y HTML
  - Sintaxis más estricta que HTML que va a permitir una correcta interpretación de la información independientemente del dispositivo desde el que se acceda a ella
  - Puede incluir otros lenguajes como MathML(basado en XML, permite incluir fórmulas), o SVG (describir gráficos vectoriales usando XML)al contrario que HTML.
  - Uso de etiquetas para marcar partes de un texto.
  - Orientado al uso de un etiquetado correcto, exige una serie de requisito básicos a cumplir→1)elementos correctamente anidados, 2)elementos cerrados correctamente, 3) etiquetas en minúsculas, 4) valores de atributos entre comillas 5)un único elemento raíz
  - XHTML no se usa para aplicar estilo al documento→hojas de estilo, CSS, se encargan de ello.

# Documentos estructurados

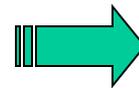
- *Ventajas* de SGML:
  - Es extensible: permite al autor definir nuevo elementos o atributos.
  - Estructura: se define un modelo de información del documento.
- *Inconvenientes* de SGML:
  - Mucho más complejo que HTML.
  - Gran número de opciones en las que deben ponerse de acuerdo el emisor y el receptor.



Nuevo lenguaje de marcas:

XML.

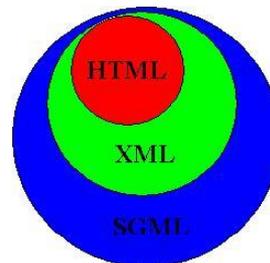
- Subconjunto de SGML, estructurado y extensible.
- Mejora HTML.
- Más fácil que SGML.



Nuevo lenguaje de marcas:

XHTML.

- Combina fortalezas HTML+XML
- Sintaxis más estricta
- Portable



# Parte II: Lenguaje de Marcas: HTML- XHTML

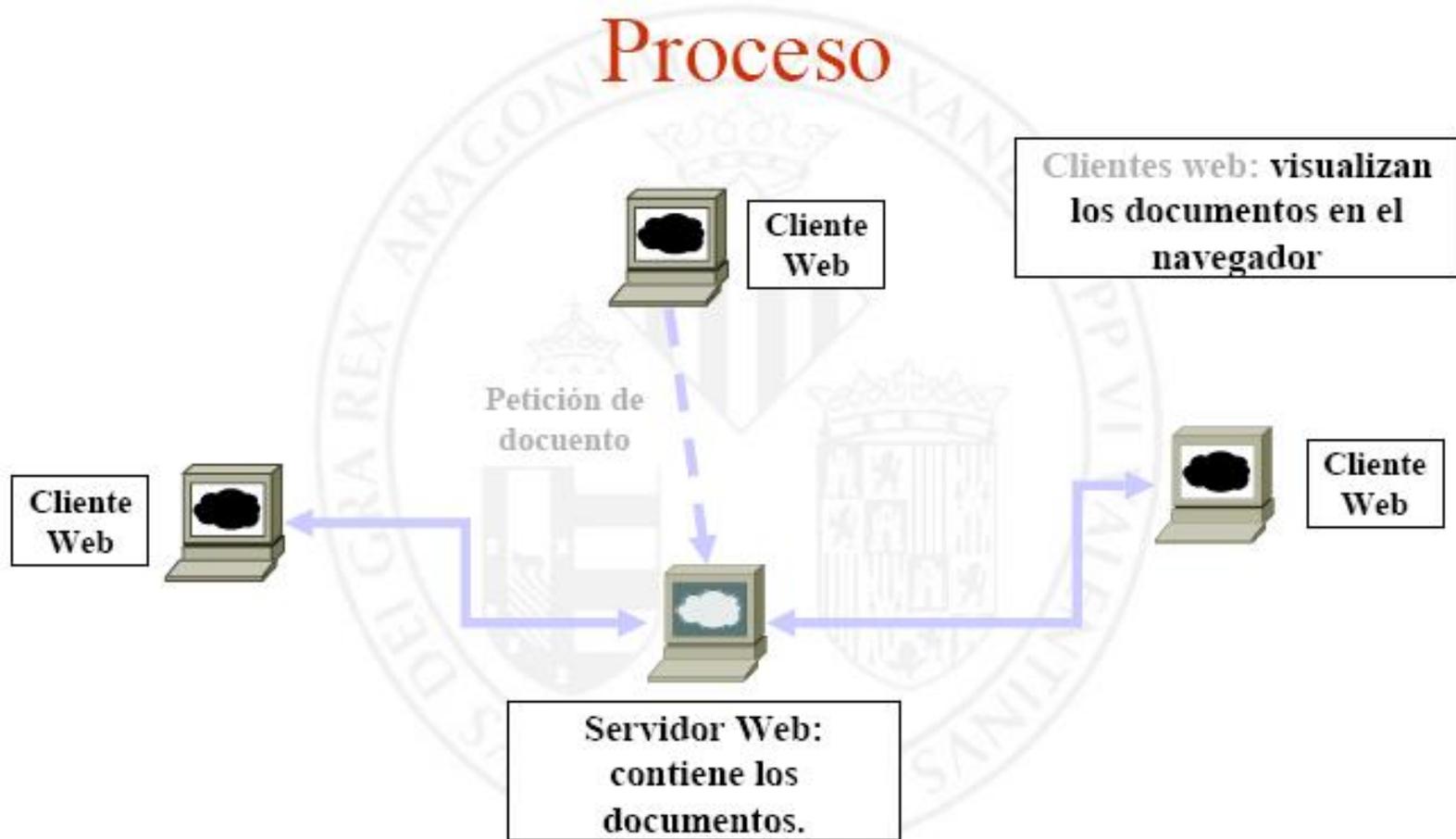
---

# Definiciones

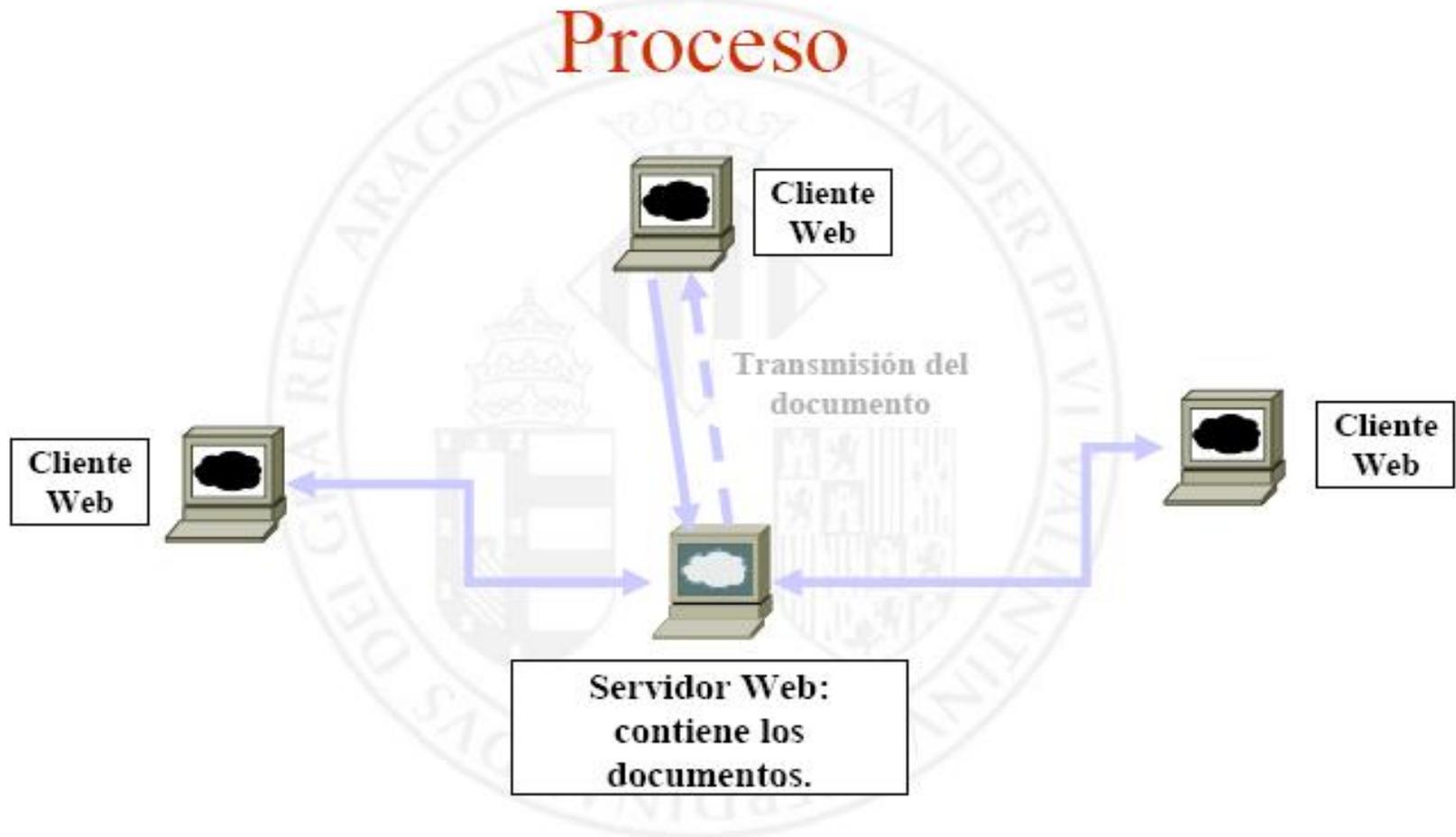
---

- *WWW (World Wide Web)*. Es un sistema de información global que permite relacionar mediante hipertexto distintas fuentes de información → configuración: red de servidores entrelazados por los hiperenlaces.
- *HTTP (HyperTextTransferProtocol)*. Protocolo usado por un cliente WWW para solicitar documentos a un servidor WWW (p.e. [www.terra.es](http://www.terra.es) [www.elpais.es](http://www.elpais.es)), y transferir estos documentos desde el servidor.
- Los documentos se visualizan en el navegador, p.e: Netscape o Internet Explorer.

# Proceso



# Proceso



# Definiciones

---

- Páginas web: textos ASCII escritos en el lenguaje HTML o XHTML.
- HTML: HyperTextMarkupLanguage.  
XHTML: Extensible HypertextMarkupLanguage.
- Lenguajes de marcas usado para diseñar las páginas web.
- Página web compuesta por distintos elementos: texto, dibujos, tablas, listas, etc. --> permiten estructurar la información para mostrarla.
- Facilita la interacción con el usuario a través de formularios-> posibilidad de introducir datos.
- Extensiones: “.html” o “.htm” o “.xhtml”

# Publicar una página Web

---

- Crear un documento HTML.
- Situarlo en un directorio accesible dentro del servidor web.
- Se accede a la página a través de su dirección URL:

<http://maquina/camino/fichero.html>

- Ej: <http://mural.uv.es/~ajojo/>

(si se omite el fichero, se supone index.html o index.htm)

# Elementos de un Documento HTML/XHTML

---

- HTML/XHTML ofrece un conjunto de etiquetas o marcas para incluir elementos en un página web.
- Un documento HTML estará formado por:
  - **Texto.**
  - **Caracteres especiales.**
    - Valores simbólicos:
      - acute. Ejemplo: á -&aacute
      - tilde. Ejemplo: ñ -&ntilde
      - cedill Ejemplo: ç -&ccedil
  - **Etiquetas HTML/XHTML** (o directivas).

# Elementos de un Documentos HTML/XHTML

---

- **Etiquetas** (directivas o comandos).

- Situadas entre dos marcas:

  - Marca de apertura <

  - Marca de cierre >

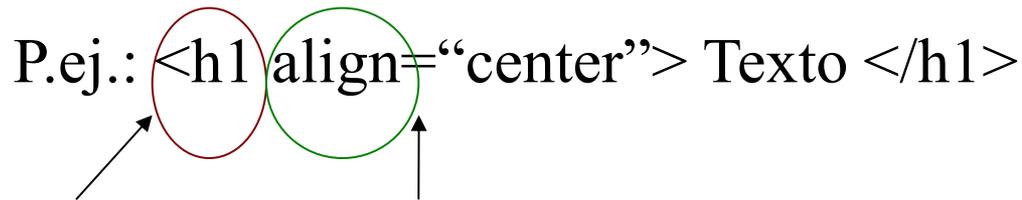
- Estructura: <etiqueta>

# Elementos de un Documento HTML/XHTML

- **Atributos:** las etiquetas tienen asociados atributos que modifican su significado o comportamiento.

-Se incluyen dentro de la etiqueta de apertura:

**<etiqueta atributo1="valor"..... atributo\_n="valor">**

P.ej.:  <h1 align="center"> Texto </h1>

etiqueta

atributo

- **Comentarios:** <!--.... -->

# Lenguaje HTML-XHTML

- HTML/XHTML describe los documentos mediante etiquetas. Etiquetas encerradas entre `<` y `>` con `/` indican fin de etiqueta.  
*Tipos de etiqueta:* Pareadas `<h1>..</h1>`  
Sin Parear `<br/>...`

Se **deben** seguir las siguientes reglas de sintaxis:

- 1) Etiquetas anidadas correctamente
- 2) Etiquetas deben estar cerradas correctamente
- 3) Elementos y atributos deben escribirse en minúscula
- 4) Un único elemento raíz
- 5) El valor de un atributo debe aparecer entre “ “
- 6) Está prohibida la utilización de atributos minimizados

# Lenguaje HTML-XHTML

1) Las etiqueta deben estar anidadas correctamente

```
<ul>
  <li>Coffee</li>
  <li>Tea
    <ul>
      <li>Black tea</li>
      <li>Green tea</li>
    </ul>
  <li>Milk</li>
</ul> This is correct:
```

**Incorrecto**

```
<ul>
  <li>Coffee</li>
  <li>Tea
    <ul>
      <li>Black tea</li>
      <li>Green tea</li>
    </ul>
  </li>
  <li>Milk</li>
</ul>
```

**Correcto**

# Lenguaje HTML-XHTML

## 2) Etiquetas deben estar cerradas correctamente

```
<p>This is a paragraph  
<p>This is another paragraph
```

Incorrecto

```
<p>This is a paragraph</p>  
<p>This is another paragraph </p>
```

Correcto

This is correct:

# Lenguaje HTML-XHTML

## 3) Etiquetas sin parear deben estar cerradas correctamente

```
A break: <br >  
A horizontal rule: <hr >  
An image: 
```

**Incorrecto**

```
A break: <br />  
A horizontal rule: <hr />  
An image: 
```

**Correcto**

This is correct:

# Lenguaje HTML-XHTML

---

4)Etiquetas siempre deben escribirse en minúscula

```
<BODY>  
<P>This is a paragraph</P>  
</BODY>
```

Incorrecto

```
<body>  
<p>This is a  
paragraph</p>  
</body>
```

Correcto

This is correct:

# Lenguaje HTML-XHTML

---

5) Los nombres de los atributos deben escribirse en minúscula y sus valores entre “ ”

```
<table WIDTH=100%>
```

Incorrecto

```
<table width="100%">
```

Correcto

This is correct:

# Lenguaje HTML-XHTML

---

5) La minimización de los atributos está prohibida

```
<input checked>  
<input disabled>
```

**Incorrecto**

```
<input checked="checked">  
<input disabled="disabled">
```

**Correcto**

# Ejercicio: Encuentra los errores??

---

<p>

Hola deciros que p es una etiqueta pareada y que tiene un atributo llamado align

</P>

<p align=left>

Esta es un ejemplo de párrafo que NO está bien formado .....y escribo lo que quiero. Si quiero darle un determinado tamaño a un texto puedo usar la etiqueta pareada font con un atributo llamado size como por ejemplo

<FONT size=10> esto es un ejemplo<FONT>

Bueno ya os he dado algunas pistas..

Aquí debería acabar mi parrafo....

# SOLUCIÓN: Encuentra los errores??

<p>

Hola, deciros que p es una etiqueta pareada y que tiene un atributo llamado align

</p>

<p align="left">

Esta es un ejemplo de párrafo que NO está bien formado .....y escribo lo que quiero. Si quiero darle un determinado tamaño a un texto puedo usar la etiqueta pareada font con un atributo llamado size como por ejemplo

<font size="10"> esto es un ejemplo</font>

Bueno ya os he dado algunas pistas.....

Aquí debería acabar mi párrafo....

</p>

# Estructura general de un documento HTML-XHTML

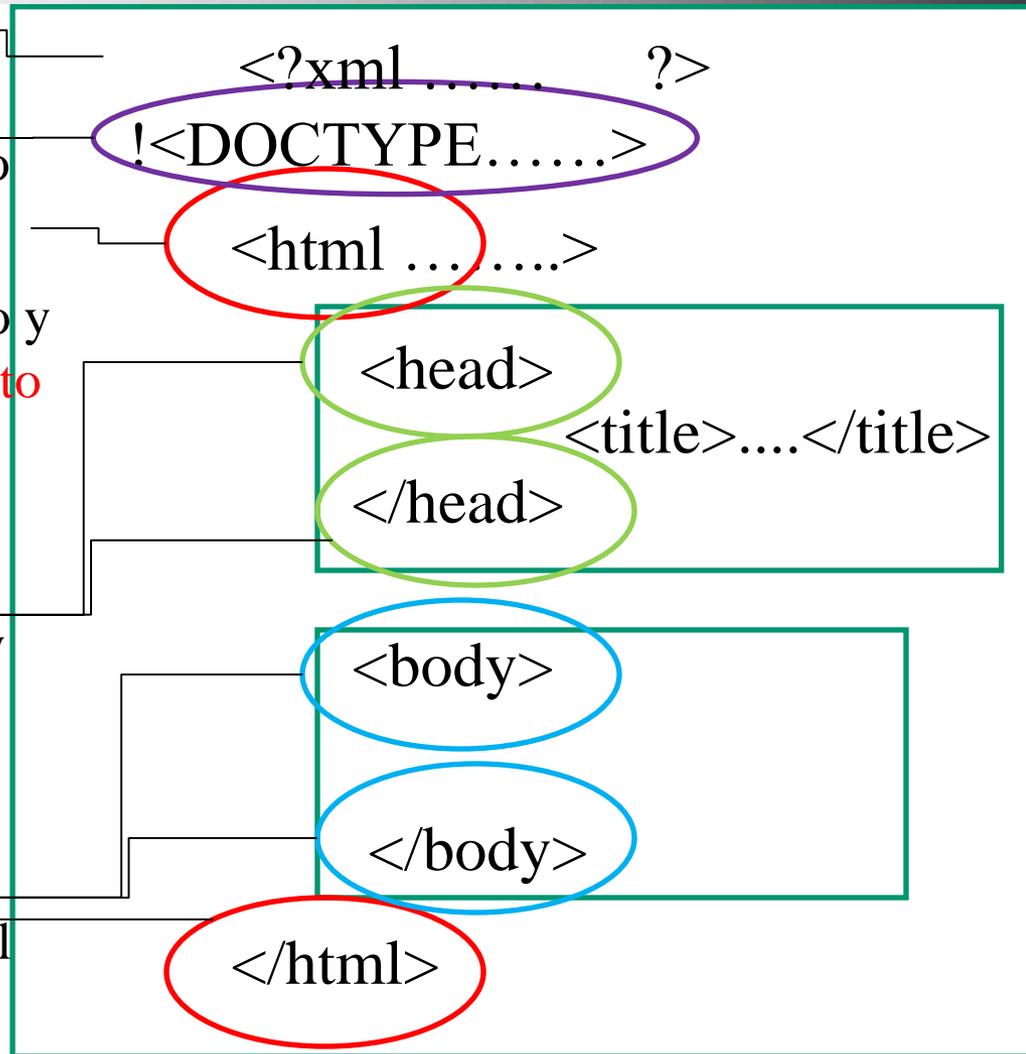
Versión XML y  
codificación

Declaración del tipo  
de documento

Etiquetas de inicio y  
final del documento

Etiquetas de inicio y  
final de la cabecera  
del documento

Etiquetas de inicio y final  
del cuerpo del  
documento



# Alternativa a XML, HTML: XHTML

- Existen tres versiones de XHTML para ayudar a la transición de HTML a XHTML. Para ello se necesita declarar un elemento llamado DOCTYPE que define el tipo de DTD a emplear en los documentos y que puede ser :

**1)Strict XHTML:** sin código desaprobado y siguiendo sintaxis estricta

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

**2)Transitional XHTML:** permite a los desarrolladores utilizar código desaprobado, pero requiere utilizar nueva sintaxis más estricta

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

**3)FrameSet XHTML:** para documentos donde se utilizan marcos

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

# Estructura general de un Documento HTML

## HTML

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

DTD Transicional

```
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
  <title> mi primera pagina de prueba HTML</title>
```

Espacio de nombres de XHTML 1.0

```
</head>
```

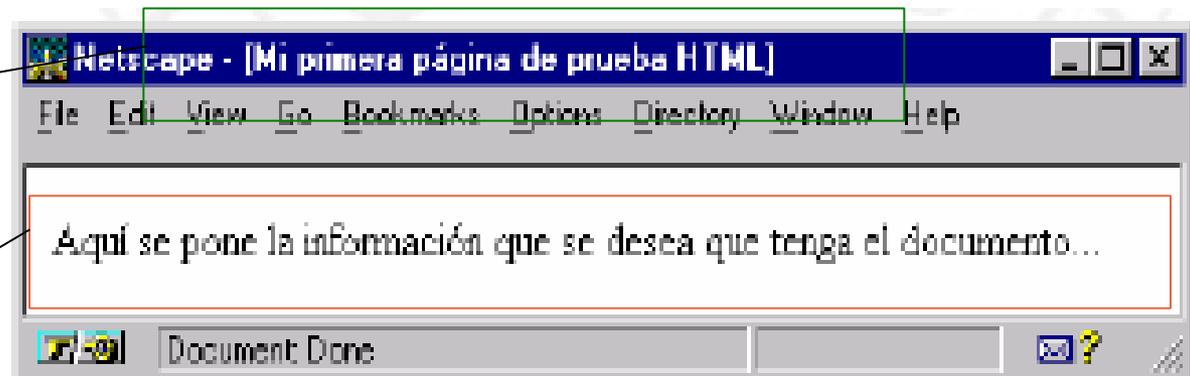
```
<body>
```

Aquí se pone la información que se desea que tenga el documento

```
</body>
```

```
</html>
```

Aquí sale lo que escribo en <title>



Aquí sale lo que escribo en el <body>

# EJERCICIO

---

Confecciona una página con las marcas mínimas que debe tener y en el cuerpo de la misma disponga tu nombre y apellido.

# Validar el documento

---

Los documentos XHTML deben:

1)Estar Bien Formado: debe seguir unas reglas de sintaxis

2)Ser validados: además de estar bien formados, cumplen las especificaciones de la DTD

El W3C tiene un validador para el estándar XHTML:

<http://validator.w3.org/>

# Etiquetas para estructurar el texto: títulos

`<h1>Este es el título 1</h1>`

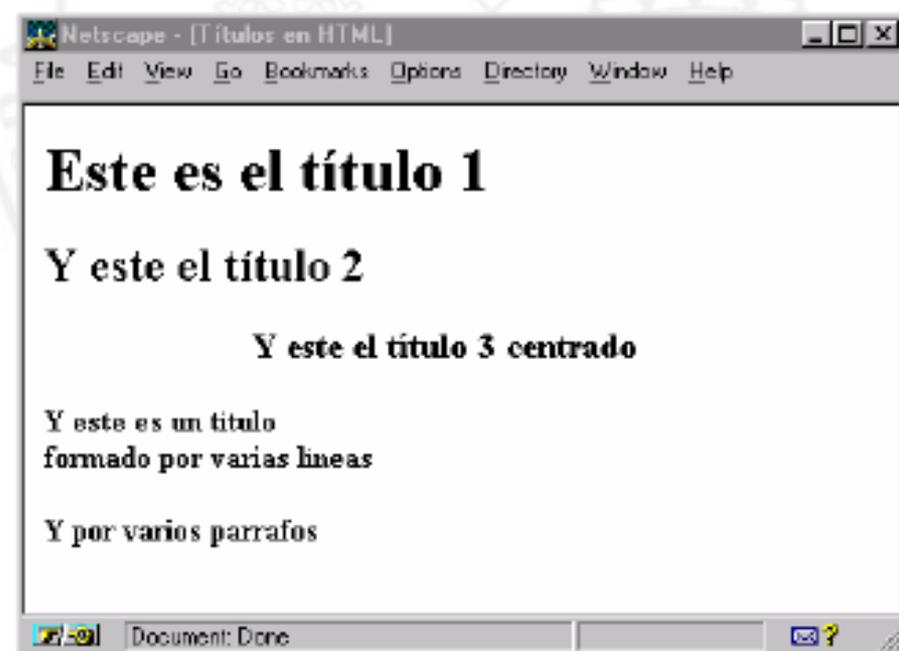
`<h2>Y este el título 2</h2>`

`<h3 align="center">Y este el título 3 centrado</h3>`

`<h4><p>Y este es un título<br/>formado por  
varias líneas</p>`

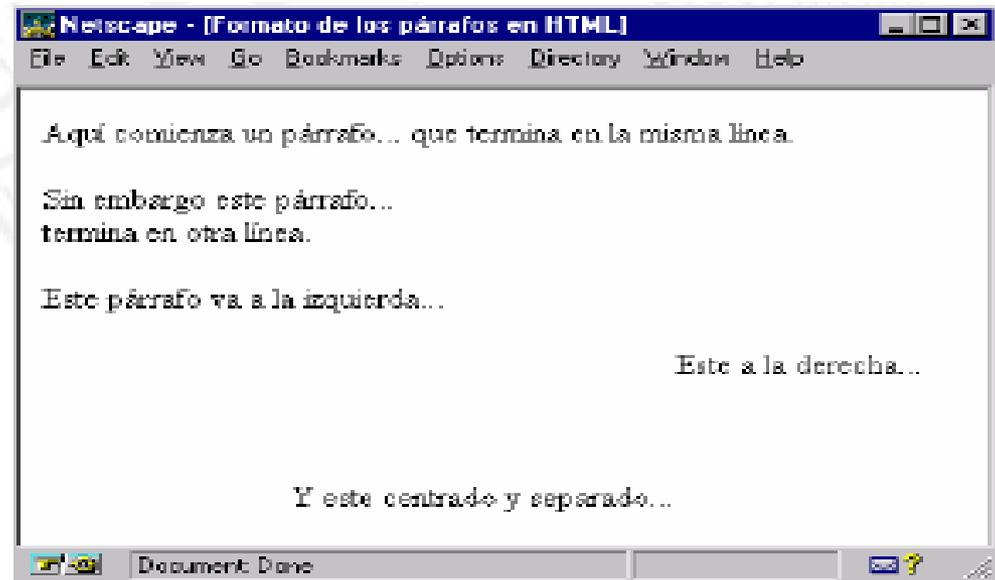
`<p>Y por varios párrafos</p>`

`</h4>`



# Etiquetas para estructurar el texto: Párrafos y saltos de línea

- `<p>` Aquí comienza un párrafo ... que termina en la misma línea `</p>`  
`<p>` Sin embargo este párrafo ...`<br/>` termina en otra línea`</p>`  
`<p align="left">` Este párrafo va a la izquierda `</p>`  
`<p align="right">` Este a la derecha `</p>`  
`<p align="center"><br/><br/>` Y este centrado y separado  
`</p>`



# EJERCICIO

---

Confecciona una página donde en un párrafo contenga tus nombres y apellidos en dos líneas diferente y en otro párrafo contenga tres títulos de tus películas favoritas, cada una en una línea diferente

# Etiquetas para estructurar el texto: Alineación de elementos

`<center>`

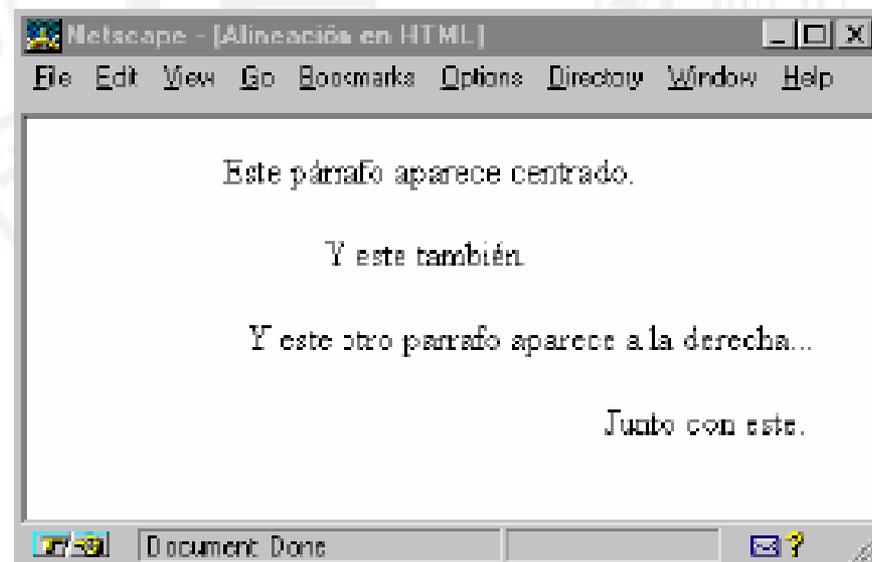
`<p>`Este párrafo aparece centrado.`</p>``<p>`Y este también.`</p>`

`</center>`

`<div align="right">`

`<p>`Y este otro párrafo aparece a la derecha...`</p>``<p>`Junto con este.`</p>`

`</div>`



# Etiquetas para estructurar el texto: líneas horizontales

`<hr/>` Esta etiqueta tiene los siguiente atributos:

Atributo	Función	Valor por Defecto
size=	Marca el grosor de la línea.	2
width=	Marca la anchura horizontal de la línea. Se puede expresar en pixels o en porcentaje	El ancho de la pantalla
align=	Especifica la alineación de la línea, puede ser left, right o center	Si no se especifica las líneas están centradas
noshade=	Muestra una línea sólida sin apariencia tridimensional Valor atributo="noshade"	Línea con apariencia tridimensional

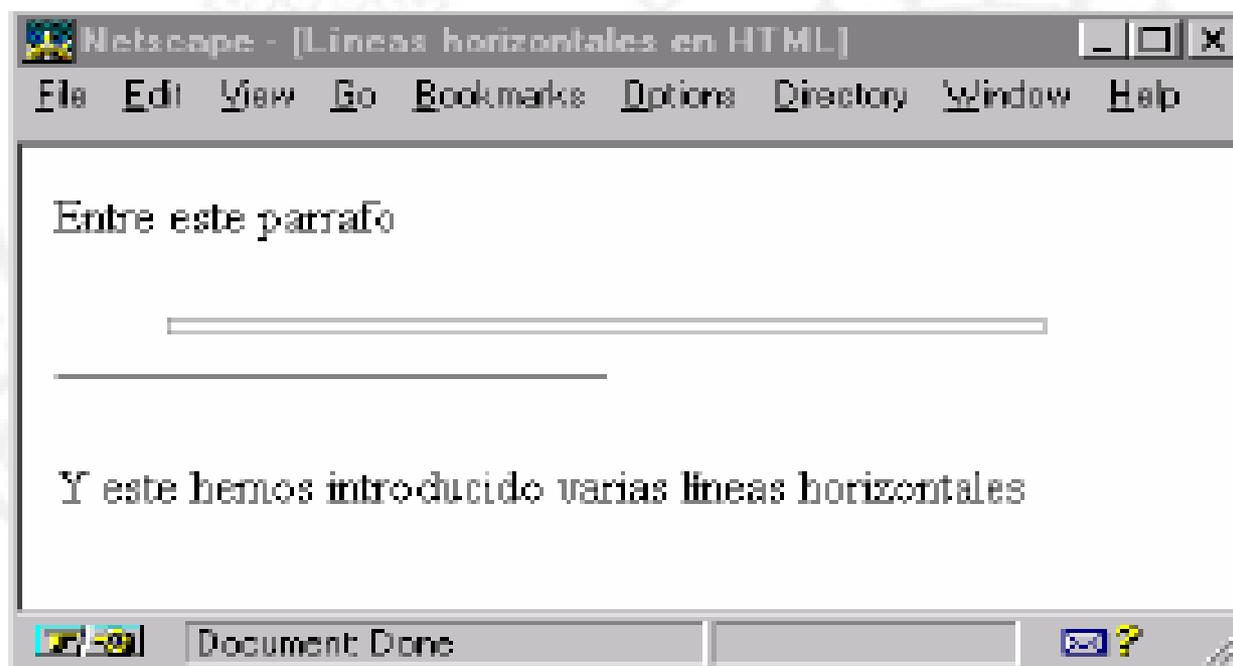
# Etiquetas para estructura el texto: líneas horizontales

`<p>Entre este parrafo</p>`

`<hr size="5" width="80%" />`

`<hr width="50%" align="left" noshade="noshade" />`

`<p>Y este hemos introducido varias líneas horizontales</p>`



# Etiquetas para estructurar el texto: sangrado de bloques

`<p>`Este párrafo no esta sangrado`</p>`

`<blockquote>`

`<p>`Pero este párrafo si`</p>`

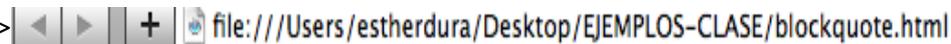
`<blockquote>`

`<hr/>`

`<p>`Y este y la línea anterior más aún`</p>`

`</blockquote>`

`</blockquote>`



Este párrafo no esta sangrado

<http://es.youtube.com/>

Pero este párrafo si

---

Y este y la línea anterior más aún

# Etiquetas para estructurar el texto: texto preformateado

<pre>

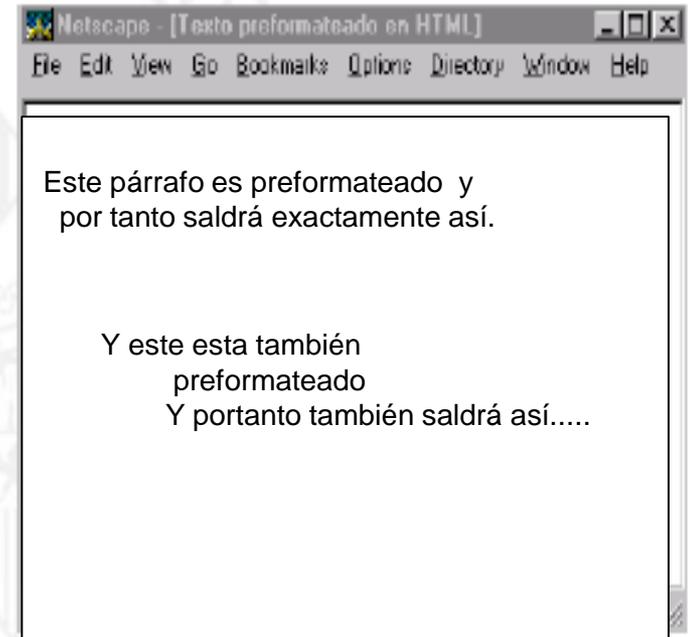
Este párrafo es preformateado y  
por tanto saldrá exactamente así.

</pre>

<pre>

Y este esta también  
preformateado  
Y portanto también saldrá así...

</pre>



# Etiquetas para estructurar el texto: Direcciones

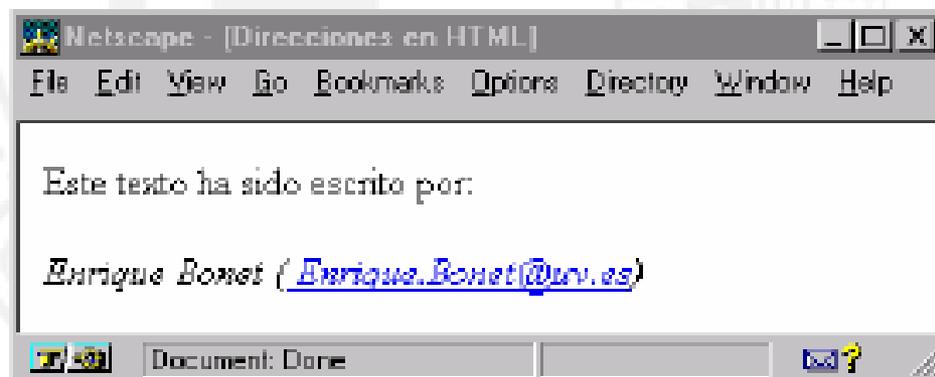
<p>Este texto ha sido escrito por:</p>

<address>

Enrique Bonet (<a href="mailto: [Enrique.Bonet@uv.es](mailto:Enrique.Bonet@uv.es)">

Enrique.Bonet@uv.es<a/>)

</address>



# Etiquetas de formato de texto

---

Dos tipos:

- **Etiquetas de formato lógico.** Clasifican el texto en una serie de clases predefinidas.
- **Etiquetas de formato físico o presentación.** Fijan un estilo en la presentación.

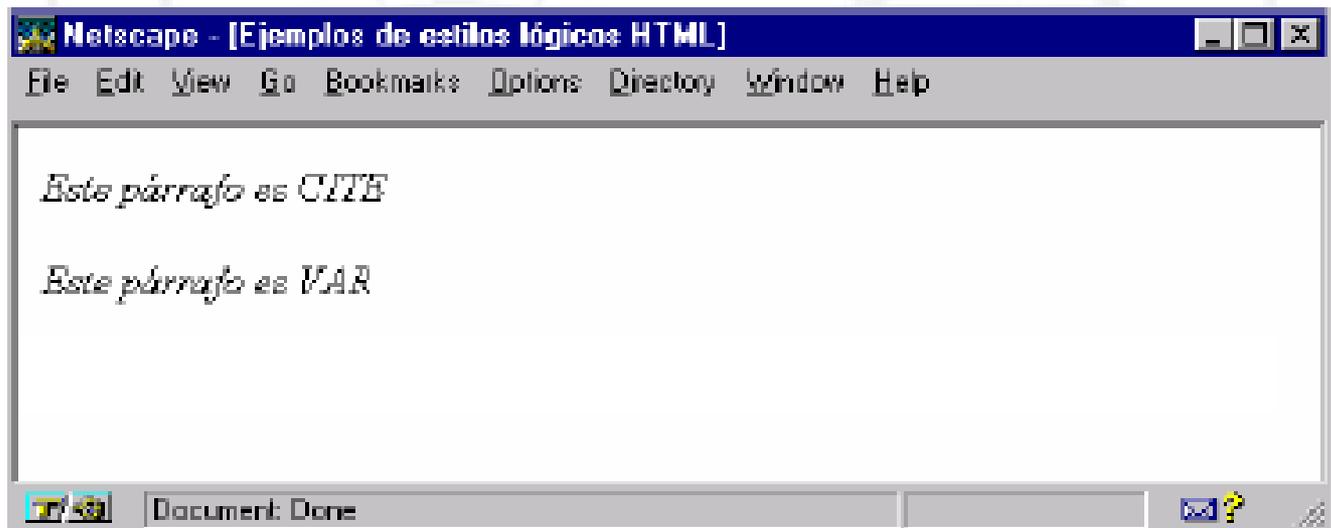
## Etiquetas de formato del texto: formato lógico

Estilo Lógico	Explicación
<code>&lt;cite&gt;&lt;/cite&gt;</code>	Cita literal de un texto
<code>&lt;code&gt;&lt;/code&gt;</code>	Presentación de código fuente
<code>&lt;em&gt;&lt;/em&gt;</code>	Aplica énfasis al texto
<code>&lt;kdb&gt;&lt;/kdb&gt;</code>	Secuencias de órdenes o comandos que debe introducir el usuario
<code>&lt;samp&gt;&lt;/samp&gt;</code>	Insertar una secuencia de caracteres literales
<code>&lt;strong&gt;&lt;/strong&gt;</code>	Aplica énfasis al texto
<code>&lt;var&gt;&lt;/var&gt;</code>	Define el nombre de una variable
<code>&lt;del&gt;&lt;/del&gt;</code>	Permite marcar un elemento de un texto como eliminado
<code>&lt;ins&gt;&lt;/ins&gt;</code>	Permite marcar la inserción de un texto en un documento

# Etiquetas de formato de texto: formato lógico

`<p><cite>Este párrafo es CITE</cite></p>`

`<p><var>Este párrafo es VAR</var></p>`



# Etiquetas de formato de texto: formato físico

ESTILO FÍSICO	EXPLICACIÓN
<code>&lt;b&gt;&lt;/b&gt;</code>	Muestra el texto en negrita
<code>&lt;i&gt;&lt;/i&gt;</code>	Muestra el texto en itálica
<code>&lt;tt&gt;&lt;/tt&gt;</code>	Muestra el texto como una máquina de escribir
<code>&lt;u&gt;&lt;/u&gt;</code>	Subraya el texto marcado
<code>&lt;sub&gt;&lt;/sub&gt;</code>	Muestra el texto como un subíndice
<code>&lt;sup&gt;&lt;/sup&gt;</code>	Muestra el texto como un superíndice
<code>&lt;big&gt;&lt;/big&gt;</code>	Muestra el texto al de un tamaño superior al por defecto
<code>&lt;small&gt;&lt;/small&gt;</code>	Muestra el texto al de un tamaño inferior al por defecto

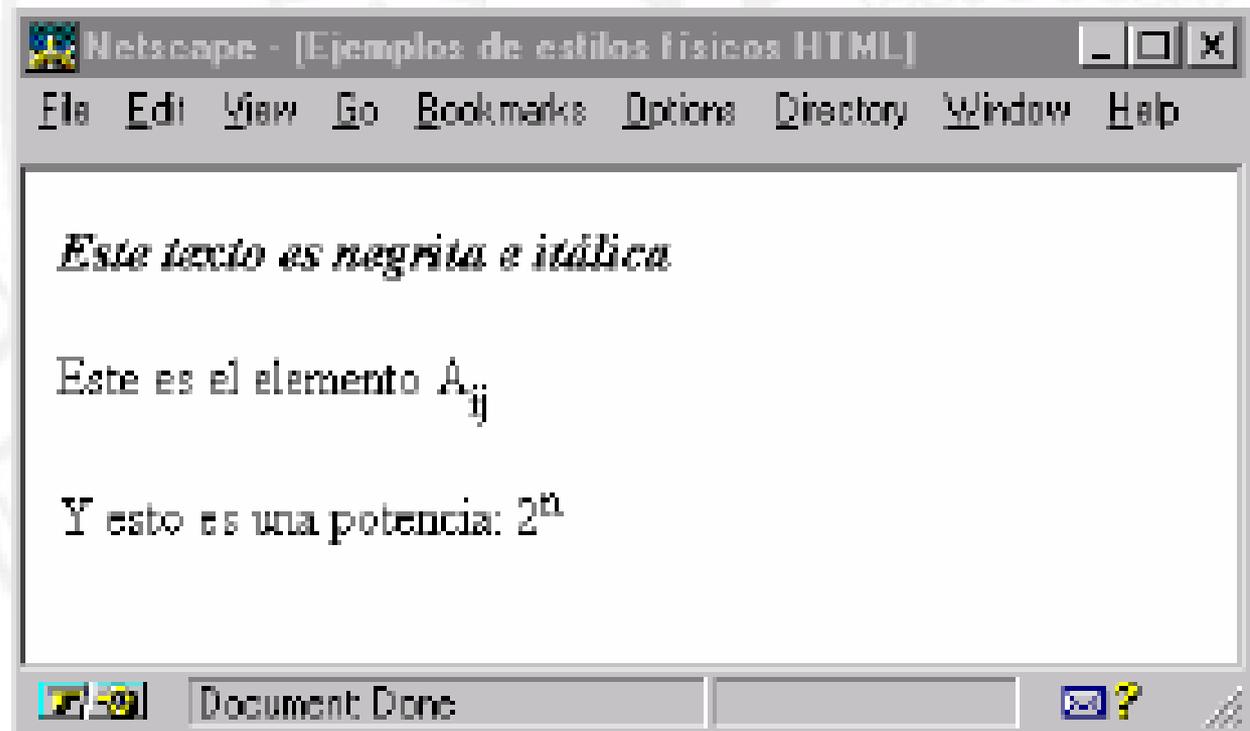
# Etiquetas de Formato del texto

## Formato Físico

`<p><b><i>` Este texto es negrita e itálica `</i></b></p>`

`<p>`Este es el elemento A`<sub>ij </sub></p>`

`<p>` Y esto es una potencia: 2 `<sup>n</sup></p>`



# Aspectos de los caracteres

- La etiqueta que permite fijar el aspecto de los caracteres es **<font>**
- Podemos cambiar los tamaños de letra con el atributo **size**, que puede tomar valores entre 1 y 7. El valor por defecto del texto es 3.
- `<font size="3">A</font><font size="4">A</font>`
- `<font size="5">A</font> <font size="6">A</font>`
- `<font size="5">A</font><font size="4">A</font>`
- `<font size="3">A</font>`
- Dará como resultado: **AAAAAAA**

# Aspectos de los caracteres

---

- Es posible utilizar en la etiqueta `<font>` el atributo *face* que permite elegir tipos de letra.
- Este atributo permite forzar el tipo de letra que el diseñador de la página quiere que vea el cliente
- El navegador mostrará el tipo de letra sólo si dicho tipo está disponible.
- Por ejemplo:
- `<font face="times newroman">Prueba con TIMES NEW ROMAN</font>`

## Resultado:

Prueba con TIMES NEW ROMAN

# Aspectos de los caracteres

---

- Si en la máquina cliente no está instalada una determinada fuente, ésta no se verá y en su lugar aparecerá la fuente por defecto del visualizador
- En la etiqueta `<font>` puede usarse el atributo `color` para definir el color de la fuente.

# EJERCICIO

Escribe un documento XHTML que permita visualizar por pantalla la siguiente información:.....



## SOBRE XHTML

Este curso ~~2010~~ **2011** voy a ser un hacha en XHTML.

No va a haber *código XHTML* que se me resista porque trabajando duro en casa, en prácticas, atendiendo y participando en clase voy a aprobar esta asignatura  
**Firmado:**

El hacha de 1º en XHTML

# Caracteres especiales

Carácter	Código	Carácter	Código	Carácter	Código	Carácter	Código
<	&lt;	>	&gt;	&	&amp;	#	&#35;
“	&quot;	á	&aacute;	é	&eacute;	í	&iacute;
ó	&oacute;	ú	&uacute;	Á	&Aacute;	ñ	&ntilde;
Ñ	&Ntilde;	®	&reg;	@	&copy;	Espacio no divisible.	&nbsp;

# Listas

---

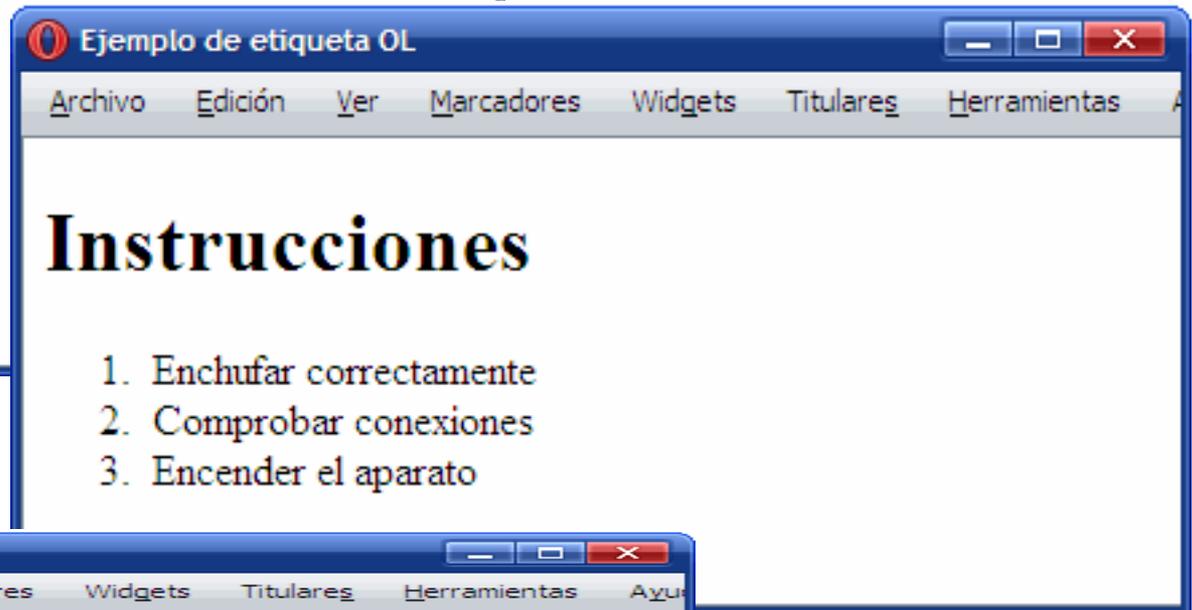
Tres tipos en XHTML:

- **No numeradas.** Aparecen como una lista con puntos.
- **Ordenadas.** Aparecen como una secuencia de números o letra en vez de una lista de puntos.
- **Definición,** permiten especificar un término y su definición

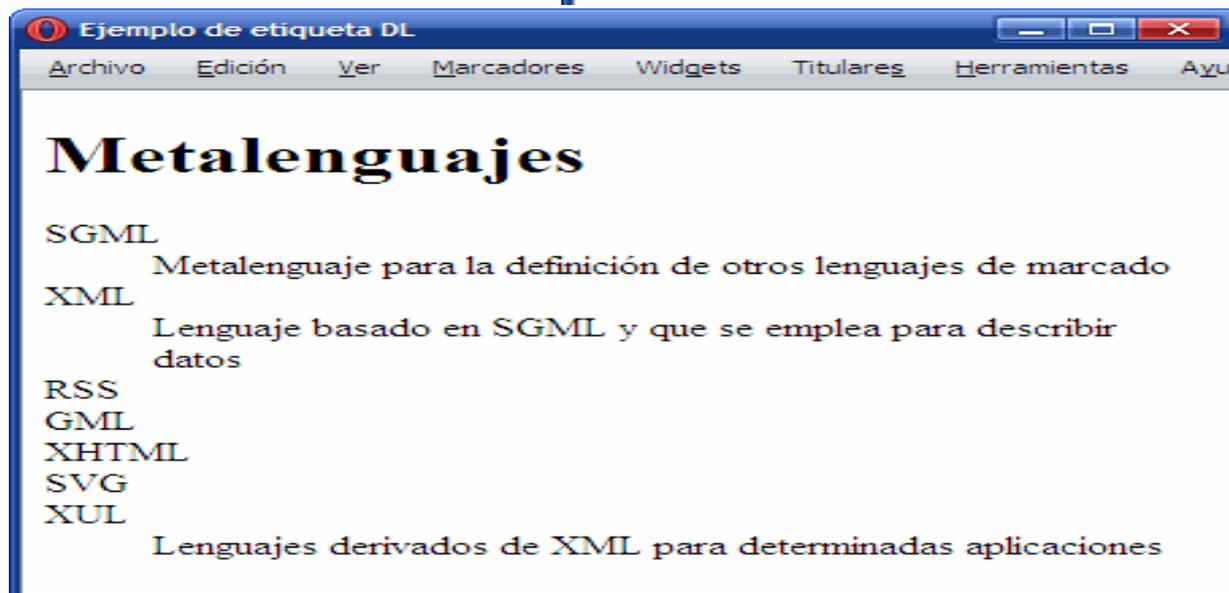
# Listas: ejemplos



Lista: no numerada



Lista: Ordenada

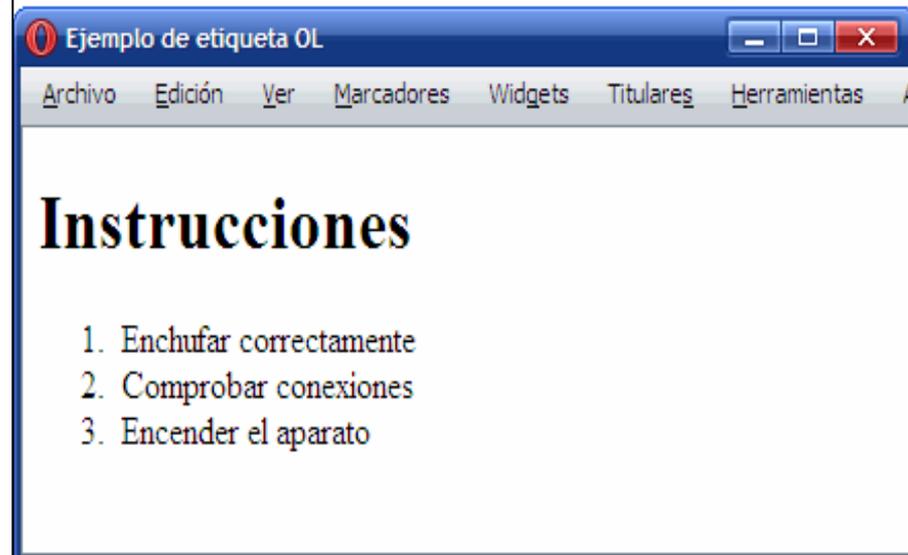


Lista: Definiciones

# Listas ordenadas

- Una lista ordenada se encuentra dentro de la etiqueta pareada `<ol>...</ol>`
- Cada elemento de la lista se debe anidar dentro del elemento `<ol>...</ol>` y situarse dentro de las etiquetas `<li>..</li>`

```
<body>
<h1>Instrucciones</h1>
  <ol>
    <li> Enchufar correctamente</li>
    <li> Comprobar conexiones</li>
    <li> Encender el aparato </li>
  </ol>
</body>
```



# Listas ordenadas o marcadas

- La etiqueta pareada <ol> puede utilizar el atributo **type** para seleccionar números, letras o números romanos en listas ordenadas

Valor para atributo <b>type</b>	Descripción	Ejemplos
1	Números arábigos (por defecto)	1,2,3,4,5
A	Letras mayúsculas	A,B,C,D,E
a	Letras minúsculas	a,b,c,d,e
I	Números romanos mayúsculas	I, II, III, IV, V
i	Números romanos minúsculas	i , ii, iii, iv, v

- Utilizar el atributo **start** para cambiar el número de inicio en las listas ordenadas

# Listas ordenadas

- Ejemplo lista ordenada utilizando atributos `type` y `start`

```
<body>
<h1>Instrucciones</h1>
  <ol type="I" start="2">
    <li> Enchufar correctamente</li>
    <li> Comprobar conexiones</li>
    <li> Encender el aparato </li>
  </ol>
</body>
```



## Instrucciones

- II. Enchufar correctamente
- III. Comprobar conexiones
- IV. Encender el aparato

# Listas No numeradas o marcadas

- Una lista se introduce mediante los la etiqueta pareada `<ul>...</ul>`
- Cada elemento de la lista se debe declarar dentro de la etiqueta `<ul>...</ul>` y situarse entre las etiquetas `<li>..</li>`

```
<body>
<h1>Menú</h1>
  <ul>
    <li>Inicio</li>
      <li>Noticias</li>
        <li>Artículos </li>
          <li>Contacto</li>
    </ul>
</body>
```



# Listas: de definición o descripción

---

- Este tipo de listas se utilizan para presentar por ejemplo glosarios, definiciones...etc
- Las lista de definiciones se construye inicialmente utilizando la etiqueta pareada:  
`<dl>.....</dl>`
- Las etiquetas `<dt>..</dt>` definen respectivamente el término que quieres definir o describir y la etiqueta `<dd>..</dd>` se utiliza para la descripción o definición de cada elemento de la lista.

# Listas: de definición o descripción

```
<body>
<h1>Metalenguajes</h1>
<dl>
  <dt>SGML</dt>
  <dd>Metalenguaje para la definición de otros lenguajes de marcado</dd>
  <dt>XML</dt>
  <dd>Lenguaje basado en SGML y que se emplea para describir datos</dd>
  <dt>RSS</dt>
  <dt>GML</dt>
  <dt>XHTML</dt>
  <dd>Lenguajes derivados de XML para determinadas aplicaciones</dd>
</dl>
</body>
```



## Metalenguajes

SGML

Metalenguaje para la definición de otros lenguajes de marcado

XML

Lenguaje basado en SGML y que se emplea para describir datos

RSS

GML

XHTML

Lenguajes derivados de XML para determinadas aplicaciones

# Listas anidadas: No ordenadas y ordenadas

```
<body>
```

```
<p><em>Elige una opción</em></p>
```

```
<ol start="1" type="A">
```

```
<li>Comprar un producto</li>
```

```
<li>Ver el estado de una compra
```

```
<ul>
```

```
<li>De este mes</li>
```

```
<li>De otros meses</li>
```

```
</ul>
```

```
</li>
```

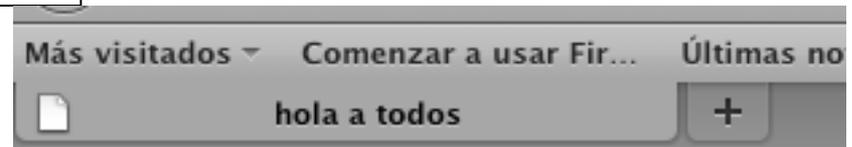
```
<li>Salir</li>
```

```
</ol>
```

```
</body>
```

```
</html>
```

Lista anidada



*Elige una opción*

A. Comprar un producto

B. Ver el estado de una compra

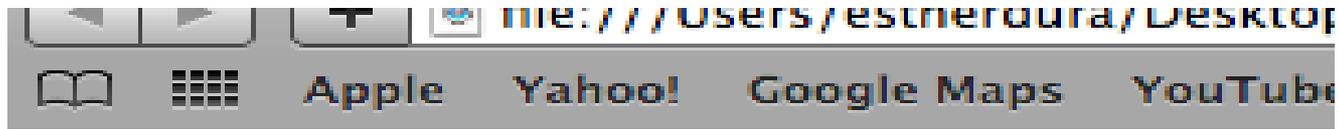
o De este mes

o De otros meses

C. Salir

# Ejercicio:

Escribe el código XHTML de tal forma que se visualice en el Navegador la siguiente lista anidada



- PHP
  1. XHTML para principiantes
  2. XHTML nivel intermedio
  3. XHTML expertos
- JavaScript
  1. Nivel 1
  2. Nivel 2
- Lenguaje C++
  1. Nivel intermedio
  2. Club Builder

# Tablas

The diagram illustrates the components of a table. The table is titled "Cursos de diseño gráfico". The first row is the header row, with cells labeled "Nombre", "Horas", "Plazas", and "Horario". The first column is the header column, with cells labeled "Introducción a XHTML", "CSS avanzado", "Taller de usabilidad", and "Introducción a AJAX". The table is annotated with arrows and labels: "título de tabla" points to the title, "cabecera de columna" points to the header row, "cabecera de fila" points to the header column, "fila" points to a data row, and "columna" points to a data column.

Nombre	Horas	Plazas	Horario
Introducción a XHTML	20	20	09:00 – 13:00
CSS avanzado	40	15	16:00 – 20:00
Taller de usabilidad	40	10	16:00 – 20:00
Introducción a AJAX	60	20	08:30 – 12:30

# Tablas

---

- Las tablas se incluyen con la con la etiqueta pareada:

**<table> </table>**

**Importante:** la tabla se va escribiendo fila a fila

- Las filas de la tabla se van incluyendo con la etiqueta pareada:

**<tr> </tr>**

- Estas filas incluyen una o varias celdas que se van especificando con la etiqueta pareada:

**<td> </td>**

- Celdas** de encabezado de columnas representado por la etiqueta pareadas

**<th> </th>**

- Para añadir título a la tabla utiliza etiqueta pareada

**<caption> </caption>**

# Tablas

`<table>`

`<caption>Gastos del mes</caption>`

`<tr><th>Concepto</th><th>Importe</th></tr>`

`<tr><td>Alimentacion</td><td>20.000</td></tr>`

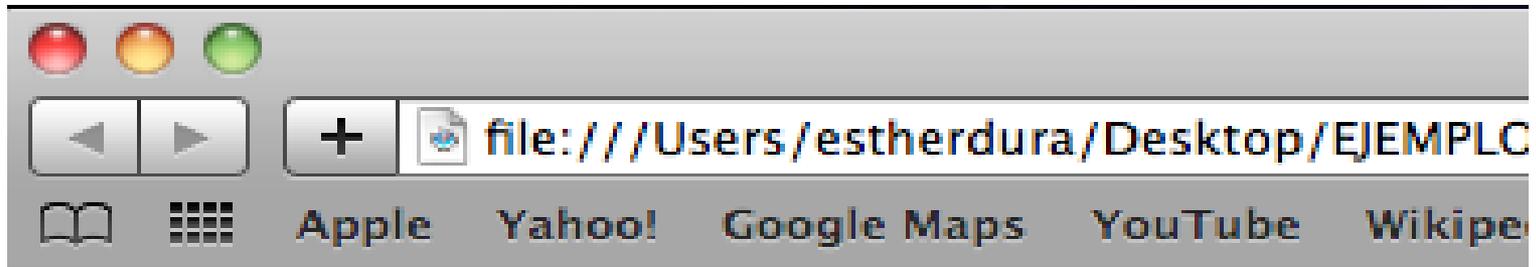
`<tr><td>Vestido</td><td>50.000</td></tr>`

`</table>`



# Ejercicio:

Escribe el código XHTML de tal forma que se visualice en Navegador la siguiente tabla



<b>Curso</b>	<b>Horas</b>	<b>Horario</b>
CSS	10	16:00-20:00
XHTML	25	18:00-22:00
XML	20	14:00-16:00

# Tablas: atributos

```
<table border="2">
  <caption align="bottom">Gastos del mes</caption>
  <tr align="center">
    <th>Concepto</th><th>Importe</th><th>Fecha</th>
  </tr>
  <tr align="right">
    <td align="left">Alimentación</td><td>20.000</td><td>1/1/1990</td>
  </tr>
  <tr align="right">
    <td align="left">Vestido</td><td>50.000</td><td>10/1/1990</td>
  </tr>
</table>
```



va

1 windows

Buscar Sidewiki

simple document

Capturar una pantalla del orde

Concepto	Importe	Fecha
Alimentación	20.000	1/1/1990
Vestido	50.000	10/1/1990

Gastos del mes

# Tablas: atributos de modificación

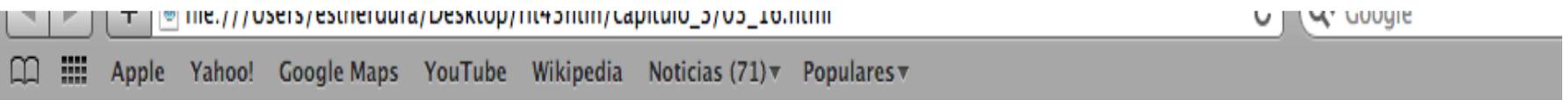
Etiqueta	Atributos	Explicación
<code>&lt;table&gt;</code>	<code>border=</code>	Fija el ancho del borde exterior de la tabla (en pixels) y cada celda individual
	<code>cellpadding=</code>	Fija la distancia (en pixels, o % (del ancho de la tabla)) entre el borde de la celda y su contenido
	<code>cellspacing=</code>	Se usa para crear un espacio (pixels o % (del ancho de la tabla)) entre los bordes de cada celda
	<code>width=</code>	Controla la anchura horizontal de la tabla. Se puede especificar en pixels o % (del espacio de la ventana del explorador)
	<code>align=</code>	Control de la alineación de la tabla en la página. Puede tomar los valores <code>left</code> , <code>right</code> y <code>center</code>
<code>&lt;caption&gt;</code>	<code>align=</code>	Especifica la posición puede ser <code>top</code> , <code>bottom</code> (arriba o abajo de la tabla)
<code>&lt;tr&gt;</code>	<code>align=</code>	Especifica la posición del contenido de todas las celdas en la fila. Puede tomar los valores: <code>left</code> , <code>right</code> , <code>center</code>
	<code>valing=</code>	Especifica la alineación vertical del texto en la fila, puede ser <code>top</code> , <code>bottom</code> , <code>middle</code> o <code>baseline</code>

# Tablas: atributos de modificación

Etiqueta	Atributos	Explicación
<code>&lt;td&gt;</code>	<code>align=</code>	Especifica la alineación horizontal del texto en la celda, puede ser <code>left</code> , <code>right</code> o <code>center</code>
	<code>valign=</code>	Especifica la alineación vertical del texto en la celda, puede ser <code>top</code> , <code>bottom</code> , <code>middle</code> o <code>baseline</code>
	<code>colspan=</code>	Indica el número de columnas que ocupará esta celda
	<code>rowspan=</code>	Indica el número de filas que ocupará esta celda
	<code>nowrap=</code>	Obliga al cliente Web a no romper las líneas de texto que contenga la celda. Valor que toma es: <code>nowrap</code>
	<code>width=</code>	Determina el ancho de la celda. Se puede especificar como un ancho en pixels o en porcentaje
	<code>height=</code>	Permite especificar la altura de una celda en pixels
<code>&lt;th&gt;</code>	Igual que <code>&lt;td&gt;</code>	Igual que los atributos <code>&lt;td&gt;</code>

# Tablas: atributo rowspan

```
<table width="80%" border="1">
  <tr>
    <td width="33%" rowspan="2">1</td>
    <td width="33%">2</td>
    <td width="34%">3</td>
  </tr>
  <tr>
    <td width="33%">2</td>
    <td width="34%">3</td>
  </tr>
</table>
```



1	2	3
	2	3

# Tablas: atributo colspan

```
<table width="80%"border="1">
<tr>
  <td colspan="3" align="center">Título</td>
</tr>

<tr>
  <td width="33%">1</td>
  <td width="33%">2</td>
  <td width="34%">3</td>
</tr>
</table>
```

Apple Yahoo! Google Maps YouTube Wikipedia Noticias (71) Populares

Título		
1	2	3

# Ejercicio:

Escribe el código XHTML de tal forma que se visualice en el Navegador la siguiente tabla



<b>Nombre y URL</b>	
<b>Periodicos de España</b>	El pais <a href="http://www.elpais.es">www.elpais.es</a>
	El mundo <a href="http://www.elmundo.es">www.elmundo.es</a>
	La vanguardia <a href="http://www.lavanguardia.es">www.lavanguardia.es</a>

# Enlaces

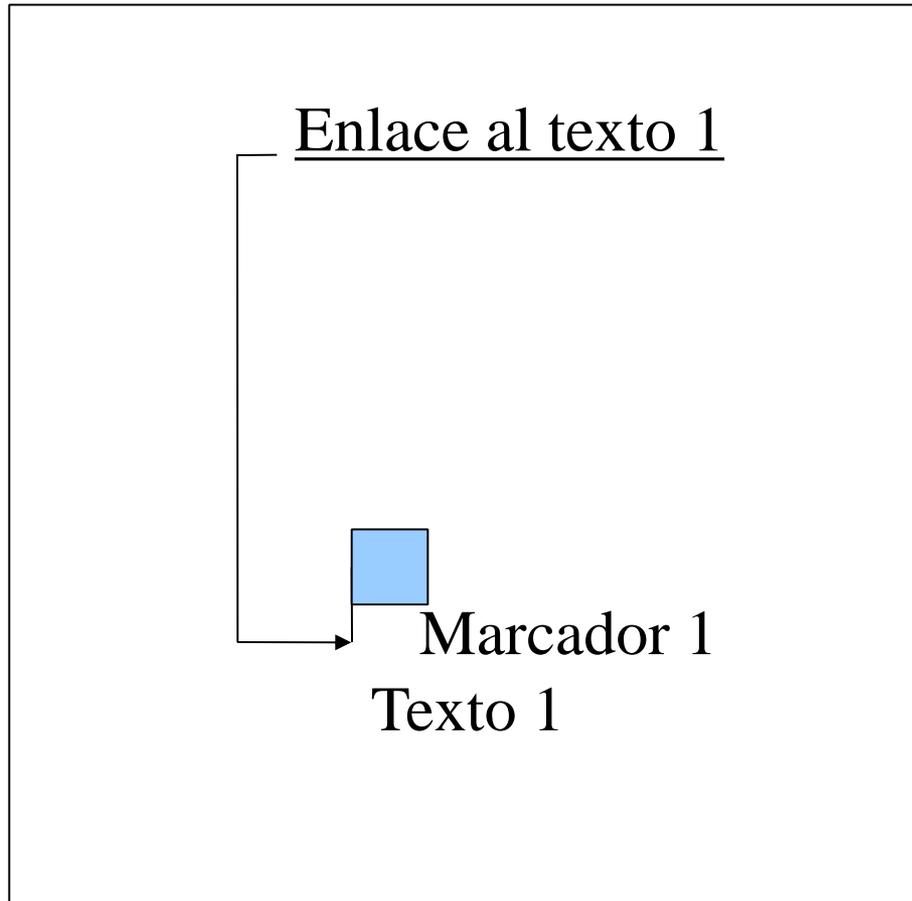
Etiquetas pareada `<a></a>`

Dos tipos:

- **Enlaces de destino:** marcan un sitio en el documento al cual se pueden ir
- **Enlaces de origen:** indican un URL o enlace de destino

Atributo	Explicación
<code>name=".."</code>	Define un <b>enlace de destino</b> . En la zona entrecomillada se debe incluir un nombre, único dentro de este documento, que servirá de referencia del bloque seleccionado.
<code>href=".."</code>	Define un <b>enlace de origen</b> . En la zona entrecomillada se debe incluir el URL que el cliente recogerá ante la activación del enlace. La URL podrá incluir un enlace de destino.

# Enlaces internos



Pagina1.html

## Pasos a seguir:

- 1) Declaración del marcador (enlace de destino)
- 2) Enlace hacia el marcador (enlace de origen)

# Enlaces internos

## Pasos a seguir:

1) Declaración del marcador-->con esto se define un lugar en la página, lo que permite establecer un enlace hacia él. Se usa el atributo **name**

```
<a name="nombre del marcador"></a>
```

2) Enlace hacia el marcador-->después de definir el marcador se define el enlace hacia este. Se usa el atributo **href** y **#**

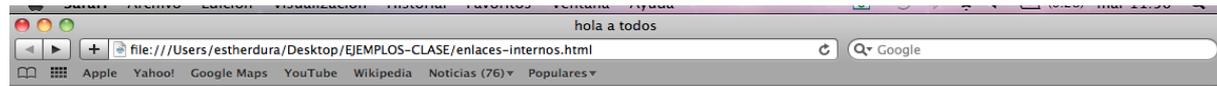
```
<a href="#nombre del marcador"></a>
```

# Enlaces interno

```
<body>  
<br/><br/>  
<a href="#cap1">capitulo 1</a><br/>  
<a href="#cap2">capitulo 2</a><br/>  
<br/><br/><br/><br/><br/>  
<a name="cap1"/> esto forma parte del capitulo 1</a>  
y puede tener varias paginas
```

```
<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="cap2"/> esto forma parte del capitulo 2</a>  
</body>
```

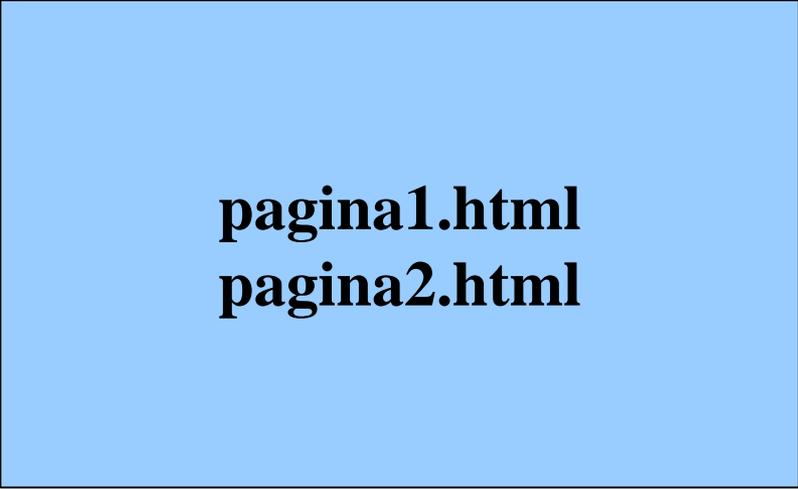


[capitulo 1](#)  
[capitulo 2](#)

esto forma parte del capitulo 1 y puede tener varias paginas

# Enlaces locales

Son enlaces hacia otra páginas que forman un sitio web. Para la creación de un sitio web es muy recomendable almacenar en una carpeta todos los archivos (pagina1.html, página2.html..etc) relativos al sitio



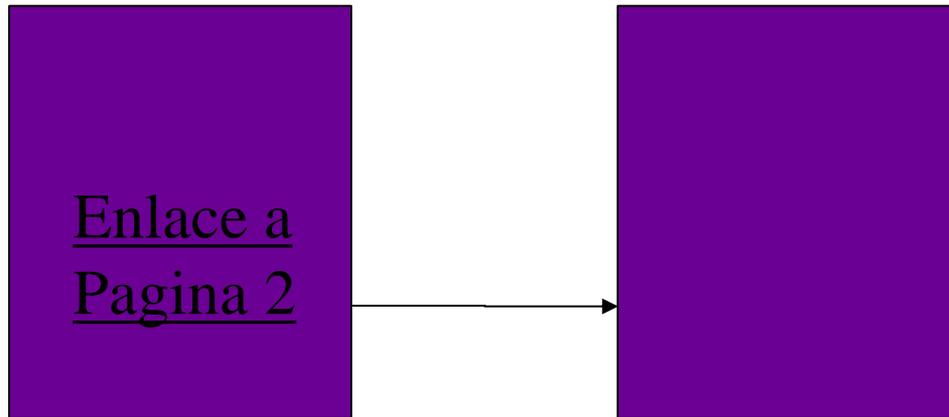
**pagina1.html**  
**pagina2.html**

**carpeta1**

**pagina1.html y pagina2.html**  
Pertenece al mismo sitio web y están almacenados en la misma carpeta llamada **carpeta1**

# Enlaces locales

Imaginemos que queremos hacer un enlace de la pagina1.html hacia la pagina2.html



pagina1.html

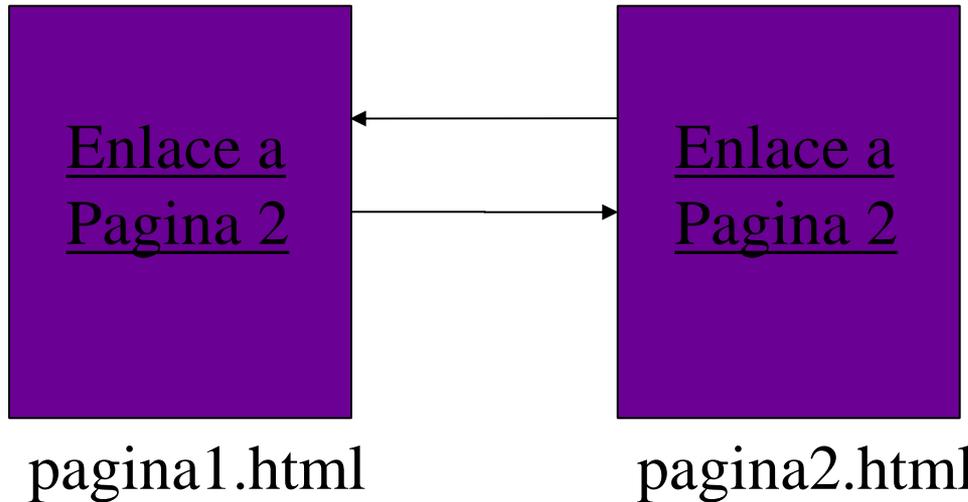
pagina2.html

```
<a href="pagina2.html" >Enlace a Pagina2</a>
```

Codigo xhtml de la pagina 1

# Enlaces locales

Imaginemos que queremos hacer un enlace de pagina1.html hacia pagina2.html y de la pagina2.html hacia la pagina1.html



```
<a href="pagina2.html" >Enlace a Pagina 2</a>
```

```
<a href="pagina1.html" >Enlace a Pagina1</a>
```

Codigo  
xhtml  
pagina 1

Codigo  
xhtml  
pagina 2

# Enlaces remotos

Un enlace puede hacer referencia a otra páginas situadas en otro dirección en la web (URL)

Por ejemplo podemos hacer referencia a la página principal de google

`<a href="http:www.google.es">Enlace hacia pagina de google</a>`  
a la página principal de la universidad

`<a href="http://www.uv.es">Enlace hacia pagina de la Universidad de Valencia</a>`

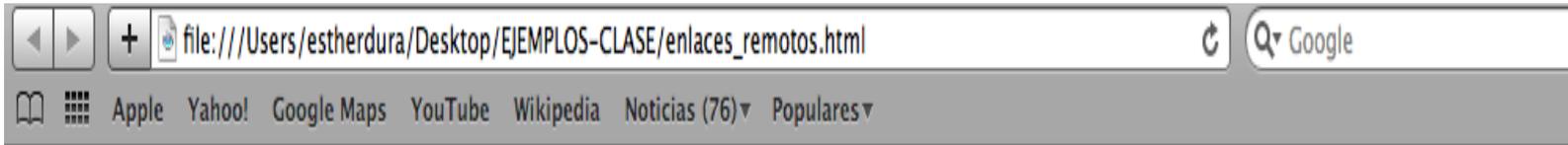
# Enlaces remotos

<body>

Para acceder a la pagina de la Universitat de valencia pincha en el siguiente enlace:<a href="http://www.uv.es">Web de la UV</a>  
<br/>

Para acceder a la pagina de Google pincha en el siguiente enlace:<a href="http://www.google.es">Web de google</a>

</body>



Para acceder a la pagina de la Universitat de valencia pincha en el siguiente enlace:Web la UV

Para acceder a la pagina de GOOGLE pincha en el siguiente enlace:Web de google

# Ejercicio

---

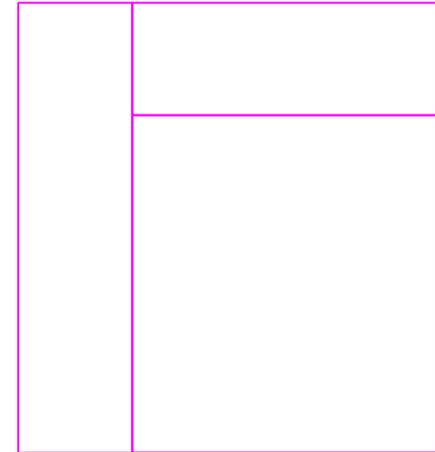
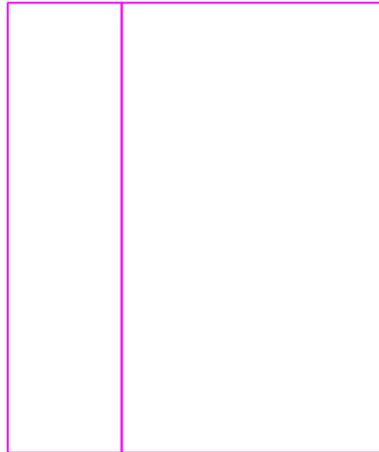
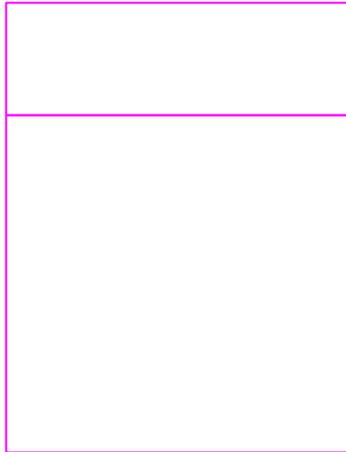
Realiza el ejercicio que se indica:

(ejemplo\_enlaces.html)

# Marcos

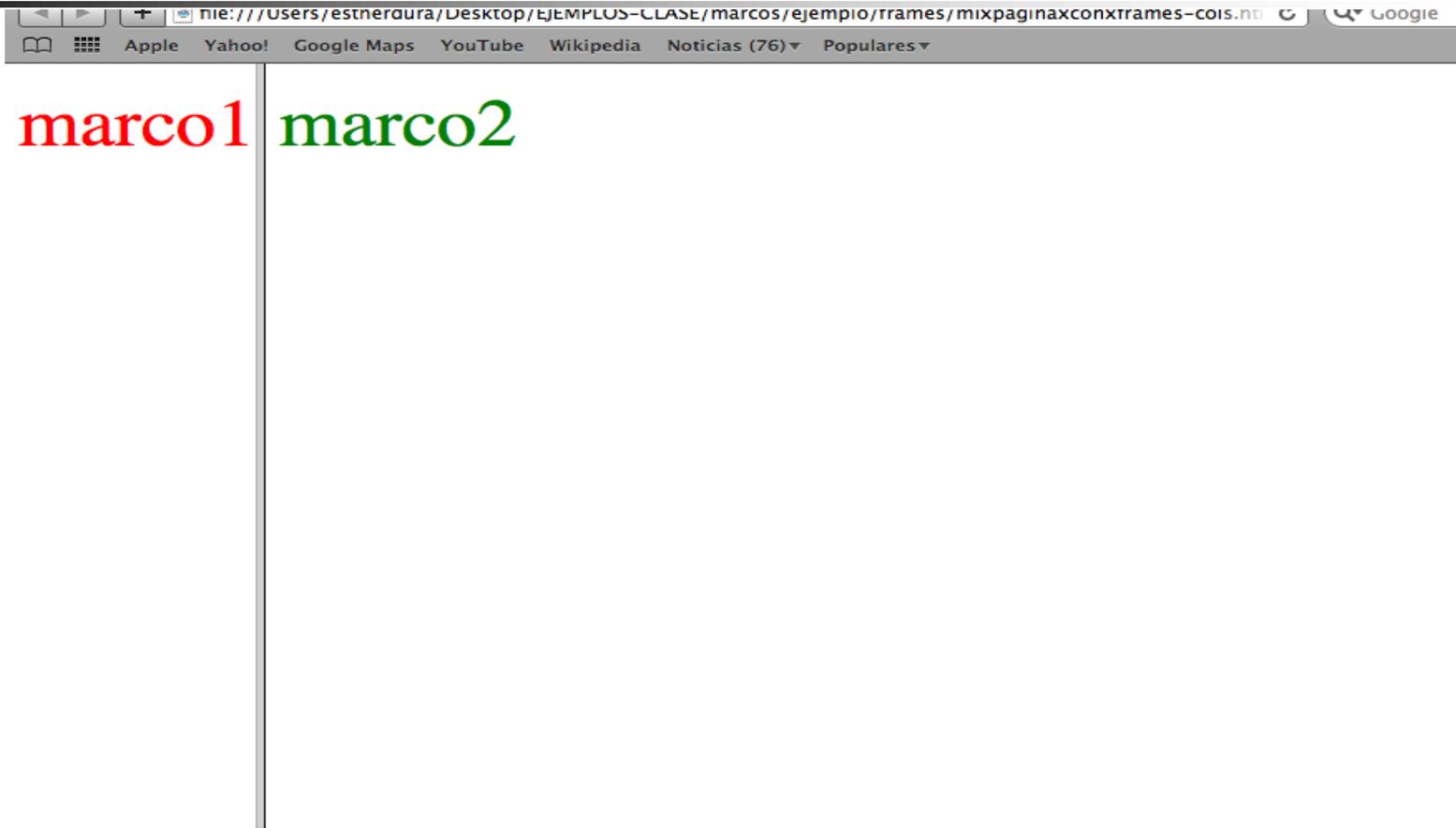
- Una de sus características más importantes es que pulsando un enlace situado en un frame, se puede cargar en otro frame una página determinada.
- Se utiliza principalmente para tener un frame estrecho en la parte lateral ( o superior) con un índice del contenido en forma de diferentes enlaces, que al ser pulsado cargan en la ventana principal las distintas páginas

# Marcos



- Frames (Marcos o cuadros) es un procedimiento del lenguaje X HTML para dividir la pantalla en diferentes zonas o ventanas, que pueden *actuar independientemente unas de otras*, como si se trataran de **páginas diferentes**.

# Marcos



Ejemplo de pagina con 2 zonas o marcos distribuidos verticalmente-->2 columnas

# Marcos



marco1

---

marco2

Ejemplo de pagina con 2 zonas o marcos distribuidos horizontalmente-->2 filas

# Marcos



Ejemplo de pagina con 3 zonas: de filas y columnas  
¿Cuántas filas y cuántas columnas?

# Marcos

## Partes de un Documento XHTML

Declaración del tipo de documento

```
<?xml version="1.0" encoding="UTF-8"?>  
!<DOCTYPE frameset.....>
```

Etiquetas de inicio y final del documento

```
<html ... ..>
```

Etiquetas de inicio y final de la cabecera del documento

```
<head>
```

```
</head>
```

La etiqueta **body** se sustituye por la etiqueta **frameset**

```
<frameset>
```

```
</frameset>
```

```
</html>
```

# Marcos

## PASOS A SEGUIR:

- 1. Diseño:** Lo primero que hay que hacer es crear un documento XHTML (principal.html) en el que se definan cuantas zonas o marcos van a haber, qué distribución y tamaño van a tener
  - 2. Contenido:** En segundo lugar se determinará cual va ser el contenido de cada una de las zonas o marcos, para ello se definirán ficheros html (pagina1.html, pagina2.html..etc).
- **NOTA:** es conveniente que tanto el fichero donde se diseña la páginas de marcos (principal.html) y los ficheros que determinan el contenido esté en el mismo directorio

# Marcos: 1.diseño

La etiqueta **frameset** es la que divide la distribución, el número y tamaño de los frames. Para ellos podemos utilizar dos atributos: **cols** (columnas) y/o **rows** (filas)

```
<frameset cols="xx, yy, zz...">
```

.....

```
</frameset/>
```

```
<frameset rows="xx, yy, zz...">
```

.....

```
</frameset/>
```

- Donde xx, yy,zz...define el tamaño de los frames. Este tamaño se puede expresar en:
  - A) Un **porcentaje** del ancho de la pantalla (para las columnas), o el alto de las filas (para las filas)
  - B) Un **número absoluto**, que representa el número de pixels que ocupará cada frame a lo ancho o a lo alto (según filas o columnas)

# Marcos: 2. especificar contenido de cada marco o zona

Tenemos otra etiqueta `<frame/>` que sirve para indicar qué documento se cargará en cada uno de los marcos o zonas de la ventana, esto se realiza incluyendo el atributo `src`. Con esta etiqueta se indica el contenido de cada marco o zona, es decir qué documento ( página html) se cargará en cada marco.

-Tiene otra etiqueta llamada `name` que sirve para identificar cada marco. Es muy recomendable utilizarlo

```
<frameset cols="xx, yy, zz...">
  <frame src="pagina1.html" name="marco1"/>
  <frame src="pagina2.html" name="marco2"/>
  <frame src="pagina2.html" name="marco3"/>
  .....
</frameset/>
```

# Marcos



Veamos como crear el siguiente marco:

**Observaciones:** Diseño: 2 zonas o marcos distribuidos verticalmente, una más estrecha que la otra-->2 columnas

Contenido: - marco izquierda contiene palabra marco1  
-marco derecha contiene

palabra marco 2

¿Cuántas páginas xhtml tendremos que generar?-->3 páginas  
(principal.html, marco1.html, marco2.html)

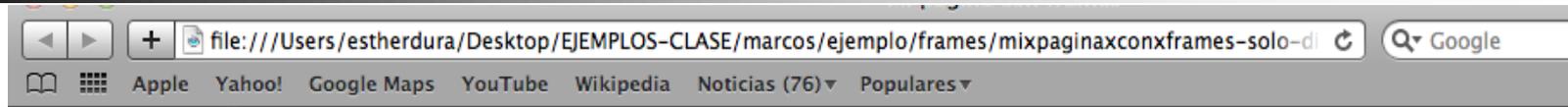
# Marcos: 1.Diseño

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Frameset//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-
frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Mi pagina primera pagina con frames</title>
</head>
<frameset cols="10%,80%">
  <frame />
  <frame />
</frameset>
</html>
```

Va a tener dos columnas, una que ocupará el 10% de la pantalla y otra el 80%

Permitirá definir el contenido de cada marco (columna)

# Marcos



# Marcos: 2. contenido

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Frameset//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-
frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Mi pagina primera pagina con frames</title>
</head>
<frameset cols="10%,80%">
  <frame src="marco1.html" name="m1"/>
  <frame src="marco2.html" name="m2"/>
</frameset>
</html>
```

Indico contenido del  
primer marco(columna)

Indico contenido del  
segundo marco (columna)

# Marcos

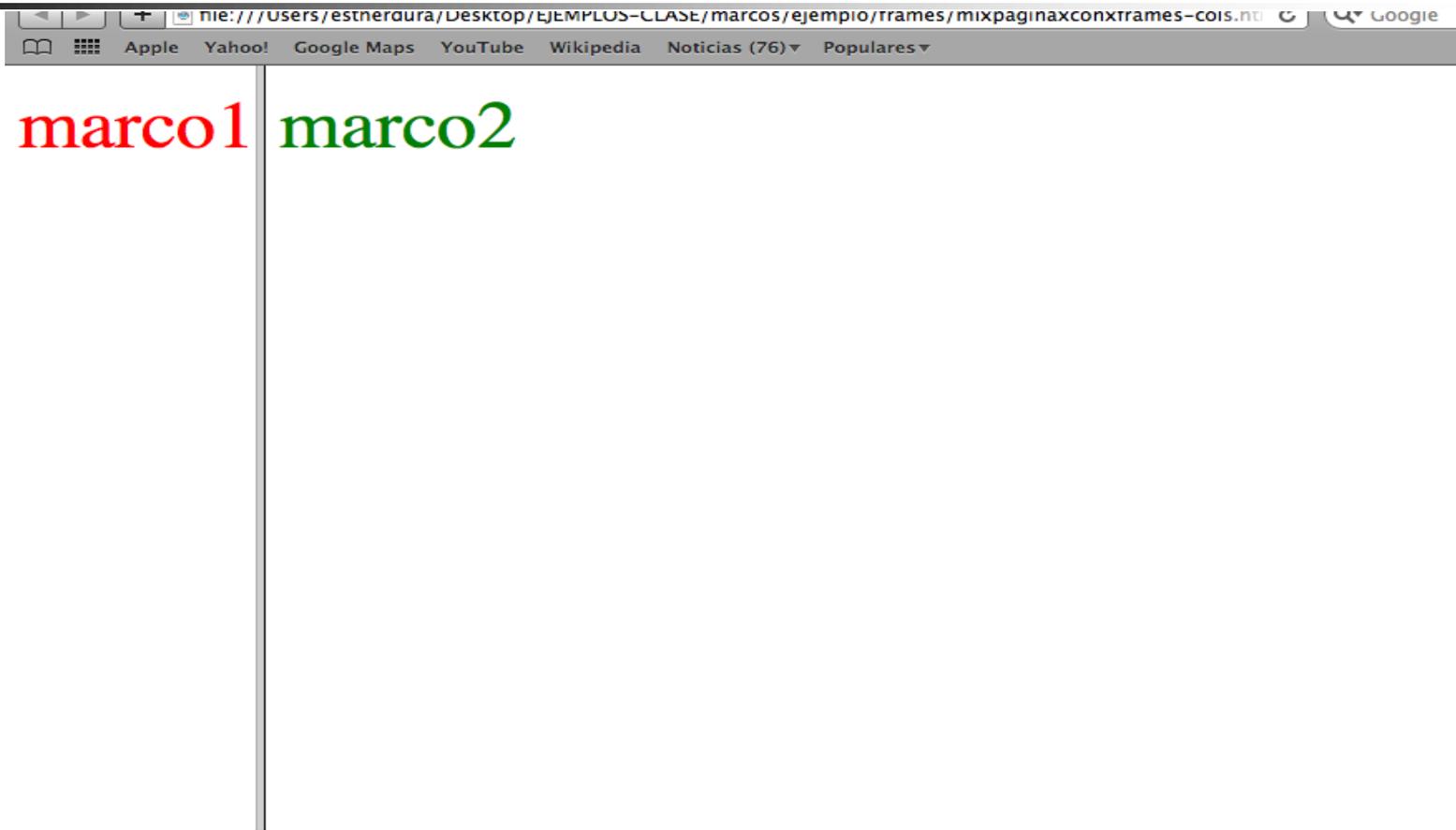
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Indice</title>
</head>
<body>
<p><font size="7" color="red"> marco1 </font></p>
</body>
</html>
```

marco1.html

```
!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>
<title>Indice</title>
</head>
<body>
<p><font size="7" color="red"> marco2 </font></p>
</body>
</html>
```

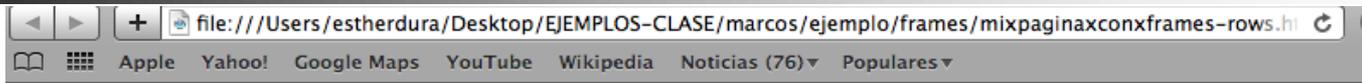
marco2.html

# Marcos



Ejemplo de pagina con 2 zonas o marcos distribuidos verticalmente-->2 columnas

# Ejercicios: crea el siguiente marco



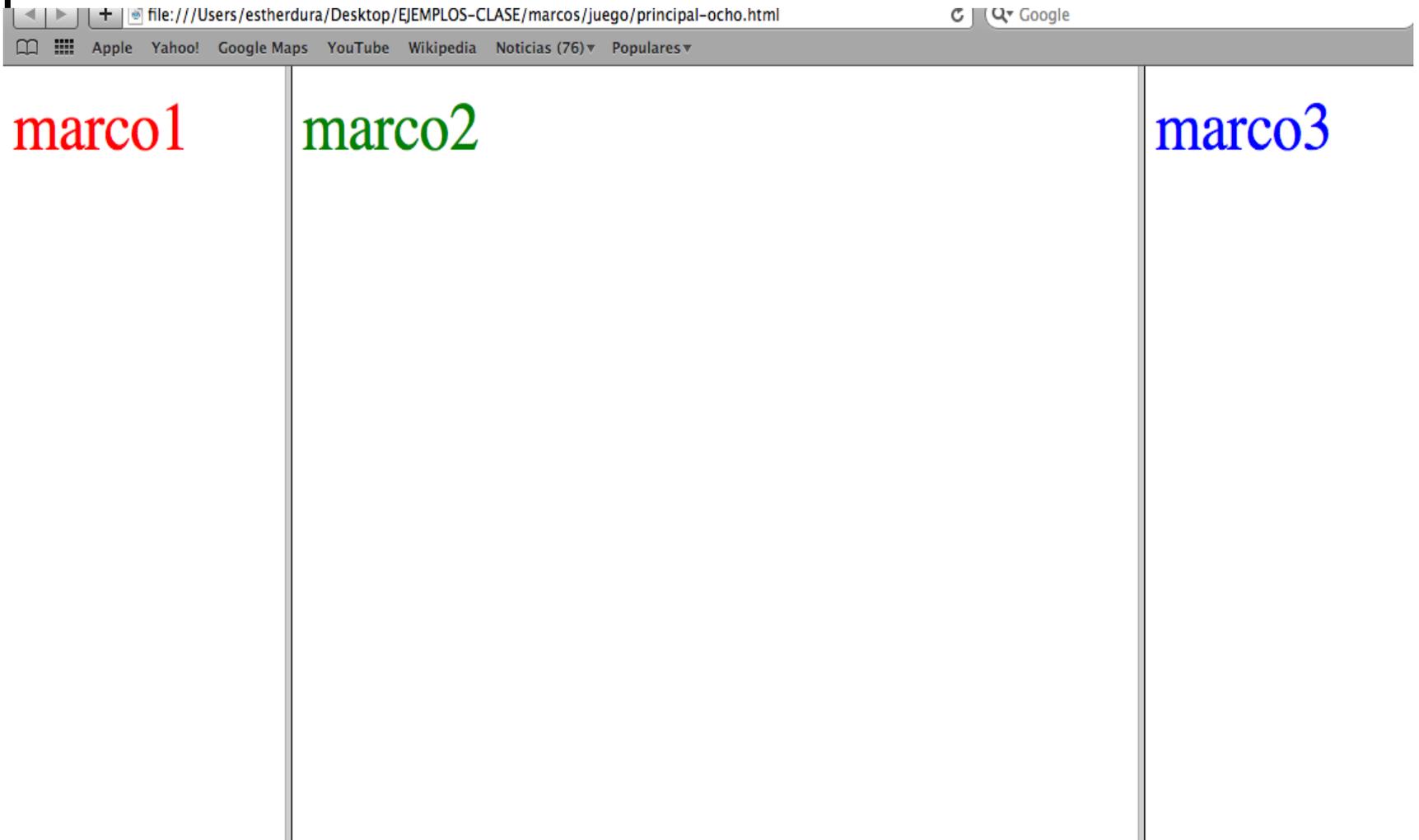
marco1

---

marco2

Ejemplo de pagina con 2 zona o marcos distribuidos horizontalmente-->2 filas

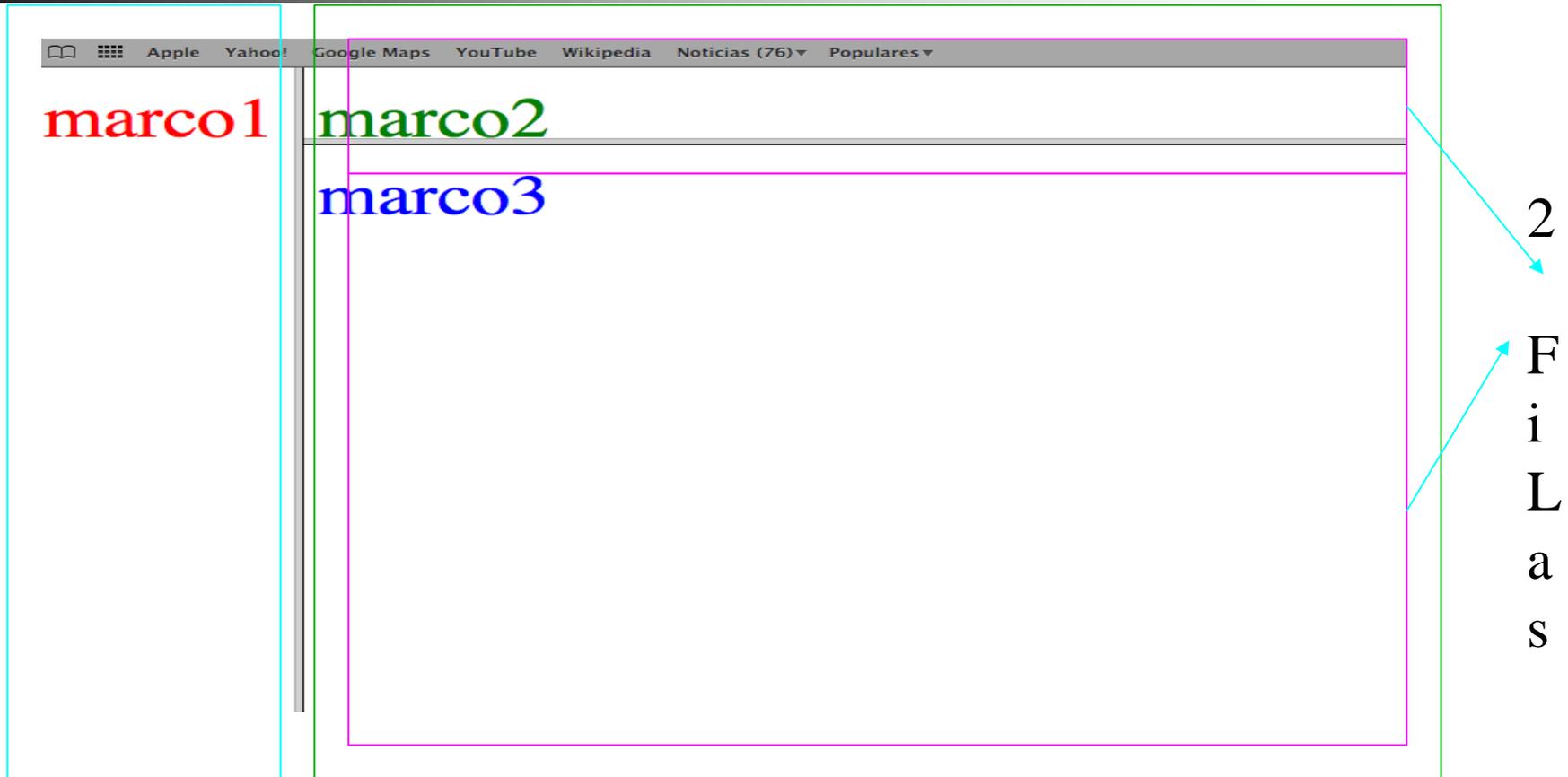
# Ejercicios: crea el siguiente marco



Columna 1

# Marcos

Columna 2



Ejemplo de pagina con 3 zonas: de 2 columnas y 2 filas en la columna de la derecha

# Marcos

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Frameset//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-
frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Mi pagina primera pagina con frames</title>
</head>
<frameset cols="10%,80%">
  <frame src="marco1.html" name="m1"/>
  <frame src="marco2.html" name="m2"/>
</frameset>
</html>
```

Indico contenido del primer marco(columna)

Esta linea se sutituye por

```
<frameset rows="10%, 80%">
  <frame src="marco2.html" />
  <frame src="marco3.html" />
</frameset>
```

# Marcos

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Frameset//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-
frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Mi pagina primera pagina con frames</title>
</head>
<frameset cols="10%,80%">
  <frame src="marco1.html" name="m1"/>
  <frameset rows="10%, 80%">
    <frame src="marco2.html" name="m2" />
    <frame src="marco3.html" name="m3"/>
  </frameset>
</frameset>
</html>
```

Indico contenido del primer marco(columna)

Indico contenido del segundo marco el cual contiene dos filas

# Ejercicio

---

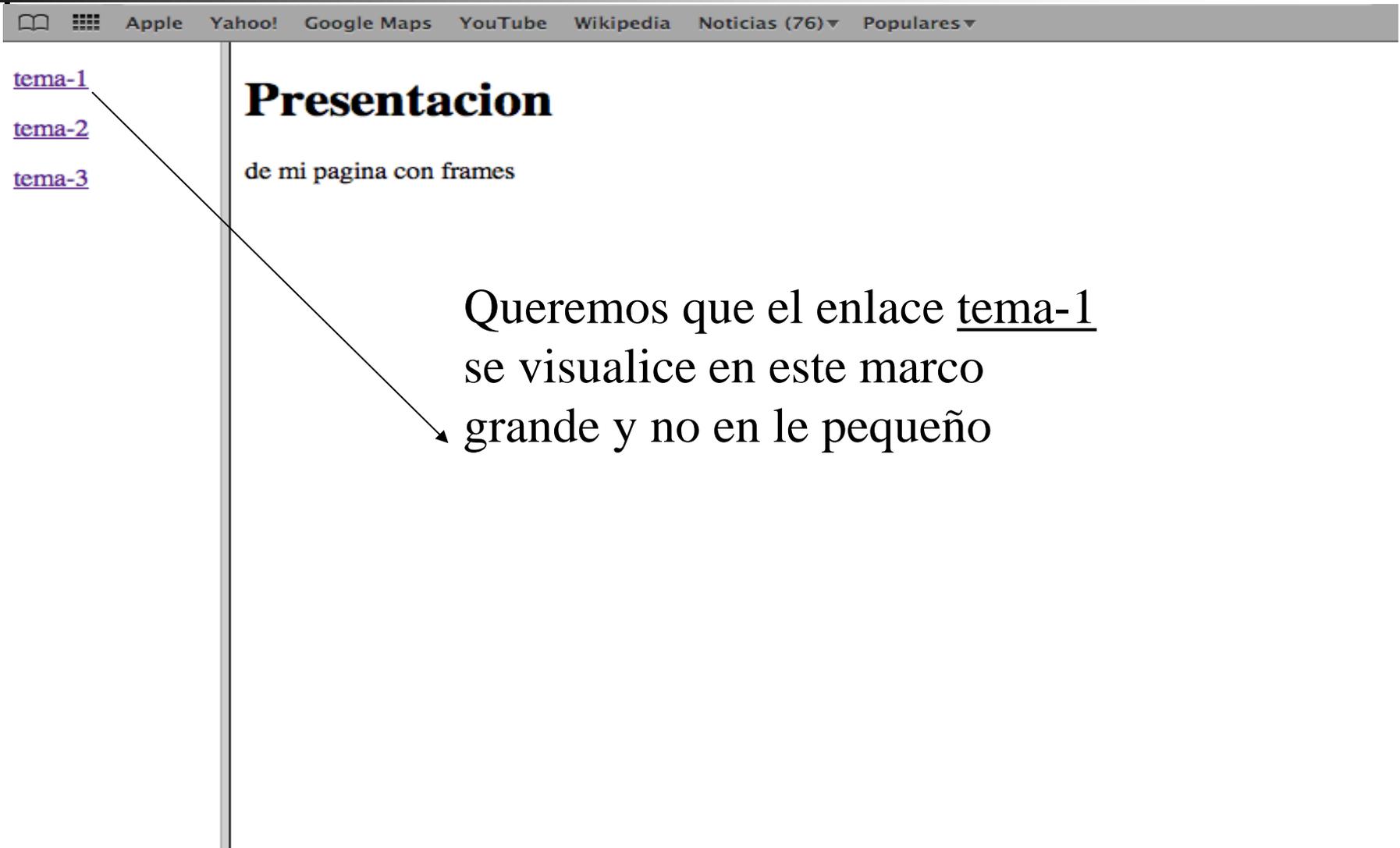
- Dado la página web basada en marcos que se muestra la profesora en pantalla identifica el código xhtml que lo genera

# Marcos

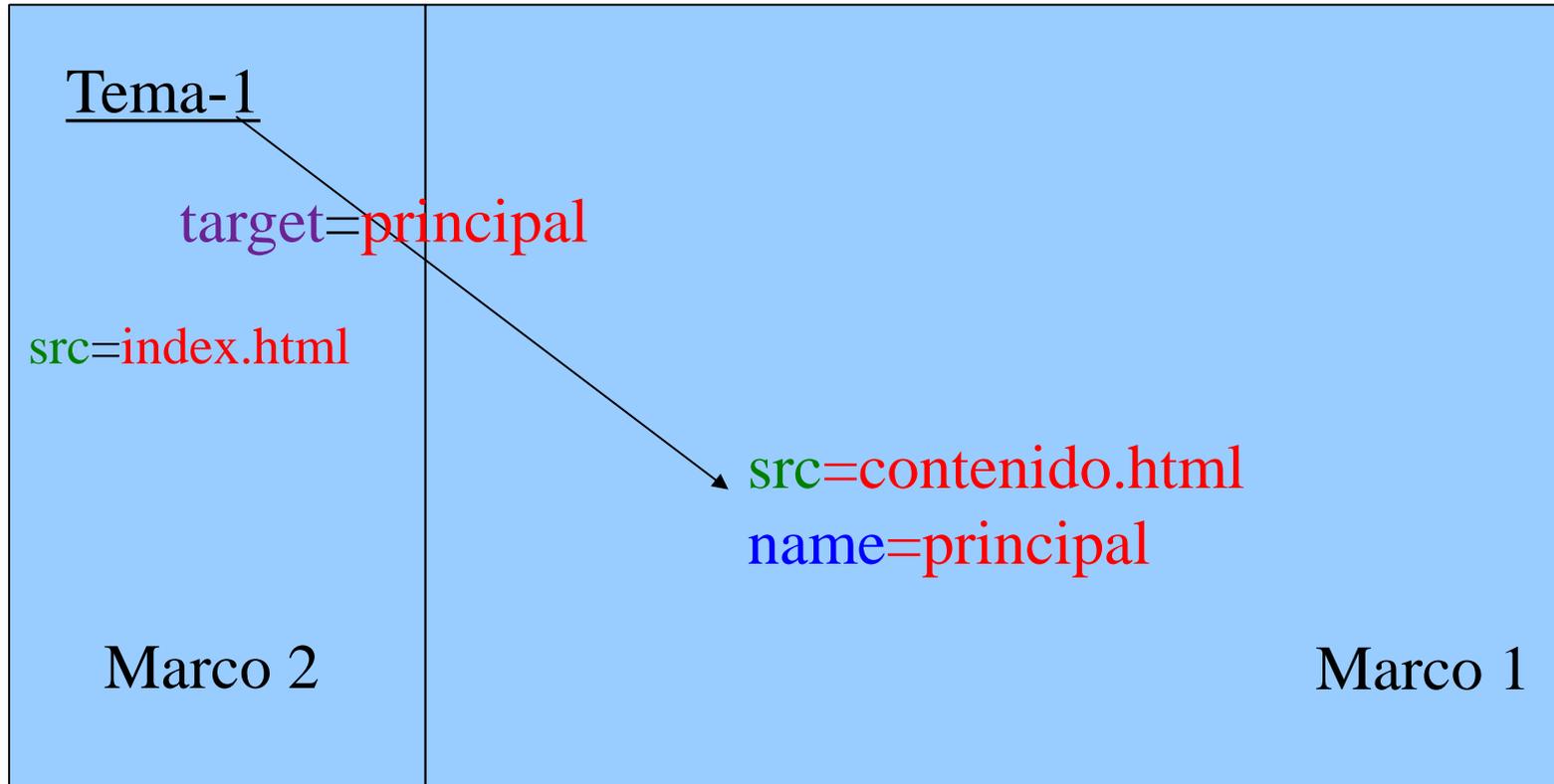
## Atributos de la etiqueta `<frame/>`

Atributos	Expiación
<code>src="direccion"</code>	La direccion puede ser la de un documento HTML o cualquier recurso web (o URL). Sino se pone nada aparecerá vacío
<code>name="nombre de la ventana"</code>	Se usa para asignar un frame
<code>Marginwidth="numer"</code>	Se utiliza cuando se quiere controlar el ancho de los márgenes dentro de un frame (en pixels)
<code>Marginheight="número"</code>	Igual que el anterior, pero referido a los márgenes en altura
<code>Scrolling="yes no auto"</code>	Este atributo se utiliza para indicar si el frame tendrá o no una barra deslizador
<code>Noresize</code>	Es un indicador para que la ventana no se pueda redimensionar
<code>Frameborder="1 0"</code>	Elimina el borde cuando toma un valor de 1

# Enlaces entre marcos



# Enlaces entre marcos



`mi_pagina_frames.html`

# Enlaces entre marcos

```
<a href="nombre-pagina.html" target="nombre_dado_a_otro_frame">  
</a>
```

Se debe usar el atributo **target** para indicar en que marco se va a visualizar el enlace

# Enlaces entre marcos

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

<head>
<title>Mi pagina con frames</title>
</head>

<frameset cols="10%,80%">
  <frame src="indice.html" name="indi"/>
  <frame src="contenido.html" name="principal"/>
</frameset>
</html>
```

mi\_pagina\_frames.html

# Enlaces entre marcos

```
<!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>
<title>Indice</title>
</head>
<body>
<p><a href="tema1.html" target="principal"> tema-1 </a></p>
<p><a href="tema2.html" target="principal"> tema-2 </a></p>
<p><a href="tema3.html" target="principal"> tema-3 </a></p>
</body>
</html>
```

indice.html

```
<!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>
<title>tema1</title>
</head>
<body>
<p>Esta es la introducción del tema 1: edici&oacute;n electr&oacute;nica</p>
</body>
</html>
```

tema1.html

# Enlaces entre marcos

```
<!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>
<title>Presentacion</title>
</head>
<body>
<h1>Presentacion</h1>
<p>de mi pagina con frames</p>
</body>
</html>
```

contenido.html

# Imágenes: mapas de pixels

`<img atributo1=" " ...atributo2=".." />`

Atributo	Explicación
<code>src=".."</code>	Contienen el URL de la imagen a incluir. Este atributo es obligatorio
<code>alt=".."</code>	Asigna un texto descriptivo relacionado con la imagen
<code>align=".."</code>	Controla la alineación de la imagen con respecto al párrafo en que esta insertada. Los valores que pueden tomar son: top, middle, bottom
<code>align=".."</code>	Control de la alineación de la imagen, puede tomar los valores left y right y permite que un párrafo fluya sobre el borde de una imagen, bloqueando ésta a la izquierda o derecha del margen. No puede utilizarse junto con la anterior
<code>height=".."</code>	Informa al Cliente Web de la altura de la imagen a insertar
<code>width=".."</code>	Informa al cliente Web de la anchura de la imagen a insertar.
<code>border=".."</code>	Fija la anchura del borde que rodeará a la imagen. Un valor de = desactiva el bloque
<code>hspace=".."</code>	Fija la anchura horizontal de los márgenes de la imagen (en pixels)
<code>vspace=".."</code>	Fija la anchura vertical de los márgenes de la imagen (en pixels)

# Imágenes: mapas de pixels

<p>Vamos a insertar una imagen a continuacion</p>

<p> Todo el texto de este párrafo se colocará a la derecha de la

imagen al estar esta alineada a la izquierda

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

<p>Pero al sobrepasar la imagen continuará de forma normal</p>



# Formularios

- Sección de un documento html que contiene elementos especiales llamados **controles** (cajas de texto, botones...etc)
- Los **usuarios** completan el texto actuando sobre los controles
- Los datos introducidos en el formulario se envían a un agente (servidor web, de correo..) que enviará esta información

The screenshot shows a web browser window titled "Rellena tu CV - Opera". The browser's menu bar includes "Archivo", "Editar", "Ver", "Marcadores", "Widgets", "Herramientas", and "Ayuda". The page content is titled "Rellena tu CV" and contains the following form elements:

- Nombre**: A text input field.
- Apellidos**: A text input field.
- Contraseña**: A text input field.
- DNI**: A text input field.
- Sexo**: Radio buttons for "Hombre" (selected) and "Mujer".
- Incluir mi foto**: A text input field followed by a button labeled "Elegir...".
- Suscribirme al boletín de novedades**
- Buttons at the bottom: "Guardar cambios" and "Borrar los datos introducidos".

# Formularios

La etiqueta principal para los formularios: `<form></form>`

## Atributos:

`action`=”url” agente que procesará el formulario (Este atributo es obligatorio)

`method`=get|post método utilizado para enviar los datos del formulario

`enctype`=designa el tipo de codificación de datos transmitidos

-valor por defecto:”`application/x-www-form-urlencoded`”

-sólo para formularios de transferencia de archivos: “`multipart/form-data`”

`accept-charset`= designa el juego de caracteres aceptados para la codificación en los campos de los elementos de los formularios. Por defecto, es el definido en el prólogo XML

`accept`=designa el tipo de archivos aceptados para los formularios de transferencia de archivos. (sólo para formularios de transferencia de archivos)

# Formularios

---

2 tipos de formularios:

1) Simples

2) Avanzados

# Formularios: Simples

La etiqueta: `<input/>` permite crear diferentes tipos de controles de formularios del atributo principal `type`

El tipo de control que definen depende del atributo `type` el cual puede tomar los siguientes valores:

-text

Nombre

-passwd

Contraseña

-checkbox

Puestos de trabajo buscados  
 Dirección  
 Técnico  
 Empleado

-radio:botón de radio

Sexo  
 Hombre  
 Mujer

-submit botón para enviar formulario

-reset: reiniciar un formulario

-hidden:control oculto

\*Los campos ocultos  
no se ven en pantalla

-file: transferencia o carga de ficheros

Fichero adjunto

# Formularios: Simples

Otros atributos de la etiqueta `<input/>` son:

`name=` asigna un nombre al control (es imprescindible para que el servidor pueda procesar el formulario)

`value='texto'`

`size='unidad_de_medida'` -Tamaño inicial de control (campos de `texto` y `passwd` se refiere al número de caracteres, en el resto se refiere a su tamaño en pixel)

`maxlength='numero'` máximo número de caracteres para los controles de texto y passwd

`checked="checked"` -Para los controles `checkbox` y `radio` button permite indicar qué opción aparece preseleccionada

`disabled="disabled"` -El control aparece deshabilitado y su valor no se envía al servidor junto con el resto de los datos

# Formularios: Simples

---

Otros atributos de la etiqueta `<input/>` son:

`Readonly='readonly'` el contenido del control no se puede modificar

`src='url'`-Para el control que permite crear botones con imágenes, indica la URL de la imagen que se emplea como botón de formulario

`Alt='texto'`-Descripción del control

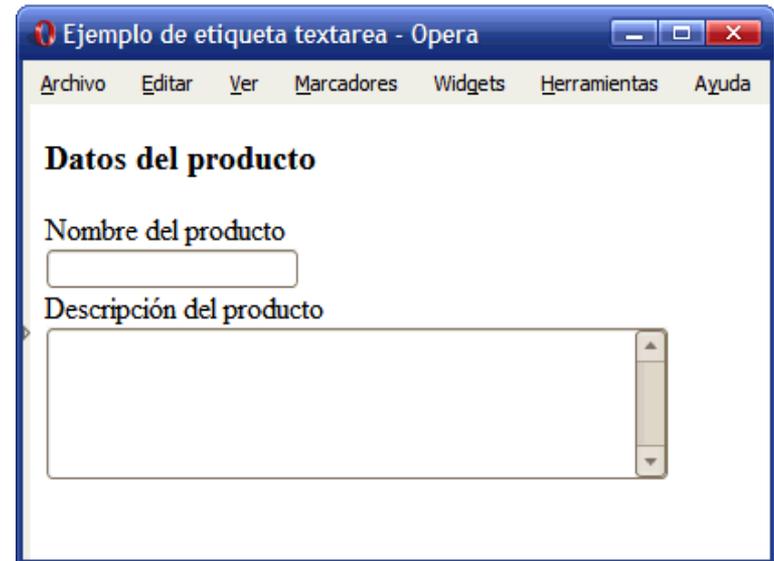
# Formularios: Simples

Algunas aplicaciones Web utilizan otros elementos de formularios que no se pueden crear con la etiqueta `<input/>`

Las listas desplegables y las áreas de texto disponen de sus propias etiquetas: `<select><option></option></select>` para listas y `<textarea></textarea>` para áreas de texto



Ejemplo formulario con listas desplegables



Ejemplo formulario con áreas de texto

# Ejercicio

---

Lee con detenimiento y posteriormente discute con el resto de tus compañeros de grupo el formulario que se te ha asignado para posteriormente ser explicado en un máximo de 5 minutos al resto tus compañeros

# Formularios: Ejercicio

Escribe el código XHTML necesario para realizar el siguiente formulario

The screenshot shows a web browser window with the title "Rellena tu CV - Opera". The menu bar includes "Archivo", "Editar", "Ver", "Marcadores", "Widgets", "Herramientas", and "Ayuda". The main content area is titled "Rellena tu CV" and contains the following form elements:

- Nombre:
- Apellidos:
- Contraseña:
- DNI:
- Sexo:  Hombre,  Mujer
- Incluir mi foto:
- Suscribirse al boletín de novedades
-

# Formularios: Avanzados

Existen unas etiquetas adicionales que permiten mejorar la estructura de los formularios

La siguiente imagen muestra un formulario que **agrupa sus campos** y asignar **un nombre** a cada grupo

La etiqueta `<fieldset></fieldset>` agrupa campos del formulario y la etiqueta `<legend> </legend>` asigna un nombre a cada grupo.

Grupo 1

The screenshot shows a browser window titled "Ejemplo de etiqueta fieldset y legend - Opera". The browser's menu bar includes "Archivo", "Editar", "Ver", "Marcadores", "Widgets", "Herramientas", and "Ayuda". The main content area is titled "Formulario estructurado" and contains two distinct sections, each enclosed in a box. The first section is titled "Datos personales" and contains three input fields labeled "Nombre", "Apellidos", and "DNI". The second section is titled "Datos de conexión" and contains three input fields labeled "Nombre de usuario", "Contraseña", and "Repite la contraseña". Red arrows point from the text "Grupo 1" to the "Datos personales" section and from "Grupo 2" to the "Datos de conexión" section. Green ovals highlight the legend text for each section.

Grupo 2

Ejemplo de formulario Avanzado

# Formularios: Avanzados

```
<form action="maneja_formulario.php" method="post">
  <fieldset>
    <legend>Datos personales</legend>
    Nombre <br/>
    <input type="text" name="nombre" value="" />
    <br/>
    Apellidos <br/>
    <input type="text" name="apellidos" value="" />
    <br/>
    DNI <br/>
    <input type="text" name="dni" value="" size="10"
maxlength="9" />
  </fieldset>
  <fieldset>
    <legend>Datos de conexión</legend>
    Nombre de usuario<br/>
    <input type="text" name="nombre" value=""
maxlength="10" />
    <br/>
    Contraseña<br/>
    <input type="password" name="password" value=""
maxlength="10" />
    <br/>
    Repite la contraseña<br/>
    <input type="password" name="password2" value=""
maxlength="10" />
  </fieldset>
</form>
```

Grupo 1

Grupo 2

Ejemplo de etiqueta fieldset y legend - Opera

Archivo Editar Ver Marcadores Widgets Herramientas Ayuda

**Formulario estructurado**

Datos personales

Nombre

Apellidos

DNI

Datos de conexión

Nombre de usuario

Contraseña

Repite la contraseña

Ejemplo de formulario Avanzado

# Parte III: Hojas de estilo: CSS

---

# Definición: CSS

---

- CSS (Cascade Style sheet ) tiene como objetivo controlar el **aspecto o presentación** de los **documentos** electrónicos definidos con HTML y XHTML.
- Procedimiento a seguir:
  - 1) Primero: utilizar el lenguaje XHTML para definir los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos, etc.
  - 2) Segundo: utilizar 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.

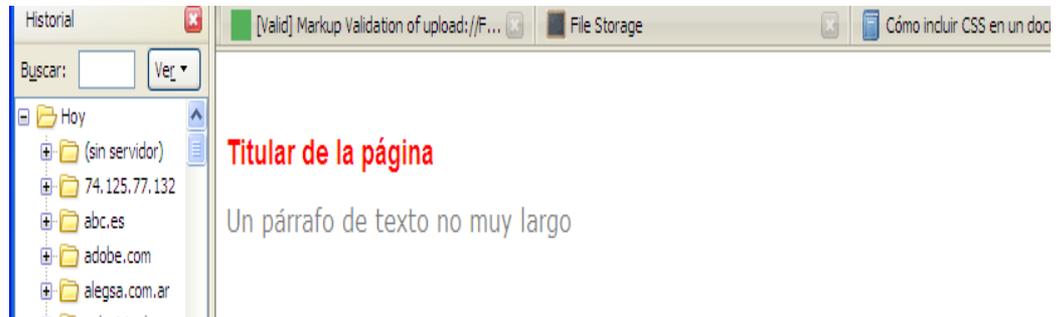
# Definición: CSS

---

- **Ventajas separar definición y aspecto de los contenidos-→**
  - 1) mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes.
  - 2) obliga a crear documentos HTML/XHTML bien definidos y con significado completo.

# Ejemplo página xhtml con estilos definidos sin usar 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>
  <title>Ejemplo de estilos sin CSS</title>
</head>
<body>
  <h1> <font color="red" face="Arial" size="5">Titular de la  página</font></h1>
  <p><font color="gray" face="Verdana" size="2">Un párrafo de texto no muy
largo.</font>
  </p>
</body>
</html>
```



# Ejemplo página xhtml con estilos definidos usando 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>
```

```
<title>Ejemplo de estilos sin CSS</title>
```

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

```
h1 { color: red; font-family: Arial; font-size:2.8em; }
```

```
p { color: gray; font-family: Verdana; font-size: medium; }
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1> Titular de la página</h1>
```

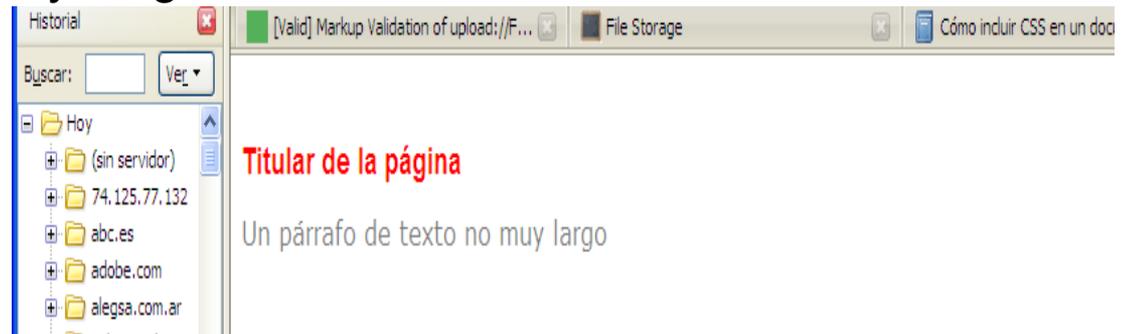
```
<p>Un párrafo de texto no muy largo
```

```
</p>
```

```
</body>
```

```
</html>
```

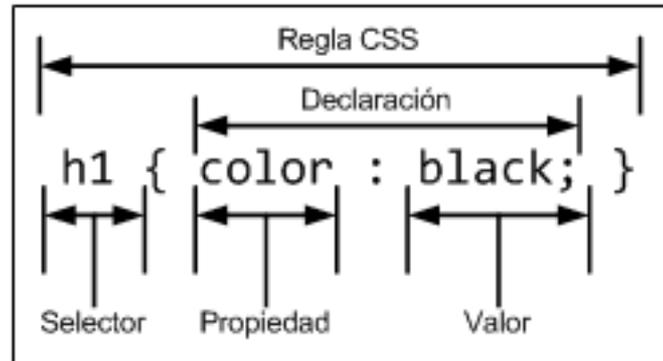
Zona  
definición  
hoja de  
estilo



# Cómo incluir CSS en un documento XHTML

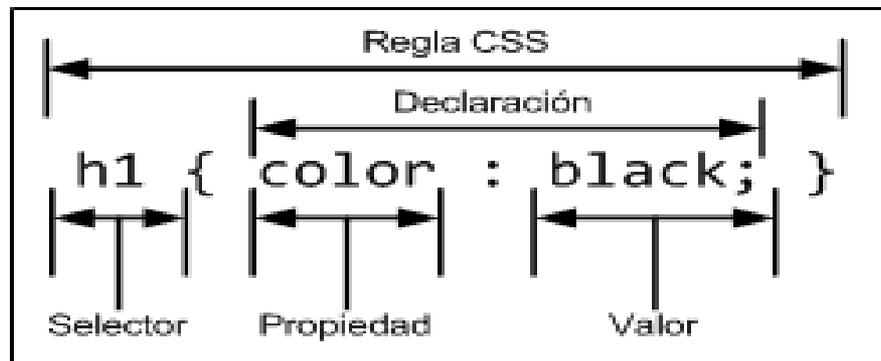
- CSS funcionan porque permiten asociar reglas con los elementos (etiquetas) que aparecen en el documentos.

- Ejemplo de regla CSS:



# Cómo incluir CSS en un documento XHTML

- **Regla:** cada uno de los estilos que componen una hoja de estilos CSS. Cada regla está compuesta de una parte de "*selectores*", un símbolo de "*llave de apertura*" (`{`), otra parte denominada "*declaración*" y por último, un símbolo de "*llave de cierre*" (`}`).
- **Selector:** indica el elemento(etiquetas) o elementos (separados por comas) XHTML a los que se aplica la regla CSS.
- **Declaración:** especifica los estilos que se aplican a los elementos(etiquetas). Está compuesta por una o más propiedades CSS.
- **Propiedad:** característica que se modifica en el elemento seleccionado, como por ejemplo su tamaño de letra, su color de fondo, etc.
- **Valor:** establece el nuevo valor de la característica modificada en el elemento.



# Cómo incluir CSS en un documento XHTML

Es posible definir varias propiedades por selector

```
h1 { propiedad1:valor; propiedad2:valor, propiedad3:valor }
```

## Ejemplo

```
<!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>
  <title>Ejemplo de estilos sin CSS</title>
  <style type="text/css">
    h1 { color:white; font-size:1.5em; }
    body{background-color:black}
  </style>
</head>
<body>
  <h1>mi primera hoja de estilo</h1>
</body>
</html>
```



# Propiedades que se pueden aplicar sobre hojas de estilo

- Veremos diferentes propiedades que se pueden aplicar a las apariencias del texto, fondo, fuente, dimensiones y posicionamiento de los elementos en textos y tablas.

FONDO	TEXTO	FUENTE	TABLAS
background-color	color	font-family	border-collapse
background-image	text-align	font-style	width, height
background-repeat	text-transform	font-size	text-align
background-attachment	text-indent		vertical-align
background-position			padding
			border
			color
			background-color

# Ejercicio

Dado el siguiente código en xhtml genera la hoja de estilo de tal forma que se visualice así:

```
<body>
  <h1> Dieta Mediterránea</h1>
  <p>Esto es una sugerencia de una dieta
  </p>
  <table border="4">
    <tr><th>Lunes</th><th>Martes</th></tr>
    <tr><td>Pollo con
  Pechuga</td><td>Lentejas</td></tr>
  </table>
</body>
```

## Dieta Mediterránea

Esto es una sugerencia de una dieta

Lunes	Martes
Pollo con Pechuga	Lentejas

# Cómo incluir CSS en un documento XHTML

---

- 1) Incluir CSS en el mismo documento XHTML. También llamadas hojas de estilo internas
- 2) Definir CSS en un archivo externo (fichero separadas). También llamadas hojas de estilo externas
- 3) Incluir CSS en los elementos XHTML ( no recomendada!!)

# 1. En el propio documento XHTML

Los estilos se definen en una zona específica del propio documento HTML. Se emplea la etiqueta `<style>` de HTML y solamente se pueden incluir en la cabecera del documento (sólo dentro de la sección `<head>`).

## EJEMPLO:

```
<!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>
  <title>Ejemplo de estilos CSS en el propio documento</title>
  <style type="text/css">
    p { color: cyan; font-family: Verdana; }
  </style>
</head>
<body>
  <p>Un párrafo de texto </p>
</body>
</html>
```

Zona de definición de la hoja de estilo



## 2. En un fichero externo

---

- En este caso, todos los estilos CSS se incluyen en un archivo de tipo CSS que las páginas HTML enlazan mediante la etiqueta `<link>`.
- Un archivo de tipo CSS no es más que un archivo simple de texto cuya extensión es `.css`
- Se pueden crear todos los archivos CSS que sean necesarios y cada página HTML puede enlazar tantos archivos CSS como necesite.

## 2.En un fichero externo: ejemplo

Para el ejemplo anterior se deben seguir los siguientes pasos:

1) Se crea un archivo de texto y se le añade solamente el siguiente contenido:

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

2) Se guarda el archivo de texto con el nombre `mi_hoja_deestilo.css` Se debe poner especial atención a que el archivo tenga extensión `.css` y no `.txt`

3) En la página HTML se enlaza el archivo CSS externo mediante la etiqueta `<link>`:

```
<!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>
  <title>Ejemplo de estilos CSS en un archivo externo</title>
  <link rel="stylesheet" type="text/css" href="mi_hoja_deestilo.css" media="screen" />
</head>
<body>
  <p>Un párrafo de texto.</p>
</body>
</html>
```

## 2. Atributos etiqueta: link

---

Normalmente, la etiqueta `<link>` incluye cuatro atributos cuando enlaza un archivo CSS:

**rel:** indica el tipo de relación que existe entre el recurso enlazado (en este caso, el archivo CSS) y la página XHTML. Para los archivos CSS, siempre se utiliza el valor `stylesheet`

**type:** Sus valores están estandarizados y para los archivos CSS su valor siempre es `text/css`

**href:** indica el archivo CSS que contiene los estilos.

**media:** indica el medio en el que se van a aplicar los estilos del archivo CSS. Puede tomar los valores de *screen* (pantalla ordenador), *tty* (dispositivos móviles), *tv*, *print* (documento impresos)

## 2. Ventajas hojas estilo externas:

---

- La misma hoja se puede reutilizar por todas las páginas Web de un sitio → garantiza la aplicación homogénea de los mismos estilos a todas las páginas que forman un sitio web.
- El mantenimiento del sitio web se simplifica al máximo, ya que un sólo cambio en un sólo archivo CSS permite variar de forma instantánea los estilos de todas las páginas HTML que enlazan ese archivo.
- Evita sobrecargar al servidor (la hoja de estilo CSS se descarga con el primer documento una sola vez)
- Reutilizar el mismo documento con diferentes hojas de estilo para diferentes necesidades de los visitantes (impresión, dispositivo portátil)

# 3.En los elementos XHTML

- Se añaden como un valor de un atributo *style* en cualquier elemento que incluya el atributo *style*. También se conocen como reglas de estilo en línea

```
<!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>
```

```
<title>Ejemplo de estilos CSS en el propio documento</title> </head>
```

```
<body>
```

```
<p style="color: black; font-family: Verdana;">
```

```
Un párrafo de texto.</p>
```

```
</body>
```

```
</html>
```

!!\*\*Ojo style es un atributo aquí y no un elemento\*\*!!

## 3. En los elementos XHTML

---

- Esta forma de incluir CSS directamente en los elementos HTML solamente se utiliza en determinadas situaciones en las que se debe incluir un estilo muy específico para un solo elemento concreto.
- No es recomendable utilizarla

# Propiedad importante de las hojas de estilo

---

- **Herencia:** las propiedades que se aplican a los elementos se heredarán por los elementos hijos (elementos contenido dentro del elemento sobre el que se han declarado sus reglas)
- Si aparece una regla más específica anulará cualquier propiedad asociada con el elemento contenedor

# Ejercicio

---

Dado la siguiente página web de un curriculum, realiza una hoja de estilo externa con las reglas necesarias para que el documento se visualice tal como se ve en el navegador.