



### Caso práctico

Despois de asociarse, **María e Félix** esperan aumentar o número de clientes do seu negocio tendo presenza en internet, para iso falan con Xoán, técnico superior en Administración de Sistemas Informáticos.

**Xoán** aconséllalles desenvolver unha sinxela páxina web para a empresa, cuxa funcionalidade pode ir aumentando a medida que aumenten as súas necesidades. Polo que o primeiro paso será encargarllo a alguén que coñeza a linguaxe HTML.

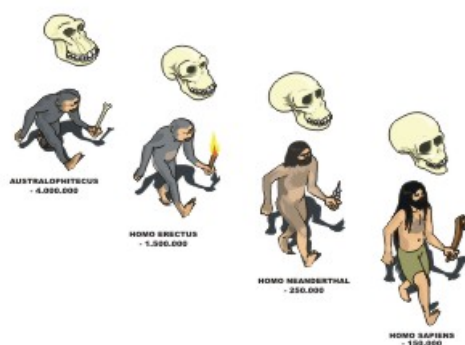
**María** pregúntalle a Xoán pola súa dispoñibilidade e capacidade para realizar a dita tarefa. **Xoán** contéstalle que aprendeu HTML en 1998. O boom de internet animouno a interesarse por este campo e nesta linguaxe realizou a súa primeira páxina web. Desde entón a linguaxe sufriu cambios nos que se foi actualizando.

# 1. De HTML a XHTML: evolución e versións

HTML é a linguaxe utilizada para crear a maior parte das páxinas web. É un estándar recoñecido en todos os navegadores, polo tanto, todos eles visualizan unha páxina HTML de forma moi similar independentemente do sistema operativo sobre o que se executan.

A evolución das súas versións, desde a súa creación ata a creación do XHTML podemos vela a continuación:

A orixe de HTML foi un sistema de hipertexto 📁 para compartir documentos electrónicos en 1980. A primeira proposta oficial para converter HTML nun estándar realizouse en 1993. Aínda que ningunha das dúas propostas de estándar que se fixeron (HTML e HTML+) conseguiron converterse en estándar oficial.



- ✓ **HTML 2.0** foi a primeira versión oficial de HTML o 📁 IETF publicou o estándar en setembro de 1995.
- ✓ **HTML 3.2** publicouse o 14 de xaneiro de 1997 polo W3C. Incorpora as 📁 miniaplicacións de Java e texto ao redor das imaxes.
- ✓ **HTML 4.0** publicouse o 24 de abril de 1998. Entre as novidades que presenta atópanse as follas de estilos CSS e a posibilidade de incluír pequenos programas nas páxinas web.
- ✓ **HTML 4.01** é a última especificación oficial de HTML publicouse o 24 de decembro de 1999. É unha actualización da versión anterior. Nese momento o W3C detivo a actividade de estandarización de HTML ata marzo de 2007, momento en que se retoma debido á forza das empresas que forman o grupo 📁 WHATWG e á publicación dos borradores de HTML 5.0, que será a seguinte versión desta linguaxe.

Tras a publicación do estándar HTML 4.01 detéctase a súa incompatibilidade con ferramentas baseadas en XML. Para evitar estes problemas créase linguaxe XHTML que combina a sintaxe de HTML 4.0 coa de XML.

- ✓ **XHTML 1.0** foi a primeira versión, publicouse o 26 de xaneiro de 2000. É unha adaptación de HTML 4.01 á linguaxe XML, polo que mantén as súas características, e engade algunhas restricións e elementos de XML.
- ✓ A versión **XHTML 1.1** xa foi publicada en forma de borrador e pretende modularizar XHTML.
- ✓ O borrador de XHTML 2.0 xa foi publicado, que presenta grandes novidades respecto das anteriores versións.



## Autoavaliación

A aparición da linguaxe XHTML fai que deixe de evolucionar a linguaxe HTML:

- ☐ Si.
- ☐ Non.

## 2. Estrutura dun documento HTML



### Caso práctico

**Félix** preguntalle a **Xoán** se existen grandes diferenzas entre XML e HTML. **Xoán** explícalle que ambas as linguaxes teñen orixe noutra linguaxe que é a SGML e que as súas diferenzas son, principalmente, funcionais xa que a estrutura do documento é semellante.

A estrutura dunha páxina HTML ha de ser coherente coa que vimos no tema anterior para calquera documento XML. Por iso terá un prólogo e un exemplar.

- **Prólogo:** todo documento HTML ha de ter unha declaración do tipo de documento onde se lle indica ao navegador o tipo de documento que se vai iniciar e a versión de HTML utilizada para a codificación deste e, ademais, permítelle interpretalo correctamente.


Para a versión HTML 4.0, hai tres prólogos distintos que definen tres tipos de documentos HTML:

- *HTML 4.0 Strict.* É a DTD utilizada por defecto con HTML 4.0. Nestes documentos non se permite o uso dos elementos declarados deprecated noutras versións ou Recomendacións HTML. A declaración do tipo de documento correspondente é:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 //EN http://www.w3.org/TR/REC-html40/strict.dtd" "">
```

- *HTML 4.0 Transitional.* Permite o uso de todos os elementos que permite o HTML 4.0 Strict, ademais dos elementos deprecated. A declaración do tipo de documento correspondente é:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 transitional//EN http://www.w3.org/TR/REC-html40/loose.dtd" "">
```

- *HTML 4.0 Frameset.* É unha variante de HTML 4.0 Transitional para documentos que usan  frames. Nestes documentos o elemento body hai que substituílo por un elemento frameset. A declaración do tipo de documento correspondente é:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN http://www.w3.org/TR/REC-html40/frameset.dtd" "">
```

- **Exemplar:** nun documento HTML está delimitado polas etiquetas <html> e </html>. O exemplar pode, á súa vez dividirse en dúas partes:
  - A *cabeceira*, delimitada polas etiquetas <head> e </head>. Contén a información sobre o título da páxina, o autor, palabras clave etc. Dentro desta sección é obrigatorio definir o título do documento, para iso úsanse as etiquetas <title> </title>. Esta información non se presentará na ventá do navegador, salvo o título que aparecerá na barra de título da parte superior.
  - O *corpo*, contén a información que se vai presentar na pantalla. Está limitado polas etiquetas <body> e </body>, salvo nos documentos de tipo HTML 4.0 Frameset onde estas se substitúen por frameset <> e </frameset>.



### Autoavaliación

Cal das seguintes afirmacións é verdadeira?

- ☐ Un documento HTML ha de ter título aínda que non teña cabeceira.
- ☐ Un documento HTML ha de ter cabeceira aínda que non teña título.
- ☐ Un documento HTML ha de ter título e cabeceira.
- ☐ Un documento HTML ha de ter exemplar aínda que non teña título.

### 3. Identificación de etiquetas e atributos de HTML



#### Caso práctico

**María**, tras escoitar que XML e HTML teñen a mesma orixe, interésase por se utilizan as mesmas etiquetas.

**Xoán** contéstalle que en HTML as etiquetas e os atributos están definidos previamente, mentres que en XML os define o programador.

Un documento HTML está formado por etiquetas e atributos.

Do mesmo xeito que en XML as etiquetas poden ser de apertura, <etiqueta>, ou de peche, </etiqueta>. Unha das diferenzas con XML é que a cantidade de etiquetas de HTML está limitada a aquelas que están definidas pola linguaxe.

Aínda que HTML define unha gran cantidade de etiquetas, estas non son suficientes para crear páxinas complexas xa que a definición completa de certos elementos, como as imaxes e as ligazóns, require información adicional. Como non é posible crear unha etiqueta por cada elemento diferente, engádese a información adicional ás etiquetas mediante os atributos, o que dá lugar aos elementos.

Para cada un dos atributos hai definido un conxunto de valores que se lle pode asignar, se o valor dun atributo non é válido, o navegador ignóralo.

Cada unha das etiquetas HTML define os atributos que pode utilizar, aínda que algúns deles son comúns a moitas etiquetas.



#### Autoavaliación

As etiquetas de HTML, do mesmo xeito que as de XML, son ilimitadas:

- ☐ Verdadeiro.
- ☐ Falso.

## 3.1. Clasificación dos atributos comúns segundo a súa funcionalidade

- **Atributos básicos:** pódense usar en case todas as etiquetas HTML.

Atributo	Descrición
name = "texto"	Permite asignar o nome "texto" a un obxecto HTML
title = "texto"	Asigna un título a un elemento HTML, mellora así a accesibilidade. O dito título é mostrado polos navegadores cando o usuario pasa o rato por encima do elemento. É especialmente útil cos elementos: a, link, img, object, abbr e acronym
id = "texto"	<p>Permite identificar o elemento HTML sobre o que se aplica de forma única mediante o identificador "texto". Só é útil cando se traballa con CSS e con Javascript.</p> <p>Non poden empezar por números e só pode conter letras, números, guións medios e/ou guións baixos.</p>
style = "texto"	Permite aplicar ao elemento HTML o estilo "texto" directamente.
class = "texto"	<p>Permite aplicar ao elemento HTML o estilo "texto" definido nas CSS.</p> <p>Non poden empezar por números e só pode conter letras, números, guións medios e/ou guións baixos.</p>

- **Atributos para internacionalización:** utilízanos as páxinas que mostran os seus contidos en varios idiomas ou aquelas que queren indicar de forma explícita o idioma dos seu contidos.

Atributo	Descrición																
dir	Indica a dirección do texto polo que só pode tomar dous valores:  ltr (left to right) de esquerda a dereita. É o valor por defecto.  rtl (right to left) de dereita a esquerda.																
lang = "codigo"	<div>Especifica o idioma do elemento mediante un código predefinido. Os posibles valores deste atributo atópanse no documento RFC 1766, algúns dos valores posibles son:</div> <table><tr><th>Código</th><th>Idioma</th><th>Código</th><th>Idioma</th></tr><tr><td>en inglés</td><td>(Gran Bretaña)</td><td>é</td><td>Español</td></tr><tr><td>en-US</td><td>Inglés americano</td><td>fr</td><td>Francés</td></tr><tr><td>xa</td><td>Xaponés</td><td>fr-CA</td><td>Francés de Canadá</td></tr></table>	Código	Idioma	Código	Idioma	en inglés	(Gran Bretaña)	é	Español	en-US	Inglés americano	fr	Francés	xa	Xaponés	fr-CA	Francés de Canadá
Código	Idioma	Código	Idioma														
en inglés	(Gran Bretaña)	é	Español														
en-US	Inglés americano	fr	Francés														
xa	Xaponés	fr-CA	Francés de Canadá														
xml:lang = "codigo"	Especifica o idioma do elemento mediante un código definido segundo a recomendación RFC 1766.																

Nas páxinas XHTML, o atributo `xml:lang` ten máis prioridade que `lang` e é obrigatorio incluílo sempre que se inclúe o atributo `lang`.

- **Atributos de eventos e atributos para os elementos que poden obter o foco:** só se utilizan nas páxinas web dinámicas creadas con JavaScript. Como non é o noso obxectivo non o imos contemplar.

## 3.2. Elementos HTML

Un elemento HTML está formado por:

- ✓ Unha etiqueta de apertura.
- ✓ Cero ou máis atributos.
- ✓ Texto encerrado pola etiqueta. É opcional, non todas as etiquetas poden encerrar texto.
- ✓ Unha etiqueta de peche.

Segundo o modo en que ocupan o espazo dispoñible na páxina os elementos poden ser de dous tipos:

- ✓ **Elementos en liña.** Só ocupan o espazo necesario para mostrar os seus contidos. O seu contido pode ser texto ou outros elementos en liña.
- ✓ **Elementos de bloque.** Os elementos de bloque sempre empezan nunha nova liña e ocupan todo o espazo dispoñible ata o final da liña, aínda que os seus contidos non cheguen ata alí. O seu contido pode ser texto, elementos en liña ou outros elementos de bloque.

Existen elementos cuxo comportamento pode ser en liña ou de bloque segundo as circunstancias.

O seguinte exemplo mostra a diferenza entre ambos os comportamentos:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 transitional//EN" "http://www.w3.org/TR/REC-html40/
<html>
  <head>
    <title>Ejemplo de la diferencia entre los elementos en línea y los elementos de blo
  </head>

  <body>
    <h1>Los encabezados son elementos de bloque.</h1>
    <p>Y los párrafos también.</p>
    <a href="http://www.infoalial.com">Los enlaces son elementos de línea</a>
    <p>Incluso si esta definido dentro de un párrafo, un texto en negrita sigue siendo un elemento en línea. </p>
  </body>
</html>
```



Código fonte da imaxe

Ao publicalo nun navegador, por exemplo no Firefox, teríamos:





## Autoavaliación

Os elementos de liña e de bloque diferéncianse en:

- ☐ Os de liña só actúan sobre unha liña de texto e os de bloque actúan sobre máis dunha liña.
- ☐ Os elementos de bloque poden actuar como elementos de liña pero os de liña non poden actuar como elementos de bloque.
- ☐ Os de liña ocupan o espazo imprescindible mentres que os de bloque non.
- ☐ Tanto os de liña como os de bloque ocupan o espazo imprescindible.



## 3.2.1. Elementos da estrutura básica do documento

A estrutura básica dun documento vén determinada polas seguintes etiquetas:

Elemento	Descrición
html	Documento HTML.
head	Cabeceira do documento.
body	Corpo do documento. Permite definir formatos que se aplican aos elementos da páxina de maneira global, como son a cor do fondo do texto, as marxes, a cor das ligazóns...

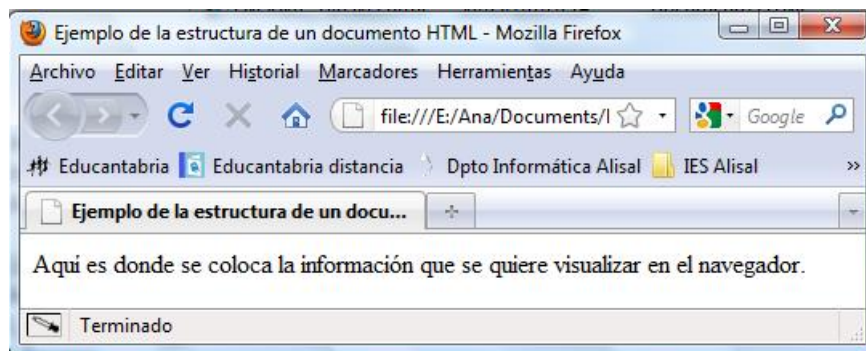
Un exemplo dun documento HTML básico que utiliza estes elementos é:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 transitional//EN" "http://www.w3.org/TR/REC-html40/
<html>
  <head>
    <title>Ejemplo de la estructura de un documento HTML</title>
  </head>
  <body>
    Aquí es donde se coloca la información que se quiere visualizar en el navegador.
  </body>
</html>
```



Código fonte da imaxe

Ao publicalo nun navegador, por exemplo no Firefox, teríamos:



### Autoavaliación

A etiqueta **body** encerra os elementos que van determinar o formato do documento:

- ☐ Verdadeiro.
- ☐ Falso.

### 3.2.2. Elementos da sección de cabeceira

---

- **Elementos colectores:**

Elemento	Descrición
title	Título do documento.
script	Script incrustado. O seu contido ha de ir situado entre as marcas de comentarios xa que non ha de ser interpretado.
style	Estilo aplicado ao documento utilizando CSS. O seu contido ha de ir situado entre as marcas de comentarios xa que non ha de ser interpretado.

- **Elementos non colectores:**

Elemento	Descrición
basee	URI base do documento
isindex	Prompt de entrada de datos.
link	Ligazóns a documentos externos de librerías
meta	Información que axiliza a procura do documento en buscadores.

### 3.2.3. Elementos que lle dan formato ao texto dun parágrafo

Os distintos elementos que podemos utilizar para lle dar formato ao noso texto son:

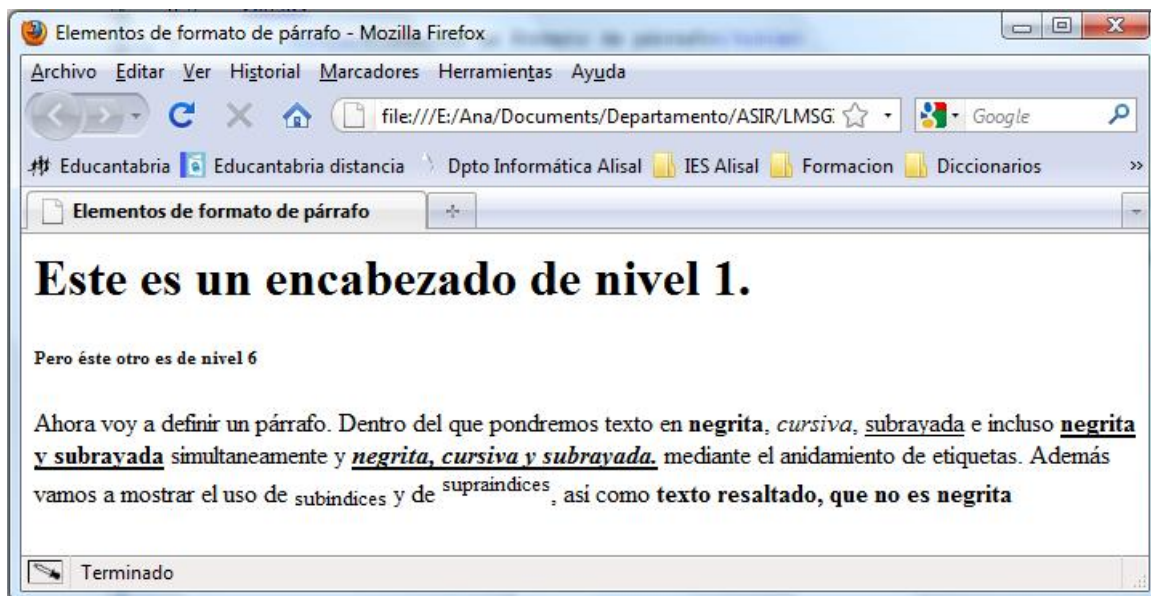
Elemento	Descrición
p	Delimita os parágrafos
h <i>i</i>	Encabezado de nivel <i>i</i> , onde <i>i</i> é un número enteiro entre 1 e 6, ambos inclusive. O tamaño da letra do encabezado é maior canto menor sexa o valor de <i>i</i> . Non deben de usarse estas etiquetas para formatear texto. Só estarán ben usadas para designar títulos de parágrafos.
b	Indica que o texto que está nese elemento se poñerá en negra.
i	Indica que o texto que está nese elemento se poñerá en itálica ou cursiva.
u	Indica que o texto que está nese elemento se poñerá subliñado.
sup	Indica que o texto que está nese elemento é un supraíndice.
sub	Indica que o texto que está nese elemento é un subíndice.
strong	Indica que o texto que está nese elemento estará resaltado. Habitualmente os navegadores resaltan o texto poñéndoo en negra aínda que podería haber algún navegador que resaltase o texto poñéndoo en cursiva e en laranxa.

Un exemplo dun documento HTML que utiliza estes elementos é:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 transitional//EN" "http://www.w3.org/TR/REC-html40/
<html>
  <head>
    <title>Elementos de formato de párrafo</title>
  </head>
  <body>
    <h1>Este es un encabezado de nivel 1.</h1>
    <h6>Pero párrafo este otro es de nivel 6</h6>
    <p>Ahora voy a definir un párrafo. Dentro del que pondremos texto en <b>negrita<
    <i>cursiva</i>, <u>subrayada</u> e incluso <b><u>negrita y subrayada</u></b> simultanea
    <b><i><u>negrita, cursiva y subrayada.</u></i></b> mediante el anidamiento de etiquetas
    Además vamos a mostrar el uso de <sub>subíndice</sub> y de <sup>supraíndice</sup>
    as párrafo como <strong>texto resaltado, que no es negrita</strong>
  </p>
</body>
</html>
```



Código fonte da imaxe



## Autoevaluación

Para poner un párrafo en negra utilizaremos a etiqueta:

- ☐ <b>
- ☐ <strong>
- ☐ Ambas.
- ☐ <n>

### 3.2.4. Elementos de listas

---

Hai tres tipos de listas: ordenadas, desordenadas e listas de definición.

Elemento	Descrición
ul	Delimita os elementos que forman unha lista desordenada
ol	Delimita os elementos que forman unha lista ordenada
li	Indica cada un dos elementos dunha lista
dl	Delimita os elementos que forman unha lista de definición
dt	Cada un dos termos que se definen dunha lista de definición.
dd	Cada unha das definicións dunha lista de definición.

Un exemplo dun documento HTML que mostra a forma de utilizar estes elementos é:

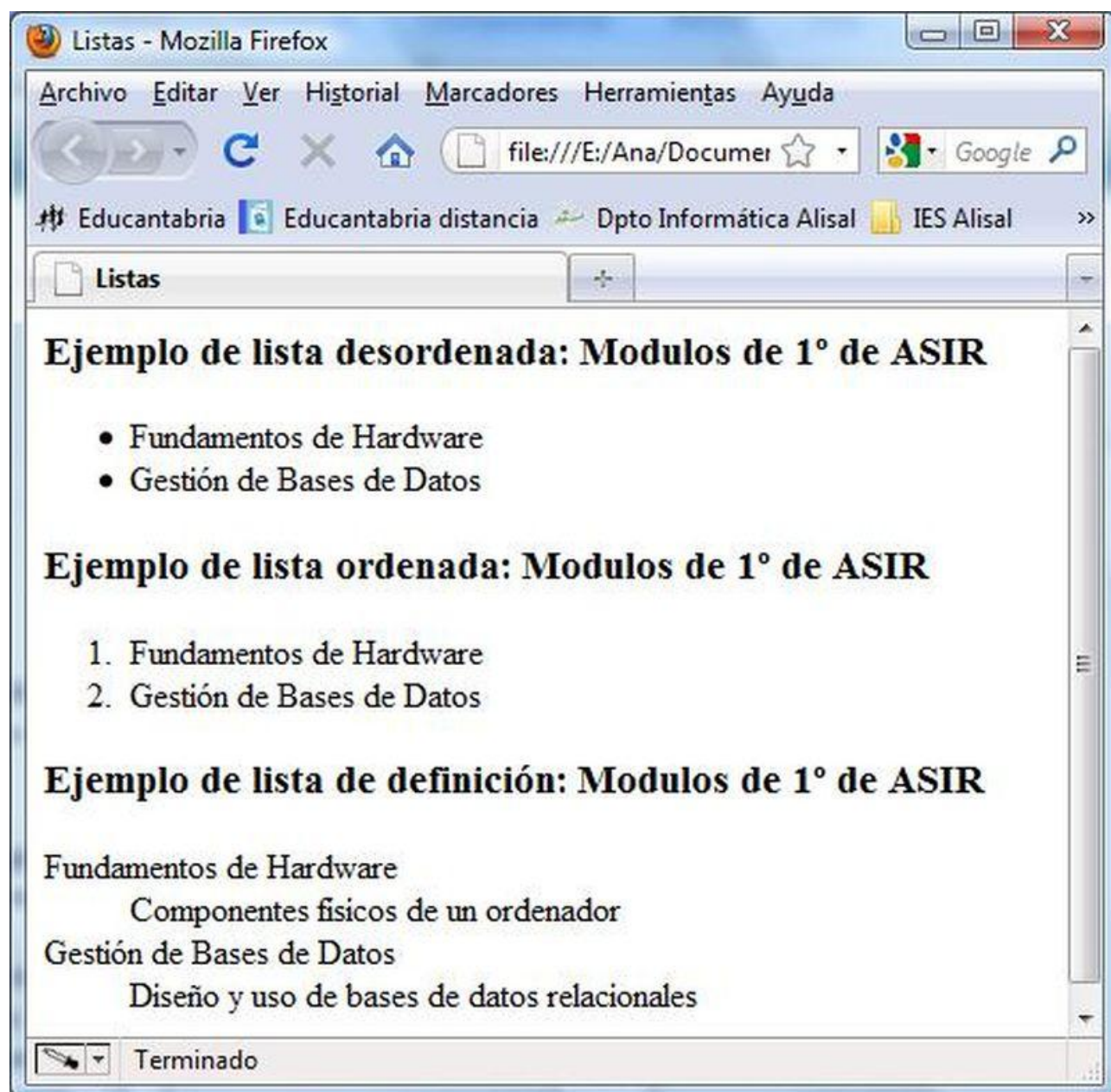
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 transitional//EN" "http://www.w3.org/TR/REC-html40/
<html>
  <head>
    <title>Listas</title>
  </head>
  <body>
    <h3>Ejemplo de lista desordenada: Modulos de 1º de ASIR</h3>
    <ul>
      <li>Fundamentos de Hardware</li>
      <li>Gesti&ocute;n de Bases de Datos</li>
    </ul>

    <h3>Ejemplo de lista ordenada: Modulos de 1º de ASIR</h3>
    <ol>
      <li>Fundamentos de Hardware</li>
      <li>Gesti&ocute;n de Bases de Datos</li>
    </ol>

    <h3>Ejemplo de lista de definici&ocute;n: Modulos de 1º de ASIR</h3>
    <dl>
      <dt>Fundamentos de Hardware</dt>
      <dd>Componentes f&iacute;sicos de un ordenador</dd>
      <dt>Gesti&ocute;n de Bases de Datos</dt>
      <dd>Diseño y uso de bases de datos relacionales</dd>
    </dl>
  </body>
</html>
```



Código fonte da imaxe





### 3.2.5. Elementos de táboas

---

Os elementos para definir unha táboa son os seguintes:

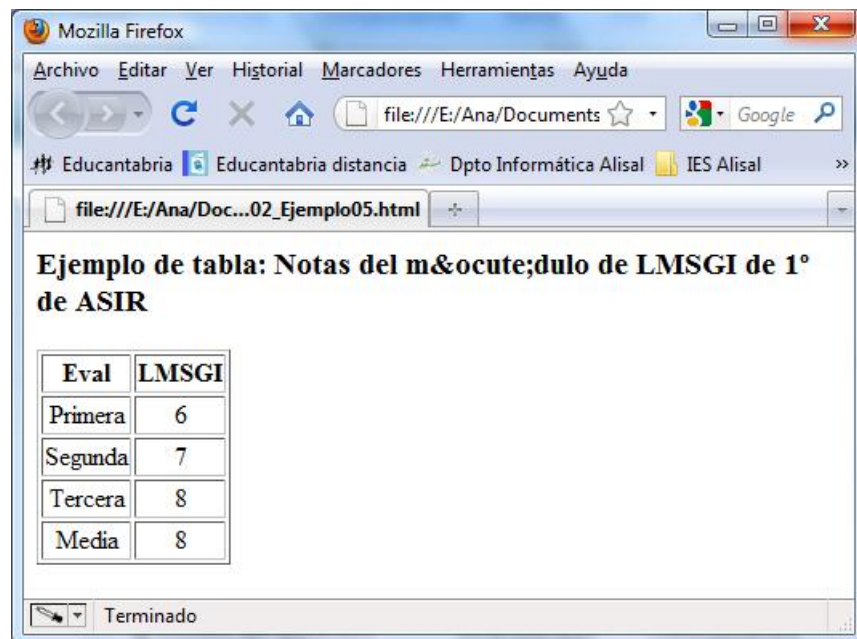
Elemento	Descrición
table	Delimita o contido dunha táboa.
tr	Delimita cada unha das liñas da táboa.
td	Delimita o contido de cada cela da táboa.
colgroup	Permite agrupar columnas.
tbody	Permite agrupar liñas da táboa.
thead	Define a liña cabeceira da táboa.
th	Delimita cada unha das celas da cabeceira
tfoot	Define a fila pé da táboa.

Un exemplo dun documento HTML que mostra o modo de utilizar algúns destes elementos é:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 transitional//EN" "http://www.w3.org/TR/REC-html40/
<html>
  <head>
    <title>Tablas</title>
  </head>
  <body>
    <h3>Ejemplo de tabla: Notas del m&ocute;dulo de LMSGI de 1º de ASIR</h3>
    <table border = "1">
      <thead> <tr> <th>Eval</th> <th>LMSGI</th> </tr> </thead>
      <tfoot align="center"> <tr> <td>Media</td> <td>8</td> </tr> </tfoot>
      <tbody align="center">
        <tr> <td>Primera</td> <td>6</td> </tr>
        <tr> <td>Segunda</td> <td>7</td> </tr>
        <tr> <td>Tercera</td> <td>8</td> </tr>
      </tbody>
    </table>
  </body>
</html>
```




[Código fonte da imaxe](#)





### 3.2.6. Elementos de formularios

O elementos que pode conter un formulario son os seguintes:

Elemento	Descrición
form	Delimita o contido do  formulario
input	Caixa de texto para texto curto. Dependendo do valor que tome o atributo type deste elemento podemos estar ante un texto sen máis, un campo de texto onde ao escribir non se visualice o contido se non que escriba asteriscos, un botón de radio que o usuario poderá elixir, unha opción que o usuario poderá activar, botóns...
textarea	Caixa de texto para texto longo.
select	Crea un menú despregable que permite elixir dunha lista de opcións que contén o elemento.
option	Delimita cada unha das opcións dun menú despregable que o contén.
button	Permite definir un botón. A súa principal vantaxe fronte aos botóns feitos con input é que este elemento permite introducir no botón calquera outro elemento de HTML, por exemplo imaxes.
fieldset	Permite agrupar elementos dun un formulario.
legend	Permite poñer un título ao fieldset.
label	Etiqueta dun campo do formulario.



#### Exercicio resolto

Cal podería ser o código HTML asociado ao documento que se mostra na seguinte imaxe?



## 3.2.7. Elementos de frames

Os elementos utilizados para traballar con frames son:

Elemento	Descrición
frameset	Define a partición da ventá do navegador en marcos. Só se pode partir en filas ou en columnas. Para partir a xanela do navegador en filas e columnas hai que aniñar frames.
Frame	Define un marco que contén información

Un exemplo dun documento HTML que utiliza estes elementos é:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN" "http://www.w3.org/TR/REC-html40/frameset.dtd">
<html>
  <head> <title>Frames</title> </head>
  <frameset rows="20%,30%,*">
    <frame src="LMSGI02_Ejemplo07_1.html" /> <frame src="LMSGI02_Ejemplo03.html" />
    <frame src="LMSGI02_Ejemplo05.html" />
  </frameset>
</html>
```



[Código fonte da imaxe](#)

O contido do ficheiro LMSGI02\_Exemplo07\_1.html é:

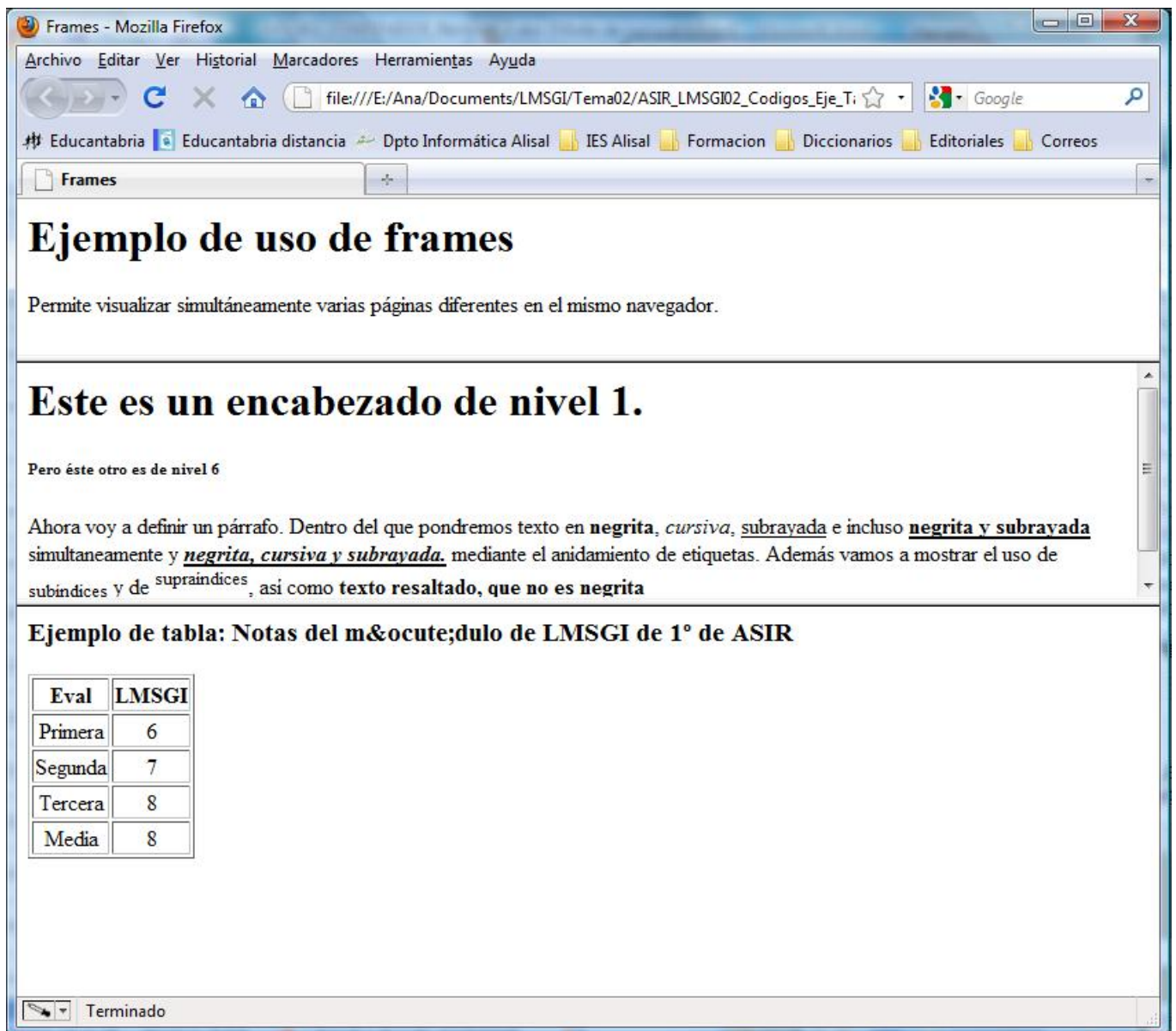
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 transitional//EN" "http://www.w3.org/TR/REC-html40/transition.dtd">
<html>
  <head> <title>Frames_2</title> </head>
  <body>
    <h1>Ejemplo de uso de frames</h1>
    <p>Permite visualizar simultaneamente varias páginas diferentes en el mismo navegador</p>
  </body>
</html>
```



[Código fonte da imaxe](#)

Os ficheiros LMSGI02\_Exemplo03.html e LMSGI02\_Exemplo05.html correspóndense cos códigos de exemplos que vimos anteriormente para mostrar o uso dos formatos de texto e as táboas.

Ao publicalo nun navegador, por exemplo no Firefox, teríamos:



### 3.2.8. Outros elementos

---

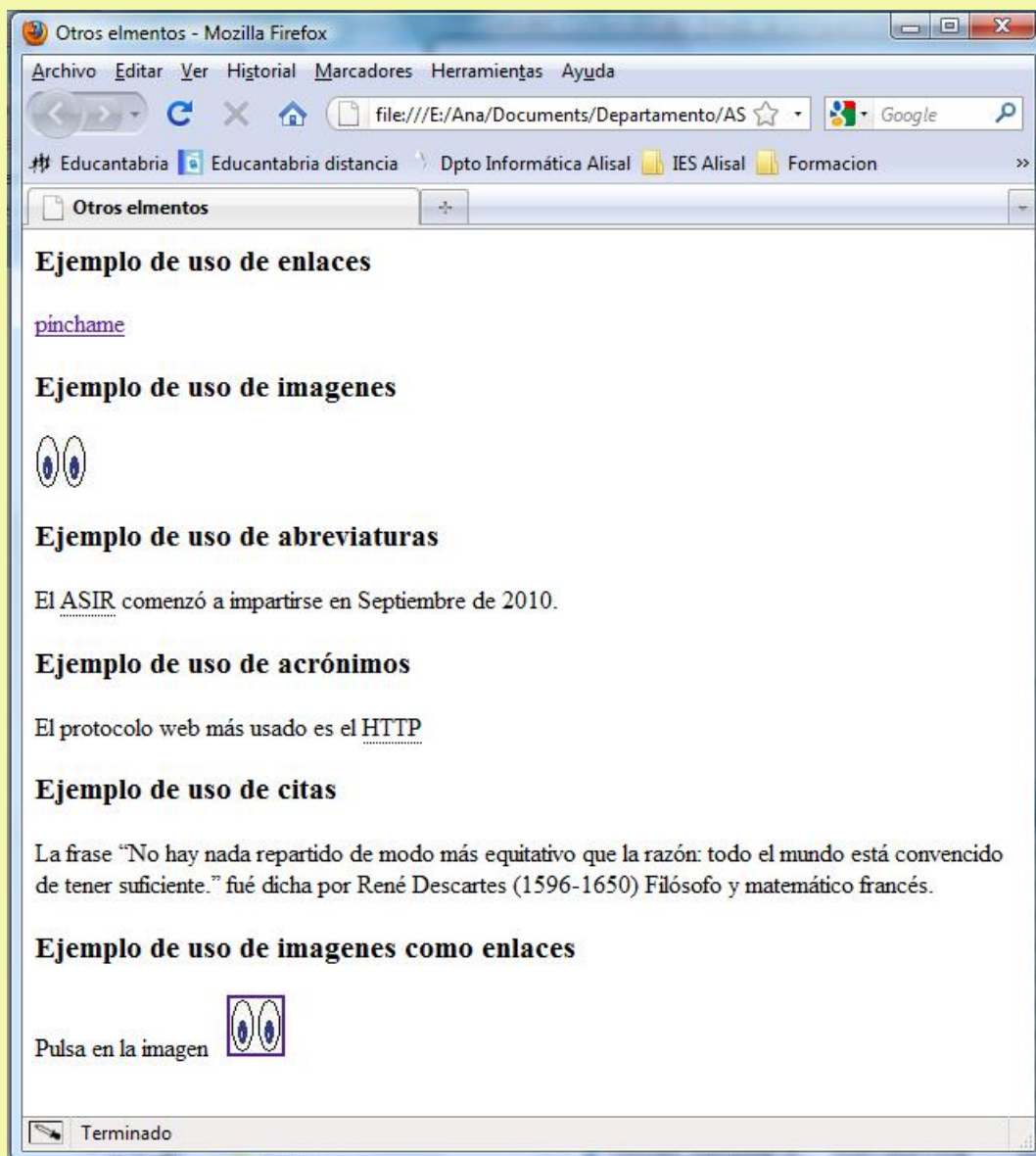
Outros elementos que podemos utilizar son:

Elemento	Descrición
a	Permite definir unha ligazón a unha páxina web, un arquivo ou un enderezo de correo.
img	Permite inserir unha imaxe nunha páxina web. É obrigatorio utilizar o atributo src para determinar o path do ficheiro de imaxe que queremos inserir.
abbr	Indica unha forma abreviada.
acronym	Indica un acrónimo.
blockquote	Contén un bloque de texto con sangría.
q	Contén unha cita polo que o navegador lle engade as marcas de citación
br	Insere unha liña en branco. Non ten etiqueta de peche, só se abre.



#### Exercicio resolto

Cal podería ser o código HTML asociado ao documento que se mostra na imaxe?



## Autoevaluación

Todos os elementos de HTML están formados por:

- ☐ Etiquetas de apertura e peche.
- ☐ Etiquetas de apertura e peche, un só atributo e o seu valor correspondente.
- ☐ Etiquetas de apertura, peche, varios atributos e os seus valores.
- ☐ Etiquetas de apertura, peche, ningún ou varios atributos e os seus valores.



## Debes saber

Deberías coñecer a especificación de HTML 4.01 que fai o W3C, así que te convidamos a que as coñezas:

[Recomendación do W3C "HTML 4.01 Specification"](#)

[Tradución desta recomendación ao castelán](#)

## 4. XHTML fronte a HTML



### Caso práctico

Posto que **Xoán** aceptou realizar a páxina web formúlase facela en HTML ou en XHTML consulta con **Mariña**, traballadora da súa empresa informática.

**Mariña** opina que, desde un punto de vista formal, non hai diferenzas substanciais entre utilizar unha ou outra linguaxe, e seguindo a evolución lóxica parécelle que sería máis apropiado utilizar XHTML e engade que esta linguaxe ten a vantaxe de ser compatible con navegadores antigos.

**Xoán** opina que é unha boa opción, pero que algúns navegadores, a pesar de ser compatibles coa linguaxe non interpretan os formatos.



A linguaxe XHTML é moi similar á linguaxe HTML. De feito, non é máis que unha adaptación de HTML á linguaxe XML, o estándar XHTML 1.0 só engade pequenas melloras e modificacións menores ao estándar HTML 4.01, polo que este último está practicamente incluído no primeiro, o que fai que pasar do HTML 4.01 Strict a XHTML non require case ningún cambio.

A linguaxe HTML ten unha sintaxe moi permisiva, polo que é posible escribir as súas etiquetas e atributos de moitas formas diferentes. As etiquetas, por exemplo, podían escribirse en maiúsculas, en minúsculas e mesmo combinando maiúsculas e minúsculas. O valor dos atributos das etiquetas pódense indicar con ou sen comiñas. Ademais, a orde no que se abrían e pechaban as etiquetas non era importante.

A flexibilidade de HTML dá lugar a páxinas cun código desordenado, difícil de manter e moi pouco profesional.

XHTML soluciona estes problemas engadindo certas normas na forma de escribir as etiquetas e atributos.



### Autoavaliación

A principal diferenza entre HTML e XHTML é:

- ☐ HTML permite usar as etiquetas en maiúsculas e XHTML non.
- ☐ XHTML dá lugar a códigos ordenados.
- ☐ XHTML é compatible con XML.
- ☐ HTML é permisivo.



## 4.1. XHTML: diferenzas sintácticas e estruturais con HTML

O esquema básico do documento, para considerarse conforme á especificación deberá cumprir as seguintes condicións:

- O elemento raíz do documento debe ser `<html>`.
- O elemento raíz do documento debe indicar o espazo nominal XHTML usando o atributo `xmlns`. O espazo nominal para XHTML é `http://www.w3.org/1999/xhtml`
- Debe haber unha declaración DOCTYPE no prólogo do documento. O identificador público incluído na declaración DOCTYPE debe facer referencia a algunha das tres DTD definidas polo W3C usando o Identificador Formal Público correspondente:

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

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

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



### Restricións básicas que introduce XHTML respecto de HTML na sintaxe das súas etiquetas:

- As etiquetas téñense que pechar en orde inversa á que se abren, nunca poden solaparse.
- Os nomes das etiquetas e atributos sempre se escriben en minúsculas.
- O valor dos atributos, incluso os numéricos, sempre se encerra entre comiñas.
- Os atributos nos que o nome coincide co seu valor, non pode darse o valor por entendido, é dicir, non se poden comprimir. Este tipo de atributos non son moi habituais.
- Todas as etiquetas deben pecharse sempre. XHTML permite que en lugar de abrir e pechar de forma consecutiva a etiqueta (`<br><br/>`) se pode utilizar a sintaxe `<br/>` para indicar que é unha etiqueta baleira que se abre e se pecha nese mesmo punto.

### Outras restricións:

Ademais das cinco restricións básicas, XHTML inclúe outros cambios máis avanzados respecto de HTML, entre elas:

- Antes de acceder ao valor dun atributo, elimínanse todos os espazos en branco que se atopan antes e despois do valor. Ademais, elimínanse todos os espazos en branco sobrantes dentro do valor dun atributo.
- O código JavaScript debe encerrarse entre unhas etiquetas especiais (`<![CDATA[ e ]]>`) para evitar que o navegador interprete de forma errónea caracteres como `&` e `<`.
- As páxinas XHTML deben prescindir do atributo `name` no seu lugar, sempre debe utilizarse o atributo `id`.
- XHTML é necesario separar o formato do contido. Os parágrafos deben separarse consistentemente e as cabeceiras `h1-h6` só deben usarse para destacar os diferentes apartados. É recomendable dar o formato aos datos por medio do uso das CSS.



## 4.2. Vantaxes e inconvenientes de XHTML sobre HTML

### • Vantaxes:

- Compatibilidade parcial con navegadores antigos: a información visualízase, aínda que sen formato.
- Un mesmo documento pode adoptar deseños radicalmente distintos en diferentes apartados.
- Sinxeleza á hora de editar e manter o código.
- É compatible cos estándares que está a desenvolver o W3C como recomendación para futuros axentes de usuario ou navegadores.
- Os documentos escritos conforme a XHTML 1.0 presentan mellor rendemento nas actuais ferramentas web que aqueles escritos conforme a HTML.
- A separación dos contidos e a súa presentación fai que os documentos XHTML se adapten mellor ás diferentes plataformas: pantallas de computador, pantallas de dispositivos móbiles...
- Como é XML pódense utilizar facilmente ferramentas creadas para procesar documentos XML xenéricos (editores, XSLT etc.).



### • Inconvenientes:

- Algúns navegadores antigos non son totalmente compatibles cos estándares, o que fai que as páxinas non sempre se mostren correctamente. Isto cada vez é menos problemático xa que estes navegadores van caendo en desuso.
- Moitas ferramentas de deseño web aínda non xeran código XHTML correcto.



### Autoavaliación

A linguaxe XHTML permite:

- ☐ A información contida no documento visualízase aínda que o navegador non interprete o código de formato.
- ☐ A súa funcionalidade presenta un rendemento máis alto que o de HTML.
- ☐ Ao ser unha linguaxe máis nova que HTML é compatible con todas as ferramentas de deseño que permiten traballar con este último.
- ☐ Ningunha resposta é válida.

## 5. Ferramentas de deseño web



### Caso práctico

**Xoán** móstralles a María e **Félix** unha primeira versión da páxina web.

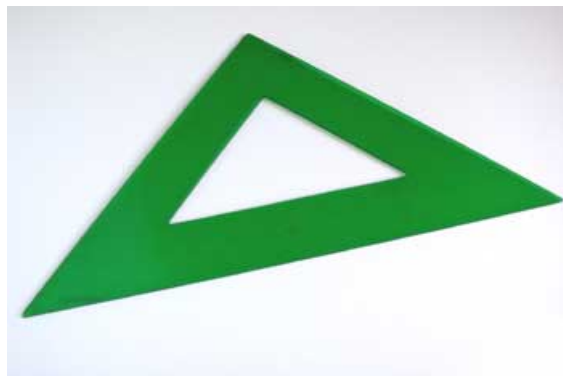
Tras vela **Félix** ten a curiosidade de saber se para codificala hai que utilizar algún software característico ou basta con usar un editor de texto plano, como no caso de XML.

**Xoán** cóntalle que pode bastar o bloc de notas, pero que existen varios editores que facilitan a tarefa.

Actualmente hai ferramentas que permiten deseñar un sitio web sen necesidade de saber programar HTML. As máis populares:

- **Macromedia Dreamweaver.**
- **Microsoft Front Page.**
- **Adobe Go live.**
- **NetObjects Fusion.**
- **Amaya** é unha ferramenta de libre distribución creada polo W3C que permite visualizar e editar páxinas HTML e XHTML con follas de estilo CSS, expresións

👉 MathML e debuxos 👉 SVG; ademais de documentos XML. Está dispoñible para plataformas Windows, GNU/Linux, Mac VOS X, entre outras. A última versión soporta HTML 4.01, XHTML 1.0, XHTML Basic, XHTML 1.1, HTTP 1.1, MathML 2.0, moitas características CSS 2, e inclúe soporte para gráficos SVG.



Ademais das anteriores, para xerar páxinas web, é conveniente ter algunhas das seguintes ferramentas:

- **Software de deseño**
  - *Macromedia Flash* para facer animacións, banners ou sitios enteiros con esta tecnoloxía.
  - *Macromedia Fireworks* ou *Adobe Illustrator* para deseñar botóns, logos, imaxes etc.
  - *Adobe Photoshop* ou *Gimp*, para retocar fotografías e traballar con imaxes.
- **Recursos: deseño web**
  - *My Fonts*, é un sitio web que nos vende fontes que poden ser utilizadas para a web.
  - *Cor Voodoo*, ten interesante información sobre o uso das cores e a súa influencia na web.
  - *moreCrayons*, unha paleta de cores web seguras (Pois non todas as cores se ven iguais en distintos sistemas operativos).
  - *Yafra Cor*, permite crear a gama para partir da cor que se elixa como principal ademais funciona como conversor entre os modelos de cor 👉 RGB e 👉 HSV.
  - *ColorJack*, permite crear unha gama a partir dunha cor e ademais calcula equivalencias de cores entre RGB, HSV e 👉 CMYK.
  - *Kuler*, é unha aplicación en liña de Adobe Labs na que se pode elixir unha combinación de cores e compartila con outros usuarios, que poden votar as súas favoritas. Require rexistro.



### Autoavaliacion

Todas as ferramentas web permiten facer animacións e retocar fotografías:

- ☐ Verdadeiro.
- ☐ Falso.



## Debes saber

A seguinte ligazón permíteche descargar a versión do editor Amaya adecuada para o teu sistema operativo

[Amaya](#)

A seguinte ligazón permíteche descargar a versión do editor html-kit adecuada para o teu sistema operativo

[html-kit](#)



## Para saber máis

Estes sitios ofrecen gratuitamente persoais (templates) de webs que soamente temos que adaptar ás nosas necesidades para realizar a nosa páxina.

[Free web templates](#)

[GUI stuff](#)

## 6. Follas de estilo ou CSS



### Caso práctico

**María** mostra a súa conformidade coa estrutura e contidos da futura web corporativa que **Xoán** realizou, aínda que lle gustaría probar outras gamas de cores e tipos de fonte.

**Félix** expón que, quizais, resulte un traballo demasiado laborioso xa que supoñería modificar todos os ficheiros da web.

**Xoán**, sorrindo, comenta que de feito non é exactamente así. En realidade os datos están separados dos seus formatos.

CSS (Cascading Style Sheets) permítelles aos desenvolvedores web controlar o estilo e o formato de múltiples páxinas web ao mesmo tempo.

Antes do uso de CSS, os deseñadores de páxinas web debían definir o aspecto de cada elemento dentro das etiquetas HTML da páxina. O principal problema desta forma de definir o aspecto dos elementos é que habería que definir o formato de cada un dos elementos que formen a páxina, o cal fai que sexa moi difícil de actualizar.

CSS permite separar os contidos da páxina e o seu aspecto. Para iso defínese nunha zona reservada o formato de cada un dos elementos da web. Calquera cambio no estilo marcado para un elemento na CSS afectará a todas as páxinas vinculadas a ela nas que apareza ese elemento. As follas de estilo están compostas por unha ou máis regras de estilo aplicadas a un documento HTML ou XML.

Ao crear unha páxina web, utilízase en primeiro lugar a linguaxe HTML/XHTML para marcar os contidos, é dicir, para designar a función de cada elemento dentro da páxina: parágrafo, cabeceira, texto destacado etc. Unha vez creados os contidos, utilízase a linguaxe CSS para definir o formato de cada elemento.

CSS obriga a crear documentos semánticos HTML/XHTML, mellora a accesibilidade do documento, reduce a complexidade do seu mantemento e permite visualizar o mesmo documento en infinidade de dispositivos diferentes.

As follas de estilos apareceron pouco despois que a linguaxe de etiquetas SGML, ao redor do ano 1970. Desde a creación de SGML, observouse a necesidade de definir un mecanismo que permitise aplicar estilos aos documentos electrónicos. A guerra de navegadores e a falta dun estándar para a definición dos estilos dificultaban a creación de documentos que tivesen igual aparencia en distintos navegadores.

O organismo W3C propuxo a creación dunha linguaxe de follas de estilos específica para a linguaxe HTML.

En 1995, o W3C engadiu ao seu grupo de traballo de HTML o desenvolvemento e estandarización de CSS.

**CSS 1**, publicouse en 1996, é a primeira recomendación oficial.

**CSS 2**, publicada en 1998, é a segunda recomendación oficial.

**CSS 3**, continúa en desenvolvemento desde 1998.

Actualmente utilízase a versión CSS 2.1, actualizada por última vez o 19 de xullo de 2007.

O deseño web sempre está limitado polas posibilidades dos navegadores que utilizan os usuarios para acceder ás súas páxinas. Por este motivo é imprescindible coñecer o soporte de CSS en cada un dos navegadores máis utilizados do mercado.





## Autoavaliación

As follas de estilo en cadoiro permiten:

- ☐ Definir formatos que se aplican sobre varias páxinas web dun sitio.
- ☐ Separar o formato da estrutura dunha páxina web.
- ☐ Estructurar o contido da páxina web.
- ☐ Ningunha resposta é válida.

## 6.1. Soporte de CSS nos navegadores

---

O soporte de CSS dun navegador vén determinado polo motor deste xa que é este o encargado de interpretar o CSS.

A seguinte táboa mostra o soporte de CSS 1, **CSS 2.1** e **CSS 3** dos cinco navegadores máis utilizados na actualidade:

Navegador	Motor	CSS 1	CSS 2.1	CSS 3
Safari	WebKit	Completo	Case completo	Parcial
Opera	Presto	Completo	Case completo	Parcial
Firefox	Gecko	Completo	Case completo	Parcial
Google Chrome	WebKit	Completo	Case completo	Parcial
Internet Explorer	Trident	Completo, desde a versión 6.0	Completo, desde a versión 8.0	Practicamente nulo



## 6.2. Como incluír CSS nun documento HTML ou XHTML

Existen tres opcións para incluír CSS nun documento HTML ou XHTML:

### 1. Definir CSS nun arquivo externo.

Neste caso, todos os estilos CSS inclúense nun, ou varios, arquivos de texto plano, cuxa extensión é .css, que as páxinas HTML enlazan mediante o elemento <link> da cabeceira do ficheiro HTML.

Posto que unha páxina web pode ter asociados varios ficheiros CSS é recomendable agrupar estes últimos nun directorio.

O navegador descarga os arquivos CSS externos, ademais da páxina web asociada a eles, e aplica os estilos aos contidos da páxina antes de mostrar os seus contidos.

Esta é a forma de incluír CSS nas páxinas HTML máis utilizada. A principal vantaxe é que se pode incluír un mesmo arquivo CSS en multitude de páxinas HTML, polo que se garante a aplicación homoxénea dos mesmos estilos a todas as páxinas que forman un sitio web.

Ademais, o mantemento do sitio web simplifícase ao máximo, xa que o cambio nun só arquivo CSS permite variar de forma instantánea os estilos de todas as páxinas HTML asociadas.

Pode facerse de dous modos diferentes:

1. *Mediante ligazóns.*
2. *Importando o ficheiro CSS.*

### 2. Incluír CSS no documento HTML.

Este método emprégase cando se definen poucos estilos ou cando se queren incluír estilos específicos nunha determinada páxina HTML que completen os estilos globais de todas as páxinas do sitio web.

Ten o inconveniente de que para modificar os estilos definidos, é necesario modificar todas as páxinas que inclúen o estilo que se vai cambiar.

### 3. Incluír CSS nos elementos HTML.

O último método para incluír estilos CSS en documentos HTML é o peor e o menos utilizado, xa que para modificar un formato hai que cambiar todos os elementos que estean asociados a el.

Soamente se utiliza en determinadas situacións nas que se debe incluír un estilo moi específico para un só elemento concreto.



## Autoavaliación

O mellor modo de aplicar formatos a unha páxina web é:

- ☐ Definindo os formatos directamente a través dos atributos dos elementos HTML.
- ☐ Incluindo o formato CSS nos elementos de HTML.
- ☐ Definindo os estilos na cabeceira do documento HTML.
- ☐ Definindo un ficheiro CSS externo.



## 6.2.1. Definir CSS nun arquivo externo enlazado

Para realizar unha páxina web usando un arquivo CSS externo, débense seguir os tres pasos seguintes:

- Créase un arquivo de texto plano coas definicións dos formatos.
- O dito arquivo de texto gárdase con extensión .css
- Enlázase o arquivo CSS externo mediante a etiqueta <link> na cabeceira da páxina web.

O elemento <link> pode ter definidos catro atributos cando se enlaza un arquivo CSS:

- rel, indica o tipo de relación que ten o arquivo enlazado e a páxina HTML. Para os arquivos CSS, sempre se utiliza o valor stylesheet
- type, indica o tipo de recurso enlazado. Para os arquivos CSS o seu valor sempre é text/css
- href, indica a URL do arquivo CSS que contén os estilos. Pode ser relativa ou absoluta e pode referenciar a un recurso interno ou externo ao sitio web.
- media, indica o medio no que se van aplicar os estilos do arquivo CSS.

Un exemplo do uso de arquivos externos CSS enlazados para a construción de páxinas web é:

O arquivo formatos.css contén:

```
h3 { color: green; }
p { color: orange; font-family: Verdana; }
```



Código fonte da imaxe

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

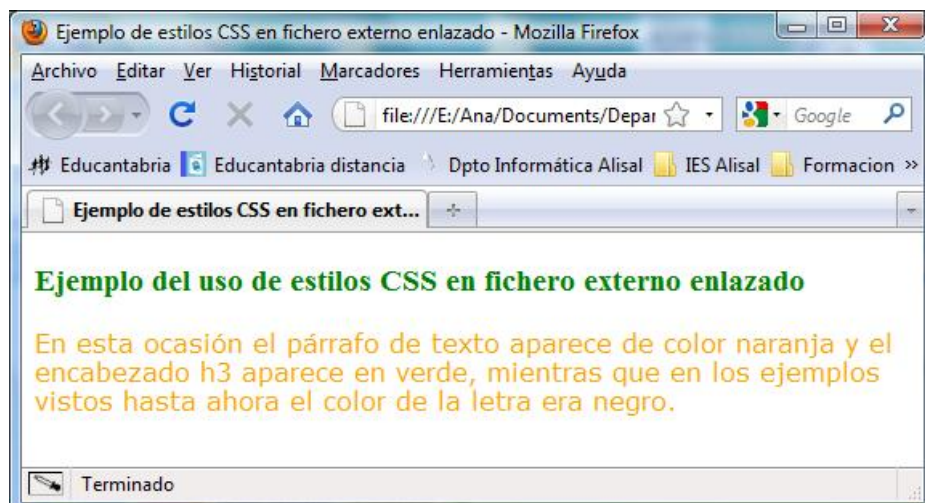
  <head>
    <title>Ejemplo de estilos CSS en fichero externo enlazado</title>
    <link rel="stylesheet" type="text/css" href="formatos.css" />
  </head>

  <body>
    <h3>Ejemplo del uso de estilos CSS en fichero externo enlazado</h3>
    <p>En esta ocasi&ocute;n el pá&aacute;rrofo de texto aparece de color naranja y el enca
aparece en verde, mientras que en los ejemplos vistos hasta ahora el color de la letra era negr
  </body>
</html>
```



Código fonte da imaxe

Ao publicalo nun navegador, por exemplo no Firefox, teriamos:





## 6.2.2. Definir CSS nun arquivo externo importado

Pódese obter o mesmo resultado anterior utilizando o elemento <style> en lugar de link <>.

Neste caso, úsase unha regra de tipo @import seguida dunha cadea de texto encerrada con comiñas simples ou dobres que se corresponde coa URL do arquivo CSS, ou de url() e contén a dita cadea entre o paréntese. As seguintes regras @import son equivalentes para un ficheiro formatos.css que está no directorio css.:

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

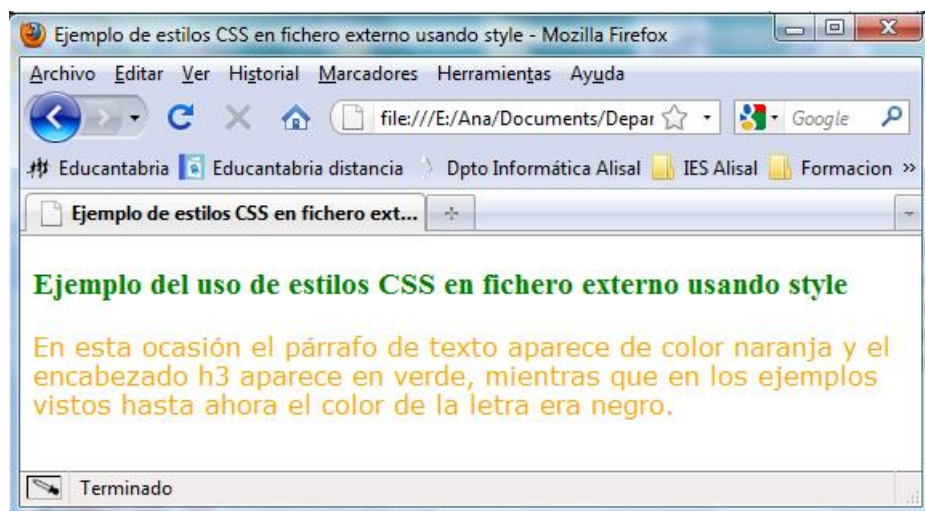
O exemplo anterior quedaría:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD  
transitional.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml">  
  
  <head>  
    <title>Ejemplo de estilos CSS en fichero externo usando style</title>  
    <style type="text/css">  
      @import 'formatos.css';  
    </style>  
  </head>  
  
  <body>  
    <h3>Ejemplo del uso de estilos CSS en fichero externo usando style</h3>  
    <p>En esta ocasi&ocuten el párrafo de texto aparece de color naranja y el encabezado h3 aparece en verde, mientras que en los ejemplos vistos hasta ahora el color de la letra era negro.</p>  
  </body>  
</html>
```



Código fonte da imaxe

Ao publicalo nun navegador, por exemplo no Firefox, teríamos:



## 6.2.3. Definir CSS no documento HTML

Neste caso os formatos dos elementos defínense na cabeceira do documento HTML, dentro do elemento <style>.

Un exemplo dun documento XHTML no que se utiliza este método para incluír formatos é:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

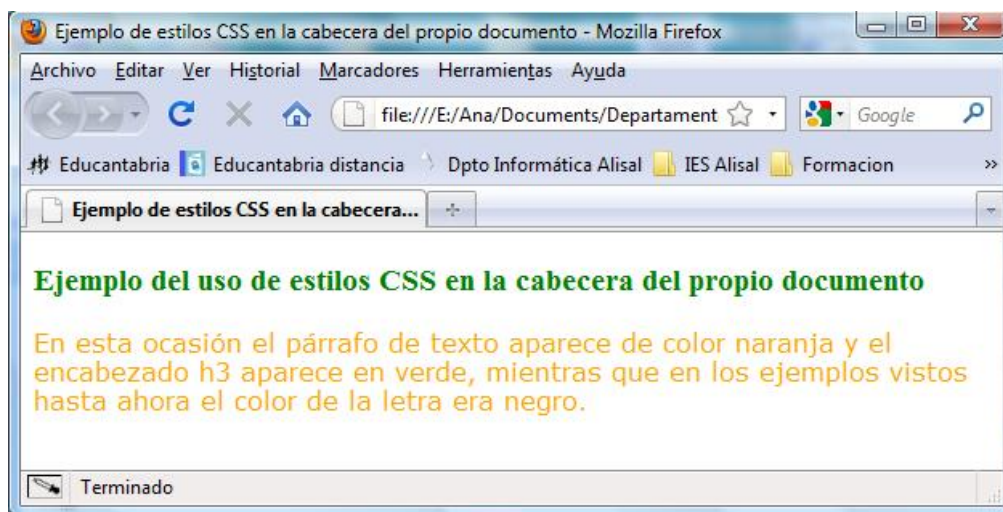
  <head>
    <title>Ejemplo de estilos CSS en la cabecera del propio documento</title>
    <style type="text/css">
      h3 { color: green; font-family: Times; }
      p { color: orange; font-family: Verdana; }
    </style>
  </head>

  <body>
    <h3>Ejemplo del uso de estilos CSS en la cabecera del propio documento</h3>
    <p>En esta ocasi&ocute;n el p&aacute;rrafo de texto aparece de color naranja y el enca
aparece en verde, mientras que en los ejemplos vistos hasta ahora el color de la letra era negr
    </body>
</html>
```



Código fonte da imaxe

Ao publicalo nun navegador, por exemplo no Firefox, teriamos:



## 6.2.4. Incluir CSS nos elementos HTML

Un exemplo dun documento XHTML no que se utiliza este método para incluír formatos é:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

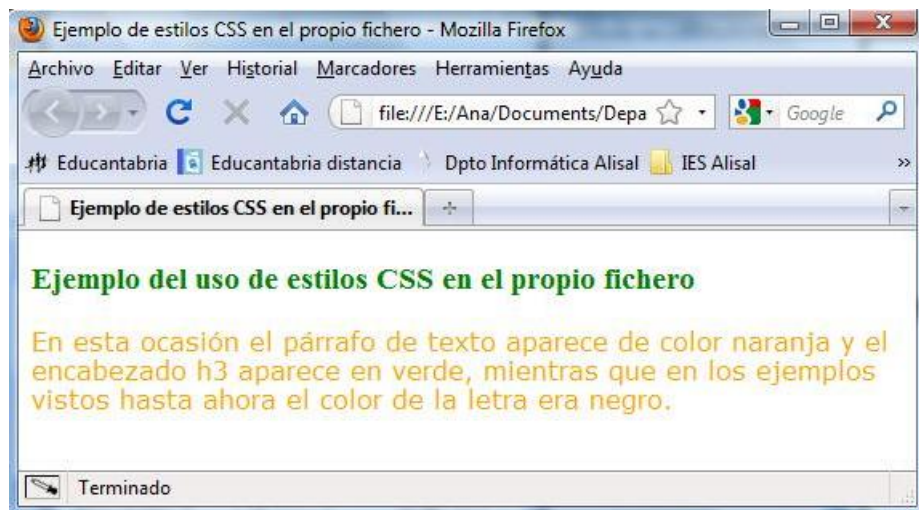
  <head>
    <title>Ejemplo de estilos CSS en el propio fichero</title>
  </head>

  <body>
    <h3 style="color: green;">Ejemplo del uso de estilos CSS en el propio fichero</h3>
    <p style="color: orange; font-family: Verdana;">En esta ocasi&ocute;n el p&aacute;rrafo aparece de color naranja y el encabezado h3 aparece en verde, mientras que en los ejemplos vist
    el color de la letra era negro.</p>
  </body>
</html>
```



Código fonte da imaxe

Ao publicalo nun navegador, por exemplo no Firefox, teriamos:



## 6.3. Sintaxe das regras de estilo

Cada un dos estilos que compoñen unha folla de estilos CSS denomínase regra. Cada regra fórmase por:

- Selector: indica o elemento ou elementos HTML aos que se aplica a regra CSS
- Chave de apertura, {
- Declaración: especifica os estilos que se aplican aos elementos.
  - Propiedade: permite modificar o aspecto dun atributo do elemento.
  - Valor: indica o novo valor do atributo modificado no elemento.
- Chave de peche, }.

Exemplo: `p{ cor : blue; }`

Neste caso o selector é "p", a declaración é: "cor : blue" e, dentro desta, podemos diferenciar a propiedade "cor" e o valor "blue".

Un arquivo CSS pode conter infinitas regras CSS, cada regra pode conter varios selectores e cada declaración pode estar formada por diferentes declaracións.

+	x	+	=	+
-	x	-	=	+
+	x	-	=	-
-	x	+	=	-
+	:	+	=	+
-	:	-	=	+
+	:	-	=	-
-	:	+	=	-



### Autoavaliación

O elemento HTML sobre o que se aplica un estilo especificase:

- ☐ En etiquetas, é dicir, entre < e >.
- ☐ Entre paréntese.
- ☐ Entre chaves.
- ☐ Non vai encerrado entre signos.

## 6.4.- Atributos principais.

---

Nos seguintes subapartados imos ver os atributos principais que se usan en CSS como son:

- ✔ Atributos de cor e fondo.
- ✔ Atributos de fonte.
- ✔ Atributos de texto.
- ✔ Atributos de caixa.
- ✔ Atributos de clasificación.

Pasemos a velos detidamente.





## 6.4.1. Atributos de cor e fondo

Os atributos de cor e fondo son os que enumeramos a continuación:

Elemento	Descrición
cor	Indica a cor do texto. Admíteno case todas as etiquetas de HTML. O valor deste atributo é unha cor, co seu nome ou o seu valor RGB.
background-cor	Indica a cor de fondo do elemento. O valor deste atributo é unha cor, co seu nome ou o seu valor RGB.
background-image	Permite colocar unha imaxe de fondo do elemento. O valor que toma é o nome da imaxe co seu camiño relativo ou absoluto
background-repeat	Indica se ha de repetirse a imaxe de fondo e, nese caso, se debe ser horizontal ou verticalmente. Os valores que pode tomar son: repeat-x, repeat-e ou non-repeat.
background-attachment	Especifica se a imaxe ha de permanecer fixa ou realizar un scroll. Os valores que poden tomar son: scroll ou fixed.
background-position	É unha medida, porcentaxe ou o posicionamento vertical ou horizontal cos valores establecidos que serve para situar unha imaxe. Os valores que pode tomar son: porcentaxe, tamaño, ou [top, center, bottom] [left, center, right]
background	Establece nun só paso calquera das propiedades de background anteriores. Os valores que pode tomar son: background-cor, background-image, background-repeat, background-attachment, background-position.

Dado que non todos os nomes de cores son admitidos no estándar, é aconsellable utilizar o valor RGB.

Un exemplo dun documento XHTML no que se utiliza este método para incluír formatos é:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

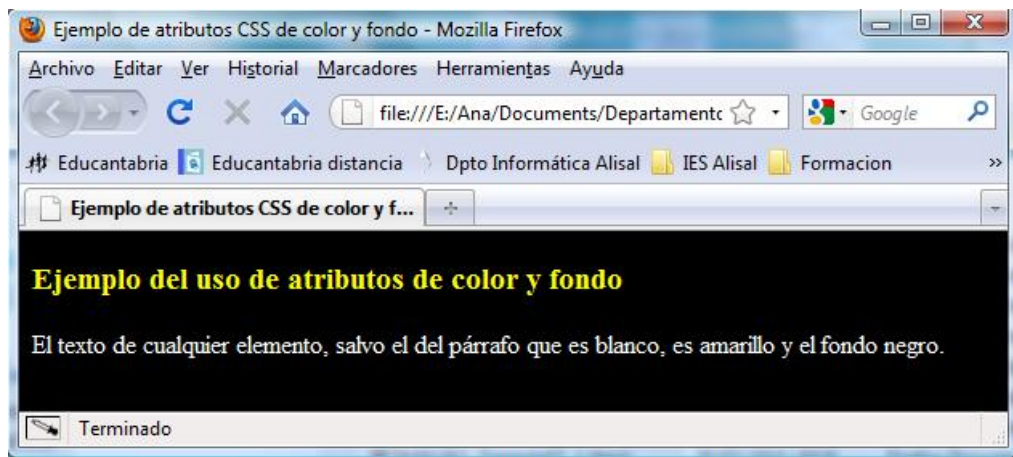
  <head>
    <title>Ejemplo de atributos CSS de color y fondo</title>
    <style type="text/css">
      body { background-color: black; color:yellow; }
      p { color: #ffffff; }
    </style>
  </head>

  <body>
    <h3>Ejemplo del uso de atributos de color y fondo</h3>
    <p>El texto de cualquier elemento, salvo el del p<b>rafo que es blanco, es amaril
negro.</p>
  </body>
</html>
```



[Código fonte da imaxe](#)

Ao publicalo nun navegador, por exemplo no Firefox, teríamos:



## 6.4.2. Atributos de fonte

Neste apartado imos ver os distintos atributos que podemos utilizar referentes ás fontes do noso documento e que son:

Elemento	Descrición
font-size	Indica o tamaño da fonte. Pode ser un tamaño absoluto, relativo ou en porcentaxe. Toma valores de unidades de CSS
font-family	Establece a familia á que pertence a fonte. Se o nome dunha fonte ten espazos utilízanse comiñas para que se entenda ben. O valor é o nome da familia fonte.
font-weight	Define o grosor dos caracteres. Os valores que pode tomar son: normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800 ou 900
font-style	Determina se a fonte é normal ou cursiva. O estilo oblique é similar ao cursiva. Os valores posibles son: normal, italic, oblique.
font-variant	Determina se a fonte é normal ou maiúsculas pequenas. Os valores que pode tomar son: normal ou small-caps
line-height	O alto dunha liña e polo tanto, o espaciado entre liñas. É unha desas características que non se podían modificar utilizando HTML.
font	Permite establecer todas as propiedades anteriores na orde que se indica a continuación: font-style, font-variant, font-weight, font-size[line-height], font family. Os valores han de estar separados por espazos. Non é obrigatorio o uso de todos os valores.

Un exemplo dun documento XHTML no que se utiliza este método para incluír formatos é:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

  <head>
    <title>Ejemplo de atributos CSS de fuente</title>
    <style type="text/css">
      body { background-color: black; color:yellow; font-family: courier }
      p { color: #ffffff; font:italic 900 12px Verdana; }
    </style>
  </head>

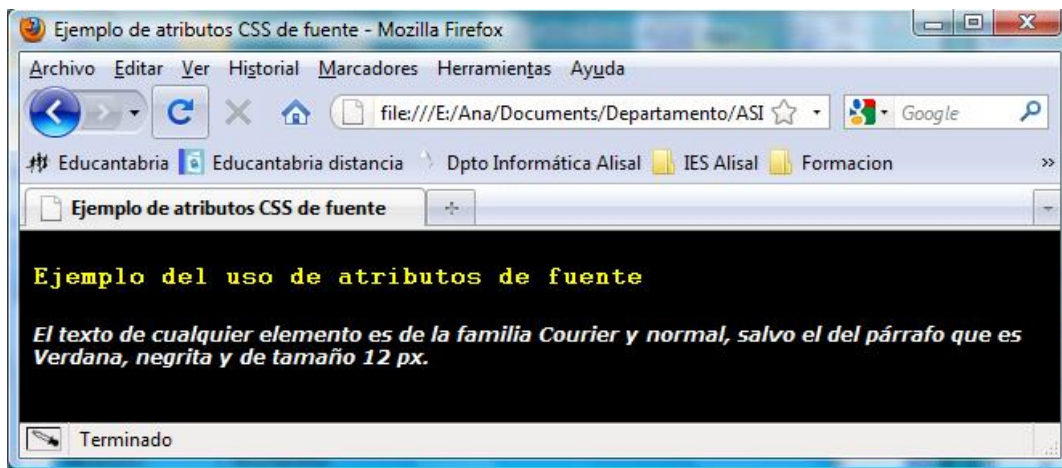
  <body>
    <h3>Ejemplo del uso de atributos de fuente</h3>
    <p>El texto de cualquier elemento es de la familia Courier y normal, salvo el del p<b>aac
Verdana, negrita y de tamaño 12 px.</p>
  </body>
</html>
```



Código fonte da imaxe

Ao publicalo nun navegador, por exemplo no Firefox, teriamos:





### 6.4.3. Atributos de texto

No apartado anterior vimos os atributos relacionados coas fontes e neste imos ver os relacionados co texto en si e son os seguintes:

Elemento	Descrición
text-decoration	Establece se o texto está subliñado, sobreraiado ou tachado. os valores que pode tomar son: none, underline, overline, line-through ou blink.
text-align	Indica a aliñación do texto. Aínda que as follas de estilo permiten o xustifico de texto non funciona en todos os sistemas. Os valores que pode tomar son: left, right, center ou justify.
text-indent	Determina a tabulación do texto. Os valores que toma son unha lonxitude, en unidades CSS, ou unha porcentaxe da establecida.
text-transform	Permítenos transformar o texto, facendo que teña a primeira letra en maiúsculas de todas as palabras, todo en maiúsculas ou minúsculas. Os valores que pode tomar son: capitalize, uppercase, lowercase ou none.
word-spacing	Determina o espaciado entre as palabras. Os valores que pode tomar é un tamaño.
letter-spacing	Determina o espaciado entre letras. Os valores que pode tomar é un tamaño.
vertical-align	Establece a aliñación vertical do texto. Os seus valores posibles son: baseline, sub, super, top, text-top, middle, bottom, text-bottom ou unha porcentaxe.
line-height	Altura da liña. Pode establecerse mediante un tamaño ou unha porcentaxe.

Un exemplo dun documento XHTML no que se utiliza este método para incluír formatos é:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

  <head>
    <title>Ejemplo de atributos CSS de texto</title>
    <style type="text/css">
      h3 { text-decoration:underline; text-align: center; text-transform: capitalize }
      p { text-indent: 50%; }
    </style>
  </head>

  <body>
    <h3>Ejemplo del uso de atributos de texto</h3>
    <p>El texto de del encabezado de tercer nivel est&aacute; subrayado, centrado y la prim
cada palabra es may&uacute;cula.</p>
    <p>El p&aacute;rrafo est&aacute; tabulado</p>
  </body>
</html>
```



[Código fonte da imaxe](#)

Ao publicalo nun navegador, por exemplo no Firefox, teríamos:



## 6.4.4. Atributos de caixa

Agora imos ver outros atributos moi importantes e que utilizaremos moi a miúdo e que non son nin máis nin menos que os atributos de caixa:

Elemento	Descrición
margin-left	Indica o tamaño da marxe esquerda. Pode usarse unha lonxitude, en unidades CSS, ou unha porcentaxe.
margin-right	Indica o tamaño da marxe dereita. Pode usarse unha lonxitude, en unidades CSS, ou unha porcentaxe.
margin-top	Indica o tamaño da marxe superior. Pode usarse unha lonxitude, en unidades CSS, ou unha porcentaxe.
margin-bottom	Indica o tamaño da marxe inferior. Pode usarse unha lonxitude, en unidades CSS, ou unha porcentaxe.
margin	Permite establecer as marxes dunha vez. Hai que seguir a orde: superior, dereito, inferior e esquerdo.
padding-left	Indica o espazo esquerdo entre o bordo e o contido. Pode usarse unha lonxitude, en unidades CSS, ou unha porcentaxe.
padding-right	Indica o espazo dereito entre o bordo e o contido. Pode usarse unha lonxitude, en unidades CSS, ou unha porcentaxe.
padding-top	Indica o espazo superior entre o bordo e o contido. Pode usarse unha lonxitude, en unidades CSS, ou unha porcentaxe.
padding-bottom	Indica o espazo inferior entre o bordo e o contido. Pode usarse unha lonxitude, en unidades CSS, ou unha porcentaxe.
padding	Establece o espazo entre os bordos e o contido dunha soa vez. Hai que respectar a orde superior, dereito, inferior e esquerdo.
border-left-color	Establece a cor do bordo esquerdo do elemento. O seu valor é unha cor RGB ou o nome da cor.
border-right-color	Establece a cor do bordo dereito do elemento. O seu valor é unha cor RGB ou o nome da cor.
border-top-color	Establece a cor do bordo superior do elemento. O seu valor é unha cor RGB ou o nome da cor.
border-bottom-color	Establece a cor do bordo inferior do elemento. O seu valor é unha cor RGB ou o nome da cor.
border-color	Establece a cor dos bordos do elemento dunha soa vez. Hai que seguir a orde superior, dereito, inferior e esquerdo. O seu valor é unha cor RGB ou o nome da cor.
border-style	Establece o estilo do bordo, os valores significan: none=ningun bordo, dotted=punteado (non funciona sempre), solid=adoitado, double=dobre bordo, os valores groove, ridge, inset e outset son bordos con varios efectos 3D.
border-left-width	Grosor do bordo esquerdo. Os seus valores posibles son: thin, médium, thick ou un tamaño.
border-right-width	Grosor do bordo dereito. Os seus valores posibles son: thin, médium, thick ou un tamaño.

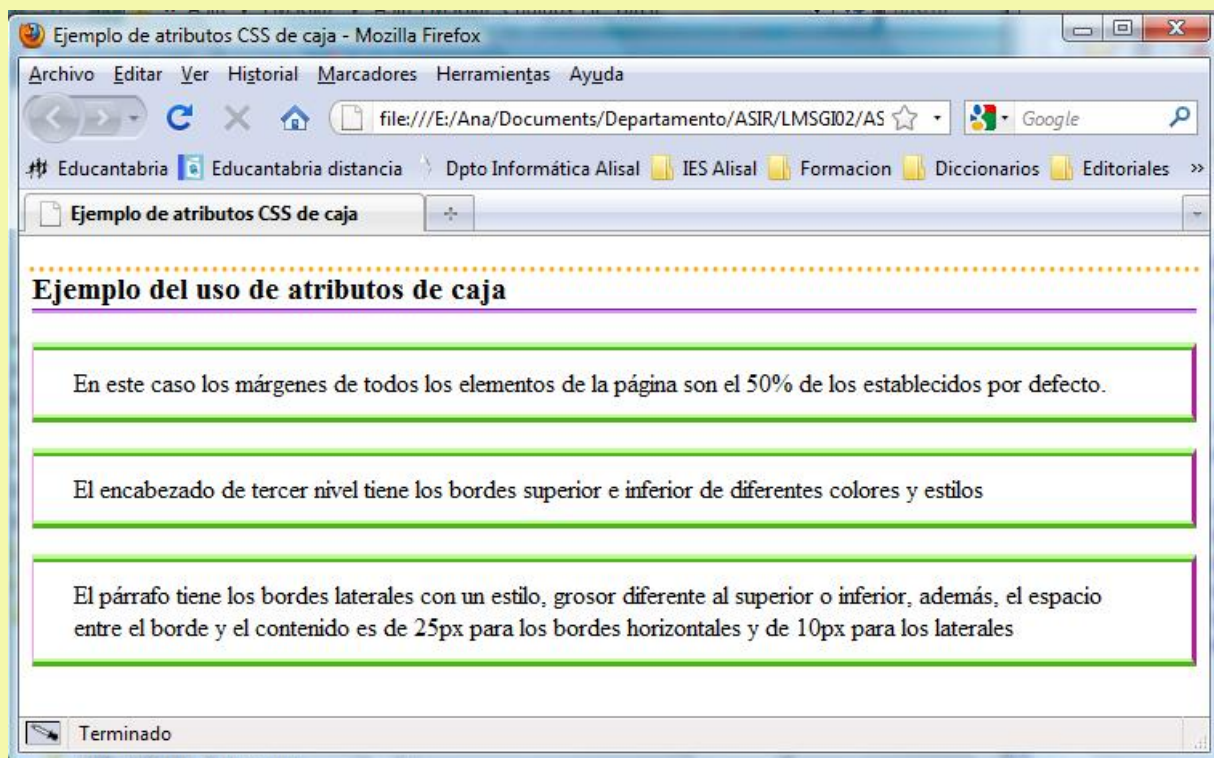
width	
border-top-width	Grosor do bordo superior. Os seus valores posibles son: thin, médium, thick ou un tamaño.
border-bottom-width	Grosor do bordo inferior. Os seus valores posibles son: thin, médium, thick ou un tamaño.
border-width	Establece o tamaño dos bordos do elemento ao que o aplicamos. Hai que seguir a orde superior, dereito, inferior, esquerdo.
width	Establece o ancho do contido do elemento. O valor é unha porcentaxe ou un tamaño.
height	Establece a altura do contido do elemento. O valor é unha porcentaxe ou un tamaño.
float	Serve para aliñar un elemento á esquerda ou a dereita facendo que o texto se agrupe ao redor do dito elemento. Toma os valores none, left ou right.
clear	Establece se un elemento ten á súa altura imaxes ou outros elementos aliñados á dereita ou á esquerda. Os seus valores posibles son: none, left, right ou both.

Para que practiques todo o aprendido recoméndoche que tentes facer o exercicio que se propón a continuación, antes de ver a súa solución. Ánimo!



## Exercicio resolto

Cal sería un posible código XHTML para o documento da imaxe que se mostra a continuación?



## Autoavaliación

**Para modificar o tamaño do bordo esquerdo dunha caixa ou utilizar o atributo:**

- ☐ border-left-width
- ☐ border-width
- ☐ Ambas.
- ☐ Ningunha.

## 6.4.5. Atributos de clasificación

Neste apartado imos ver outros atributos que etiquetamos como atributos de clasificación e que son os seguintes:

Elemento	Descrición
display	Determina se o elemento é de bloque, liña, lista ou ningún deles. Os valores que pode tomar son: block, inline, list-item ou none.
white-space	Indica o modo en que se han de xestionar os espazos en branco que hai no elemento, é dicir, se se manteñen todos os existentes tal e como estean no documento ou se se anulan a un as secuencias de brancos, é o valor por defecto e o da opción normal. Valores que pode tomar son: pre, nowrap, normal.
list-style-type	Indica cal é o símbolo que se utiliza como marcador nas listas. Valores que pode tomar son: disc, circle, square, decimal, lower-roman, upper-roman, lower-alpha, upper-alpha, none.
list-style-image	Permite utilizar o uso dunha imaxe como marcador nunha lista. O valor que toma é a ruta do ficheiro imaxe
list-style-position	Determinan a posición do marcador nunha lista. Pode tomar os valores: outside ou inside.
list-style	Permite establecer dunha única vez todas as características dunha lista. Hai que seguir a orde seguinte: list-style-type, list-style-position e list-style-image.

Un exemplo dun documento XHTML no que se utiliza este método para incluír formatos é:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTDtransitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

  <head>
    <title>Ejemplo de atributos CSS de clasificaci&oaacute;n</title>
    <style type="text/css">
      h3 { white-space:pre;    }
      p { white-space:normal;  }
      ul { list-style:square; list-style-position: outside;}
    </style>
  </head>

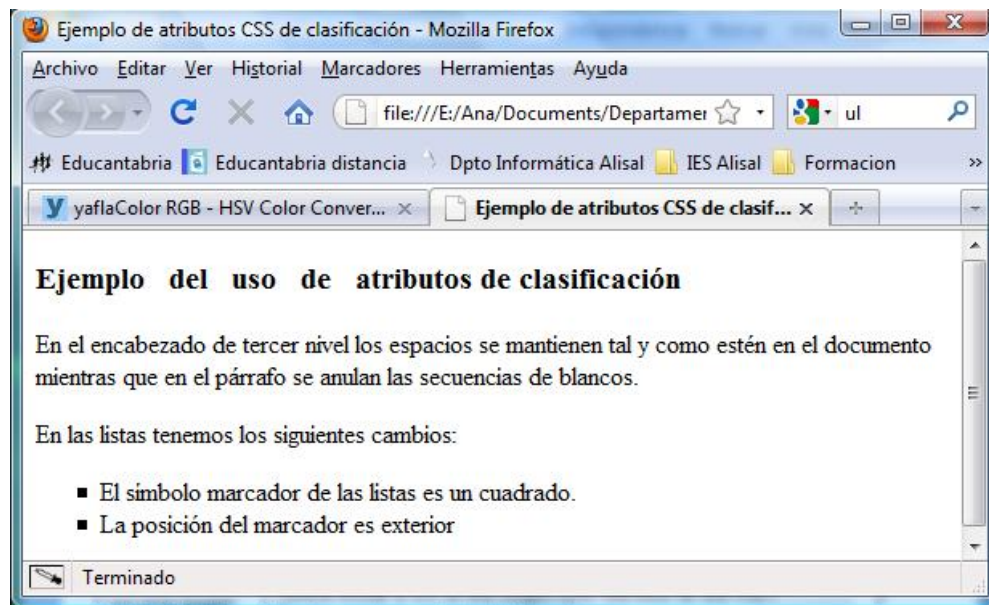
  <body>
    <h3>Ejemplo del uso de atributos de clasificaci&oaacute;n</h3>
    <p>En el encabezado de tercer nivel los espacios se mantienen tal y c
    en el documento mientras que en el p&aacute;rrafo se anulan las secuencias de blancos.</p>
    <p> En las listas tenemos los siguientes cambios:</p>
    <ul>
      <li>El s&iacute;mbolo marcador de las listas es un cuadrado.</li>
      <li>La posici&oaacute;n del marcador es exterior</li>
    </ul>
  </body>
</html>
```



[Código fonte da imaxe](#)

Ao publicalo nun navegador, por exemplo no Firefox, teriamos:





## 6.5. CSS de posicionamento

É un engadido a CSS que permite determinar o modo en que se ha de colocar un determinado elemento. As propiedades definidas en CSS-P son as seguintes:

Elemento	Descrición
clip	Permite seleccionar unha zona. Os valores que pode tomar son: shape ou auto.
height	Permite establecer a altura dun elemento. Os valores que pode tomar son: auto ou un tamaño.
width	Permite establecer a anchura dun elemento. Os valores que pode tomar son: auto ou un tamaño ou porcentaxe.
visibility	Indica se o elemento sobre o que actúa será visible ou non. Os valores que pode tomar son:
left	Indica a posición ao lado esquerdo do elemento. Os valores que pode tomar son: auto ou un tamaño ou porcentaxe.
top	Indica a posición ao lado superior do elemento. Os valores que pode tomar son: auto ou un tamaño ou porcentaxe.
overflow	Indica se o elemento será visible ou non en caso de superar os límites do colector. Os valores que poden tomar son: visible, hidden, scroll ou auto.
position	Determinan se o posicionamento dun elemento é absoluto, relativo ou estático. Os valores que pode tomar son: absolute, relative ou static.
z-index	Define a posición do elemento no terceiro eixo de coordenadas e permite superpoñer uns elementos sobre outros coma se fosen capas.

Un exemplo dun documento XHTML no que se utiliza este método para incluír formatos é:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

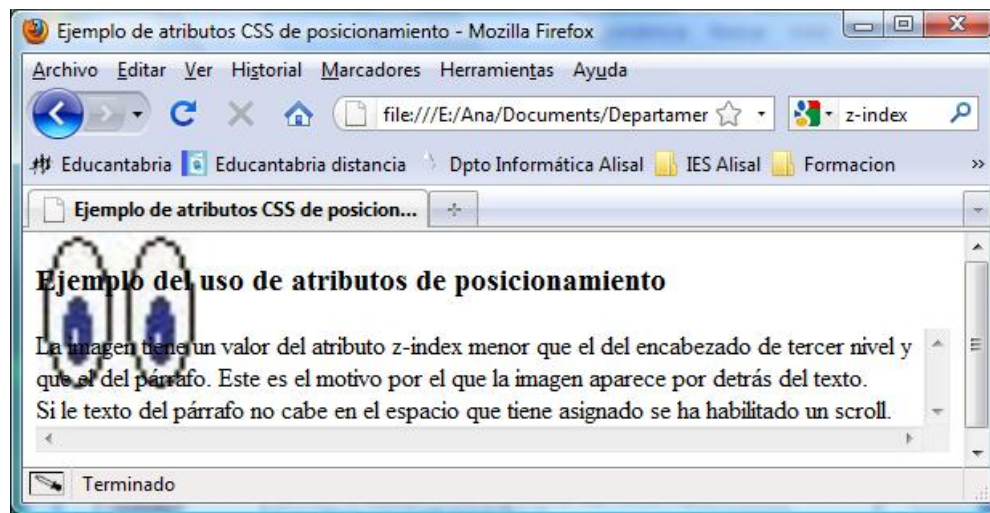
  <head>
    <title>Ejemplo de atributos CSS de posicionamiento</title>
    <style type="text/css">
      img{ position:absolute; left: 10px; top: 0px; z-index:-1;}
      p{overflow:scroll;}
    </style>
  </head>

  <body>
    <h3>Ejemplo del uso de atributos de posicionamiento</h3>
    </img>
    <p> La imagen tiene un valor del atributo z-index menor que el del encabezado de tercer
del p&aacute;rrafo. Este es el motivo por el que la imagen aparece por detr&aacute;s del texto.
Si le texto del p&aacute;rrafo no cabe en el espacio que tiene asignado se ha habil
scroll.</p>
  </body>
</html>
```



Código fonte da imaxe

Ao publicalo nun navegador, por exemplo no Firefox, teriamos:



## 6.6. Unidades de tamaño

As distintas unidades que podemos utilizar para indicar tamaños son as seguintes:



- **Relativas**

- *Element (em)*: expresa o tamaño relativo ao tamaño da fonte utilizada.
- *X-height (ex)*: expresa o tamaño relativo ao da letra "x".
- *Pixel (px)*: expresa o tamaño relativo á resolución do monitor.

- **Absolutas**

- *Milímetros (mm)*.
- *Centímetros (cm)*: cada centímetro son 10 mm.
- *Polgadas (in)*: cada polgada equivale a 2,54 cm.
- *Puntos (pt)*: cada punto son 1/72 in.
- *Picas (pc)*: cada pica son 12 pt.



### Autoavaliación

As unidades relativas especifican o tamaño en relación ao tamaño dunha letra determinada que escolle o programador:

- ☐ Verdadeiro.
- ☐ Falso.

## 6.7. Definición e uso de clases

---

Cando as regras de estilos asóciase a un documento HTML utilizando un ficheiro externo ou incluíndoas no contido da etiqueta STYLE na cabeceira do documento, poden definirse estilos e asocialos a determinados elementos do documento.

Para definir unha clase hai que usar a sintaxe seguinte:

**.clase\_azul{cor:blue}**

Para asociar un elemento HTML a unha clase haberá que usar o atributo CLASS ao usar o dito elemento no documento HTML do seguinte modo:

**<h3 class="clase\_azul">O encabezado de terceiro nivel é agora azul</h3>**

Para restrinxir a clase a un determinado elemento basta poñer o elemento diante do punto ao definir a regra. Por exemplo, para restrinxir o uso da clase a parágrafos teremos:

**p.clase\_azul{cor:blue}**

Un exemplo dun documento XHTML no que se utiliza este método para incluír formatos é:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

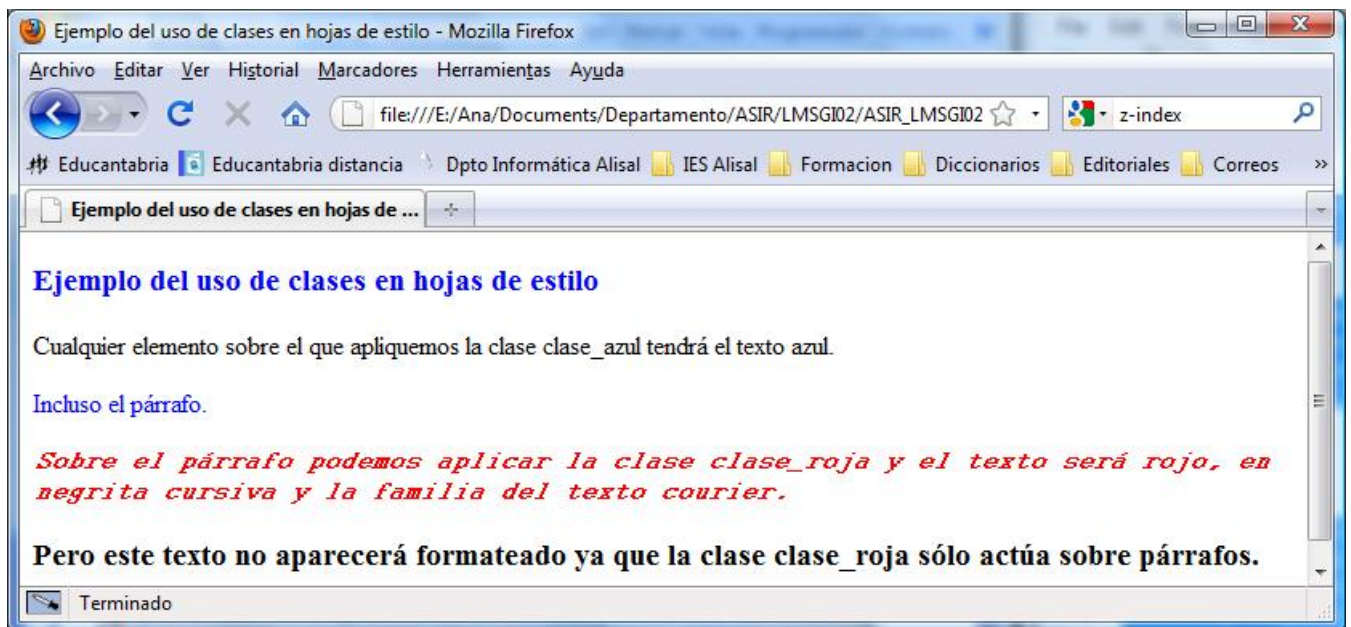
  <head>
    <title>Ejemplo del uso de clases en hojas de estilo</title>
    <style type="text/css"> .clase_azul{color:blue} p.clase_roja{color:#ff0000; font-style:
weight:bold; font-family:courier;} </style>
  </head>

  <body>
    <h3 class="clase_azul">Ejemplo del uso de clases en hojas de estilo</h3>
    <p>Cualquier elemento sobre el que apliquemos la clase clase_azul tendr&aacute; el text
    <p class="clase_azul"> Incluso el p&aacute;rrafo.</p>
    <p class="clase_roja">Sobre el p&aacute;rrafo podemos aplicar la clase clase_roja y el
rojo, en negrita cursiva y la familia del texto courier.</p>
    <h3 class="clase_roja"> Pero este texto no aparecer&aacute; formateado ya que la clase c
s&aacute; lo act&aacute;a sobre p&aacute;rrafos.</h3>
  </body>
</html>
```



[Código fonte da imaxe](#)

Ao publicalo nun navegador, por exemplo no Firefox, teríamos:



## Autoevaluación

Cal dos seguintes valores do atributo cor non é válido:

- ☐ rgb(33,33,0).
- ☐ rede.
- ☐ #f06.
- ☐ #ff06.

## Anexo. Licenzas de recursos

### Licenzas de recursos utilizados na unidade de traballo

Recursos (1)	Datos do recurso (1)	Recurso (2)	Datos do recurso (2)
	<p>Autoría: Firefox baixo windowx</p> <p>Licenza: copyright (Cita)</p> <p>Procedencia: captura de pantalla do navegador Firefox baixo Windows.</p> <p>Todas as capturas utilizadas nos exemplos teñen as mesmas credenciais.</p>		<p>Autoría: Bulefish baixo Ubuntu</p> <p>Licencia: GPL</p> <p>Procedencia: captura de pantalla de Bluefish baixo Ubuntu.</p> <p>Todas as capturas utilizadas nos exemplos de código teñen as mesmas credenciais.</p>
	<p>Autoría: Ana Polo Arozamena</p> <p>Licenza: uso educativo non comercial</p> <p>Procedencia: montaxe utilizando imaxes openclipart-0.18-full (<a href="http://openclipart.org/">http://openclipart.org/</a>) cuxa licenza é dominio público.</p>		<p>Autoría: Ana Polo Arozamena</p> <p>Licenza: uso educativo non comercial</p> <p>Procedencia: montaxe utilizando imaxes openclipart-0.18-full (<a href="http://openclipart.org/">http://openclipart.org/</a>) cuxa licenza é Dominio público.</p>