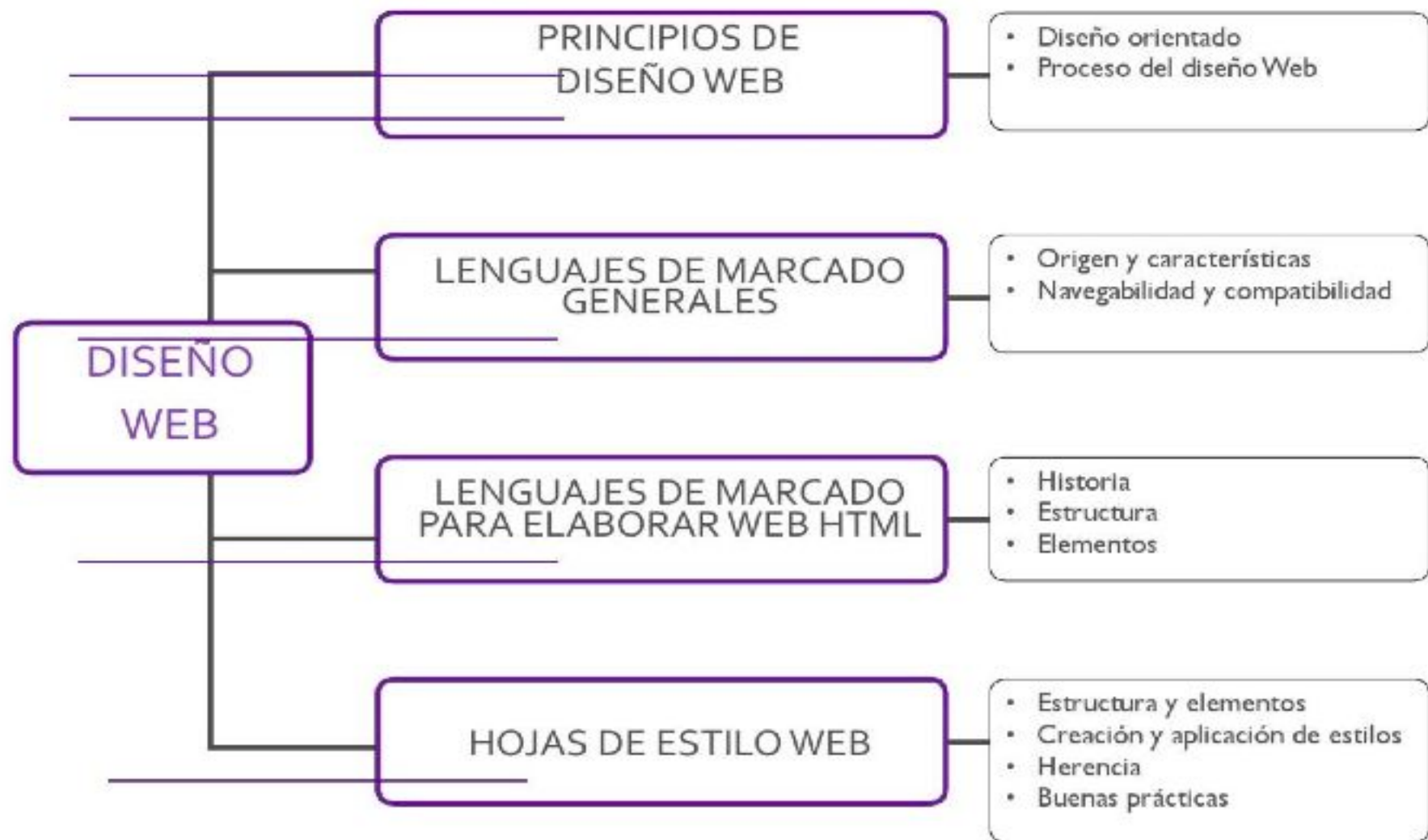


Creació de documents web mitjançant llenguatges d' etiqueta

UF 1841

60 hores



1. Disseny web

El terme **disseny web** es refereix habitualment al disseny de la part client (front-end) d'un lloc web.

Comprèn diverses disciplines: disseny gràfic, disseny d'interfícies d'usuari i conceptes d'usabilitat i accessibilitat.

Els dissenyadors han de dominar diverses tecnologies. Les bàsiques són **HTML**, **CSS** i **Javascript**.

1.1 Principis de disseny web

Una pàgina ha de ser dissenyada per tal de facilitar la tasca dels usuaris. En aquest sentit hi ha tres normes que s'ha de seguir en la seva elaboració:

- Usabilitat. L'usuari ha de poder trobar la informació que l'interessa ràpidament.
- Consistència. S'ha de mantenir un disseny coherent en totes les pàgines i seccions del site web.
- Velocitat de càrrega. S'han d'aplicar tècniques d'optimització per tal que la pàgina es carrega al més aviat possible i evitar la marxa dels lectors.

1.1.1 Disseny orientat a l'usuari

L'objectiu de tot disseny es troba centrat en l'experiència de l'usuari, sobretot en les tasques que necessiten dur a terme sobre la nostra web al seu entorn.

En els **processos iteratius** d'elaboració de webs les propostes són provades a nivell d'usabilitat pels usuaris reals. Aleshores els dissenyadors prenen nota d'aquest comportament i a la següent iteració incorporen les suggerències dels usuaris sobre el disseny.



1.1.2 Disseny orientat a objectius

El disseny orientat a objectius posa èmfasi al propòsit de la web, allò que els propietaris volen aconseguir amb ella: pot ser una botiga online, una pàgina d'empresa, un mitjà de comunicació, etc.

Ens ajuden a decidir quines funcionalitats s'han d'incloure a la web i solen plantejar-se amb anterioritat al disseny orientat a l'usuari.

1.1.3 Disseny orientat a la implementació

Aquest enfocament es fixa en com s'ha de fer la pàgina web. Aquí s'han de seguir els estàndars del W3C, el consorci que publica les guies per HTML i CSS.

El bon compliment d'aquests estàndars és una bona pràctica que ajuda la compatibilitat del nostre disseny amb tots els navegadors i facilita el manteniment del codi.

També aquí es tenen en compte les guies d'accessibilitat per aquells usuaris amb limitacions sensorials, tot seguint les pautes que en aquest sentit també marca la W3C.

1.2 El procés de disseny web

El procés del disseny web sol començar amb un esquema o plànol del site web (wireframe).

Al començament l'esquema se centra en el layout (disposició dels elements), deixant per fases posteriors detalls com la tipografia, colors, imatges, etc.



1.2.1 Estructura d'un lloc web i navegabilitat

El disseny de navegació permet que l'usuari es mogui entre les diferents pàgines del lloc web.

Normalment sol haver més d'una forma de navegació. Per exemple en un diari digital trobem els articles organitzats per seccions, tot i que també se'ns podem mostrar ordenats segons les darreres notícies publicades, les més llegides o les més comentades.

1.2.2 Estructura i composició de les pàgines

Un dels principals objectius de l'esquema és el disseny de la informació. Es tracta de com s'estructura i mostra la informació de la pàgina, de manera que sigui fàcil d'entendre.

La jerarquia visual dels continguts facilita que l'usuari trobi la informació que cerca de manera ràpida i contribueix a què la informació s'entengui millor.

1.2.3 Compatibilitat amb navegadors

Un dels principals maldecaps dels dissenyadors ha estat la compatibilitat entre navegadors, sobretot els més antics. En aquest aspecte s'ha de remarcar IE6, per la seva poca compatibilitat amb els estàndards HTML.

Sortosament, l'arribada d'HTML5 ha fet conscienciar els fabricants sobre la necessitat de respectar aquests estàndards pel bé comú de la xarxa.



1.2.4 Diferència entre disseny presentació/impressió

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

Mitjançant l'atribut media de l'etiqueta <link>, que importa fulls d'estils, és possible especificar un aspecte de la impressió diferent del que es veu per pantalla. En aquests casos el que es fa normalment és:

- Revisar els tipus de lletra
- Treure elements de navegació
- Eliminar videos, animacions, anuncis, etc.
- Incloure la URL, l'autoria, copyright.

2. Llenguatges de script generals

Els llenguatges de script generals són els que defineixen les regles per als llenguatges de script més específics.

Per això SGML i XML són els llenguatges genèrics de HTML (fins la versió 4) i XHTML respectivament.



2.1 Origen dels llenguatges script SGML i XML

SGML va néixer com a llenguatge de marcat per al formateig de les impressions. Va esdevenir estàndard els 1986 i era hereu de l'SGL, d'IBM.

En canvi XML (1996) és estàndard del W3C i va solucionar les limitacions del SGML, per fer-lo més senzill i flexible. A més incloïa l'SGML com a subconjunt seu, de manera tot document SGML és també XML.

2.2 Característiques generals del llenguatge script

En ambdós casos, SGML i XML, barregen contingut i etiquetes en un mateix document.

EJEMPLO DE DOCUMENTO EN SGML

```
<!DOCTYPE article PUBLIC "-//OASIS//DTD DocBookV3.1//EN">
<article>
<sect1 id="introduction"><title>Introducción a SGML</title>
<para>
Bienvenido a SGML
</para>
</sect1>
</article>
MAX_LENGTH="30"
COLLATION="SQL_Latin1_General_CI_AS"/>
</RECORD>
<ROW>
<COLUMN SOURCE="1" NAME="age"
xsi:type="SQLINT"/>
<COLUMN SOURCE="2" NAME="firstname"
xsi:type="SQLVARYCHAR"/>
<COLUMN SOURCE="3" NAME="lastname"
xsi:type="SQLVARYCHAR"/>
</ROW>
</BCPFORMAT>
```

2.2 Característiques generals del llenguatge script

Per altra banda les etiquetes es diferencien de la resta del contingut mitjançant **caracters especials**.

Els documents s'emmagatzemen en fitxers de **text pla**.

EJEMPLO DE DOCUMENTO EN XML

```
<?xml version="1.0"?>
<BCPFORMAT
  xmlns="http://schemas.microsoft.com/sqlserver/2004/bulkload/format"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <RECORD>
    <FIELD ID="1" xsi:type="CharTerm" TERMINATOR="\t"
      MAX_LENGTH="12"/>
    <FIELD ID="2" xsi:type="CharTerm" TERMINATOR="\t"
      MAX_LENGTH="20" COLLATION="SQL_Latin1_General_CP1_CI_AS"/>
    <FIELD ID="3" xsi:type="CharTerm" TERMINATOR="\r\n"
```


2.3 Estructura general d'un doc. amb lleng. script

En qualsevol fitxer d'aquestes característiques, com per exemple XML, es distingeixen dues parts:

- Un pròleg, opcional, amb metadades i instruccions de procés
- Un element arrel que contindrà la resta dels elements

2.3.1. Metadades i instruccions de procés

Les instruccions de procés donen informació sobre el document (metadades) al navegador. És per això que la primer línia d'un document XML sol ser:

```
<?xml version='1.0' encoding='UTF-8'>
```

Dona informació sobre la versió de l'especificació i el conjunt de caracters. També és possible associar l'XML a un DTD, un fitxer que especifica la seva norma de formació.

```
<!DOCTYPE personas SYSTEM "personas3.dtd"> <?xml-  
stylesheettype="text/xsl" href="incidencias.xslt"?>
```

2.3.2. Codificación de caracteres. Caracteres especiales.

La codificació de caràcters és la manera com aquests són emmagatzemats. Donat que hi ha diversos conjunts de caràcters s'ha d'especificar el propi del document, normalment a la web és UTF-8.

Els caràcters utilitzats per crear etiquetes dins del document han de tenir un tractament especial i per això són codificats de manera diferent.

CARACTER	ENTIDAD
>	>
<	<
"	"
'	'
&	&

2.3.3. Etiquetes o marques

A XML y SGML les etiquetes s'escriuen entre els símbols '<' y '>'

```
<!--etiqueta de apertura--> <a>
```

```
<!--etiqueta de cierre--> </a>
```

```
<!--etiqueta de apertura y cierre--> <meta/>
```

2.3.4. Elements

En general, els elements tenen una etiqueta d'apertura i una altra de tancament.

El text contingut entre ambdues és el text marcat i es troba afectat per l'etiqueta.

En alguns llenguatges, com HTML, es permeten elements sense etiqueta de tancament (és el cas de ``)

A l'interior de les etiquetes també trobarem atributs, que especifiquen característiques de l'element.

```

```

2.3.5 Atributs

Efectivament, els elements poden tenir atributs que se situen en l'etiqueta d'apertura.

Això sí, es troben reglats per especificació d'aquell element, però el seu ordre a l'interior no importa.

La majoria de casos tenen la forma `nom="valor"` (entre cometes)

```
<span id="div1" class="normal">text</span>
```

2.3.6 Comentarís

Els comentaris es fan servir per aclarir el codi a un altre dissenyador que se'l trobi posteriorment.

Incloure comentaris interessants és una bona pràctica, fins i tot quan els escrivim, per nosaltres mateixos, un cop haguem de revisar el codi mesos després.

```
<!--comentario-->
```

2.4 Documents vàlids i ben formats. Esquemes.

Un document XML està ben format si compleix les regles de sintaxi d'XML.

Entre altres característiques ha de complir que:

- Element arrel que contingui tots els elements
- Totes les etiquetes tancades
- Valors dels atributs entre cometes

2.4 Documents vàlids i ben formats. Esquemes.

Per exemple, aquest document sí està ben format:

```
<?xml version="1.0" encoding="UTF-8"?>
<trabajadores>
  <trabajador cod="t100">
    <nombreCompleto>Antonio Pérez</nombreCompleto>
    <nacimiento>1970</nacimiento>
    <carnetConducir/>
  </trabajador>
  <trabajador cod="t101">
    <nombreCompleto>Ana Gómez</nombreCompleto>
    <nacimiento>1980</nacimiento>
  </trabajador>
</trabajadores>
```

Aquest document NO està ben format:

```
<?xml version="1.0" encoding="UTF-8"?>
<trabajador cod="t100">
  <nombreCompleto>Antonio Pérez</nombreCompleto>
  <nacimiento>1970</nacimiento>
  <carnetConducir/>
</trabajador>
<trabajador cod="t101">
  <nombreCompleto>Ana Gómez</nombreCompleto>
  <nacimiento>1980</nacimiento>
</Trabajador>
```

2.4 Documents vàlids i ben formats. Esquemes.

Un document és vàlid respecte un DTD (Document Type Definition), és a dir, a més de estar ben format ha de respectar la seva especificació, segons la qual ha estat dissenyat.

Ha de respectar els possibles atributs o els possibles valor que aquests tinguin, i els document poden aleshores ser validats també de manera automàtica mitjançant esquemes (són una alternativa als DTD)

http://www.w3schools.com/xml/xml_validator.asp

http://www.w3schools.com/xml/xml_dtd.asp

2.4 Documents vàlids i ben formats. Esquemes.

DTD

```
<!DOCTYPE note
[
<!ELEMENT note (to,from,heading,
body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
]>
```

http://www.w3schools.com/xml/xml_dtd.asp

XML vàlid DTD

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE note [
<!ENTITY nbsp "&#xA0;">
<!ENTITY writer "Writer: Donald Duck.">
<!ENTITY copyright "Copyright: W3Schools."
>
]>

<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
<footer>&writer;&nbsp;&copyright;</footer>
</note>
```

3. Llenguatges d'etiquetes per presentació de pàgines web

HTML5, HTML (Hypertext Markup Language) i el seu germà XHTML (Extensible HTML) són els principals llenguatges d'etiquetes per crear pàgines web



3.1 Història d'HTML i XHTML. Diferències.

La primera versió d'HTML és de començaments dels anys 90, gràcies al treball de Tim Berners-Lee. A partir del 1996 la seva especificació passa a ser controlada pel W3C.

La darrera versió, HTML5, ja és un estàndard de facto i és la que porten tots els navegadors mòbils.

3.1 Història d'HTML i XHTML. Diferències.

La versió 4.01 HTML tenia 3 variants, que es poden distingir encara als encapçalaments de les pàgines antigues:

- Strict: no permeten l'ús d'etiquetes deprecated (obsoletes)
- Transitional: Sí que en permeten
- Frameset: Per a pàgines amb marcs (frames), que veurem més endavant.

3.1 Història d'HTML i XHTML. Diferències.

XHTML és la versió 'XML' d'HTML, donat que comparteix característiques de documents ben formats d'XML, com per exemple la prohibició de deixar etiquetes obertes, l'ús de cometes en els valors dels atributs, etc.



3.1 Història d'HTML i XHTML. Diferències.

Finalment tenim HTML5, que ja no és XHTML, una versió tanmateix molt més semàntica (introducció de article, section, footer, header, etc.) que a més inclou noves etiquetes per funcionalitats que abans només es podien fer amb Javascript, com la validació de camps de formularis.

3.2 Estructura d'un document

La primera línia és la declaració del tipus de document.

A continuació el tag <html> d'apertura de document, la capçalera <head> (on posarem els imports a Javascript i CSS), amb el <title> anidat, i finalment l'etiqueta <body>, on comença el disseny visible de la pàgina amb la resta d'etiquetes HTML.

Estructura bàsica d'un document HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Primeras pruebas</title>
  </head>
  <body>
    Hola mundo
  </body>
</html>
```

3.2.1. Versions

La primera línia de qualsevol document HTML és la declaració del tipus de document o doctype (no és una etiqueta) i indicar al navegador el llenguatge i la versió.

Versión	Declaración de DOCTYPE
HTML 5	<code><!DOCTYPE html></code>
HTML 4.01 Strict	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd"></code>
HTML 4.01 Transitional	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"></code>
HTML 4.01 Frameset	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd"></code>

3.2.2 Elements de capçalera

La capçalera està formada per l'element head.

Aquí s'inclouen metadades (per exemple termes claus per al SEO) i també es vinculen els fulls d'estil css i els js de Javascript.

Així doncs inclou:

<title>, <style>, <meta>,
<link>, <script>, and <base>.

EJEMPLO

```
<HEAD>  
<TITLE>Todo Jardin </TITLE>  
<META NAME="keywords" CONTENT="jardineria, plantas, cuidados, consejos ">  
</HEAD>
```

http://www.w3schools.com/html/html_head.asp

3.2.3 Elements del cos del document

El contingut visible de la pàgina es troba envoltat per les etiquetes <body>.

Excepte amb etiquetes <frameset>, <body> és una etiqueta obligatòria.

La resta d'elements dins <body> poden ser de bloc (block) o inline. Aquesta distinció és important perquè afecta la visualització per defecte d'aquestes etiquetes. Si son block ocuparan tot l'espai d'una mateixa línia, provocant el salt dels elements anteriors o posteriors en una altra: p. ex. DIV, H1 són block.

Si són inline poden conviure amb altres etiquetes dins la mateixa línia: SPAN, IMG, STRONG són inline.

3.3. Color

En HTML5 els atributs relacionats amb el color han desaparegut.

Com la resta de propietats relacionades amb la presentació, actualment s'ha especificar mitjançant CSS.

Color	RGB	HSL
Blanco	255, 255, 255	0, 0%, 100%
Negro	0, 0, 0	0, 0%, 0%
Rojo	255, 0, 0	0, 100%, 50%
Verde	0, 255, 0	120, 100%, 50%
Azul	0, 0, 255	240, 100%, 50%
Verde oscuro	0, 128, 0	120, 100%, 25%
Verde claro	128, 255, 128	120, 100%, 75%

3.3.1. Codificació de colors

A la web trobareu diferents formes de codificar els colors:

- RGB numérica: cada color Red-Green-Blue té un rang entre 0 i 255 per aconseguir la resta de colors.
 - rgb(0,0,0): negre pur
 - rgb(255, 255, 255): blanc pur
 - rgb(0, 0, 255): blau pur

-És possible afegir una quarta component per especificar l'opacitat.

- RGB hexadecimal: igual sistema additiu de colors que l'anterior però cada color té el rang especificat en aquesta notació hexad. entre 00 i FF.

- #000000
- #FFFFFF
- #0000FF

3.3.1. Codificació de colors

Menys usats és el sistema HSL (Hue-Saturation-Lightness). La primera component, entre 0 i 360, representa el color base, i les altres dues, entre 0 i 1 (o percentatge), la saturació i la lluminositat.

Ex.: `body {background-color: rgba(255, 0, 0, 0.5) }`

-També pot presentar una quarta component pel canal alpha de transparència



Recurs web: <http://hslpicker.com/>

3.3.2 Colors tipus

- Actualment hi ha 16 colors en HTML5 reconeguts pel seu nom. Són els següents:

aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white y yellow.

- CSS3, al seu torn, suporta 140 noms de colors.

Recurs web: [Llista de colors CSS3](#)

3.3.3 Colors segurs

Els **Web Safe Colors** eren considerats els 216 colors estàndard que antigament es considerava que podien mostrar les targetes gràfiques de tots els ordinadors connectats a Internet.

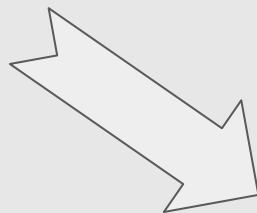
Per això aquest conjunt va ser etiquetat com a tal i els dissenys més acurats els tenien en compte. Tanmateix avui dia aquesta denominació ha deixat de tenir sentit perquè pràcticament tots els aparells no tenen cap problema per mostrar colors de fins a 16 bits.

3.4 Text

En HTML els salts de línia s'ignoren i els espais col·lapsen.

```
<!doctype html5>  
<html>  
<head>  
  <title>Primeras pruebas</title>  
  <meta charset="UTF-8">  
</head>  
  <body>  
    Todo en  
    una línea  
    y los espacios  
  </body>  
</html>
```

colapsan



Primeras pruebas



Todo en una línea y los espacios colapsan

3.4 Text. Entitats predefinides HTML

Alguns caracters especials (ñ, ç, caracters accentuats, etc.), per evitar problemes de mala codificació en segons quins servidors, és probable que te'ls trobis referenciats mitjançant codis especials:

Carácter	Referencia	Entidad	Carácter	Referencia	Entidad
Espacio	 	 	Salto de línea	
	
Á	Á	Á	á	á	á
É	É	É	é	é	é
Í	Í	Í	í	í	í
Ó	Ó	Ó	ó	ó	ó
Ú	Ú	Ú	ú	ú	ú
Ñ	Ñ	Ñ	ñ	ñ	ñ

3.4.1 Capçaleres. Jerarquia i estructura.

HTML disposa de sis elements per les capçaleres: h1, ..., h6, de header, ordenats de major a menor importància. Tots els navegadors, en el seu full d'estil per defecte, donen una mida de lletra més gran a h1 i més petita a h6.

Per motius SEO, **només hauria d'haver un encapçalament <h1> a cada pàgina**, i hauria de reflectir el títol del document amb fidelitat al contingut.

3.4.2. Paràgrafs

Podem delimitar paràgrafs de text amb l'etiqueta `<p>`, que són elements de bloc.

És per això una de les etiquetes més usades, sobretot si ens trobem en una pàgina web de tipus diari o llibre online.

No és apropiat fer-la servir només per marcar un salt de línia.

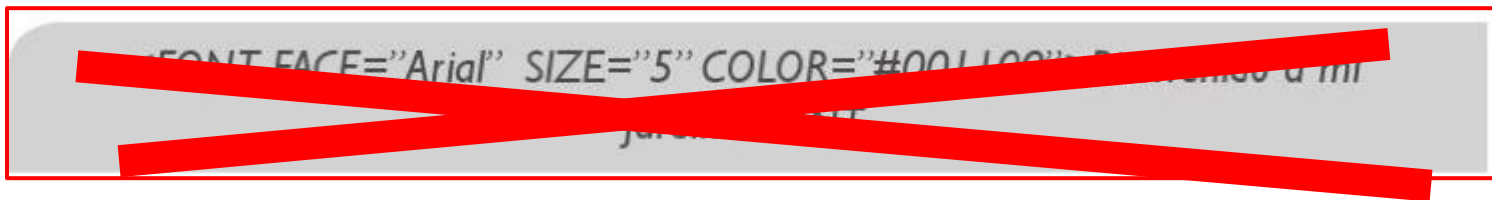
3.4.3 Alineació, espaciat i sangrat de text

Aquests tres elements es controlen en la nova especificació HTML5 mitjançant CSS, tal i com veurem més endavant, de manera que els antics atributs que es feien servir a HTML són ara obsolets.

3.4.4. Característiques de lletra: tipus, mides, colors

Igualment, els antics atributs per especificar aquestes característiques del text són obsolets avui dia, i s'ha d'assignar amb CSS3, tal i com veurem més endavant.

NO



3.4.5. Separadors de text

L'element `<hr>` es fa servir per introduir una separació visual, una línia horitzontal, entre dos blocs de contingut dins una mateixa secció.

Aquesta etiqueta es continua mantenint però atenció perquè ara ja no és correcte tampoc fer servir atributs interiors per estilar-la, haurem de fer servir CSS3.

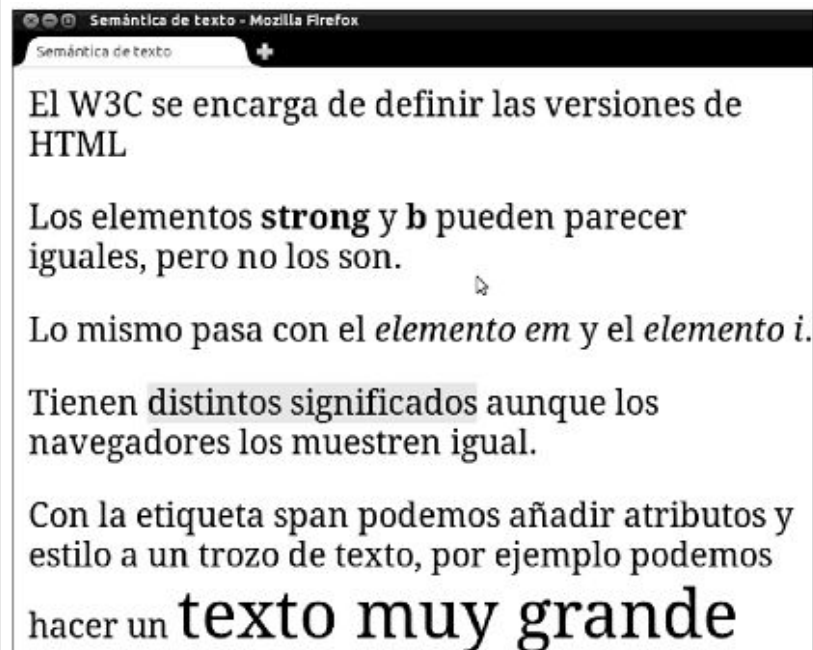
3.4.6. Etiquetes marca de text. Estils lògics.

Les etiquetes de semàntica a nivell de text, o estils lògics, es fan servir per indicar quina mena d'informació representa el text que contenen.

Elemento	Descripción
small	Texto que normalmente aparece en letra pequeña (copyright, aclaraciones)
span	No aporta significado por sí sola, se usa para añadir atributos o estilo
strong	Contenido importante
em	Énfasis
b	Contenido resaltado, pero no por ser especialmente importante (por ejemplo, la primera frase de un párrafo)
i	Representa una parte de texto cualitativamente diferente del texto que le rodea (por ejemplo, términos en técnicos o en otro idioma)
cite	Referencia a una obra creativa
abbr	Abreviatura o acrónimo
dfn	Representa una definición
mark	El texto marcado es especialmente importante dentro del documento
code	Código de programación, rutas de ficheros o en general texto que un ordenador reconocería

3.4.6. Etiquetas marca de text. Estils lògics.

```
<!DOCTYPE html>
<html>
<head>
  <title>Semántica de texto</title>
  <meta charset='UTF-8'>
</head>
<body>
  El <abbr>W3C</abbr> se encarga de definir las versiones de <abbr>HTML</abbr>
  <p>Los elementos <strong>strong</strong> y <b>b</b> pueden parecer
  iguales, pero no los son.</p>
  <p>Lo mismo pasa con el <em>elemento em</em> y el <i>elemento i</
  i>.</p>
  <p>Tienen <mark>distintos significados</mark> aunque los navegadores
  los muestren igual.</p>
  <p>Con la etiqueta span podemos añadir atributos y estilo a un trozo de
  texto, por
    ejemplo podemos hacer un <span style="font-size:xx-large">texto muy
    grande</span>
  </p>
</body>
</html>
```



3.5. Enllaços d'hipertext

La presència d'enllaços a HTML és tan característica al llenguatge que de fet li dóna nom.

Per defecte tots els fulls d'estil propis dels navegadors ja li confereixen un aspecte, normalment els sotslineat.

3.5.1. Estructura d'un enllaç: adreça internet o URL

Per localitzar un recurs a Internet es fa servir de manera genèrica l'anomenat URL (*Uniform Resource Locator*), que té la següent estructura:



3.5.2. Estils dels enllaços

Tal i com s'ha esmentat abans els enllaços solen tenir un estilat predefinit pel full d'estils del propi navegador, normalment un sotslineat.

Tal i com veurem posteriorment a CSS els enllaços a més poden tenir un estil propi segons el seu estat active, visited o hover.

3.5.3. Diferències entre enllaços absoluts i relatius

Aquest és una de les característiques sovint costa més entendre a qui acaba d'arribar a HTML.

Un enllaç és **relatiu** quan la seva ruta (atribut href) apunta a un objectiu a partir de la ubicació del propi enllaç en el sistema de fitxers.

```
<a href="../../../images/foto1.jpg">La meva foto relativa</a>
```

Un enllaç és **absolut** quan la seva ruta apunta al seu objectiu basant-se en la posició d'aquest en la pròpia màquina, independentment de la posició de l'enllaç. Per això aquesta ruta començarà des de l'arrel del sistema de fitxers. També es considera absoluta una ruta que apunta a un recurs d'un domini d'Internet.

```
<a href="C:\Imágenes\Fotos\foto1.jpg">La meva foto absoluta</a>
```

3.5.4. Àncores o enllaços interns

Per fer referència a enllaços interns i posicionar la pàgina a l'altura d'un d'ells es fan servir les àncores.

A aquest efecte, a HTML5 l'atribut name ja és obsolet, ara es fa servir id. De manera que si tenim un paràgraf com aquest:

```
<p id="capitol3">Capítol 3</p>
```

podem referenciar-lo des de la mateixa pàgina amb aquest enllaç:

```
<a href="#capitol3">Enllaç al Capítol 3</a>
```


3.5.5. Enllaços especials. E-mail. Descàrrega

Quan no existien llenguatges servidor com PHP que poguessin enviar e-mails amb formularis, era molt freqüent crear un **enllaç de tipus correu electrònic** que enllacés directament amb el client de correu de l'equip del propi usuari:

```
<a href="mailto:micorreu@miweb.com">Enviar email</a>
```

Aquesta pràctica no és la millor perquè l'usuari podria no tenir accés a aquest programa de correu client de l'equip des d'on opera... o no tenir-lo configurat.

.....

Més pràctic és fer servir l'**atribut download** per indicar la descàrrega directa d'un enllaç:

```
<a download href="fitxer.zip">Descarregar fitxer</a>
```

Atributs específics: títol, destinació, shortcuts, etc.

HTML defineix per cada etiqueta un seguit d'atributs. <a> no podia ser menys:

Atributo	Valores posibles	Descripción
href	Una URL relativa o absoluta	URL del recurso enlazado
target	_blank, _self, _top, _parent id de un frame o iframe	Dónde se abrirá el enlace
rel	alternate, autor, bookmark, help, license, next, nofollow, noreferrer, prefetch, prev, search, tag	Relación entre el documento y el documento enlazado
hreflang	Identificador del idioma	Idioma del documento enlazado
type	Un tipo MIME válido	Tipo del documento enlazado
download	Atributo booleano	Si se usa, el recurso enlazado se descarga

3.6. Imatges

Les imatges a un document HTML s'inclouen mitjançant l'etiqueta

No formen part del propi codi html sinó que resten vinculades com un element extern mitjançant la seva ruta.

En trobar una etiqueta img,
el servidor demanarà per la
seva ruta a l'atribut src.



3.6.1. Formats d'imatges

HTML suporta diversos formats:

- GIF
- JPEG
- PNG
- PDF
- SVG
- GIFS animats
- AJPEG (animat)

3.6.2. Característiques de les imatges. Atributs.

L'etiqueta té dos atributs molt importants, src i alt, i d'altres opcionals com per exemple:

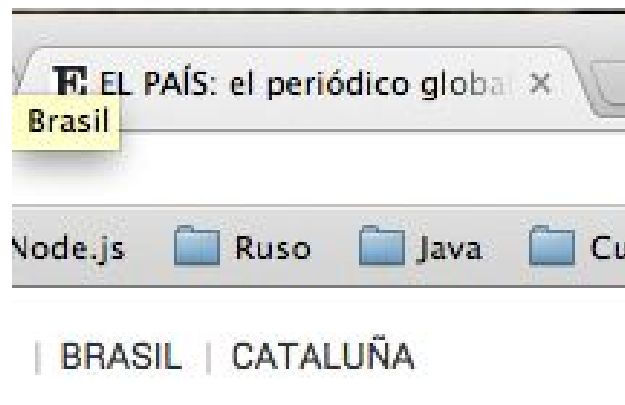
Atributo	Descripción	Valor por defecto	Obligatorio
src	URL del fichero	No tiene	Sí
alt	Texto alternativo	No tiene	Sí
height	Alto de la imagen	El de la imagen	No
width	Ancho de la imagen	El de la imagen	No

3.6.2. Característiques de les imatges. Favicons.

Favicon és una imatge que identifica una web com si fos un logo. Sol tenir un mida de 16px per 16px i els navegadors els reconeixen posant-los a la barra d'adreces a l'esquerra de la url.

Perquè el reconeguim hem de posar aquesta etiqueta a la capçalera (<head>) de la pàgina:

```
<link rel="shortcut icon" href="
favicon.ico" type="image/gif"/>
```



3.6.3. Enllaços en imatges

Es possible que una imatge, de la manera que ho fa un text, estigui vinculada a un altre recurs a través d'un enllaç. Per fer-ho només hem d'embolcallar la imatge amb l'etiqueta <a>:

```
<a href="http://www.ub.edu"></a>
```

3.6.4. Imatges de fons

A HTML5 ja no és correcte fer servir l'atribut background a l'etiqueta <body> per crear una pàgina amb imatge de fons. Ara s'ha de fer amb regles CSS, tal i com veurem més endavant.

~~<body background="fons.jpg">~~ NO ÉS CORRECTE A HTML5

3.7.1. Característiques

A HTML tenim tres tipus de llistes:
ordenades, no ordenades i definició.

- Inicio
- Noticias
 - [Recientes](#)
 - [Más leídas](#)
 - [Más valoradas](#)
- Artículos
 1. XHTML
 2. CSS
 3. JavaScript
 4. Otros
- Contacto

Email

nombre@direccion.com

Teléfono

900 900 900

Fax

900 900 900

3.7.2. Ordenació de llistes

- Ordered Lists (Llistes ordenades)

Atributo	Valores posibles	Descripción
start	Un número entero	Primer número de la lista
type	1- números decimales a- orden alfabético A- orden alfabético en mayúsculas i - números romanos I- números romanos en mayúsculas	Tipo de marcador de la lista
reversed	Atributo booleano	Si está presente, ordena la lista de mayor a menor

```
<ol start="3" type='l'>  
  <li>Peras</li>  
  <li>Manzanas</li>  
  <li>Limones</li>  
</ol>
```

3.7.2. Ordenació de llistes

``- Unordered Lists (Llistes no ordenades) (etiqueta sin atributos)

```
<ul>
```

```
  <li>Patates</li>
```

```
  <li>Peres</li>
```

```
  <li>Pomes</li>
```

```
</ul>
```

`<dl>`-Definition Lists (Llistes de definició) (a l'interior `<dt>`, `<dd>`)

```
<dl>
  <dt>Coffee</dt>
  <dd>Black hot drink</dd>
  <dt>Milk</dt>
  <dd>White cold drink</dd>
</dl>
```

3.7.3. Anidament de llistes

Les llistes es fan servir sovint per crear menús de navegació que posteriorment seran estilats per CSS.

Quan hi ha submenús al menú principal és perquè s'ha fet servir l'anidament de llistes.

```
<ul>  
  <li>Ensalada</li>  
  <ul>  
    <li>Caprese</li>  
    <li>César</li>  
  </ul>  
</ul>
```

3.7.4. Altres tipus de llistes: llistes de definició

Les llistes de definició inclouen agrupaments de terme-definició, com a les entrades d'un diccionari:

`<dl>` defineix la llista

`<dt>` correspon al terme a definir

`<dd>` correspon a la definició

```
<dl>  
  <dt>Tomate</dt>  
  <dd>Fruto de la tomatera</dd>  
  <dd>Juego de naipes</dd>  
  <dt>Perro</dt>  
  <dt>Can</dt>  
  <dd>Animal de compañía</dd>  
</dl>
```

3.8. Taules

Les taules permeten agrupar dades en files i columnes de cel·les.

Durant molts anys van servir també per configurar l'estructura de les pàgines, el seu layout, però avui dia aquesta pràctica ja no és correcta, i s'opta pel posicionament flotant amb CSS.

3.8.1. Taules. Estructura bàsica.

table: per crear la taula

tr: per crear les files

td: per crear cel·les

th: cel·les capçalera de columna

caption: llegenda de la taula

Recurs web: [Taules](#)

```
<!DOCTYPE html>
<html>
<head>
  <title>Tablas</title>
  <meta charset='UTF-8'>
  <style>table,td,th,tr{ border: 1px black solid}</style>
</head>
<body>
  <table>
    <caption>Tabla de películas, directores y guionistas</caption>
    <tr>
      <th>Película</th>
      <th>Director</th>
    </tr>
    <tr>
      <td>Los otros</td>
      <td>Alejandro Amenábar</td>
    </tr>
    <tr>
      <td>Los pájaros</td>
      <td>Alfred Hitchcock</td>
    </tr>
  </table>
</body>
</html>
```

3.8.2. Format de taules: vores, alineació, mida, etc

És molt important recordar que a HTML5 són obsolets tots els atributs que es feien servir per formatejar les taules.

Ara només és correct fer-ho amb CSS, com veurem més endavant.

3.8.3. Format de contingut de cel·les.

Els elements `<thead>`, `<tbody>` i `<tfoot>` permeten agrupar files. Són etiquetes semàntiques.

Amb els atributs `colspan` i `rowspan` (aplicats a `td` i `th`) podem aconseguir agrupar columnes i files, respectivament.

El elemento `<colgroup>` nos permite aplicar las mismas reglas de estilo a varias columnas. `<col>` se anida dentro de `<colgroup>` cuando queremos especificar el estilo de una columna concreto dentro de un grupo de columnas.

3.8.4. Taules anidades

L'anidament d'una taula dins una altra s'aconsegueix mitjançant la definició d'un nou element `<table>` dins una cel·la.

Tablas anidadas					
Columna 1	Columna 2				
Celda normal	<table> <tr> <td>1</td><td>2</td></tr> <tr> <td>3</td><td>4</td></tr> </table>	1	2	3	4
1	2				
3	4				

```
<table>
  <caption>Tablas anidadas</caption>
  <tr>
    <th>Columna 1</th>
    <th>Columna 2</th>
  </tr>
  <tr>
    <td>Celda normal</td>
    <td>
      <table>
        <tr>
          <td>1</td>
          <td>2</td>
        </tr>
        <tr>
          <td>3</td>
          <td>4</td>
        </tr>
      </table>
    </td>
  </tr>
</table>
```

3.8.5. Bones pràctiques en l'ús de taules

Per tal de fomentar la llegibilitat d'una taula és recomanable:

- Distingir els encapçalaments de la resta de cel·les
- Alternar els color de fons de les files (pijama)
- Fer servir vores
- Fer servir elements semàntics com caption, thead, tbody, tfoot.

... i sobretot no fer servir taules per estructura pàgines, tal i com vam esmentar.

3.9. Marcs (Frames)

Antigament els frames eren habituals també per estructura el layout d'una pàgina web. A HTML5 ja han desaparegut, i per maquetar fem servir etiquetes semàntiques com

```
<frameset cols="25%, 50%, 25%">
```

```
<frame src="frame_a.htm">
```

```
<frame src="frame_b.htm">
```

```
<frame src="frame_c.htm">
```

```
</frameset>
```

3.9.1. Creació de marcs

Per crear marcs fem servir els elements `<frameset>` i `<frame>`. El primer defineix un conjunt de marcs i el segon cada un dels marcs.

De fet a una pàgina web `frameset` substitueix `body` i no poden aparèixer tots dos.

`Frameset` conté a més dos atributs, `cols` i `rows`, per especificar el nombre i mida de les seccions en què dividirem la pàgina.

3.9.2. Avantatges i inconvenients en l'ús de marcs

Avantatges:

- Forma ràpida d'implementar estructures de pàgines senzilles
- Permet organitzar el contingut de la pàgina en diversos fitxers html

Inconvenients:

- Inclou informació de presentació dins HTML
- Problemes de compatibilitat mab dispositius mòbils
- Problemes pels motors de cerca i penalització a nivell SEO.

3.9.3 Suport dels navegadors

Els navegadors d'escriptori moderns suporten marcs, el problema ve amb els mòbils, que no el suporten perquè no és HTML5.

```
<noframes>  
  <p>Lo sentimos, su navegador no soporta marcos </p>  
</noframes>
```

3.9.4. Formateig de marcs

Atributo	Valores posibles	Descripción
frameborder	0 - sin borde 1 - con borde, es el valor por defecto	Para mostrar un borde alrededor del marco
marginheight	Un número	Altura del margen en píxeles
marginwidth	Un número	Anchura del margen en píxeles
scrolling	yes no auto - valor por defecto	Indica si se deben mostrar barras de desplazamiento en el marco
noresize	Atributo booleano	Sí aparece este atributo, no se permite modificar el tamaño del marco

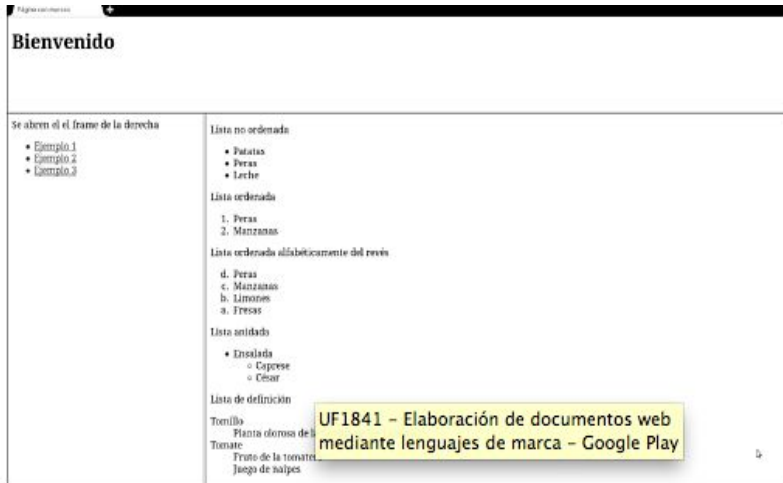
3.9.5. Enllaços entre contiguts de marcs

És possible crear enllaços que, mitjançant l'atribut `target` enllacin cap a pàgines que es carregin en els marcs que nosaltres volgüem.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Para la barra de navegación</title>
  </head>
  <body>
    Se abren el el frame de la derecha
    <ul>
      <li> <a href="imagenes.html" target="principal">Ejemplo 1</a></li>
      <li> <a href="listas.html" target="principal">Ejemplo 2</a></li>
      <li> <a href="tablas_colgroup.html" target="principal">Ejemplo 3</a></li>
    </ul>
  </body>
</html>
```

3.9.6. Marcs anidats

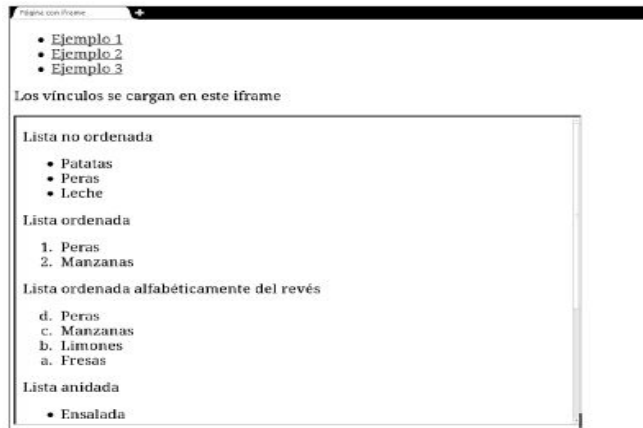
Si s'escau, sobretot quan es componen pàgines amb elements com ara capceleres o barres de navegació és possible anidar framesets dins d'altres framesets.



```
<!--Un primer frameset para crear una parte superior de cabecera y una
parte central-->
<frameset rows="20%,80%">
<!--marco para la parte superior-->
<frame src="cabecera.html">
<!--partimos la parte inferior en dos, la izquierda para vínculos la
derecha para el contenido-->
<frameset cols="25%,75%">
<!--frame para los vínculos de navegación-->
<frame src="navegacion.html">
<!--frame en el que se mostrará el contenido, con contenido inicial-->
<frame name="principal" src="listas.html">
</frameset>
</frameset>
```

3.9.7. Marcs incrustats. IFrames

`<iframe>` és l'alternativa en HTML5 als antics `<frame>`. Per inserir-los i donar-los mida es fan servir fulls d'estil. El contingut inicial s'especifica amb l'atribut `src` i també soporta l'atribut `name` per fer que el target d'un enllaç sigui un marc en concret.



```
<body>
  <ul>
    <li> <a href="cap3_ejemplo19_listas.html" target="principal">Ejemplo
      1</a></li>
    <li> <a href="cap3_ejemplo20_tablas_basicas.html"
      target="principal">Ejemplo 2</a></li>
    <li> <a href="cap3_ejempl10_imagenes.html"
      target="principal">Ejemplo 3</a></li>
  </ul>
  <p>Los vínculos se cargan en este iframe</p>
  <iframe height="400" width="600" name="principal" src ="cap3_
    ejemplo19_listas.html">
</body>
```

3.10. Formularis

HTML5 aporta moltes novetats respecte l'ús dels formularis. S'ha de recordar que pel correcte funcionament d'un formulari hem de comprovar el seu bon funcionament al client i també al processament del servidor.


HTML5 Contact Form with Google Maps

Contact Us

Name

Email

Phone

Contact Reason 

Message


GET IN TOUCH

PHONE
(404) 123 - 4567

EMAIL
info@27and42.com

ADDRESS
123 Peachtree St
Atlanta GA, 30313

[Get Directions](#)



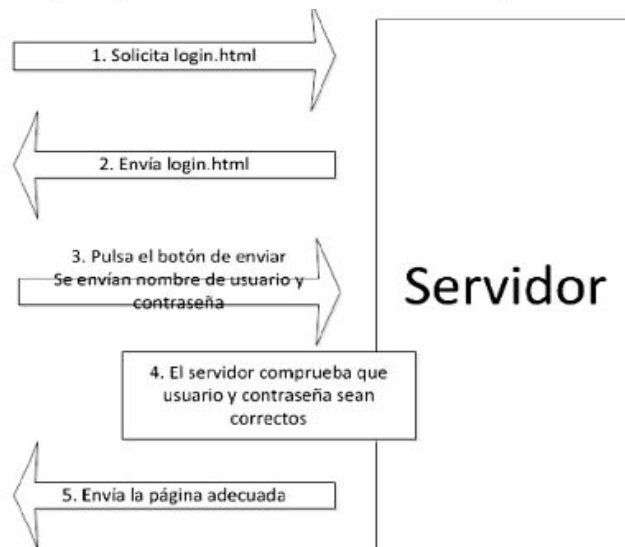
3.10.1. Descripció general i ús dels formularis

Tot i que associem els formularis a la sol·licitud d'informació per part d'un usuari, la seva versatilitat els ha permès ocupar altres funcions d'interaccions amb els usuaris, com per exemple fer un calculador en Javascript.

Tenen un botó d'enviament (submit) i d'altres camps per obtenir informació de manera diversa, como ara input text, dropdowns, botons radio, botons checks, etc.

3.10.1. Descripció general i ús dels formularis

Exemple de login a un correu web:



```

<!doctype html>
<html>
  <head>
    <title>Formularios</title>
  </head>
  <body>
    Por favor, introduzca usuario y contraseña
    <form name="login" autocomplete action="procesar.php"
    method="post">
      Usuario: <input type="text" name="user"><br>
      Contraseña: <input type="password"
      name="passw"><br>
      <input type="submit" value="Enviar">
      <input type="reset" value="Limpiar">
    </form>
  </body>
</html>

```

3.10.2. Elements bàsics d'un form: text, botons, etc.

Formularios. Elemento input

Bienvenido al IES Clara de

Formulario de Registro

Datos personales

Nombre y apellidos:

Hombre ☒ Mujer ☐

Mayor de edad ☐ Menor de edad ☒

Antiguo alumno ☒

Solicita convalidación ☒

Adjunte fichero con la fotografía: No se ha seleccionado ningún archivo.

Seleccione su color favorito:

Seleccione número de hermanos o hermanas:

```

<h1>Bienvenido al IES Clara del Rey</h1>
<h2>Formulario de Registro</h2>
<form action="procesar.php" method="get">
  <h3>Datos personales</h3>
  Nombre y apellidos: <input required name="apellido">
  <!--botones de radio, dos grupos diferentes-->
  <p>Hombre<input type="radio" name="sexo" value="hombre">
  Mujer<input type="radio" checked name="sexo" value="mujer">
  <p>Mayor de edad<input type="radio" name="edad" value="mayor">
  Menor de edad<input type="radio" checked name="edad"
  value="menor"></p>
  <!--checkboxbox-->
  <p>Antiguo alumno <input name="antiguo" type="checkbox"></p>
  <p>Solicita convalidación<input name="conva" type="checkbox"></p>
  <!--para seleccionar un fichero-->
  <p>Adjunte fichero con la fotografía<input name="foto" type="file"
  ></p>
  <!--para seleccionar un color-->
  <p>Seleccione su color favorito<input name="color" type="color"></p>
  <p>Seleccione número de hermanos o hermanas:
  <!-- este campo solo admite numeros entre 0 y 99-->
  <input name="hermanos" type='number' max="99" min="0"></p>
  <input type="submit">
  <input type="reset">
</form>

```

3.10.2. Elements bàsics d'un form: text, botons, etc.

Alguns valors de l'atribut type en un element <input>:

Valor de type	Descripción
text	Campo de texto, es el valor por defecto
password	Contraseña, el texto tecleado se muestra enmascarado
radio	Botones de radio
checkbox	Casillas de verificación
submit	Botón para enviar el formulario
reset	Botón para limpiar todos los campos del formulario
file	Permite seleccionar uno o varios ficheros del cliente
button	Botón genérico

3.10.2. Elements bàsics d'un form: text, botons, etc.

Més exemples d'elements en formularis:

Formularios. Más controles +

Elemento textarea

Aquí caben varias líneas de texto

Lista desplegable

Desarrollo de aplicaciones multiplataforma :
Informática
 Desarrollo de aplicaciones web
 Desarrollo de aplicaciones multiplataforma
Administración
 Gestión administrativa
 Comercio internacional

Caja de texto asociada a un *datalist*

Enviar Borrar

```
<form name="formu" action="procesar.php" method="post">
  Elemento textarea<br>
  <textarea name="texto" cols="80" rows="6"></textarea>
  <p>Lista desplegable</p>
  <select name="ciclo">
```

```
    <optgroup label="Informática">
      <option value="daw">Desarrollo de aplicaciones web</option>
      <option value="dam">Desarrollo de aplicaciones multiplataforma</option>
    </optgroup>
    <optgroup label="Administración">
      <option value="ga">Gestión administrativa</option>
      <option value="ci">Comercio internacional</option>
    </optgroup>
  </select>
  <!--saltos de línea para que se vea bien el formulario-->
  <br><br><br><br>
  <p>Caja de texto asociada a un <em>datalist</em></p>
  <datalist id="provincias">
    <option value="Palencia"/>
    <option value="Valencia"/>
    <option value="Madrid"/>
  </datalist>
  <input name="prov" list="provincias">
  <br>
  <!--botones-->
  <button type="submit">Enviar</button>
  <button type="reset">Borrar</button>
```

3.10.3. Processament de formularis

L'atribut `action` de l'etiqueta `<form>` determina la destinació del fitxer en servidor que processarà el formulari.

Ara bé, tant si és per Javascript com per les noves etiquetes HTML5(required) el formulari passa per una validació prèvia en client.

L'atribut `method` de `<form>` defineix el mètode HTTP amb què serà enviada la petició: els més usats són GET i POST.

Exemple de petició url de mètode GET:

```
procesar.php?apellido=Manuel+Pez&sexo=hombre&edad=mayor&antiguo=on&foto=india.JPG&color=%23000000&hermanos=5
```

3.10.4. Formatejat de formularis

La millor manera de construir un formulari és afegint etiquetes semàntiques del tipus label, legend o fieldset, així com altres etiquetes d'accessibilitat. Observeu el següent exemple:

```
<form action="procesar.php" method="post" >
  <fieldset>
    <!--label implícito: el control esta dentro del elemento label-->
    <label>Nombre <input type="text" name="nombre" tabIndex="2"
accesskey="n"></label>
    <!--label explícito usando el atributo for, contiene el valor del atributo
id del control-->
    <label for="campoApellido">Apellido</label>
    <input tabIndex="1" name="apellido" autofocus accesskey="a p"
id="campoApellido">
    <!--botones-->
    <legend>Datos personales</legend>
  </fieldset>
  <button tabIndex="3" type="submit">Enviar</button>
  <button tabIndex="4" type="reset" >Limpiar</button>
</form>
```

Formularios: atajos de tec...

Datos personales

Nombre Apellido

Enviar Limpiar

3.11. Elements específics per tecnologies mòbils

Amb CSS3 és possible crear webs que s'adaptin a diferents mides de pantalla, o fins i tot a una orientació, de manera que podem fer que una mateixa pàgina web es vegi diferent, **adaptada**, en versió escriptora, tauleta i mòbil.



3.12. Elements en desús, obsolets (deprecated)

Les etiquetes i tècniques que exposem a continuació s'han d'evitar perquè ja no corresponen a la nova especificació HTML5 i per tant ni són correctes ni responsive.



3.12.1. Text parpadejant (blink)

Aquest és un element ben estrany en la història de HTML. De fet mai no va formar part de cap especificació, i va ser famós durant un temps perquè el va introduir Netscape en els seus navegadors. Va adquirir força mala fama entre els dissenyadors perquè destorbava la lectura.

Avui dia aquest efecte es pot aconseguir amb Javascript i CSS.

Just don't do it...

\

```
<blink>Look at me!</blink>
```

3.12.2 Marquesines

Un altre element polèmic, encara més que el blink. Es tracta d'un element, normalment text, que es mou d'un d'un lloc a un altre, també destorbant la lectura de la pàgina web.



Recurs web: [Try Marquee](#)

3.12.3 Alineacions

Fins a HTML 4.01 es feien servir els atributs `valign`, `align` i l'etiqueta `<center>` per alinear i centrar. Ara s'han d'evitar i fer servir les propietats corresponents de CSS3.

3.12.4. Altres elements en desús

Altres elements pertanyents a versions anteriors a HTML5 i que hem d'evitar són:

- `font`: per tipus de lletra
- atributs `background` i `background-color`
- `applet` (ara es fa servir `object`)
- `frame` i `frameset`

4. Fulls d'estil web

Mitjançant els fulls d'estil podem:

- Asignar tipus de lletra, colors i mida als elements d'una pàgina
- Crear l'estructura d'una pàgina (layout)
- Adaptar el contingut al tipus de dispositiu amb què s'accedeix a la pàgina (mòbil, ordinador, lector de pantalla per cecs, etc.)
- Crear animacions, transicions i altres efectes avançats

4.1. Tipus de fulls d'estil: estàtics i dinàmics

Els fulls d'estil estàtics són aquells el contingut del qual no varia. No tenen modificació respecte de com van ser escrits.

Els fulls d'estil dinàmics són els generats pel servidor, i s'adapten a cada petició del client. S'usen molt menys.

També podem diferenciar entre:

- Fulls d'estil d'autor (les que crea el dissenyador web)
- Fulls d'estil de l'usuari (les que crea l'usuari per personalitzar l'aspecte)
- Full d'estil d'agent d'usuari (navegador): les que venen de sèrie amb cada navegador i que serveixen per pre-formatejar algunes etiquetes com `<h1>...` `<h6>`, ``, etc.

4.2. Elements i estructura d'un full d'estil

Estan formats per una o més **regles**. Cada regla consta d'un **selector** i un bloc de declaracions, entre claus. Un **bloc** consisteix en una o més **declaracions** separades per punt i coma. Cada declaració està formada per una **propietat** i un **valor**, separats per dos punts(:).



4.2.1. Creació de fulls d'estil

És possible incloure fulls d'estil en una pàgina de 3 maneres diferents:

- Fulls d'estil **externs**

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="ej1.css" media="all">
  </head>
  ...
```

- Fulls d'estils **interns** o incrustats, embedded, fent servir <style>

```
<head>
  <style>p{ color: red }</style>
</head>
```

- Estils **inline**, fent servir l'atribut *style*, sense que calgui el selector

```
<p style="color: red">Texto en rojo</p>
```

4.2.2. Aplicació d'estils

La primera part d'una regla és el selector. S'usa per determinar quins elements de la pàgina afecta. Els més importants són:

- Selector universal `{color: red}`
- Selector per nom element `p {color: red}`
- Selector per atribut id: `#principal {color: red}`
- Selector per atribut class: `.importante {color: red}`
- Selector elements d'una determinada classe: `p.important {color: red}`
- Selector per atribut: `[target='blank']{color: red}` (admet expressions regulars)
- Selector per element i atribut: `a[target='blank']{color: red}`

4.2.2. Aplicació d'estils

- Selector descendent: `nav a {color: red}`
- Selector fill: `nav>a {color: red}` (només fills... no 'néts')
- Selector múltiple: `h1, .importante {color: red}` (és possible estilar dos selectors, separats per coma)

4.2.2. Aplicació d'estils. Exemple.

```
<style>
/*para elementos con clase grande*/
.grande{ font-size: x-large }
/*para elemento con id enorme*/
#enorme{ font-size: xx-large }
/*para elementos a con atributo target="_blank"*/
a[target='_blank']{ font-size: x-large }
/*para elementos p hijos de section */
section > p{ text-decoration: underline }
/*para elementos p descendentes de section */
section p{font-size: x-large }
</style>
</head>
<body>
<p c >Párrafo normal</p>
<p class="grande">Párrafo con clase grande</p>
<p id="enorme">Párrafo con id enorme</p>
<p>Vínculo en la misma ventana <a href="http://www.w3.org">W3C</
```

```
<p>Vínculo en ventana nueva <a href="http://www.w3.org" "target="_
blank">W3C</a> </p>
<section>
<p>Párrafo hijo de section</p>
<div>
<p>Párrafo nieto de section</p>
</div>
</section>
```

Ejemplo de selectores

+

Párrafo normal

Párrafo con clase grande

Párrafo con id enorme

Vínculo en la misma ventana W3C

Vínculo en nueva ventana W3C

Párrafo hijo de section

Párrafo nieto de section

4.2.2. Aplicació d'estils. Pseudoclasses.

Les pseudo-classes eixamplen les possibilitats dels selectors, refinant encara més. Són fàcils de reconèixer perquè el seu nom comença per dos punts i poden ser de dos tipus:

Estructurals: Seleccionen a partir de la posició de l'element dins d'una jerarquia arbòria:

- :first-child (section > p:first-child)
- :nth-child (nth-child(3), nth-child(odd))

Dinàmiques: Aquí incloem les de vincles (link y visited), y també les d'usuari:

- :hover
- :active
- :focus

Recurs web: [Try Pseudoclasses](#)

ex: p:hover{font-size: x-large;}, a:active{font-size: x-large;}, input[type='text']:focus{font-size: x-large;}

4.2.2. Aplicació d'estils. Pseudoelements.

Els pseudoelements permeten seleccionar elements que no han estat explícitament definits en el document:

- `::first-line`: primera línia de text d'un element \rightarrow `p::first-line{font-size: x-large;}`
- `::first-letter`: primera letra dels elements `p` \rightarrow `p::first-letter{font-size: x-large;}`

4.2.2. Aplicació d'estils. Esquema: selectors bàsics

Tipo de selector	Ejemplo	Descripción
Universal	<code>*{ color: red }</code>	Todos los elementos
Por etiqueta	<code>p{ color: red }</code>	Todos los elementos <code>p</code>
Por id	<code>#principal{ color: red }</code>	Elemento con <code>id='principal'</code>
Por clase	<code>.importante{ color: red }</code>	Elementos con <code>class='importante'</code>
Selector múltiple	<code>p, h1, .importante { color: red }</code>	Elementos <code>p</code> , <code>h1</code> o con <code>class='importante'</code>
Selector hijo	<code>section>p{color: red }</code>	Elementos <code>p</code> que sean hijos de un elemento <code>section</code>
Selector descendente	<code>section p{color: red }</code>	Elementos <code>p</code> que estén dentro de un elemento <code>section</code>
Selector adyacente	<code>header + section</code>	Elementos <code>section</code> adyacentes a un elemento <code>header</code>
Selector de atributos	<code>a[target="_blank"]</code>	Vínculos que se abren en ventana nueva

4.2.3. Herència d'estils i aplicació en cascada

Aquestes són les dues propietats fonamentals dels fulls d'estil CSS.

Herència. `html{color: red}` afecta tots els elements de la pàgina. Si per algun motiu, propi del navegador, no s'aplica per defecte aquesta propietat, podem forçar-la amb l'atribut `inherit` de la propietat `display`: `→ p{display: inherit}`

Cascada. Quan un seguit de regles afecten un element s'aplica la més específica. De vegades sol ser complicat endevinar quina és formalment la prioritària però per exemple les `inline` són més específiques que les de full d'estil interns. Podem també marca un estil com a prioritari mitjançant l'equiteta `!important`: ex: `#enrojo {color: red !important}`

4.2.4. Formatejat de pàgines amb estils. Unitats.

Les unitats de mesura més comunes a CSS són:

- Píxels: depèn de la mida de la pantalla. Cada pantalla té un nombre de píxels segons el model. No és recomanable per mides de lletres o seccions de pàgina, sí per gruix de vores o ombres.
- Em: Aquesta unitat equival a l'amplada de la lletra 'm' al tipus de lletra actual del document. És la ideal per especificar mides de lletra. El valor s'interpreta en relació a l'element contenidor.

```
<section style="font-size : 2em">  
  Sección  
  <p style="font-size : 2em">Hola</p>  
</section>
```



4.2.4. Formatejat de pàgines amb estils. Unitats

- Percentatges: permeten expressar la mida d'un element en funció del seu contenidor. Si volem un disseny adaptable, aquesta és la forma més apropiada de donar mides a les seccions de pàgina:

Altres unitats menys freqüents:

- Punts (pt) i piques (pc): per indicar mides de tipus de lletres
- Polzades (in), centímetres (cm) i milímetres (mm)

4.2.4. Formatejat de pàgines amb estils. Display.

Els elements HTML poden ser inline o block. Amb la propietat CSS display podem modificar el seu comportament per defecte:

```
/*los vínculos dentro del elemento nav serán de bloque*/  
nav a{ display : block }  
/*los elementos p serán de línea*/  
p{ display : inline }  
/*el elemento con id oculto no se mostrará*/  
#oculto{ display : none }
```

4.2.4. Formatejat de pàgines amb estils. Tipografia

font-family és la propietat que es fa servir per indicar un tipus de lletra o fins i tot una família de tipus de lletra. Com no tots els tipus de lletra es troben disponibles, si no els importem, és recomanable indicar-ne més d'un.

font-size estableix la mida del tipus de lletra, el més convenient en unitats em. També hi ha altres mides predefinides:

- **absolutes** (calculades a partir de les taules de fonts del navegador): `medium` | `xx-small` | `x-small` | `small` | `large` | `x-large` | `xx-large`
- **relatives** (en relació a l'element pare) : `smaller` | `larger`

4.2.4. Formatejat de pàgines amb estils. Tipografia

font-weight: estableix el gruix del tipus de lletra. Es pot fer servir un número de 100 a 900 o bé una de les següents paraules: `normal`, `bold`, `bolder`, `lighter`.

font-variant: només ofereix el valor `smallcaps`, per mostrar el text en majúscules petites.

font-style: ofereix els valors `italic` i `oblique`.

font: és la propietat shorthand de la tipografia. És a dir, aplega les anteriors propietats en una línia de text.

```
p.ex2 {font: italic bold 12px/30px Georgia, serif;}
```

4.2.4. Formatejat de pàgines amb estils. Text

- color: color del text.
- text-align: posiciona horitzontalment un text. left|right|justify|center
- white-space: determina com es tracten els espais en blanc i els salts de línia.
- text-decoration: underline|text-through|overline Recurs web: [Try white-space](#)
- text-transform: uppercase|lowercase|capitalize

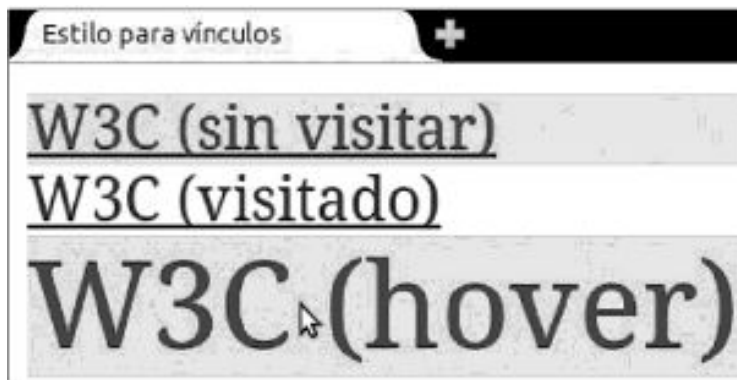
4.2.4. Formatejat de pàgines amb estils. Fons

- **background-color:** color de fons de l'element.
- **background-image:** imatge de fons de element.
- **background-position:** posiciona la imatge en relació al fons. Es pot especificar la seva posició horitzontal (right, center, left) o vertical (top, center, bottom).
- **background-repeat:** es fa servir per crear patrons de fons. Podem repetir el patró tant horitzontalment, com verticalment com en ambdós sentits.
- **background-attachment:** fixa la imatge de fons, que roman inalterada en la seva posició mentre fem scroll.

background és la propietat shorthand que inclou les propietats de fons.

4.2.4. Formatejat de pàgines amb estils. Enllaços

Amb les pseudoclasses :
link i **:visited** podem
 assignar estils diferents a
 enllaços que hagin estat
 visitat prèviament.



```
<!DOCTYPE html>
<html>
  <head>
    <meta charset='UTF-8'>
    <title>Estilo para vínculos</title>
    <style>
      /*para mostrar uno debajo de otro*/
      a{ display: block }
      /*no visitados*/
      a:link{ background-color: lightblue }
      /*visitados*/
      a:visited{ background-color: white }
      /* al pasar el ratón por encima*/
      a:hover{ font-size: 2em;
                text-decoration: none;}
    </style>
  </head>
  <body>
    <a href="http://www.w3.org">W3C [sin visitar]</a>
    <a href="http://www.w3.org/TR">W3C [visitado]</a>
    <a href="http://www.w3.org">W3C [hover]</a>
  </body>
</html>
```

4.2.4. Formatejat de pàgines amb estils. Llistes

Tenim tres propietats específiques per llistes:

Recurs web: [Try list style](#)

- **List-style-type:** permet escollir l'estil del marcador entre diverses opcions
- **List-style-image:** permet personalitzar el marcador amb una imatge pròpia

```
li{list-style-image: url("imagen.jpg") }
```

- **List-style-position:** posició del marcador respecte el text.

La propietat **list-style** és la shorthand que permet englobar les tres anteriors:

```
ul {list-style: square url("sqpurple.gif");}
```

4.2.4. Formatejat de pàgines amb estils. Taulas

Les taules solen mostrar-se amb vores, i en aquest sentit fan servir la mateixa propietat `border` que fa servir qualsevol altre element, tal i com veurem més endavant: `table, th, td{border: black solid 1px}`

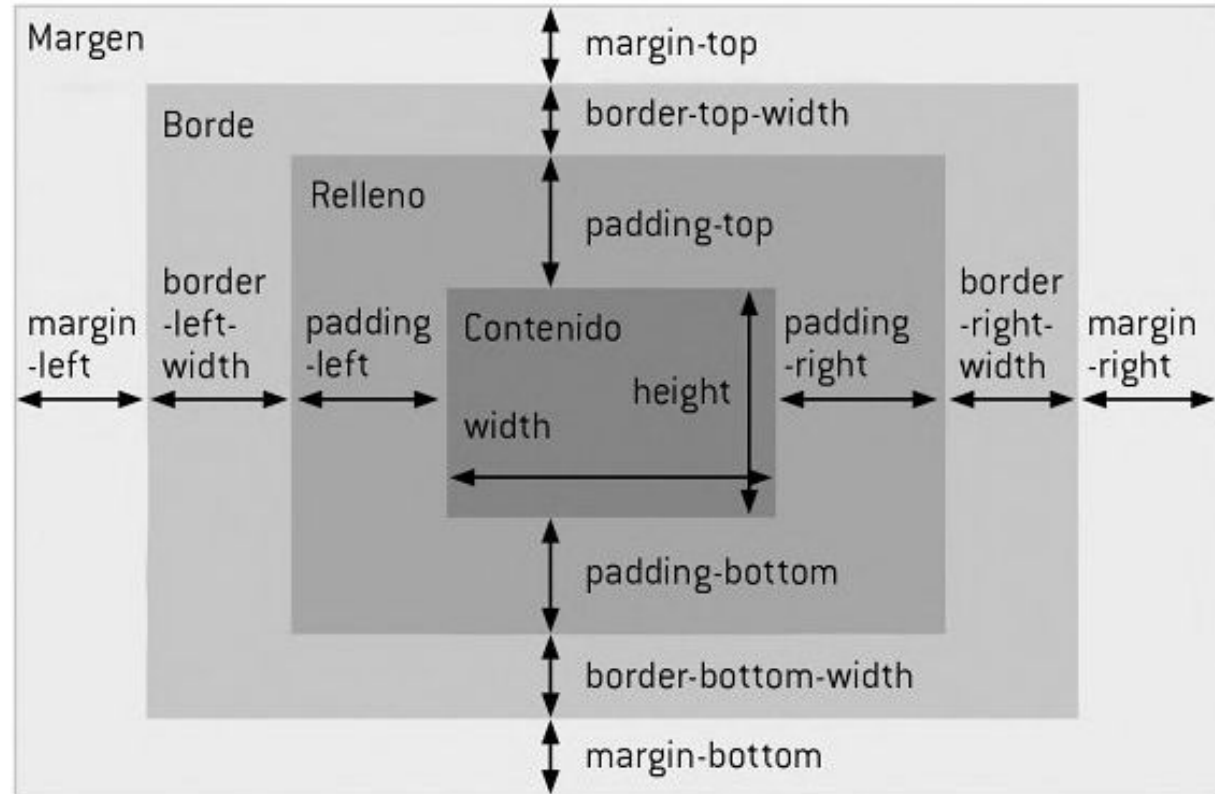
`border-collapse` permet millorar l'aspecte del contorn de la taula.

Una de les millors tècniques per millorar la visibilitat de les dades d'un taula és fer servir l'estil 'pijama', en què s'alterna el color de les files. Per a això es fa servir:

```
table { caption-side : bottom }  
tr:nth-child(odd) { background-color: yellow }  
tr:nth-child(even) { background-color: lightblue }
```

4.2.5. Estructura de pàgines. Model de caixa.

Cada element, inline o block, es representan dins una **caixa**. El seu aspecte bàsic ve definit per l'amplada, l'alçada, el marge, el padding (farciment), el border (vores o contorn) i el marge.



4.2.5. Estructura de pàgines. Model de caixa (1)

Padding: farciment que hi ha des del contingut fins al border de la caixa.

Border: envolta la caixa i el padding. És possible canviar-ne el gruix, el color i fins i tot l'estil de la línia (dashed, solid, double, ridge...) i a més per cada costat.

Margin: espai que deixa la caixa amb els elements adjacents. El seu marge és transparent i el fons el del contenidor de la caixa.

```
header{ padding: 1em }  
section{ padding-right: 0.5em;  
          padding-left: 1em;  
          padding-bottom: 1.5em;  
          padding-top: 2em; }
```

```
nav{ border: 1px dotted black }
```

```
header{ margin: 1em }  
section{ margin-right: 0.5em;  
          margin-left: 1em;  
          margin-bottom: 1.5em;  
          margin-top: 2em; }
```


4.2.5. Estructura de pàgines. Model de caixa (y 2)

Contorns arrodonits: `border-top-left-radius`, `border-top-right-radius`, `border-bottom-right-radius`, `border-bottom-left-radius` y `border-radius`.

Ombres: `div {box-shadow: 10px 10px 5px #888888;}`

```
/*todas las esquinas igual*/
header{ border-radius: 1em }
/*esquinas con radio diferente*/
section{ border-radius: 1em 2em 0.5em 1em }
/*solo las esquinas superiores*/
footer{ border-top-left-radius: 1em;
        border-top-right-radius: 1em }
```

```
section{ box-shadow: 10px 10px }
header{ box-shadow: 10px 10px 5px 5px black}
footer{ box-shadow: 10px 10px inset}
```

4.2.5. Estructura de pàgines. Posicionament

CSS no tan sols serveix per formatejar el contingut d'una pàgina HTML. També serveix per **posicionar** els seus elements, és a dir, **distribuir** cada una de les caixes que aquests elements conformen per tal de confegir un **layout** de característiques típiques de pàgina: amb una barra lateral de navegació, una capçalera, un header, un contingut central, etc.

El posicionament és la part de CSS que s'encarrega d'això, i hi ha cinc possibilitats de fer-ho: de forma **estàtica**, **relativa**, **absoluta**, **fixa** i **flotant**. Les veiem a continuació.

En els quatre primers les propietats involucrades són position, left, top, right i bottom. En el flotant, float i clear.

4.2.5. Estructura de pàgines. Posicionament

Estàtic o normal: és el mode per **defecte**. El navegador mostra els elements en el seu ordre i fa salts de línia si son de block.

Relatiu: permet desplaçar els elements en relació a la posició que tindrien amb el mètode estàtic. S'indica amb les propietats left, right, bottom i top. Si la resta d'elements actuen amb la posició estàtica o normal es poden produir solapaments.

```
section { position: relative;  
  background-color: lightblue;  
  top: 2em;  
  left: 3em; }
```

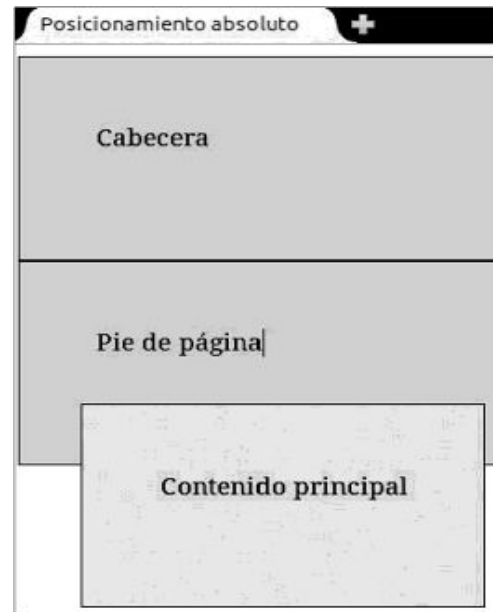


4.2.5. Estructura de pàgines. Posicionament

Absolut: s'especifica la posició en relació a un altre element, en general, el primer antecessor amb posicionament no estàtic. De tota manera, si no en troba cap agafa com a referència `<body>` tal i como passa en el nostre exemple:

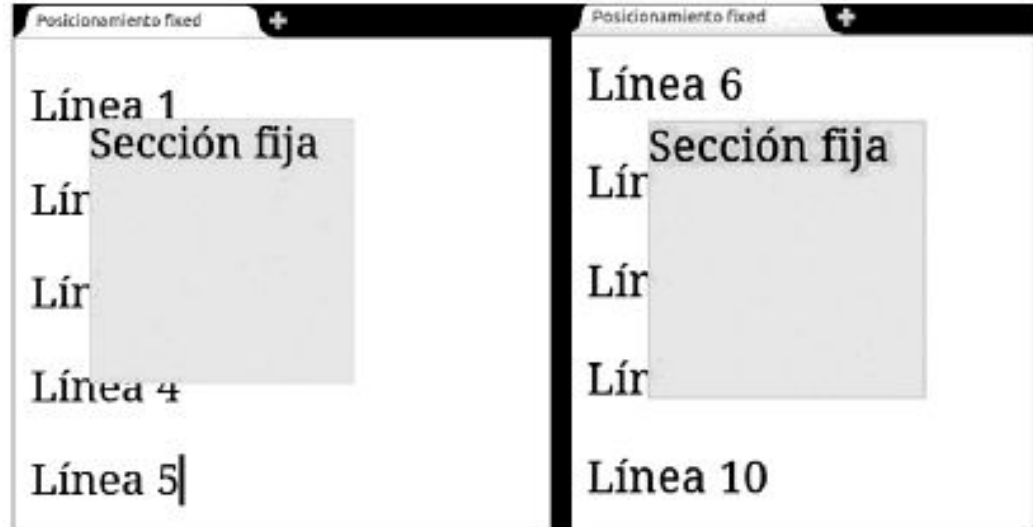
```
section { background-color: lightblue;
           position: absolute;
           top: 16em;
           left: 3em; }
```

La resta d'elements no tenen en compte el posicionament absolut i per això es solapen. A més l'amplada de l'element canvia i no s'ajusta per ocupar tot l'espai disponible, tal i com fan els elements amb posicionament estàtic i relatiu.



4.2.5. Estructura de pàgines. Posicionament

Fix: Deixar fix un element en pantalla, tot i que fem scroll. La posició s'especifica com en el mètode absolut.



Recurs web: [Try posicionament](#)

4.2.5. Estructura de pàgines. Posicionament

Flotant: es fan servir les propietats float i clear, que poden prendre els valors right, left o none (més both en el cas de clear). Aquests elements es desplacen a l'esquerra o la dreta, i la resta d'elements continuen aquesta inèrcia si no s'interromp amb la propietat clear.

Recurs web: [Try posicionament flotant](#)

Recurs web: [Try exemples layout](#)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Posicionamiento flotante</title>
    <meta charset='UTF-8'>
    <style>
      section { float:left;
                height: 6em;
                width: 4em;
                padding: 1em;
                border: 1px solid black; }
    </style>
  </head>
  <body>
    <section><p>Elemento flotado a la izquierda</p></section>
    <section><p>Elemento flotado a la izquierda</p></section>
    <section><p>Elemento flotado a la izquierda</p></section>
    <section><p>Elemento flotado a la izquierda</p></section>
  </body>
</html>
```

4.3. Disseny d'estils per diferents dispositius

Els **media queries** permeten adaptar la presentació d'una pàgina a les característiques del dispositiu amb què s'accedeix. Entre d'altres característiques podem demanar per:

- Tipus de dispositiu
- Orientació de la pantalla
- Amplada finestra navegador

Valor	Descripción
all	Todos los dispositivos (modo por defecto)
braille	Líneas braille
embossed	Impresoras braille
handheld	Móviles
print	Para impresión
projection	Proyectores
screen	Pantalla de ordenador normales
speech	Lector de pantalla
tty	Para acceso desde terminal o dispositivos con pantalla muy limitada
tv	Para pantallas de televisión (baja resolución, color)

4.3. Disseny d'estils per diferents dispositius. MQ.

Una media query (lit. 'consulta sobre el mitjà') és una expressió lògica que s'avalua com a vertadera o falsa. Si és vertadera s'aplicaran les regles associades a aquella consulta. Les propietats admeten els prefixos min i max.Ex:

- Canvia el color de fons del header quan, a qualsevol dispositiu, es posa orientació portrait (més ampla que llarga) la seva pantalla:
- Regles per quan l'amplada de pantalla sigui menor o igual a 800px:
- Condicions múltiples:

```
@media all and (orientation: portrait) {  
    header { background-color: lime }  
}
```

```
@media all and (max-width: 800px) {  
    ...  
}
```

```
@media all and (max-width: 800px) and (orientation: portrait) {  
    ...  
}
```


4.3. Disseny d'estils per diferents dispositius. MQ.

També és possible incloure media queries en vincular el full d'estils de la pàgina, de manera que només aplica quan es compleix l'expressió lògica:

```
<link rel="stylesheet" href="estilo.css" media="@media all and (min-width: 800px)">  
<!-- la siguiente hoja se aplica solo a pantallas normales-->  
<link rel="stylesheet" href="estilo2.css" media="screen">  
<!-- la siguiente hoja se aplica solo a modo impresión-->  
<link rel="stylesheet" href="estilo3.css" media="print">
```

4.3. Disseny d'estils per diferents dispositius. MQ.

Propietats per les media queries:

Propiedad	Significado	Valores	¿Acepta prefijos min y max?
width/height	Anchura y altura de la ventana del navegador	Un tamaño en cualquier unidad de medida CSS	Sí
orientation	Orientación	<i>landscape</i> si la ventana es más ancha que alta y <i>portrait</i> si es más alta que ancha	No
device-width/device-height	Anchura y altura de la pantalla del dispositivo	Un tamaño en cualquier unidad de medida CSS	Sí
aspect-ratio	Ratio entre anchura y altura en la ventana del navegador	Un número real	Sí
device-aspect-ratio	Ratio entre anchura y altura de la pantalla del dispositivo	Un número real	Sí

color	Bits por cada componente de color	Un número entero	Sí
color-index	Número de colores disponibles en la paleta	Un número entero	Sí
monochrome	Número de bits por píxel en el buffer (para pantallas monocromo)	Un número entero	Sí
resolution	Resolución de la pantalla	Las medidas de resolución son <i>dpi</i> , <i>dppx</i> y <i>dpcm</i> Ej: 300dpi	Sí
scan	Tipo de escaneo en dispositivos tipo "tv"	<i>progressive</i> <i>interlace</i>	No
grid	Tipo de pantalla (dispositivo de salida)	<i>bitmap</i> <i>grid</i>	No

4.4. Bones pràctiques per l'ús de full d'estil

- Fer servir comentaris i escriure fulls d'estils fàcils de llegir
- Fer servir fulls d'estil externs
- Fer-les servir modularment, separant codi.
- Combinar elements amb el selector múltiple per no repetir propietats
- Fer servir les shortcode
- Fer servir eines específiques per codi CSS (p. ex. EMMET)

4.4. Bones pràctiques per l'ús de full d'estil. Eines

Validadors CSS:

Com que els fulls d'estils segueixen unes regles és convenient validar-los. Es pot fer online mitjançant eines com el servei de validació de CSS de W3: <https://jigsaw.w3.org/css-validator/>

¡Enhorabuena! No error encontrado.

¡Este documento es [CSS versión 3](#) válido!

Puede mostrar este icono en cualquier página que valide para que los usuarios vean que el XHTML que puede usar para añadir el icono a su página Web:



```
<p>  
  <a href="https://jigsaw.w3.org/css-validator/check/referer">  
      
  </a>  
</p>
```

Vegeu també:

<http://csslint.net/>

4.4. Bones pràctiques per l'ús de full d'estil. Eines

Preprocessadors:

El món del desenvolupament web s'està transformant amb eines cada cop més productives i sofisticades. Una d'elles són els preprocessadors de CSS com Sass, que tenen la seva pròpia sintaxis i que un cop escrits esdevenen CSS per ser usats en producció.

4.4. Bones pràctiques per l'ús de full d'estil. Eines

Organitzadors i netejadors d'estil:

Els optimitzadors de codi són capaços de detectar no tan sols errors, sinó també inexactituds, redundàncies (regles repetides o que no s'apliquen mai), etc.

Un dels més coneguts és:

<http://www.csstidyonline.com/>

4.4. Bones pràctiques per l'ús de full d'estil. Eines

Tècniques per millorar el temps de càrrega de les pàgines web:

Per optimizar el temps de càrrega no tan sols hem d'atendre al pes del codi i de les imatges associades, sinó també al seu número, és a dir, al nombre de peticions que ha de fer el client navegador per descarregar la pàgina. És més ràpida carregar un fitxer de 10k que 10 fitxers de 1k. En aquest sentit es recomana **minificar** el codi i crear **sprites**. Hi ha eines online que faciliten aquestes tasques:

[JSCompress](#)

[SpriteCow](#)

4.4. Bones pràctiques per l'ús de full d'estil. Eines

Frameworks per CSS:

Hi ha diferents frameworks que faciliten i donen coherència en l'elaboració de pàgines web amb estils. Un dels més conegut, web responsive, és **Bootstrap**, un projecte de l'equip de desenvolupament de Twitter.

Bootstrap té un components web reutilitzables (botons, jumbos, galeries, etc.) i també estils amb classes predefinides, una interessant eina per maquetar la pàgina basada en graelles, etc.

Recurs web: [Try Bootstrap](#)

Recurs web: [Bootstrap Oficial](#)