

Llenguatge de marques i sistemes de gestió de la informació

R.A 2 : Utilitza llenguatges de marques per a la transmissió i presentació de informació a través del web analitzant l'estructura dels documents i identificant-ne els elements.

Utilització dels llenguatges de marques en entorns web

XML està pensat perquè un usuari pugui entendre les dades però no per ser llegides còmodament. Algú que no sigui expert en informàtica, en intentar veure les dades d'un document no espera veure això:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="notes.css" ?>
<institut nom="Institut Cendrassos">
  <cicles>
    <especialitat nom="Informàtica">
      <cicle id="ASIX">
        Administració de Sistemes Informàtics i Xarxes
      </cicle>
      <cicle id="SMX" >
        Sistemes Microinformàtics i Xarxes
      </cicle>
    </especialitat>
  </cicles>
  <notes>
    <classe nom="ASIX">
      <alumne aprovat="NO"><nom>Jaume Capmany</nom></alumne>
      <alumne aprovat="SI"><nom>Mohamed Polih</nom><nota>5</nota></alumne>
      <alumne aprovat="SI"><nom>Juan Pérez</nom><nota>6</nota></alumne>
      <alumne aprovat="SI"><nom>Frederic Pi</nom><nota>5</nota></alumne>
    </classe>
  </notes>
</institut>
...
```

Per contra, el que aquest usuari espera trobar-se és més aviat una cosa com:

Administració de Sistemes Informàtics i Xarxes		
ioc	institut obert de catalunya	
Jaume Capmany		5
Mohamed Polih		6
Juan Pérez		5
Frederic Pi		9
Bernat Alegria		
Samuel Barnadas		5
Felip Martí		10
Juanjo Roig		
Magdalena Serra		

Fulls d'estil

Els fulls d'estil seran els responsables de fer que els documents XML puguin ser visualitzats de maneres diferents per adaptar-se al medi per al qual seran visualitzats.

Els fulls d'estil adapten el contingut d'un document per fer que pugui ser visualitzat d'una manera alternativa.

En documents XML els fulls d'estil més usats són:

- **Cascading style sheets (CSS):** un mecanisme simple per afegir estil als documents. És el que es fa servir en HTML/XHTML.
- **XSL formatting objects (XSL-FO):** un llenguatge XML que només té com a objectiu donar format als documents XML. Es fa servir sobretot per transformar la informació en altres formats com PDF (*portable document format*).

CSS (Cascading style sheets)

CSS intenta donar una manera de definir les regles de presentació d'un document sense barrejar les dades amb el contingut i permetent que els estils es puguin aplicar a múltiples pàgines.

Els fulls d'estil CSS van ser creats per posar una mica d'ordre en l'entorn web i, per tant, es van pensar per donar format a HTML.

Han sortit diverses versions de CSS:

- **CSS1:** va ser la primera versió que va sortir i ja no està suportada pel W3C.
- **CSS2:** va expandir les característiques de CSS1 afegint-li moltes funcionalitats. Tot i que va sortir el 1998 encara és la versió més suportada pels programes i està en constant revisió.
- **CSS3:** CSS3 a més d'afegir noves funcionalitats converteix l'especificació en modular, de manera que alguns mòduls s'han convertit en recomanacions i d'altres encara no. No està gaire suportada.

CSS intenta donar una manera de definir les regles de presentació d'un document sense barrejar les dades amb el contingut i permetent que els estils es puguin aplicar a múltiples pàgines.

Navegador web

CSS va sorgir de les tecnologies web els programes que en tenen un suport més gran són els **navegadors web**.

El suport dels navegadors a CSS ha estat molt variable al llarg del temps però en general els navegadors interpretaran les regles CSS majoritàriament de la mateixa manera (però no sempre) i, per tant, en crear un full d'estil CSS s'haurà de comprovar com es veu amb diferents navegadors per assegurar-se que no té problemes.

En carregar un document XML que no tingui full d'estil els navegadors solen carregar un full d'estil per defecte que sol ser en forma d'arbre

```
- <classe>
  - <alumnes>
    - <alumne aprovat="si" delegat="si">
      <nom>Pere </nom>
      <cognom>Garcia</cognom>
    </alumne>
    - <alumne aprovat="si">
      <nom>Jordi</nom>
      <cognom>Ferrarons</cognom>
    </alumne>
    - <alumne aprovat="no">
      <nom>Manel</nom>
      <cognom>Puigdevall</cognom>
    </alumne>
    - <alumne aprovat="si">
      <nom>Frederic</nom>
      <cognom>Pi</cognom>
    </alumne>
  </alumnes>
</classe>
```

Associar un full d'estil a un document

Com que l'XHTML és HTML i a més està basat en XML, es pot associar un full d'estil CSS a XHTML fent servir qualsevol dels sistemes.

L'HTML defineix dues etiquetes per mitjà de les quals es poden definir fulls d'estils per a un document:

1. `<style>` permet que es puguin afegir regles CSS directament en un document HTML
2. `<link>` defineix el full d'estil com un fitxer a part amb l'element `<link>`

L'atribut **type** sempre serà el valor MIME de CSS (`text/css`).

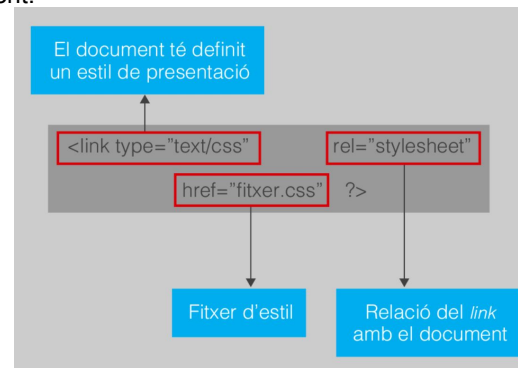
rel serveix per determinar quina és la relació de l'arxiu que estem enllaçant amb el document.

I l'atribut **href** és el que es fa servir per enllaçar amb l'arxiu que conté les regles CSS.

Normalment se li posa extensió `.css` però pot tenir l'extensió que es vulgui.

Totes dues regles s'han de definir dins de l'etiqueta `<head>` del document.

```
<html>
  <head>
    <title>Test</title>
    <style type="text/css">
      ... regles ...
    </style>
  ...
```



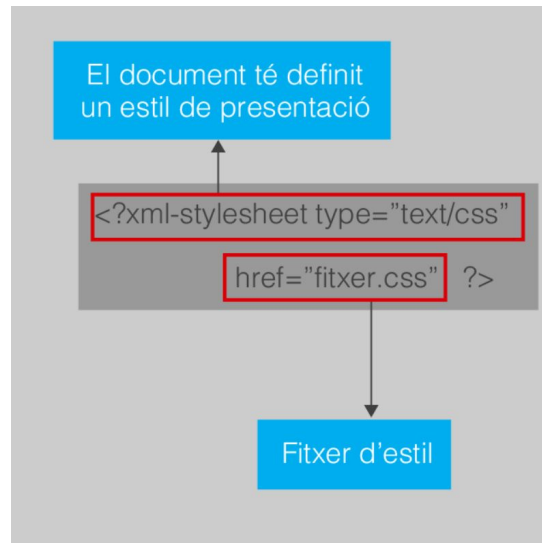
Associar un full d'estil a un document XML (I)

L'XML permet a l'usuari especificar les seves etiquetes pròpies i no en porta cap de predefinida, o sigui que en XML les etiquetes `<style>` i `<link>` no existiran, o si existeixen estaran dotades d'un significat diferent del que tenen en HTML.

Per aquest motiu els fulls d'estil en documents basats en XML es defineixen fent servir la instrucció de procés `xml-stylesheet`

Tant `<link>` com `<?xml-stylesheet ?>` permeten tota una sèrie d'atributs a part dels dos bàsics (`type` i `href`) que tenen per objectiu passar més informació al programa per representar el document

Atribut	Descripció
title	El nom o el títol del full.
media	Indica sota quin medi s'ha de carregar el <u>CSS</u> .
charset	Codi de caràcters que fa servir el full d'est'l.
alternate	Indica si el <u>CSS</u> és preferit o no.



Associar un full d'estil a un document XML (II)

Dels atributs que hi ha el més usat és **media**, que permet que s'apliqui un estil o un altre en funció del dispositiu amb què s'està intentant visualitzar el document.

Gràcies a l'atribut **media** es poden fer fulls d'estil que s'aplicaran en funció de quin sigui el dispositiu que vol llegir el document.

valor	dispositius
screen	Per ser visualitzat en un monitor de PC
print	Sortida en paper: impressores, etc.
tty	Sortida en terminals, teletips, etc.
handheld	PDA, telèfons mòbils, etc.
braille	La sortida es generarà en dispositius tàctils per a cecs.
aural	Per sortir en lectors de pantalla, sintetitzadors de veu, etc.
tv	Televisors, consoles de videojocs, etc.

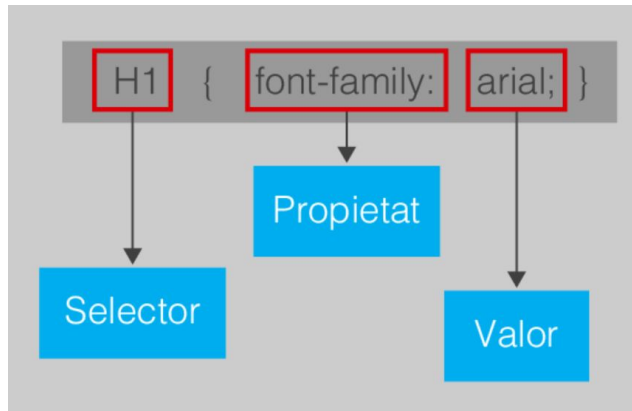
Si no s'especifica cap valor a l'atribut **media** s'està indicant que el que s'està definint és el full d'estil per defecte.

En cas d'especificar diversos fulls d'estil es barrejaran les regles de tots i en cas de conflicte es faran servir les definicions especificades més tard.

```
<?xml-stylesheet href="web.css"?>
<?xml-stylesheet media="print" href="imprimir.css"?>
<?xml-stylesheet media="handheld" href="mobils.css"?>
```


Regles CSS

Els documents de CSS normalment seran una llista de regles en què cada una té una forma semblant a la que es mostra en la figura:



CSS permet que en una regla es puguin especificar tantes propietats com calgui, sempre que es vagin separant l'una de l'altra amb un punt i coma (;).

```
persona {  
  color: red;  
  margin: 5px;  
}
```

Regles CSS - Selectors

Els selectors són un dels components bàsics de CSS, ja que serveixen per determinar quines regles s'han d'aplicar a cada element i quines no.

Tipus:

- Universal
- D'etiquetes
- De fills
- De descendents
- De germans adjacents
- Per ID
- De classes
- D'atributs
- De pseudoclasses

Regles CSS - Selectors - Tipus (I)

Selector universal

El selector universal es representa amb un '*' i es fa servir per seleccionar tots els elements d'un document.

```
* { color: red; }
```

```
<alumnes>  
  <nom>Pere</nom>  
  <cognom>Garcia</cognom>  
</alumnes>
```

Pere Garcia

Selector d'etiquetes

Es pot seleccionar un element específic simplement amb el nom. En aquest codi se seleccionen totes les etiquetes **nom** del document:

```
nom { color: red; }  
cognom { color: blue; }
```

```
<classe>  
  <alumnes>  
    <alumne>  
      <nom>Pere</nom>  
      <cognom>Garcia</cognom>  
    </alumne>  
    <alumne>  
      <nom>Manel</nom>  
      <cognom>Puig</cognom>  
    </alumne>  
  </alumnes>  
</classe>
```

Pere Garcia Manel Puig

Regles CSS - Selectors - Tipus (II)

Selector de fills

El selector de fills es fa servir per seleccionar les etiquetes filles directes d'una altra. Es fa servir el símbol `>` per definir el pare i el fill.

```
alumne > nom { color: red; }
```

```
<classe>
  <professor>
    <nom>Marcel</nom>
    <cognom>Puig</cognom>
  </professor>
  <alumnes>
    <alumne>
      <nom>Pere </nom>
      <cognom>Garcia</cognom>
    </alumne>
    <alumne>
      <nom>Joan</nom>
      <cognom>Ferrarons</cognom>
    </alumne>
  </alumnes>
</classe>
```

Marcel Puig **Pere** Garcia **Joan** Ferrarons

Perquè Marcel no és vermell?

Selector de descendents

Una possibilitat alternativa consisteix a definir que es vol que l'estil s'apliqui a algun dels descendents d'un element.

```
alumnes cognom { color: red; }
```

Marcel Puig Pere **Garcia** Joan **Ferrarons**

Regles CSS - Selectors - Tipus (II)

Selector de germans adjacents

El selector de germans adjacents serveix per seleccionar elements que comparteixin el mateix pare i estiguin situats immediatament l'un rere l'altre.

```
delegat+nom { color: red; }
```

```
<classe>
  <alumne>
    <delegat/>
    <nom>Pere</nom>
  </alumne>
  <alumne>
    <nom>Joan</nom>
  </alumne>
  <alumne>
    <nom>Marcel</nom>
  </alumne>
</classe>
```

Pere Joan Marcel

<delegat/> és una etiqueta xml que té contingut buit però sí que indica qui és el delegat

Regles CSS - Selectors - Tipus (III)

Selector per ID

L'atribut `id` es fa servir per poder identificar elements concrets dins d'un document XML o HTML. Per tant, l'atribut `id` només especificarà un sol element dins d'un document.

```
alumne#delegat { color:red; }
```

```
<classe>
  <professor>
    <nom>Marcel</nom>
    <cognom>Puig</cognom>
  </professor>
  <alumnes>
    <alumne id="delegat">
      <nom>Pere </nom>
      <cognom>Garcia</cognom>
    </alumne>
    <alumne>
      <nom>Joan</nom>
      <cognom>Ferrarons</cognom>
    </alumne>
  </alumnes>
</classe>
```

|Marcel Puig Pere Garcia Joan Ferrarons

Regles CSS - Selectors - Tipus (IV)

Selector de classes

Podria passar que, en algun document, a etiquetes que tenen el mateix nom a vegades els volem aplicar un estil i a vegades un altre.

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

```
<p class="normal">  
Els fitxers de marques només són arxius de text que  
es poden obrir amb un <span class="important">editor de textos</span>  
senzill.  
</p>  
  
<p class="important">  
Tot i que els editors especialitzats ens ajudaran a no  
cometre errors.  
</p>  
  
<p class="normal">  
Per tant, la creació de documents XML està  
a l'abast de tothom.  
</p>
```

Els fitxers de marques només són arxius de text que es poden obrir amb un editor de textos senzill.

Tot i que els editors especialitzats ens ajudaran a no cometre errors.

Per tant, la creació de documents XML està a l'abast de tothom.

Regles CSS - Selectors - Tipus (V)

Selector d'atributs

També hi ha un selector que permet triar les etiquetes en funció dels seus atributs.

```
<classe>
  <alumnes>
    <alumne aprovat="si" delegat="si">
      <nom>Pere</nom>
      <cognom>Garcia</cognom>
    </alumne>
    <alumne aprovat="si">
      <nom>Jordi</nom>
      <cognom>Ferrarons</cognom>
    </alumne>
    <alumne aprovat="no">
      <nom>Manel</nom>
      <cognom>Puigdevall</cognom>
    </alumne>
    <alumne aprovat="si">
      <nom>Frederic</nom>
      <cognom>Pi</cognom>
    </alumne>
  </alumnes>
</classe>
```

```
alumne[delegat] { color: green; }
```

Pere Garcia Jordi Ferrarons Manel Puigdevall
Frederic Pi

```
alumne[aprovat="si"] { color:blue; }
alumne[aprovat="no"] { color:red; }
```

Pere Garcia Jordi Ferrarons Manel Puigdevall
Frederic Pi

Regles CSS - Selectors - Tipus (VI)

Selectors de pseudoclasses

La idea d'aquests selectors és permetre seleccionar elements a partir d'una condició que no sigui el seu nom.

Les pseudoclasses se separen del nom de l'element al qual s'aplicaran amb dos punts (:).

Hi ha diverses pseudoclasses disponibles, que es poden veure en la taula

Pseudoclasse	Significat
first-child	Permet seleccionar el primer element de la classe actual.
link	Selecciona l'element si és l'origen d'un enllaç ja visitat.
visited	Per als enllaços visitats d'un tipus determinat.
active	Per als elements que s'estan activant (essent clicats, etc.).
linking	Per a enllaços en XLink.
hover	Elements que tenen el cursor al damunt.
focus	Tria l'element que té el focus.
lang	Permet triar els elements amb l'etiqueta especificant un idioma determinat.

Exemple:

```
alumnes:first-child {  
    color: red;  
}
```

```
<alumnes>  
  <nom>Frederic Pi</nom>  
  <nom>Pere Garcia</nom>  
  <nom>Manel Puigdevall</nom>  
</alumnes>
```

Regles CSS - Selectors - Resolució conflictes

Quan a un element se li pot aplicar la mateixa propietat segons regles diferents i aquestes es contradiuen, el CSS defineix que la regla per aplicar ha de ser la més específica.

Per exemple, com que les regles `nom` de l'exemple següent es contradiuen s'agafarà la primera regla perquè és més concreta –defineix més clarament a quins elements s'està fent referència:

```
alumnes nom { color: red; }  
nom { color: blue; }
```

Si hi ha regles que són igual d'específiques, aleshores **s'aplicarà la que s'hagi especificat més tard.**

Regles CSS - Selectors - Declaracions i herència

La declaració és el segon terme de les regles CSS i és la part que s'encarregarà de donar el format a les etiquetes. Es defineix entre claus "{ }" i estarà formada per una o més propietats. A cada propietat se li assignarà quin és el valor desitjat.

```
alumne { color: red; }
```

Podem afegir tantes propietats com calguin en una regla. Es poden fer declaracions tan llargues com calgui:

```
alumne { color: red;  
padding: 35px;  
border-style: solid;  
border-color: black;  
border-width: 1px;  
}
```

La primera lletra de CSS és de *cascading* ('cascada', en català), i indica que les propietats de les etiquetes no cal que es vagin repetint indefinidament, ja que s'hereten. Això vol dir que si apliquem unes propietats a un element, els seus elements fills automàticament les adquireixen.

```
<alumni>  
  <alumne>  
    <nom>Frederic</nom>  
    <cognom>Pi</cognom>  
  </alumne>  
</alumni>
```

```
alumne { color: red; }
```

(nom i cognom són vermells)

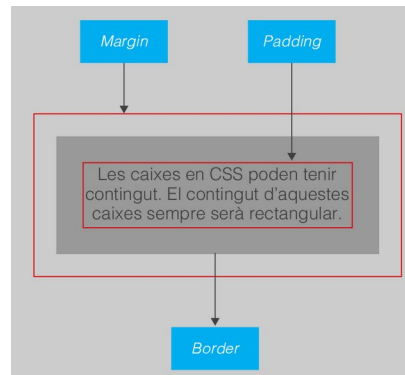
```
alumne { color: red; }  
cognom { color: blue; }
```

(especifica cognom blau)

CSS - Model de caixes

El CSS tracta tots els elements com si estiguessin dins d'una caixa rectangular que envolta el contingut. En aquesta caixa es poden definir una sèrie de marges:

- **padding**: marge intern entre el contingut i la caixa.
- **margin**: marge entre la caixa i les altres caixes adjacents.

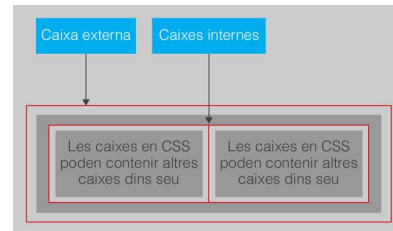


Però com que tant l'XML com l'HTML fan servir etiquetes niuades normalment tindrem caixes que estan dins d'altres caixes

En XML quan es posicionen les caixes si no es diu el contrari sempre es col·loquen una al costat de l'altra fins que s'acaba l'espai horitzontal. Quan no queda més espai es representa la caixa següent a sota

En HTML algunes etiquetes fan com les d'XML i d'altres no permeten tenir altres caixes al costat.

Podem canviar aquest comportament amb **display** i **posicionant les caixes**



CSS - Model de caixes - display

Els elements tenen una propietat `display`, que entre d'altres coses determina com es posiciona la caixa respecte a les altres:

- `block`: treballa amb blocs de contingut. Fa que les caixes acabin com si tinguessin un salt de línia darrere seu. Per tant, es trenca amb el posicionament per defecte.
- `inline`: permet que les etiquetes flueixin amb la resta del contingut. Aquest és el comportament per defecte en representar contingut XML.

```
<alumnes>
  <nom>Pere</nom>
  <nom>Frederic</nom>
  <nom>Manel</nom>
</alumnes>
```

Pere Frederic Manel

Si ara apliquem

```
nom { display:block; }
```

obtenim

Pere
Frederic
Manel

CSS - Model de caixes - posicionament de caixes

Posicionament normal

El posicionament normal és el posicionament per defecte i consisteix a anar posant les caixes una al costat de l'altra horitzontalment fins que no hi hagi espai per posar-ne una altra. Quan això passi les següents es posen al principi de la línia següent i tornem a començar.

```
alumne { margin: 20px; }
```

Amb aquest exemple, **els marges horitzontals de les caixes es respecten i s'acumulen**. Per tant, dos elements `alumne` que tinguin la definició de format següent estaran separades 40 píxels. En canvi **verticalment els navegadors mesclen els marges**, de manera que no se sumen els marges sinó que **s'agafa el més gran**. Per tant, amb la mateixa definició anterior les caixes estaran separades només 20px verticalment.

Posicionament relatiu

Si se segueix el posicionament normal les caixes ja tenen un lloc en el qual posicionar-se però es pot canviar aquest lloc convertint el posicionament de la caixa amb `position: relative`. Amb això s'aconsegueix que la caixa es mogui de la posició que li tocava indicant-li en quina quantitat.

Ens podem moure de les posicions normals amb `left`, `right`, `top`, `bottom`.

```
<alumn>  
  <nom>Pere</nom>  
  <nom delegat="si">Frederic</nom>  
  <nom>Manel</nom>  
</alumn>
```

```
nom { display: block; }  
nom[delegat] {  
  position: relative;  
  left: 30px;  
}
```



Pere
Frederic
Manel

CSS - Model de caixes - posicionament de caixes

Posicionament flotant

El posicionament flotant ens permet col·locar la caixa de manera que quedi surant en la part de la pantalla que li especifiquem.

La propietat que es fa servir per marcar posicionament com a flotant és **float**, que pot tenir els valors de la taula

Valor	Significat
left	La caixa es quedarà tant a l'esquerra com pugui.
right	La caixa es quedarà tant a la dreta com pugui.
none	La caixa no és flotant.
inherit	Ha de fer el mateix que faci el seu element pare.

```
<classe>
  <professor>
    <nom>Marcel Puig</nom>
  </professor>
  <alumnes>
    <nom>Pere Gallerí</nom>
    <nom delegat="si">Frederic Pi</nom>
    <nom>Manel Fontcoberta</nom>
    <nom>Llucia Martí</nom>
  </alumnes>
</classe>
```

```
professor {
  float: right;
  width: 50%;
  color: red;
}
```

Pere Gallerí
Frederic Pi
Manel Fontcoberta
Llucia Martí

Marcel Puig

CSS - Model de caixes - posicionament de caixes

Posicionament absolut

El posicionament absolut serveix per posicionar exactament la caixa en un lloc determinat. Les caixes que es posicionen amb posicionament absolut surten del flux normal de representació.

Per fer posicionament absolut es fa servir l'atribut `position`, que només pot tenir dos valors: `absolute` i `fixed`

Valor	Funcionament
<code>absolute</code>	Posiciona la caixa exactament en la posició especificada.
<code>fixed</code>	La caixa es col·loca en una posició fixa de la pantalla. Això fa que no es mogui mai encara que el document pugi i baixi.

```
professor {  
  position: absolute;  
  left: 200px; top: 20px;  
  width: 100px;  
  height: 50px;  
  background-color: blue;  
}
```

Pere Gallerí
Frederic Pi
Manel Fontcoberta
Lucia Martí

Marcel Puig

Les caixes amb posicionament absolut, en no estar dins del flux normal de representació, poden quedar per sota o per sobre de les altres, i per aquest motiu se solen especificar amb l'amplada i alçada.

Pere Gallerí
Frederic Pi
Manel Fontcoberta
Lucia Martí

CSS - Propietats

Podem dividir les propietats en tres grans grups:

1. **Propietats de caixa:** permetran definir característiques de la representació de les caixes que contenen els elements.
2. **Propietats de text:** ens permetran definir com ha de ser representat el contingut textual del document.
3. **Propietats de color:** definiran el color en què s'han de representar els continguts i les caixes.

Colors

Per especificar colors es poden fer servir diversos sistemes

```
nom {  
  color: rgb(250,20,10);  
}  
cognom {  
  color: rgb(80%, 40%, 0%);  
}
```

```
nom {  
  color: #cc6600;  
}
```

```
alumne {  
  color:red;  
}
```

Mides

Quan alguna propietat demana que se li especifiqui una mida es poden especificar de dues maneres diferents:

- **Valors absoluts** són aquelles en les quals s'especifica la mida exacta en la qual es vol que es representi aquella propietat.
- **Valors relatius**

Les mides en **valors absoluts** És corrent definir-ho en píxels (px) però també es poden fer servir una sèrie de valors de text (**small, medium, large...**)

CSS - Propietats Caixa (I)

El CSS representa totes les marques com a caixes rectangulars. Per defecte les caixes tenen unes propietats per defecte però es poden canviar simplement redefinint-les. Es poden veure algunes de les propietats:

Propietat	Ús
margin	Serveix per definir quin és l'espai entre caixes. Per defecte és 0.
padding	Permet definir el marge intern d'una caixa. Per defecte és 0.
border	Permet definir si s'ha de representar la línia que envolta la caixa. Per defecte no es fa.
width	Permet definir l'amplada d'una caixa.
height	Permet definir l'alçada d'una caixa.

Exemple d'ús:

```
nom {  
  border: 1px black solid;  
  padding: 10px;  
}
```

Frederic Pi

Pere Garcia

Maria Bosch

CSS - Propietats Caixa (II)

Padding : Es poden definir els valors dels marges d'un costat en concret especificant els valors de cada costat (dalt, dreta, baix, esquerra: sempre en aquest ordre, com les agulles del rellotge).

```
nom {  
  padding: 30px 50px 1px 1px;  
}
```

=

```
nom {  
  padding-left: 1px;  
  padding-top: 30px;  
  padding-right: 50px;  
  padding-bottom: 1px;  
}
```

Frederic Pi

Pere Garcia

Border: Permet especificar el gruix de la línia, el color i el tipus de línia que es vol. També es poden especificar les línies de cada un dels costats de manera individual

```
nom {  
  border-top: 2px red solid;  
  border-bottom: 2px blue dashed;  
  border-left: 1px green dotted;  
  border-right: 1px yellow outset;  
}
```

Frederic Pi

Pere Garcia

CSS - Propietats Text (I)

Un dels aspectes fonamentals a l'hora de donar format serà poder-ho fer al contingut de text dels documents. En CSS això es fa de dues maneres:

- **Propietats de tipus:** fan referència a com es veuran les lletres
- **Propietats text:** fan referència a com es mostrarà el bloc de text en conjunt

Propietats de tipus

Aquestes propietats permetran canviar tot el que fa referència al tipus de lletra que es fa servir per mostrar el text. Es poden canviar la majoria dels paràmetres de cop fent servir la propietat **font**.

Propietat	Ús
font-family	Permet especificar quin tipus de lletra fem servir i la família a la qual pertany. Normalment se n'hi posen diverses per si l'equip no té instal·lat el tipus de lletra demanat.
font-size	Serveix per definir la mida del tipus de lletra que es fa servir.
font-weight	Defineix el pes de la lletra. Generalment es fan servir negretes (bold) o rodones (normal), però hi ha més valors possibles.
font-style	Permet definir l'estil de lletra. Els valors més corrents són cursiva (italic) o obliqua (oblique).
font-variant	Sobretot es fa servir per convertir el contingut a versaletes (small-caps).

```
nom {  
  font: italic bold 12px Verdana, Geneva, Arial, sans-serif;  
}
```

CSS - Propietats Text (II)

Exemple:

```
<alumnes>
  <alumne>
    <nom>Frederic</nom>
    <cognom>Pi</cognom>
  </alumne>
  <alumne>
    <nom>Pere</nom>
    <cognom>Garcia</cognom>
  </alumne>
</alumnes>
```

```
alumne {
  display:block;
}

nom {
  font:italic bold 30px Georgia, serif;
}

cognom {
  font-family:Times,serif;
  font-size:15px;
}
```

Frederic Pi

Pere Garcia

CSS - Propietats Text (III)

Propietats de text

A més de les propietats de les lletres també es poden canviar les propietats de com es visualitzarà el text. Podem canviar l'orientació, el color, l'espaiat entre línies, etc.

Propietat	Ús
word-spacing	Espaiat entre paraules.
letter-spacing	Espaiat entre lletres.
text-decoration	Permet decorar el text amb subratllats o línies diverses.
vertical-align	Alineació vertical del contingut dins de la caixa.
text-align	Alineació del text en la caixa.
line-height	Alçada de la línia de text.
color	Color amb què es mostrarà el text.

```
nom {  
  text-decoration:underline;  
}  
  
cognom {  
  letter-spacing:20px;  
}
```

Frederic P i
Pere G a r c i a

CSS - Propietats de color i imatge (I)

Colors: Per especificar colors es poden fer servir diversos sistemes

```
nom {  
  color: rgb(250,20,10);  
}  
cognom {  
  color: rgb(80%, 40%, 0%);  
}
```

```
nom {  
  color: #cc6600;  
}
```

```
alumne {  
  color: red;  
}
```

```
nom {  
  width: 50px;  
}  
cognom {  
  width: large;  
}
```

Mides: Quan alguna propietat demana que se li especifiqui una mida es poden especificar de dues maneres diferents:

- **Valors absoluts:** són aquelles en les quals s'especifica la mida exacta en la qual es vol que es representi aquella propietat. És corrent definir-ho en píxels (px) però també es poden fer servir una sèrie de valors de text (`small`, `medium`, `large`...).
- **Valors relatius:** s'expressa l'increment o decrement de la mida per defecte del tipus de lletra de la caixa. Podem definir aquestes mides en percentatge (%) o en esalat (em).

```
alumne {  
  width: 100px;  
}  
nom {  
  width: 50%;  
}  
cognom {  
  width: 1.2em;  
}
```

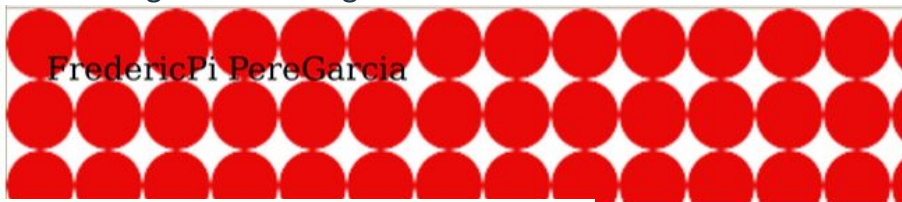
CSS - Propietats de color i imatge (II)

El color de fons es defineix amb **background-color** i s'hi especifica el color que es vol:

```
nom {  
  background-color: blue;  
}
```

```
nom {  
  background-image: url('fons.png');  
}
```

El funcionament per defecte és omplir el fons repetint la imatge tantes vegades com calgui fins a cobrir-lo tot



Propietats fons de caixa:

Propietat	Ús
background-repeat	Defineix de quina manera s'ha de repetir la imatge, si s'ha de fer.
background-position	Es fa servir per posicionar la imatge en un punt concret.
background-attachment	Defineix si el fons es mourà en desplaçar la imatge o no.

CSS - Amagar contingut

A vegades pot interessar que no tot el contingut del document XML sigui mostrat, ja sigui perquè no aporta res a un lector humà o per altres raons.

Això es pot fer mitjançant dues propietats:

- **visibility** permet fer que algun element no es mostri quan es defineix en un determinat element.

```
<persona>
  <nom>Pere</nom>
  <cognom>Gallerí</cognom>
</persona>
```

```
persona {
  display: block;
}

nom {
  visibility: hidden;
}
```



No es mostra el nom si aquest té el valor **visibility: hidden**. Aplicant **display: block**; a l'element **<persona>**, ens assegurem que aquest es comporti com un element de bloc, és a dir, ocupa tot l'ample disponible i es mostri en una nova línia.

- **display: none** Amb aquesta propietat el que s'aconsegueix és que l'etiqueta no sigui mostrada.

```
persona {
  display: block;
}

nom {
  display: none;
}
```



CSS - Taules

L'atribut `display` de CSS permet crear taules des del full d'estil amb les etiquetes. Això fa que es pugui associar cada una de les etiquetes del document a files, cel·les, etc.

Valor	Descripció
<code>display:table</code>	Defineix que comença una taula.
<code>display:table-row</code>	Defineix l'etiqueta com una línia de la taula.
<code>display:table-cell</code>	L'etiqueta amb aquest valor serà una cel·la.

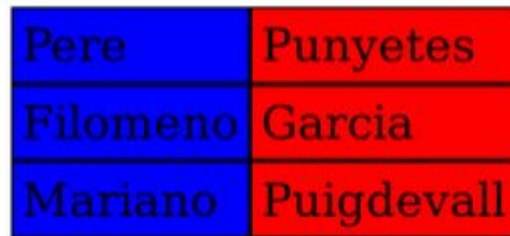
```
<?xml version="1.0" encoding="UTF-8"?>
<classe>
  <alumnes>
    <alumne>
      <nom>Pere</nom>
      <cognom>Punyetes</cognom>
    </alumne>
    <alumne>
      <nom>Filomeno</nom>
      <cognom>Garcia</cognom>
    </alumne>
    <alumne>
      <nom>Mariano</nom>
      <cognom>Puigdevall</cognom>
    </alumne>
  </alumnes>
</classe>
```

```
alumnes {
  display:block;
  display:table;
  margin:10px;
}

alumne {
  display:table-row;
}

nom, cognom {
  display:table-cell;
  border: 1px solid #000000;
  padding: 3px;
  border-spacing: 0px;
}

nom { background-color:blue; }
cognom {background-color:red; }
```



Pere	Punyetes
Filomeno	Garcia
Mariano	Puigdevall

CSS - Llistes

A partir de determinades etiquetes es poden crear llistes de resultats fent servir la propietat `display:list-item`. Normalment aquests elements es mostren amb un cercle davant del seu valor

```
<persones>
  <persona>
    <nom>Pere Gallerí</nom>
  </persona>
  <persona>
    <nom>Frederic Pi</nom>
  </persona>
</persones>
```

```
persona {
  display:block;
  margin-left:40px;
}
persona {display:list-item; }
```

- Pere Gallerí
- Frederic Pi

Propietat	Significat
list-style-type	Permet definir de quin tipus serà la llista. Pot tenir una sèrie de valors numèrics o símbols. Per exemple: decimal, circle, square, disc ...
list-style-position	Defineix si les marques han d'aparèixer abans o en el flux de text.
list-style-image	Permet definir una imatge com a marca de la llista.

CSS - Pseudoselectors

Es pot afegir text nou en mostrar un document XML fent servir els **pseudoselectors**. Els pseudoselectors permetran afegir informació complementària a les etiquetes. Els més usats:

pseudo-selector	Descripció
:first-letter	Permet definir estils especials a la primera lletra del contingut seleccionat pel selector.
:first-line	Es fa servir per donar format a la primera línia d'un contingut de dades de bloc.
:before	Permet afegir contingut abans d'un element.
:after	Permet afegir contingut després d'un element.

```
<persones>
  <professor>
    <nom>Pere Gallerí</nom>
  </professor>
  <alumnes>
    <nom>Frederic Pi</nom>
    <nom>Filomena Garcia</nom>
    <nom>Manel Puigdevall</nom>
  </alumnes>
</persones>
```

```
professor nom:before {
  content: "Professor: ";
  color: red;
}
nom {display: block;}
```

El CSS no té cap manera senzilla de mostrar les dades que hi ha en els atributs. Fent servir els pseudoselectors es podran mostrar els valors dels atributs fent servir la funció `attr()`.

```
<alumnes>
  <nom>Frederic Pi</nom>
  <nom carrec="delegat">Pere Garcia</nom>
  <nom>Manel Puigdevall</nom>
  <nom carrec="subdelegat">Maria Bosch</nom>
</alumnes>
```

```
alumnes {
  display: block;
}

nom[carrec]:after
{
  font-size: 8px;
  content: " (" attr(carrec) ")";
}
```

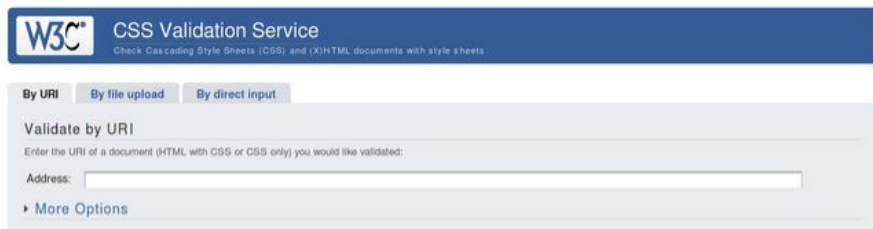
Frederic Pi
Pere Garcia (delegat)
Manel Puigdevall
Maria Bosch (subdelegat)

CSS - Validació

Generalment els navegadors web que es troben amb una línia que conté algun error (una propietat mal escrita, unitats mal definides, etc...) o que no entenen (perquè no l'han implementat) la solen ignorar.

En conseqüència, una de les coses importants per no tenir problemes de representació del CSS és que el document segueixi tant com pugui els estàndards.

Els validadors de CSS ens permetran comprovar que el document de regles que estem definint no incompleix cap estàndard ni conté errors.



The image shows the W3C CSS Validation Service interface. At the top, there is a blue header with the W3C logo and the text "CSS Validation Service" and "Check Cascading Style Sheets (CSS) and (X)HTML documents with style sheets". Below the header, there are three tabs: "By URI", "By file upload", and "By direct input". The "By URI" tab is selected. Under the "By URI" tab, there is a section titled "Validate by URI" with the instruction "Enter the URI of a document (HTML with CSS or CSS only) you would like validated:". Below this instruction, there is a text input field labeled "Address:". At the bottom of the form, there is a link that says "More Options".

<https://jigsaw.w3.org/css-validator/>

HTML / XHTML

A pesar que es pot representar la informació fent servir XML i CSS, el més habitual sol ser representar-la en algun tipus de llenguatge de marques que estigui pensat per a la representació. El llenguatge de marques per excel·lència a l'hora de presentar la informació és l'HTML (*hypertext markup language*).

L'HTML és un llenguatge de marques que permet publicar informació molt diversa pensant en l'estructura dels documents i en com es representaran les dades que contenen. Ha tingut tant d'èxit que pràcticament s'ha convertit en una forma de comunicació universal.

Actualment la creació d'una pàgina web consistirà a definir l'estructura del document per mitjà d'HTML i definir-ne el format per mitjà dels fulls d'estil CSS.

L'HTML està pensat per definir l'estructura d'un document de text a partir d'una sèrie d'etiquetes predefinides formades per un nom envoltat dels símbols `<` i `>`. Cada una de les etiquetes servirà per definir l'estructura d'un document a part d'aportar-li informació semàntica sobre el contingut respecte al document.

La separació de la informació de l'estructura de les dades és un dels components clau per donar dinamisme a les pàgines web.

L'HTML intenta definir l'estructura d'un document de manera que sigui senzill per a un programa interpretar què són cada una de les dades que va trobant. Per fer-ho fa servir tota una sèrie d'etiquetes que indicaran quin és el paper que tenen cada una de les dades en el document.

HTML / XHTML - Títols

Es poden definir **títols** fent servir les etiquetes `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` i `<h6>`. Aquestes etiquetes serveixen per definir diferents nivells de títols en un document.

Els navegadors, en carregar un document HTML, ja “comprenen” per defecte quin és el significat dels diferents nivells de títols i els representen convenientment: amb mides diferents i amb un salt de línia després

```
<h1>Primer títol</h1>  
<h2>Segon títol</h2>  
<h3>Tercer títol</h3>  
<h4>Quart títol</h4>  
<h5>Cinquè títol</h5>  
<h6>Sisè títol</h6>
```

Primer títol

Segon títol

Tercer títol

Quart títol

Cinquè títol

Sisè títol

HTML / XHTML - Paràgrafs

Una de les parts bàsiques que formen un document solen ser els **paràgrafs**, que en HTML es defineixen fent servir l'etiqueta `<p>`.

Per defecte els paràgrafs acaben amb un salt de línia encara que no s'indiqui explícitament en el document

```
<h1>Primer títol</h1>  
<p>Aquest és el primer paràgraf</p><p>Segon paràgraf</p>
```

Primer títol

Aquest és el primer paràgraf

Segon paràgraf

HTML / XHTML - Agrupació de contingut

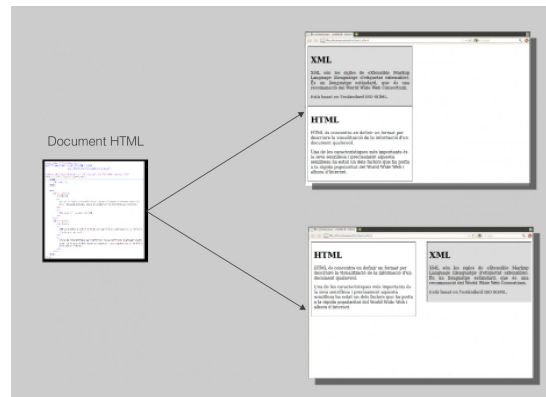
Gairebé sempre, els títols i un nombre determinat de paràgrafs d'un text estan lligats per un contingut semàntic comú. Però com que la representació de les dades pot reordenar el contingut del document cal algun sistema per mantenir aquestes parts unides.

L'HTML permet agrupar continguts que tenen algun tipus de lligam per mitjà de l'etiqueta `<div>`.

Generalment els elements `<div>` se solen identificar per mitjà d'atributs `id` o `class` perquè el full d'estil hi pugui fer referència de manera més senzilla.

Un cop es té el contingut agrupat convenientment es farà més fàcil per als fulls d'estil presentar el document reordenant els continguts d'un document sense que se'n perdi la coherència interna

```
<div>
  <h1>Titol 1</h1>
  <p>Contingut de text</p>
  <p>Més contingut de text</p>
</div>
```



HTML / XHTML - Ressaltar text

Hi ha tota una sèrie d'etiquetes més que tenen diverses funcions per a parts concretes del text:

- Definir text com a citacions: `<blockquote>`, `<q>` i `<cite>`.
- Marcar que s'ha de donar èmfasi a una part del text: `` i ``
- Definir contingut com abreviacions, acrònims i definicions: `<abbr>`, `<acronym>`, `<dfn>`
- Permetre l'entrada de text preformatat: `<pre>` i `<code>`

Però moltes d'aquestes etiquetes poden ser substituïdes per l'etiqueta ``. L'element `` aporta una manera de separar lògicament contingut dins d'un paràgraf. Gràcies a aquesta separació es podrà donar un format diferent a parts del seu contingut.

```
<h1>Cicles formatius de formació professional</h1>
<p>dimarts, 12 d'abril de 2011 12:23</p>
<p>Els cicles formatius són els ensenyaments que preparen per a l'exercici d'una professió determinada;
s'agrupen en famílies professionals i poden ser de <span class="negreta">grau mitjà</span> o de <span class="negreta">grau superior</span>.
</p>
```

Tingues en compte que 'negreta' estigui definit com a negreta en un .css

HTML / XHTML - Imatges

Les imatges són un dels elements importants en les pàgines web. Podem classificar les imatges que trobem en una pàgina web en dos grans grups:

- **Imatges d'adornament:** no són essencials per al contingut de la pàgina, simplement hi són per millorar-ne la presentació. Per tant, s'haurien de carregar des del CSS.
- **Imatges de contingut:** formen part del contingut de la pàgina. Per tant, s'han de definir en el document.

És important saber que les imatges no s'afegeixen realment al document sinó que en aquest només s'hi desa una referència al lloc on les podem trobar via una URL.

Per definir imatges en un document es fa servir l'element ``, que té dos atributs obligatoris:

- L'atribut `src`, en què s'especificarà l'URL de la imatge.
- L'atribut `alt`, que contindrà un text que visualitzaran els programes que accedeixin al document HTML però no tinguin capacitat gràfica per mostrar imatges.

```

```

HTML / XHTML - Llistes

Les llistes són una altra manera de representar la informació en HTML. Hi ha tres tipus de llistes disponibles

Les **llistes numerades** permeten especificar una sèrie de valors precedits per un nombre. Aquestes llistes comencen amb l'etiqueta `` i després cada un dels elements de la llista es col·loquen en un ``.

En les **llistes no numerades** cada un dels elements de la llista està precedit per un símbol, que per defecte és un cercle. Es comença la definició per `` i posteriorment els elements es posen dins d'un ``.

Les **llistes de definició** són les menys conegudes però permeten especificar dos valors per cada element: un per a un terme o una frase, i l'altre per a la definició. En aquestes llistes es comença la llista per `<dl>`, els termes per `<dt>` i la definició per `<dd>`.

Llista numerada	Llista no numerada	Llista de definicions
<ol style="list-style-type: none">1. XML2. HTML3. XHTML	<ul style="list-style-type: none">• XML• HTML• XHTML	<p>XML Extensible Markup Language</p> <p>HTML HyperText Markup Language</p> <p>XHTML Extensible HyperText Markup Language</p>

HTML / XHTML - Taules

Les taules sempre es consideren un grup de files en què cada fila té unes quantes caselles. Hi ha un grup més d'etiquetes per representar capçaleres, peus de taula, unir caselles, però la idea sempre es basa en el mateix.

Les etiquetes bàsiques per representar les taules són tres: `<table>` per marcar l'inici d'una taula, `<tr>` per crear files, i la que representarà cada una de les cel·les dins d'una fila, `<td>`.

```
<table>
  <tr>
    <td>1</td>
    <td>2</td>
  </tr>
  <tr>
    <td>3</td>
    <td>4</td>
  </tr>
</table>
```

1	2
3	4

HTML / XHTML - Esquema bàsic html (I)

Els documents HTML només tenen una arrel, que serà l'element `<html>`. Aquesta etiqueta es fa servir per informar a qui llegeixi el document que el contingut del document és un fitxer HTML.

L'arrel només pot tenir dos elements fills:

- `<head>`: és el lloc en el qual es pot posar informació sobre el document HTML. El títol, el creador, l'idioma, el full d'estil, etc. Molta d'aquesta informació no es visualitzarà en el navegador.
 - En la capçalera hi ha un element obligatori, `<title>`, en el qual s'especificarà el nom del document.
- `<body>`: conté el contingut i l'estructura del document. Són les parts que es “veuen” en visualitzar el document HTML.

HTML5 és la darrera versió del llenguatge de marcatge HTML (HyperText Markup Language) i representa un avenç significatiu respecte a les versions anteriors, com HTML 4.01. HTML5 va ser desenvolupat per oferir una estructura més robusta i flexible per al disseny de pàgines web i aplicacions. Introduït al voltant de 2014, HTML5 aporta noves funcionalitats que milloren la manipulació de continguts multimèdia, la semàntica del document i la interoperabilitat amb les aplicacions web modernes.

HTML / XHTML - Esquema bàsic html (II)

Exemple codi bàsic html:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="description" content="Descripció breu del contingut de la pàgina.">
  <title>Títol de la Pàgina</title>
  <!-- Enllaços a fitxers CSS externs -->
  <link rel="stylesheet" href="styles.css">
  <!-- Enllaços a fonts externes, icones, etc. -->
  <link rel="icon" href="favicon.ico" type="image/x-icon">
</head>
<body>
  <!-- Contingut de la pàgina aquí -->
</body>
</html>
```

Explicació de la Capçalera

<!DOCTYPE html>: Declara que el document utilitza HTML5, assegurant que el navegador renderit el contingut en el mode estàndard.

<html lang="es">: L'atribut **lang** especifica l'idioma principal del contingut de la pàgina, en aquest cas, espanyol (es).

<meta charset="UTF-8">: Defineix la codificació de caràcters com UTF-8, que és compatible amb la majoria de llengües i símbols especials.

<meta name="viewport" content="width=device-width, initial-scale=1.0">: Ajusta el disseny per a dispositius mòbils, assegurant que la pàgina s'adapti correctament a diferents amplades de pantalla.

<meta name="description" content="...">: Proporciona una descripció breu del contingut de la pàgina, útil per a motors de cerca i xarxes socials.

<title>Títol de la Pàgina</title>: Defineix el títol que apareixerà a la pestanya del navegador.

<link rel="stylesheet" href="styles.css">: Enllaça un fitxer CSS extern per a aplicar estils a la pàgina.

<link rel="icon" href="favicon.ico" type="image/x-icon">: Defineix la icona que apareixerà a la pestanya del navegador.

HTML / XHTML - HTML 5 (I)

Amb **HTML5** apareixen noves etiquetes, les més destacades:

Veure exemple amb cadascun d'ells

1. <header>

Descripció: Utilitzat per definir un encapçalament per a un document o una secció. **Ús típic:** Normalment conté títols, logotips, menús de navegació, etc.

2. <footer>

Descripció: Utilitzat per definir un peu de pàgina per a un document o una secció. **Ús típic:** Normalment conté informació sobre l'autor, enllaços de drets d'autor, enllaços de política de privadesa, etc.

3. <article>

Descripció: Representa un contingut independent i autònom que podria ser distribuït o reutilitzat, com articles de notícies, entrades de blog, comentaris d'usuaris, etc. **Ús típic:** Per a articles, publicacions de blog, notícies, etc.

4. <section>

Descripció: Utilitzat per definir seccions dins d'un document, com capítols, capítols de llibres, pestanyes en una interfície, etc. **Ús típic:** Per a seccions temàtiques dins d'un document.

HTML / XHTML - HTML 5 (I)

- 5. **<nav>** **Descripció:** Utilitzat per definir una secció de navegació que conté enllaços de navegació. **Ús típic:** Per a menús de navegació.
- 6. **<aside>** **Descripció:** Utilitzat per definir contingut que està lleugerament relacionat amb el contingut principal i que pot ser considerat com una barra lateral. **Ús típic:** Per a barres laterals, enllaços de context, informació addicional, etc.
- 7. **<main>** **Descripció:** Representa el contingut principal del document. Només hauria d'haver un element **<main>** per pàgina. **Ús típic:** Per a l'àrea de contingut principal de la pàgina.
- 8. **<figure>** i **<figcaption>** **Descripció:** Utilitzat per encapsular contingut de mitjans visuals com imatges, diagrames, il·lustracions, etc., amb una llegenda associada. **Ús típic:** Per a imatges amb llegendes.
- 9. **<mark>** **Descripció:** Utilitzat per ressaltar text que és de rellevància o d'interès particular. **Ús típic:** Per ressaltar text important.
- 10. **<time>** **Descripció:** Representa un valor temporal o una data. **Ús típic:** Per a dates i hores.

HTML / XHTML - Enllaços

La possibilitat que els documents s'enllacin entre ells ha estat un dels aspectes clau en la popularitat de les pàgines web. L'objectiu dels enllaços és permetre tenir un apuntador unidireccional cap a una destinació qualsevol i que s'hi pugui accedir per mitjà d'una URL.

Hi ha diverses maneres de fer enllaços en HTML però la més corrent és l'etiqueta `<a>`. L'objectiu d'aquesta etiqueta és definir origen i destinacions dels enllaços.

```
<a href="https://bemen3.com/">Institut Bemen 3</a>
```

O si volem accedir a un document

```
<a href="https://example.com/document-notes.pdf">Document Notes</a>
```

O si volem desplaçar en un punt dins de la mateixa web

```
<a href="#LLOC">Document Notes</a>
```

per anar a la marca que indiquem amb

```
<a id="LLOC"></a>
```

HTML / XHTML - Formularis

Un formulari en HTML és una part especial que conté uns elements anomenats *controls* que permeten que l'usuari hi introdueixi algun tipus de dades.

Els formularis sobretot estan pensats per permetre que els usuaris enviïn informació emplenant o seleccionant els controls que conté.

```
<form action="resultat.php" method="post">
  <p>
    <label for="nom">Nom:</label>
    <input type="text" id="Nom" />
    <label for="cognom">Cognom:</label>
    <input type="text" id="Cognom" />
    <input type="radio" name="tipus" value="professor"/>
    <input type="radio" name="tipus" value="alumne"/>
    <input type="submit">
  </p>
</form>
```

Nom:

Cognom:

☐ Professor ☐ Alumne

Tramet la consulta

Pasar de HTML a XHTML

Tot i que visualment l'HTML i l'XHTML són molt semblants, la diferència més important entre els dos llenguatges és que l'XHTML compleix les regles XML i per tant no permet que hi hagi etiquetes que s'obrin i no es tanquin o que no es mantingui l'ordre a l'hora de tancar les etiquetes obertes.

El compliment de les normes XML fa que els documents XHTML tinguin tendència a ser més grans però també tenen l'avantatge de que poden ser processats d'una forma més senzilla pels programes i que això els fa més fàcils de mantenir

XHTML parteix de la idea actual de separar la presentació de les dades del seu contingut i per tant elimina algunes de les etiquetes d'HTML que es feien servir exclusivament per donar format (`font`, `center`, `u`, `s`, `strike`, `color`, `align`...). L'XHTML ja no es preocupa de l'aspecte de les coses sinó de què signifiquen. La manera com s'ha de representar es deixa per als navegadors o bé per als fulls d'estil.

Pasar de HTML a XHTML - Capçalera

Per definir un document com HTML al principi de tot es posava:

```
<!DOCTYPE html>
```

Ara per definir XHTML ho farem així:

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

Pasar de HTML a XHTML - Eines

Els documents HTML/XHTML només són documents de text i, per tant, es poden crear fent servir un simple **editor de text**.

A pesar d'això hi ha tota una sèrie d'editors que ofereixen diverses ajudes per crear documents HTML de manera més ràpida i menys subjecta a errors.

Hi ha una gran llista d'editors d'aquest tipus, i no para de créixer:

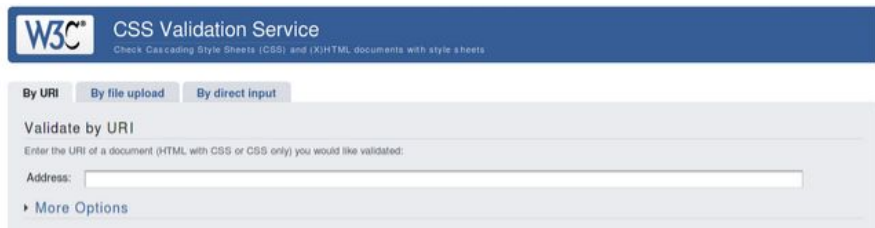
- Adobe Dreamweaver
- Microsoft Expression Studio
- CoffeCup HTML Editor
- Kompozer
- Amaya
- BlueFish
- Quanta Plus
- etc.

Pasar de HTML a XHTML - Característiques extra i validació

La major part dels editors solen oferir la possibilitat de fer més tasques a part de l'edició d'HTML assistida:

- Permetre l'edició de fulls d'estil.
- Facilitar la incorporació de codi en llenguatges de programació a la pàgina: Javascript, PHP...
- Penjar la pàgina al servidor on ha de ser-hi.
- etc.

Un dels validadors en línia més populars és el que ofereix el W3C a validator.w3.org



The screenshot shows the W3C CSS Validation Service interface. At the top, there is a blue header with the W3C logo and the text "CSS Validation Service" and "Check Cascading Style Sheets (CSS) and (X)HTML documents with style sheets". Below the header, there are three tabs: "By URI", "By file upload", and "By direct input". The "By URI" tab is selected. Under the "Validate by URI" section, there is a text input field labeled "Address:" and a "More Options" link.