

---

# M04 LLENGUATGE DE MARQUES

## UF1. PROGRAMACIÓ AMB XML

— A1: XML BEN FORMAT —

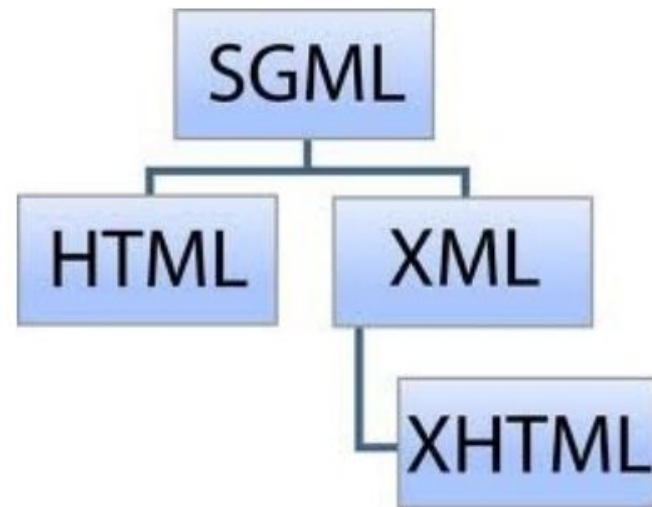
---

# ÍNDEX

1. [Definició i classificació](#)
2. [XML: Estructura i sintaxi](#)
3. [Eines d'edició](#)
4. [Elaboració de documents XML ben formats](#)
5. [Utilització d'espais de noms en XML](#)
6. [Gestió de diferents idiomes](#)

# 1. Definició i classificació

- ❖ XML prové d'un llenguatge anomenat GML (General Markup Language).
- ❖ Aquest llenguatge va agradar a l'ISO, i el 1986 van començar a treballar per a normalitzar-lo, creant el llenguatge SGML (Standard General Markup Language).



# 1. Definició i classificació

- ❖ SGML (Standard Generalized Markup Language) •
  - Estàndard internacional per a la definició de l'estructura i contingut de documents.
  - Força complicat per a ser adoptat com a llenguatge de la web.

# 1. Definició i classificació

- ❖ HTML (HyperText Markup Language)
  - Per a compartir informació per Internet.
  - Utilitza marques ja definides i no en permet de noves.
  - Dissenyat per estructurar i relacionar textos.
  - Les seves regles sintàctiques no son gaire estrictes.
  - Ineficient per a poder interpretar quines dades conté el document i fer cerques intel·ligents.
  - Bé per a presentar informació als humans.

# 1. Definició i classificació

- ❖ La tecnologia XML busca donar solució al problema d'expressar informació estructurada de la manera més abstracta i reutilitzable possible.

## eXtensible Markup Language (XML)

```
<?xml version="1.0"?>
<quiz>
  <qanda seq="1">
    <question>
      Who was the forty-second
      president of the U.S.A.?
    </question>
    <answer>
      William Jefferson Clinton
    </answer>
  </qanda>
  <!-- Note: We need to add
    more questions later.-->
</quiz>
```

**XML**

ÍNDEX

# 1. Definició i classificació

- ❖ XML (eXtensible Markup Language)
  - Permet crear les nostres pròpies marques.
  - Dissenyat per descriure dades i no preocupar-se de com es presentaran als usuaris.
  - Per a la presentació de les dades es farà servir CSS, XML-FO,...
  - Les regles sintàctiques són molt estrictes.
  - XML pot ser validat amb un DTD o XML-Schema .

# 1. Definició i classificació

❖ XML (eXtensible Markup Language)

➤ Usos actuals:

- Mostrar continguts de pàgines web amb XHTML (és un llenguatge XML)
- Comunicar sistemes distribuïts i tot executin sistemes operatius diferents
- Molts programes que emmagatzemagen les dades en binari ara ho fan en XML de manera que qualsevol que conegui les seves etiquetes pugui llegir/escriure/modificar les dades. Ex: Microsoft Office i OpenOffice.



# 1. Definició i classificació

- ❖ XML (eXtensible Markup Language)
  - Problema principal: els fitxers XML ocupen més espai que els equivalents en format binari.
  - Per a generar fitxers més petits van apareixer els llenguatges de marques lleugers com JSON.

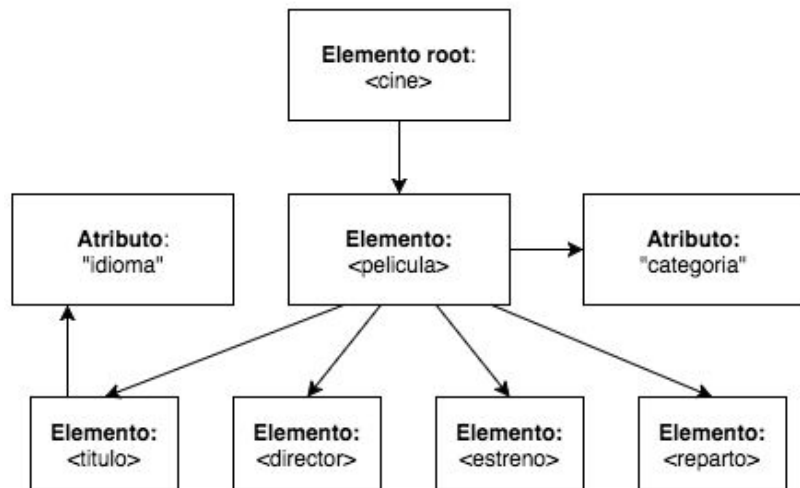
# 1. Definició i classificació

- ❖ XHTML (eXtensible Markup Language)
  - Pensat com a successor de l'HTML amb la sintaxis estricta derivada de l'XML.
  - Està pensat com un llenguatge que realment separi la informació de la seva representació.

## 2. XML estructura i sintaxi

- ❖ Representem l'XML amb una estructura jeràrquica en forma d'arbre en la que existeix un element principal de qui deriva la resta d'elements.
- ❖ Aquest element principal o arrel ha de ser el primer i al mateix temps únic.
- ❖ El diferents elements poden contenir o no atributs.

## 2. XML estructura i sintaxi

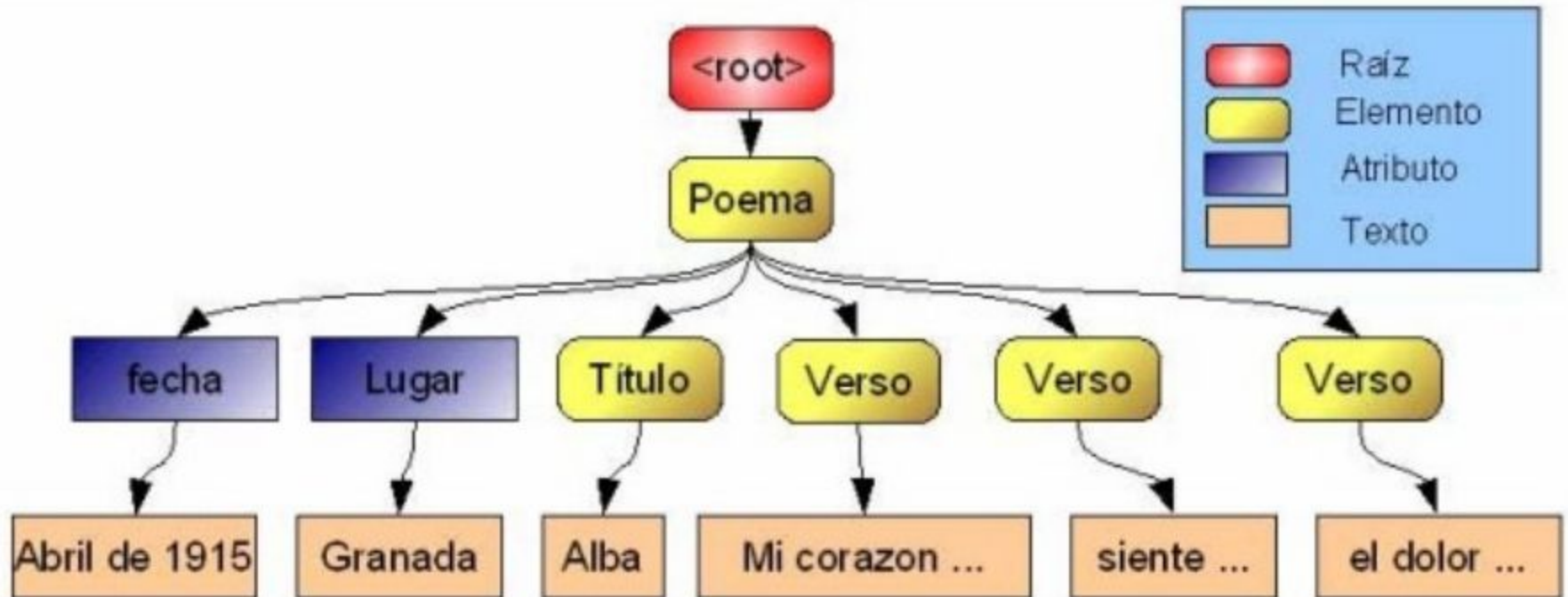


- Els atributs són part dels elements XML.
- Un element pot tenir varis atributs únics.
- Un atribut serveix per a proporcionar informació extra sobre l'element que el conté

## 2. XML estructura i sintaxi

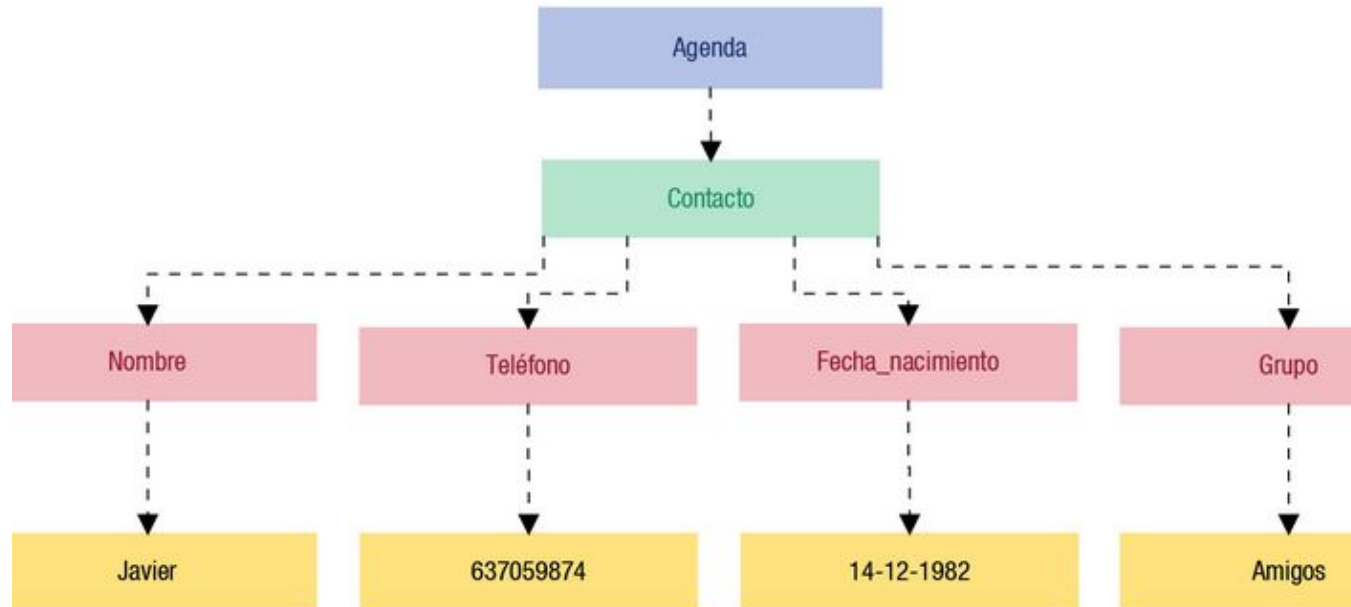
```
<?xml version="1.0" encoding="UTF-8"?>
<cine>
  <pelicula categoria="accion">
    <titulo idioma="ingles">Mad Max</titulo>
    <director>George Miller</director>
    <estreno>15 mayo 2015</estreno>
    <reparto>Tom Hardy</reparto>
    <reparto>Charlize Theron</reparto>
    <reparto>Nicholas Hoult</reparto>
  </pelicula>
  <pelicula categoria="animacion">
    <titulo idioma="ingles">Inside Out</titulo>
    <director>Pete Docter</director>
    <estreno>17 junio 2015</estreno>
    <reparto>Amy Poehler</reparto>
    <reparto>Phyllis Smith</reparto>
    <reparto>Bill Hader</reparto>
  </pelicula>
</cine>
```

## 2. XML estructura i sintaxi

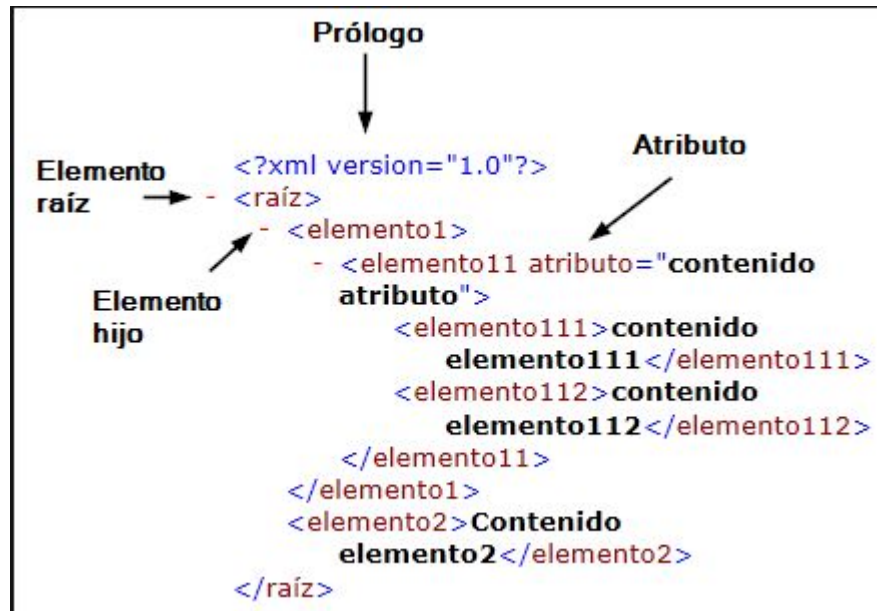


## 2. XML estructura i sintaxi

### ESQUEMA DE ÁRBOL DE UN DOCUMENTO XML



## 2. XML estructura i sintaxi





## 2. XML estructura i sintaxi

### Etiquetas (*tags*)

Una etiqueta es un identificador que empieza por el carácter `<` y termina por `>`. Existen tres tipos de etiquetas:

- las etiquetas de apertura (*start-tag*), que empiezan por el carácter `<` y terminan por `>`. Por ejemplo:

```
<apartado>
```

- las etiquetas de cierre (*end-tag*), que empiezan por los caracteres `</` y terminan por `>`. Por ejemplo:

```
</apartado>
```

- las etiquetas vacías (*empty tag*), que empiezan con el carácter `<` y terminan por `/>`. Por ejemplo:

```
<línea />
```

## 2. XML estructura i sintaxi

- ❖ Com ja s'ha nomenat abans, XML és un llenguatge estricte amb la seva sintaxis. Les normes bàsiques per crear documents XML són:
  - El document XML conté un encapçalament estàndard
  - Els elements comencen amb una lletra o bé un guió baix “\_” i no contenen espais.
  - XML és sensible a les majúscules i minúscules.

## 2. XML estructura i sintaxi

- ❖ Com ja s'ha nomenat abans, XML és un llenguatge estricte amb la seva sintaxis. Les normes bàsiques per crear documents XML són:
  - Els atributs d'un element s'han d'escriure entre cometes.
  - Els comentaris es formen com a l'HTML.
  - Totes les etiquetes s'han de tancar.
  - Les etiquetes s'han de niar correctament. Si obres l'etiqueta A i després la B, 1r has de tancar la B.

## 2. XML estructura i sintaxi

Exemple:

<extern>

<intern> Text per mostrar </intern>

Contingut de l'element <intern>

</extern>

<document Data="13/042009">

"Data" és un atribut de <document>

<titol>Tractat de biologia </titol>

<autor>Pere Rives </autor>

<pagines>140</pagines>

</document>

<linia-horitzontal/>

Element buit

## 2. XML estructura i sintaxi

- ❖ Es defineixen cinc elements o entitats per representar caràcters especials i que no siguin interpretats com a part de l'etiquetat:

ENTITAT	CARACTER
&amp;	&
&lt;	<
&gt;	>
&apos;	'
&quot;	"

### 3. Eines d'edició

- ❖ Existeix un espectre molt ample d'eines d'edició, tant gratuïtes com de pagament.
- ❖ Cadascuna d'aquestes eines es diferencien en la quantitat i diversitat de funcions que ofereixen a l'usuari. Començant pel nivell més baix, podem trobar el Wordpad o bloc de notes, Gedit, Notepad++, etc.
- ❖ Aquests editors no ens ofereixen la possibilitat de comprovar si un document XML està o no ben format.

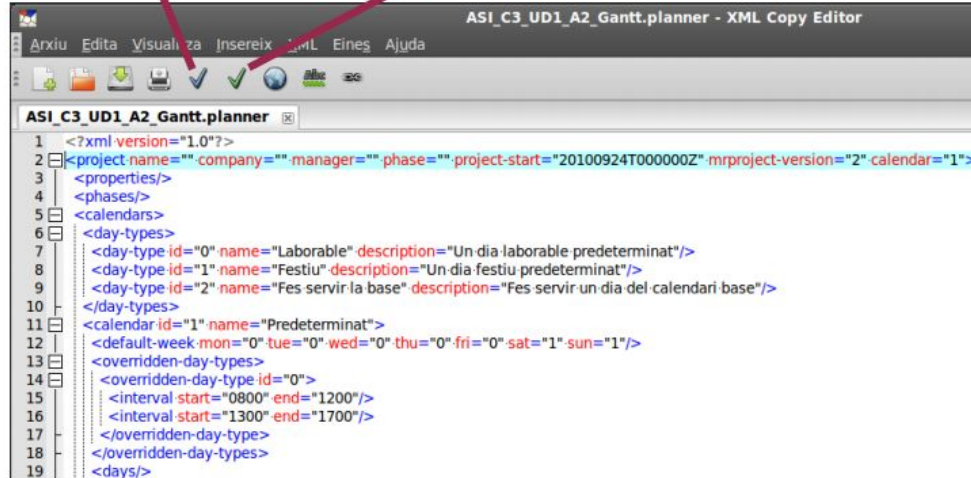
## 3. Eines d'edició

- ❖ Una eina més avançada i gratuïta que utilitzarem a Linux és “XML Copy Editor”. Aquest editor ens permet realitzar la comprovació de que un document XML està ben format i, a més a més, amb el DTD adient, també ens pot dir si el document és vàlid. (A2: XML vàlid)

# 3. Eines d'edició

Botó per a comprovar si està ben format

Botó per a comprovar si és vàlid (necessita DTD)



```
1 <?xml version="1.0"?>
2 <project name="" company="" manager="" phase="" project-start="20100924T000000Z" mrproject-version="2" calendar="1">
3   <properties/>
4   <phases/>
5   <calendars/>
6   <day-types/>
7     <day-type id="0" name="Laborable" description="Un dia laborable predeterminat"/>
8     <day-type id="1" name="Festiu" description="Un dia festiu predeterminat"/>
9     <day-type id="2" name="Fes servir la base" description="Fes servir un dia del calendari base"/>
10  </day-types/>
11  <calendar id="1" name="Predeterminat">
12    <default-week mon="0" tue="0" wed="0" thu="0" fri="0" sat="1" sun="1"/>
13    <overridden-day-types/>
14      <overridden-day-type id="0">
15        <interval start="0800" end="1200"/>
16        <interval start="1300" end="1700"/>
17      </overridden-day-type/>
18    </overridden-day-types/>
19  </calendar/>
20 </project>
```

Informació



ASI\_C3\_UD1\_A2\_Gantt.planner és ben format

Informació



libro.xml és vàlid

ÍNDIX



### 3. Eines d'edició

- ❖ D'un altre costat, tenim el “MS XML Notepad” de Microsoft, una eina també gratuïta encara que té una filosofia diferent de treballar.

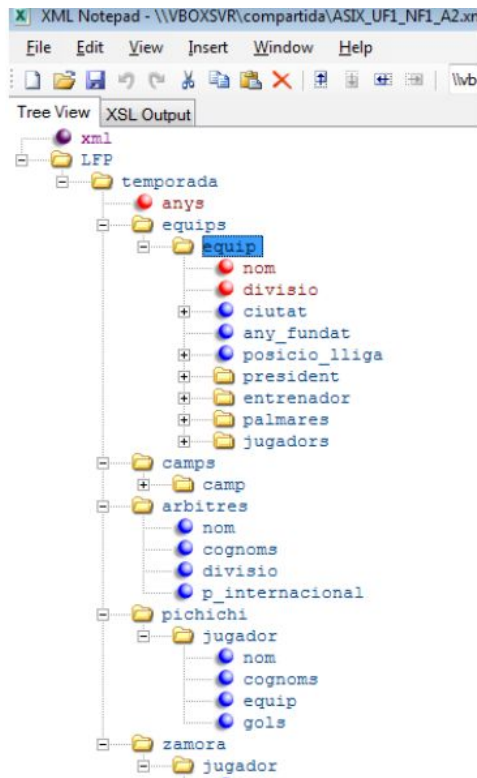


### 3. Eines d'edició

- ❖ Amb aquest editor, creem l'arbre d'elements amb els seus atributs, etc i després podem veure a la pestanya XSL Output el codi XML que resulta del nostre arbre.



# 3. Eines d'edició



Your XML document contains no xml-stylesheet processing instructions.

<?xml-stylesheet type="text/xsl" href="..."

You can also enter the XSLT file name using the above text box.

The following HTML is provided by the default XSLT transformation.

```
<LFP>
<temporada anys="2009-2010">
  <equips>
    <equip nom="FC Barça" divisio="primera">
      <ciutat>Barcelona</ciutat>
      <any_fundat/>
      <posicio_lliga>1</posicio_lliga>
      <president>
        <nom>Sandro</nom>
        <cognoms>Rosell</cognoms>
      </president>
      <entrenador>
        <nom>Pep</nom>
        <cognoms>Guardiola</cognoms>
      </entrenador>
      <palmares>
        <titol temporada=""/>
      </palmares>
    </jugadors>
    <jugador demarcacio="porter">
      </jugador>
    </equips>
  </temporada>
</LFP>
```

Error List Dynamic Help

### 3. Eines d'edició

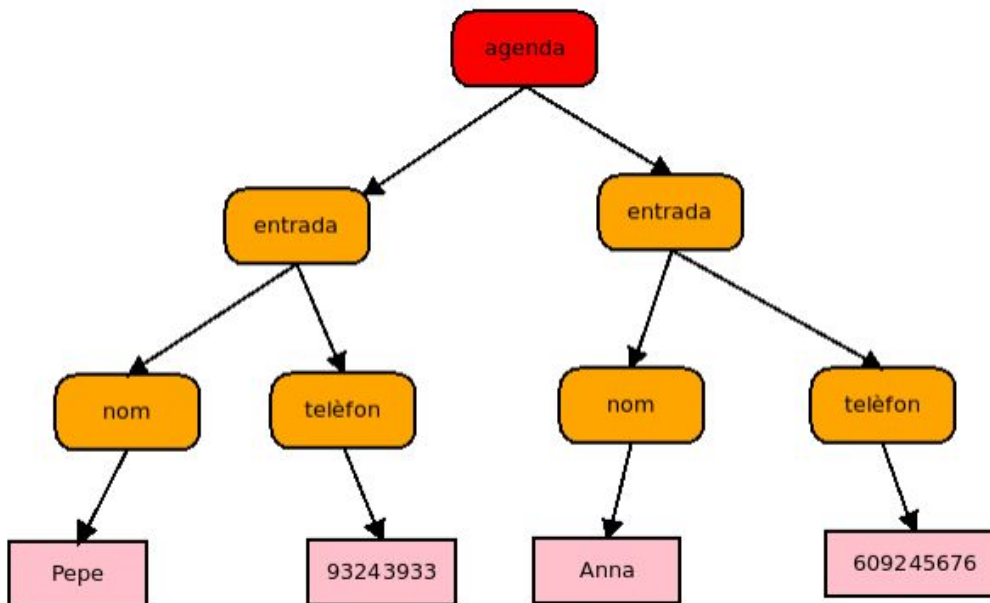
Altres eines de pagament:



### 3. Eines d'edició

Exercici resol't 1:

Donat el següent diagrama de blocs, determineu el codi XML corresponent.



### 3. Eines d'edició

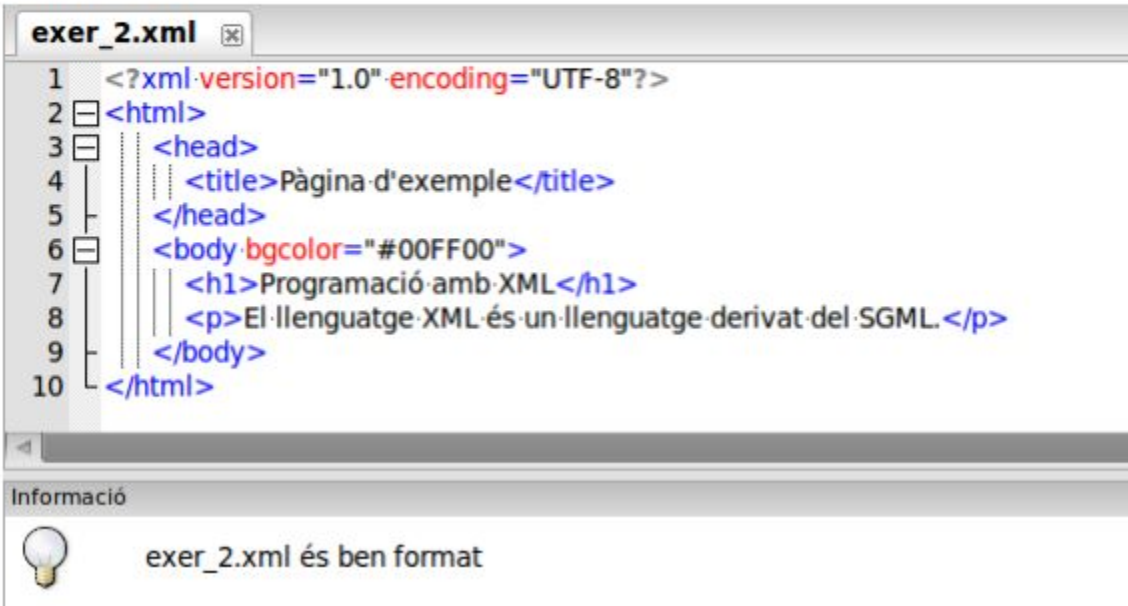
Solució:

```
*exemple_XML.xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <agenda>
3    <entrada>
4      <nom>Pepe</nom>
5      <telèfon>93243933</telèfon>
6    </entrada>
7    <entrada>
8      <nom>Anna</nom>
9      <telèfon>609245676</telèfon>
10   </entrada>
11 </agenda>
```

## 3. Eines d'edició


Exercici resolt 2:

Dibuixa l'estructura que correspon al següent codi XML.



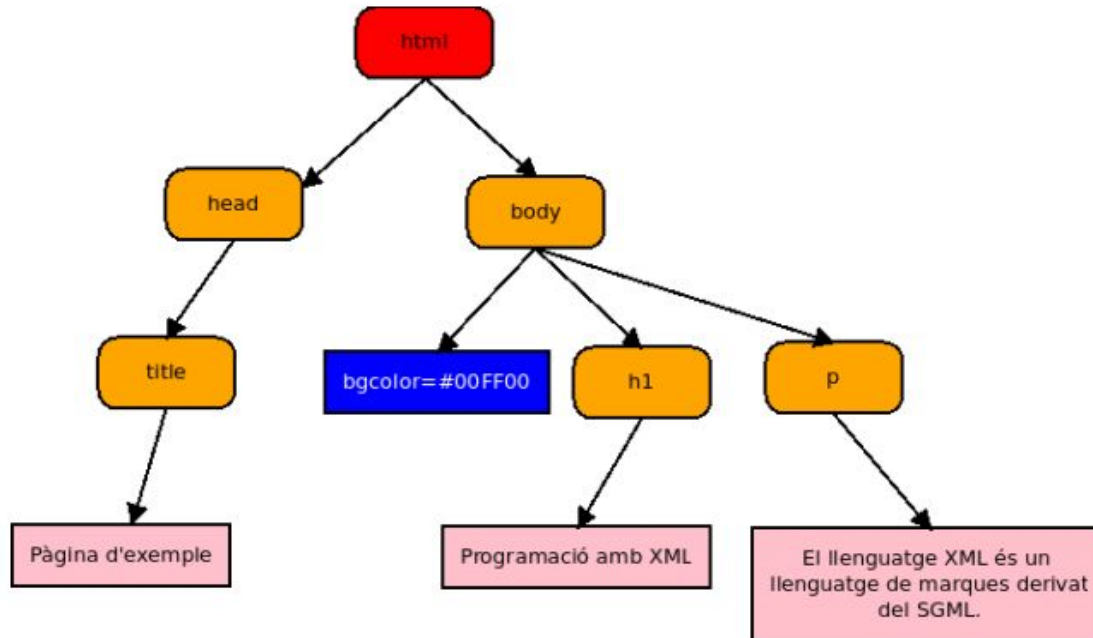
```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <html>
3   <head>
4     <title>Pàgina d'exemple</title>
5   </head>
6   <body bgcolor="#00FF00">
7     <h1>Programació amb XML</h1>
8     <p>El llenguatge XML és un llenguatge derivat del SGML.</p>
9   </body>
10 </html>
```

Informació

 exer\_2.xml és ben format

### 3. Eines d'edició

Solució:





# 4. Elaboració de documents XML ben formats

Com ja s'ha comentat en l'apartat d'estructura i sintaxi, per crear documents XML ben formats hem de respectar les següents condicions:

- Tot document XML, tindrà una capçalera `<?xml version="1.0" ?>` i de manera opcional, podem fer referencia al tipus de codificació del fitxer xml com per exemple utf-8 o ISO-8859-1. La forma correcta de especificar-ho a la capçalera és `<?xml version="1.0" encoding="ISO-8859-1"?>`.
- Qualsevol element ha d'estar tancat sigui o no buit `<agenda></agenda>` o `<br/>`.
- Els documents XML tindran un únic element arrel.
- Els elements estaran niats correctament `<a><b>...</b></a>`.
- No s'ha d'oblidar que XML és sensible a les majúscules i minúscules. Aquest exemple seria per tant, incorrecte `<p> ... </P>`.
- Els atributs s'escriuran entre cometes `<persona dni="12.098.322"> ... </persona>`.
- Els comentaris segueixen el mateix format que els de HTML `<!-- Comentari de l'XML -->`.

## 5. Utilització d'espais de noms en XML

La motivació dels espais de noms bé donat per aprofitar la modularitat del llenguatge, es a dir, si tenim un vocabulari ben format i aquest pot servir per a altres aplicacions, és millor reutilitzar-lo i no tornar a crear-ne un de nou.

Pot ser que en una mateixa aplicació coexisteixin dos vocabularis compatibles entre ells i es dona el cas que alguna o algunes de les etiquetes puguin ser comunes. Aquesta coincidència de noms en les etiquetes crearia una col·lisió que s'ha d'evitar i aquí és on entra en joc l'espai de noms. Si a cada vocabulari se li dóna un espai de noms, l'ambigüitat queda resolta entre els elements que s'anomenen igual. Això sí, dins del mateix espai de noms no poden existir dos elements que s'anomenen de la mateixa forma i tinguin diferent significat.

# 5. Utilització d'espais de noms en XML

Podem definir un espai de noms utilitzant un atribut reservat de l'XML anomenat xmlns (de XML name space) i assignant-li un valor que serà un URI (Uniform Resource Identifier o Identificador Uniforme de Recurs):

xmlns="<http://www.w3c.com/espai1>" (espai de noms genèric)



URI

S'ha de notar que l'URI no ha de contenir realment cap tipus de vocabulari, simplement l'utilitzem per fer la diferenciació dels dos vocabularis.

També podem utilitzar un prefix per associar els elements i atributs de cada vocabulari:

xmlns:vocabulari1="<http://www.w3c.com/articles>"

xmlns:vocabulari2="<http://www.w3c.com/consonants>"

# 5. Utilització d'espais de noms en XML



Un exemple que en mostra un document xml amb una ambigüitat no resolta:

```
<?xml version="1.0" encoding="UTF-8"?>
<tenda>
  <client>
    <nom></nom>
    <cognoms></cognoms>
  </client>
  <venedor>
    <nom></nom>
    <cognoms></cognoms>
  </venedor>
</tenda>
```

Si ens fixem, nom i cognoms són etiquetes comunes tant per al client com per al venedor. En un moment donat, pot ser no podríem diferenciar a quin dels dos pertany un d'aquests elements.

# 5. Utilització d'espais de noms en XML



Un altre exemple més clar en quant a l'ambigüitat es mostra a continuació:

```
<?xml version="1.0" encoding="UTF-8"?>
<inversions>
  <delegacio nom="BaixLlobregat">
    <capital>Barcelona</capital>
    <capital>350.000€</capital>
  </delegacio>
</inversions>
```

Hem de fixar-nos que existeixen dos elements anomenats <capital> que realment fan referència a informació diferent però que crea un problema d'identificació. Podem resoldre aquesta ambigüitat de la següent manera:

```
<?xml version="1.0" encoding="UTF-8"?>
<inversions xmlns:ubica="http://www.gencat.edu/ubicacions"
  xmlns:econom="http://www.gencat.edu/disposicions">
  <delegacio nom="BaixLlobregat">
    <ubica:capital>Barcelona</ubica:capital>
    <econom:capital>350.000€</econom:capital>
  </delegacio>
</inversions>
```

Hem creat dos espais de noms -ubica i econom – per poder diferenciar cada element i resoldre així el problema d'ambigüitat que teníem. Podem veure ara que cada element <capital> té associat un identificador diferent que fa referència cadascun al seu espai de noms.

# 5. Utilització d'espais de noms en XML

L'us dels espais de noms, realment és opcional per a documents XML senzills. Serà aconsellable utilitzar els espais de noms quan el nostre document hagi de ser compartit amb altres documents XML i potencialment puguin existir elements amb el mateix nom en dos o més documents.

En l'àmbit dels atributs, s'ha de tenir en compte què quan apliquem un espai de noms a l'element, NO afecta a l'atribut, per tant si fem:

```
<!-- http://www.w3.org està lligat a n1 i n2 -->  
<x xmlns:n1="http://www.w3.org"  
  xmlns:n2="http://www.w3.org" >  
  <mal a="1" a="2" />  
  <mal n1:a="1" n2:a="2" />  
</x>
```

Tenim dos errors, un és què els espais de noms n1 i n2 fan referència al mateix URI. L'altre és un element amb dos atributs amb el mateix nom.



# 5. Utilització d'espais de noms en XML

Una manera correcta de fer-ho és:

```
<!-- http://www.w3.org está lligat a n1 i és el valor per defecte -->  
<x xmlns:n1="http://www.w3.org"  
  xmlns="http://www.w3.org" >  
  <bien a="1" b="2" />  
  <bien a="1" n1:a="2" />  
</x>
```

Si ens fixem, l'espai de noms general no s'aplica als atributs del primer element (<bien>). En el segon, tenim dos atributs que s'anomenen igual, però un d'ells té definit un espai de noms que el fa vàlid.

## 6. Gestió de diferents idiomes

En el processament de documents, pot ser útil identificar el llenguatge en el que està escrit el contingut. Podem inserir un atribut especial anomenat **xml:lang** que ens permetrà especificar el llenguatge utilitzat en el contingut i valors d'atributs d'un element XML.

```
<p xml:lang="es"> Hola, una prueba </p>
```

```
<p xml:lang="en"> Hi, this is a test </p>
```

En aquest cas, “es” (espanyol) i “en” (english) són definits en la ISO-639.