

4.- Atributs. Declaracions de llistes d'atributs per als tipus d'elements.

Ja sabem declarar elements, ara vorem com es declaren els **atributs** associats a un element.

Recordem...

Els atributs en Document Type Definition (DTD) en XML són **elements clau que es poden utilitzar per proporcionar informació addicional** sobre els elements dins del document XML.

- Els atributs van insertats en els elements.
- Estan formats per parells **NOM-VALOR**.
- Els valors van entre " ".
- Poden ser obligatoris, opcionals, fixes...
- Normalment accepten valors de text, números, llistes de possibles valors, etc.

En XML podem definir els nostres propis atributs dins dels elements.

Per exemple l'atribut **país** a l'etiqueta **f1**

```
<deportistes>
  <f1 país="Països Baixos">Max Verstappen</f1>
  <f1>Carlos Sainz</f1>
  <tenis>Carlos Alcaraz</tenis>
</deportistes>
```

Els Atributs:

- Permeten afegir informació addicional als elements d'un document.
- No poden contenir sub-atributs.
- S'usen per afegir informació curta, senzilla i desestructurada.

Cadascun dels atributs

- Només es pot especificar una vegada
- Pot apareixer en qualsevol ordre.

- **Declaració d'atributs en DTD** La sintaxis per declarar un atribut és:

<!ATTLIST nomElement nomAtribut tipusAtribut valorInicialAtribut >



1. "<!ATTLIST"
2. Identificador de l'element al que s'aplica l'atribut.
3. Nom del atribut
4. Tipus
5. Valor per defecte.

• Exemple:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mensajes [
  <!ELEMENT mensajes (mensaje+)>
  <!ELEMENT mensaje (de, a, texto)>
  <!ATTLIST mensaje prioridad (normal|urgente) "normal">
  <!ELEMENT de (#PCDATA)>
  <!ELEMENT a (#PCDATA)>
  <!ELEMENT texto (#PCDATA)>
  <!ATTLIST texto idioma CDATA #REQUIRED>
]>

<mensajes>
  <mensaje prioridad="urgente">
    <de>Alfredo Regno</de>
    <a>Hans van Parijs</a>
    <texto idioma="holandés">
      Trobo Hans, hoe gaat het? ...
    </texto>
  </mensaje>
</mensajes>
```

- L'atribut **prioridad** pot estar en **<mensaje>** i pot tenir el valor "normal" o "urgente" sent "normal" el valor per defecte si no especifiquem l'atribut.
- Podem establir la necessitat o no d'especificar cada atribut.
- L'atribut **idioma**, pertany a **<texto>**, i pot contenir dades de caràcter (CDATA).
- **#REQUIRED**: indica que és obligatori especificar aquest atribut.

• Tipus d'Atributs

Els atributs poden ser de diferents tipus:

- **CDATA**: Caràcters de dades (text). Són els més senzills, poden contindre qualsevol cosa

```
<!ATTLIST element atribut CDATA #REQUIRED>
```

- **NMTOKEN**: Aquest tipus d'atribut només permet valors que siguin noms de tokens (seqüències de caràcters alfanumèrics, guions baixos i altres caràcters especials, però no espais en blanc).

```
<!ATTLIST empleado codigo NMTOKEN #REQUIRED>
```

- **NMTOKENS**: Aquest és similar a NMTOKEN, però permet una llista de noms de tokens separats per espais en blanc.

```
<!ATTLIST contacto telefonos NMTOKENS #IMPLIED>
```

Exemple:

```
<!ATTLIST mensaje fecha CDATA #REQUIRED>
<mensaje fecha="15 de Julio de 1999">
<!ATTLIST mensaje fecha NMTOKEN #REQUIRED>
<mensaje fecha="15-7-1999">
```

- **ENUMERAT:** Només es poden utilitzar valors específics.

```
<!ATTLIST coche marca (Ford | Toyota | Honda) "Ford">
```

- **ID:** Identificador únic per a l'atribut dins del document.

```
<!ATTLIST elemento id ID #REQUIRED>
```

- **IDREF:** Referència a un ID d'un altre element.

```
<!ATTLIST empleado jefe IDREF #IMPLIED>
```

- **IDREFS:** Llista de referències a IDs d'altres elements.

```
<!ATTLIST pedido productos IDREFS #IMPLIED>
```

Exemple:

El tipus **ID** permet que un atribut determinat tingui un nom únic que podrà ser referenciat per un atribut d'un altre element que siga de tipus **IDREF**.

```
<!ELEMENT enlace EMPTY>
<!ATTLIST enlace destino IDREF #REQUIRED>

<!ELEMENT capitulo (parrafo)*>
<!ATTLIST capitulo referencia ID #IMPLIED>
```

- **ENTITAT:** L'atribut amb aquest tipus fa referència a una entitat definida al DTD.

```
<!ATTLIST libro autor ENTITAT #IMPLIED>
```

- **ENTITATS:** Aquest tipus és similar a ENTITAT, però permet una llista de noms d'entitats.

```
<!ATTLIST documento referencias ENTITATS #IMPLIED>
```

En resum...

Tipus d'Atribut	Descripció
CDATA	Permet qualsevol caràcter de dades (text).
ENUMERAT	Només permet valors específics.
NMTOKEN	Només permet noms de tokens (seqüències de caràcters alfanumèrics, guions baixos i altres caràcters especials, sense espais).
NMTOKENS	És similar a NMTOKEN, però permet una llista de noms de tokens separats per espais.
ID	Ha de ser un identificador únic per a l'atribut dins del document.
IDREF	Fa referència a l'ID d'un altre element.
IDREFS	És una llista de referències a IDs d'altres elements.
ENTITAT	Fa referència a un nom d'entitat definida al DTD.
ENTITATS	És una llista de noms d'entitats definides.

• Valors d'Atributs

- Els valors dels atributs poden ser literals o referències.
- En cas de literals, els valors han de coincidir exactament amb els especificats en **ENUMERAT** o ser de tipus **CDATA**.
- Les referències s'utilitzen per a **IDREF**, **IDREFS** i **ENTITAT**, i fan referència a altres elements definits al DTD.

```
<!ATTLIST coche marca (Ford | Toyota | Honda) "Ford">  
<!ATTLIST libro autor CDATA #REQUIRED>  
<!ATTLIST empleado jefe IDREF #IMPLIED>
```

• Valors Predeterminats d'Atributs

- Pots especificar **un valor predeterminat** per a un atribut que s'utilitzarà si no es proporciona un valor al document XML.
 - **valor** entre comilles dobles (") o simples ('): L'atribut té valor per defecte.
- Els valors predeterminats poden ser:
 - **#REQUIRED**: L'atribut és obligatori.
 - **#IMPLIED**: L'atribut és opcional.
 - **#FIXED valor**: L'atribut té un valor fix que no es pot canviar.

Exemple:

```
<!ATTLIST persona edad CDATA #REQUIRED>  
<!ATTLIST libro idioma CDATA #IMPLIED>  
<!ATTLIST coche color CDATA #FIXED "negro">
```

5.- Entitats. Declaració i Tipus

Les entitats en DTD s'utilitzen per **definir valors que es poden reutilitzar a tot el document XML**. Aquestes entitats poden ser **símbols, fragments de text** o fins i tot caràcters especials que vulguis utilitzar en diversos llocs del teu document XML.

Són objectes que no han de ser analitzats sintàcticament segons les regles de XML. Els **seus principals usos són per fitxers, pàgines web, imatges, etc**

Les entitats es declaren amb **<!ENTITY**

- **Ús d'Entitats:**

Les entitats s'utilitzen al document XML posant un **ampersand &** seguit del **nom de l'entitat** i després un **punt i coma ;**.

Per exemple:

```
<autor>&autor;</autor>  
<contingut>&llibre;</contingut>
```

- **Avantatges de les Entitats:**

- ✓ **Faciliten la reutilització de dades** i fragments de text en diversos llocs del document XML.
- ✓ **Milloren la llegibilitat del document XML**, especialment quan s'utilitzen caràcters especials o fragments extensos de text.
- ✓ **Permeten gestionar i mantenir fàcilment el contingut** que s'utilitza en diversos documents XML.

• Tipus de Entitats

Les entitats poden ser de diversos tipus:

- ✓ Internes o Externes
- ✓ Analitzades o No analitzades
- ✓ Generals o de Paràmetre

• Entitats Internes:

- ✓ Les entitats internes es defineixen directament en el DTD del document XML.
- ✓ S'utilitzen principalment per definir caràcters especials o fragments de text que es reutilitzaran en el document.
- ✓ Són sempre analitzades.

Analitzades: Una volta reemplaçada la referència a l'entitat pel seu contingut, passa a ser part del document XML i com a tal, es analitzada pel processador XML

La sintaxi bàsica per declarar una entitat interna és la següent:

```
<!ENTITY nom "valor">
```

On "nom" és el nom de l'entitat i "valor" és el valor que representa.

Exemple:

```
<!DOCTYPE texto[  
  <!ENTITY ovni "Objeto Volante No identificado">  

```

Després, pots utilitzar **&ovni;** en qualsevol part del document XML per inserir " **Objeto Volante No identificado** ".

```
<texto><titulo> Un dia en la vida d'un &ovni; </titulo></texto>
```

• Entitats generals Externes Analitzades:

Les entitats externes es defineixen en un fitxer separat es poden fer servir en diversos documents XML. S'utilitzen per a la reutilització de valors comuns en múltiples documents.

La sintaxi per declarar una entitat externa és la paraula SYSTEM seguida d'un URI (Universal Resource Identifier) o una ruta a un arxiu:

```
<!ENTITY nom SYSTEM "URI o ruta de l'arxiu">
```

On **nom** és el nom de l'entitat i **"URI o ruta de l'arxiu"** és la ubicació de l'arxiu extern.

Exemple:

```
<!ENTITY intro  
SYSTEM "http://server.com/intro.xml">  

```

Llavors, pots fer referència a l'entitat **&intro;** al teu document XML per inserir tot el contingut de l'arxiu **intro.xml** en aquest lloc.

• Entitats No Analitzades:

- Si el contingut de l'entitat és binari, no s'hauria d'intentar interpretar-ho com si fora text XML.
- Sempre són entitats generals i externes.

```
<!ENTITY logo SYSTEM "http://server.com/logo.gif">
```

• Entitats Paràmetre:

- Només poden usar-se en la DTD, i **NO** en el document XML.
- S'usa el símbol **"%"** en lloc de **"&"** tant per declarar-les com per usar-les.

Tambe tenim internes i externes:

• Entitats Paràmetre Internes:

```
<!DOCTYPE texto[  
  <!ENTITY % elemento-alf "<!ELEMENT ALF (#PCDATA)>">  
  %element-alf;  

```


- **Entitats Paràmetre Externes:**

```
<!DOCTYPE texto[
  <!ENTITY % elemento-alf SYSTEM "alf.ent">
  ...
  %element-alf;
]>
```

6.- Exemple DTD

En este exemple tenim declarat un document DTD extern [LISTA.DTD](#)

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

  <!ELEMENT lista (persona)+>
  <!ELEMENT persona (nombre, email*, relacion?)>
  <!ATTLIST persona id ID #REQUIRED>
  <!ATTLIST persona sexo (hombre | mujer) #IMPLIED>
  <!ELEMENT nombre (#PCDATA)>
  <!ELEMENT email (#PCDATA)>
  <!ELEMENT relacion EMPTY>
  <!ATTLIST relacion amigo-de IDREFS #IMPLIED enemigo-de IDREFS #IMPLIED>
```

I el XML basat en el DTD anterior:

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

<!DOCTYPE lista SYSTEM "LISTA.dtd">
<lista>
  <persona sexo="hombre" id="alvaro">
    <nombre>Álvaro Álvarez</nombre>
    <email>alvaroa@hotmail.com</email>
    <relacion amigo-de="beatriz" />
  </persona>
  <persona sexo="mujer" id="beatriz">
    <nombre>Beatriz Bayo</nombre>
    <email>bea@terra.com</email>
  </persona>
</lista>
```

Explicació Detallada:

- **Document DTD**

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

→ Declaració XML que especifica la codificació del document com a UTF-8.

```
<!ELEMENT lista (persona)+>
```

→ Estableix que el teu document XML ha de començar amb un element **lista** que pot contenir un o més elements **persona**.

```
<!ELEMENT persona (nombre, email*, relacion?)>
```

→ Defineix l'element `persona`. Un element `persona` ha de contenir tres subelements: `nombre`, zero o més elements `email`, i zero o un element `relacion`.

```
<!ATTLIST persona id ID #REQUIRED>
```

→ Afegeix un atribut `id` a l'element `persona` que ha de ser de tipus ID i és requerit (és a dir, tota persona ha de tenir un `id` únic).

```
<!ATTLIST persona sexo (hombre | mujer) #IMPLIED>
```

→ Defineix l'atribut `sexo` a l'element `persona`. L'atribut `sexo` pot tenir els valors "hombre" o "mujer" i és opcional (marca `#IMPLIED`).

```
<!ELEMENT nombre (#PCDATA)>
```

→ Estableix que l'element `nombre` ha de contenir dades de text (PCDATA), és a dir, és un element de text.

```
<!ELEMENT email (#PCDATA)>
```

→ Estableix que l'element `email` també ha de contenir dades de text.

```
<!ELEMENT relacion EMPTY>
```

→ Defineix l'element `relacion` com buit (no pot contenir res més que atributs).

```
<!ATTLIST relacion amigo-de IDREFS #IMPLIED enemigo-de IDREFS #IMPLIED>
```

→ Afegeix dos atributs a l'element `relacion`: `amigo-de` i `enemigo-de`. Tots dos han de ser de tipus IDREFS (poden fer referència a IDs en altres elements) i són opcionals.

- **Document XML:**

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

→ Declara la versió de l'XML i la codificació que utilitza el document.

```
<!DOCTYPE lista SYSTEM "LISTA.dtd">
```

→ Enllaça el document XML amb la DTD anomenada "LISTA.dtd," que conté les regles de validació per al document XML.

```
<lista>
```

→ Obre l'element principal del document, `lista`.

```
<persona sexo="hombre" id="alvaro">
```

→ Obre un element `persona` i li afegeix dos atributs: `sexo` amb el valor "hombre" i `id` amb el valor "alvaro."

```
<nombre>Álvaro Álvarez</nombre>
```

```
<email>alvaroa@hotmail.com</email>
```

```
<relacion amigo-de="beatriz" />
```

→ Aquestes línies defineixen els subelements de l'element `persona`. Tenim un `nombre`, un `email` i un `relacion` amb l'atribut `amigo-de` que fa referència a "beatriz."

```
</persona>
```

→ Tanca l'element `persona`.

```
<persona sexo="mujer" id="beatriz">
```

→ Defineix una altra persona anomenada Beatriz, amb un atribut `sexo` igual a "mujer" i un atribut `id` igual a "beatriz."

```
<nombre>Beatriz Bayo</nombre>
```

```
<email>bea@terra.com</email>
```

→ Aquestes línies defineixen el nom i l'adreça de correu electrònic de Beatriz.

```
</persona>
```

→ Finalment, aquesta línia tanca l'element `persona`, i així acaba el document XML.