

Declaración de elementos

Sintaxis:

```
<!ELEMENT nombreElemento TipoContenido>
```

donde "nombreElemento" es el nombre del elemento en el documento XML, y "contenido" una expresión que describe el contenido del elemento.

Tipos de contenido

- **EMPTY**: significa que el elemento es vacío, es decir, que no puede tener contenido. Los elementos vacíos pueden escribirse con etiquetas de apertura y cierre sin nada entre ellos, ni siquiera espacios, o con una etiqueta vacía. Ejemplo de definición:

```
<!DOCTYPE ejemplo [  
    <!ELEMENT ejemplo EMPTY>  

```

Los siguientes elementos son correctos:

```
<ejemplo></ejemplo>  
<ejemplo/>
```

Estos no son correctos:

```
<ejemplo>Esto es un ejemplo</ejemplo>  <!-- No es correcto porque contiene  
texto -->  
<ejemplo><a></a></ejemplo>  <!-- No es correcto porque contiene un  
elemento <a> -->
```

- **(#PCDATA)**: significa que el elemento puede contener texto (debe escribirse entre paréntesis). Ejemplo de definición:

```
<!DOCTYPE ejemplo [  
    <!ELEMENT ejemplo (#PCDATA)>  

```

Elementos correctos:

```
<ejemplo/>  
<ejemplo>Esto es un ejemplo</ejemplo>
```

Elementos incorrectos:

```
<ejemplo><a></a></ejemplo>  <!-- No es correcto porque contiene un elemento <a> -->
```

- **ANY**: significa que el elemento puede contener cualquier cosa (texto y otros elementos). Por ejemplo:

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo ANY>  

```

Ejemplos de elementos:

```
<ejemplo/>  
<ejemplo>Esto es un ejemplo</ejemplo>  
<ejemplo><a></a></ejemplo>
```

Secuencias

Si el elemento debe tener una lista de elementos, estos se deben indicar separados por comas. Por ejemplo:

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo (a, b)>  
  <!ELEMENT a EMPTY>  
  <!ELEMENT b EMPTY>  

```

Indica que el elemento contendrá la lista de elementos indicada en el **mismo orden** de la secuencia. Es decir b no se puede poner delante de a.

Elemento correcto:

```
<ejemplo><a /><b /></ejemplo>  
<ejemplo><a /><b ></b></ejemplo>
```

Elementos incorrectos:

```
<ejemplo><a/><b/><c/></ejemplo>    <!-- No es correcto porque contiene  
un elemento <c /> -->  
<ejemplo><a/></ejemplo>    <!-- No es correcto porque falta el elemento  
<b /> -->  
<ejemplo><b/><a/></ejemplo>    <!-- No es correcto porque el orden no es  
correcto -->
```

Elecciones

Los elementos pueden contener elementos opcionales. Para indicarlo se utiliza la barra vertical | (**o lógico**), que indica que el elemento contiene uno de los dos elementos.

Por ejemplo:

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo (a | b)>  
  <!ELEMENT a EMPTY>
```

```
<!ELEMENT b EMPTY>
]>
```

Elementos correctos:

```
<ejemplo><a/></ejemplo>
<ejemplo><b/></ejemplo>
```

Elementos incorrectos:

```
<ejemplo><a/><b/></ejemplo>  <!-- No es correcto porque están los dos
elementos -->
<ejemplo></ejemplo>  <!-- No es correcto porque no hay ningún elemento
-->
```

Combinaciones

Por supuesto puede haber combinaciones, para ello se utilizan los paréntesis. Por ejemplo, podemos tener un DTD llamado *coordenada.dtd* con este contenido:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE coordenada [
<!ELEMENT coordenada ((longitud, latitud) | coordUniversal)>
<!ELEMENT longitud (#PCDATA)>
<!ELEMENT latitud (#PCDATA)>
<!ELEMENT coordUniversal (#PCDATA)>
]
```

Sería válido este documento:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE coordenada SYSTEM "coordenada.dtd">
<coordenada>
  <longitud>234</longitud>
  <latitud>-23</latitud>
</coordenada>
```

Sería válido también:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE coordenada SYSTEM "coordenada.dtd">
<coordenada>
  <coordUniversal>1232332</coordUniversal>
</coordenada>
```

Cardinalidad

La **cardinalidad** es el número de veces que puede aparecer un determinado contenido en un elemento. Se indica mediante estos símbolos:

- **?:** contenido opcional. Significa que el elemento puede aparecer o no, pero sólo una vez. Por ejemplo:

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo (a, b?)>
  <!ELEMENT a EMPTY>
  <!ELEMENT b EMPTY>
]>
```

Elementos correctos:

```
<ejemplo><a/></ejemplo>
<ejemplo><a/><b/></ejemplo>
```

Elementos incorrectos:

```
<ejemplo><b/></ejemplo>  <!-- No es correcto porque falta el elemento
<a/> -->
<ejemplo><b/><b/></ejemplo>  <!-- No es correcto porque el elemento
<b/> aparece dos veces -->
```

- *****: contenido opcional y repetible. Significa que el elemento puede no aparecer o aparecer una o más veces. Por ejemplo:

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo (a*, b)>
  <!ELEMENT a EMPTY>
  <!ELEMENT b EMPTY>
]>
```

Elementos correctos:

```
<ejemplo><b/></ejemplo>
<ejemplo><a/><b/></ejemplo>
<ejemplo><a/><a/><b /></ejemplo>
```

Elementos incorrectos:

```
<ejemplo><b/><a /></ejemplo>  <!-- No es correcto porque el elemento
<a/> aparece después de <b/> -->
```

- **+**: Contenido obligatorio y repetible. Significa que el elemento tiene que aparecer una o más veces (no puede no aparecer). Por ejemplo:

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo (a+, b)>
  <!ELEMENT a EMPTY>
  <!ELEMENT b EMPTY>
]>
```

Elementos correctos:

```
<ejemplo><a/><b/></ejemplo>
<ejemplo><a/><a/><b /></ejemplo>
```

Elemento incorrecto:

```
<ejemplo><b/></ejemplo> <!-- No es correcto porque falta el elemento  
<a/> -->
```

Declaración de atributos

Sintaxis:

```
<!ATTLIST nombreElemento nombreAtributo tipoAtributo valorPorDefecto>
```

Donde:

"nombreElemento" es el nombre del elemento para el que se define un atributo.

"nombreAtributo" es el nombre del atributo.

"tipoAtributo" es el tipo de datos.

"valorPorDefecto" es el valor predeterminado del atributo (aunque también puede indicar otras cosas).

Para definir varios atributos de un mismo elemento, se puede utilizar una o varias declaraciones de atributos. Los siguientes ejemplos son equivalentes:

```
<!ATTLIST nombreElemento nombreAtributo1 tipoAtributo1 valorInicialAtributo1>  
<!ATTLIST nombreElemento nombreAtributo2 tipoAtributo2 valorInicialAtributo2>
```

Equivalente a:

```
<!ATTLIST nombreElemento  
  nombreAtributo1 tipoAtributo1 valorInicialAtributo1  
  nombreAtributo2 tipoAtributo2 valorInicialAtributo2  
>
```

Tipos de atributo

- **CDATA:** el atributo puede contener caracteres (sin restricciones). A diferencia de los datos PCDATA de los elementos, los CDATA admiten cualquier carácter del tipo que sea (PCDATA no admite caracteres inválidos para la sintaxis de XML, como el signo < por ejemplo, CDATA sí los admite).

Por ejemplo:

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo EMPTY>  
  <!ATTLIST ejemplo color CDATA #REQUIRED>  

```

Atributos correctos:

```
<ejemplo color=""></ejemplo>  
<ejemplo color="amarillo"></ejemplo>
```

```
<ejemplo color="azul marino #000080"></ejemplo>
```

Atributo incorrecto:

```
<ejemplo></ejemplo>    <!-- No es correcto porque falta el atributo  
"color" -->
```

- **ID:** el valor del atributo (no el nombre) servirá para identificar al elemento que lo contiene, por lo que debe ser único y no se puede repetir en otros elementos o atributos. Los valores de estos atributos no pueden empezar por un número. Por ejemplo:

Los atributos ID sólo pueden indicar #IMPLIED o #REQUIRED en el apartado del valor por defecto.

Los ID son especialmente útiles para las herramientas de maquetación, análisis y programación de aplicaciones XML.

Ejemplo:

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo (libro*)>  
  <!ELEMENT libro (#PCDATA) >  
  <!ATTLIST libro codigo ID #REQUIRED>  

```

Atributos correctos:

```
<ejemplo>  
  <libro codigo="L1">Poema de Gilgamesh</libro>  
  <libro codigo="L2">Los preceptos de Ptah-Hotep</libro>  
</ejemplo>
```

Atributos incorrectos:

```
<ejemplo>  
  <libro codigo="1">Poema de Gilgamesh</libro>    <!-- No es correcto  
  porque el valor de un atributo de tipo ID no puede empezar con un  
  número -->  
  <libro codigo="L2">Los preceptos de Ptah-Hotep</libro>  
</ejemplo>
```

```
<ejemplo>  
  <libro codigo="L1">Poema de Gilgamesh</libro>  
  <libro codigo="L1">Los preceptos de Ptah-Hotep</libro>    <!-- No es  
  correcto porque el valor "L1" ya se ha utilizado -->  
</ejemplo>
```

- **IDREF:** el valor del atributo debe coincidir con el valor del atributo ID de otro elemento. Es decir, será una referencia a otra etiqueta. Por ejemplo:

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo ((libro|prestamo)*)>  
  <!ELEMENT libro (#PCDATA)>  
  <!ATTLIST libro codigo ID #REQUIRED>  
  <!ELEMENT prestamo (#PCDATA) >  
  <!ATTLIST prestamo libro IDREF #REQUIRED>
```

```
]>
```

Atributos correctos:

```
<ejemplo>
  <libro codigo="L1">Poema de Gilgamesh</libro>
  <prestamo libro="L1">Numa Nigerio</prestamo>
</ejemplo>
```

Atributos incorrectos:

```
<ejemplo>
  <libro codigo="L1">Poema de Gilgamesh</libro>
  <prestamo libro="L2">Numa Nigerio</prestamo>  <!-- No es correcto
    porque el valor "L2" no es ID de ningún elemento -->
</ejemplo>
```

- **IDREFS:** el valor del atributo es una serie de valores separados por espacios que coinciden con el valor del atributo ID de otros elementos. Es decir, es igual que el anterior sólo que permite indicar varias referencias (que deben existir en el documento XML) a otros ID.

Ejemplo:

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo ((libro|prestamo)*)>
  <!ELEMENT libro (#PCDATA) >
  <!ATTLIST libro codigo ID #REQUIRED>
  <!ELEMENT prestamo (#PCDATA) >
  <!ATTLIST prestamo libro IDREFS #REQUIRED>
]>
```

Atributos correctos:

```
<ejemplo>
  <libro codigo="L1">Poema de Gilgamesh</libro>
  <libro codigo="L2">Los preceptos de Ptah-Hotep</libro>
  <prestamo libro="L1 L2">Numa Nigerio</prestamo>
</ejemplo>
```

Atributos incorrectos:

```
<ejemplo>
  <libro codigo="L1">Poema de Gilgamesh</libro>
  <libro codigo="L2">Los preceptos de Ptah-Hotep</libro>
  <prestamo libro="L3">Numa Nigerio</prestamo>  <!-- No es correcto
    porque el valor "L3" no es ID de ningún elemento -->
</ejemplo>
```

- **ENTITY:** el valor del atributo es alguna entidad definida en la DTD.

```
<!ENTITY nombre_de_la_entidad "literal">
```

- **ENTITIES:** el valor del atributo es alguna de las entidades de una lista de entidades definida en la DTD.

- **Enumeración:** el atributo sólo puede contener uno de los valores de una lista. La lista se escribe entre paréntesis, con los valores separados por una barra vertical "|". Los valores deben ir entre comillas simples o dobles si contienen espacios en blanco. Por ejemplo:

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
  <!ATTLIST ejemplo color (azul|"blanco roto"|rojo) #REQUIRED>
]>
```

Atributos correctos:

```
<ejemplo color=""></ejemplo>
<ejemplo color="azul"></ejemplo>
```

Atributo incorrecto:

```
<ejemplo color="verde"></ejemplo> <!-- No es correcto porque "verde" no
está en la lista de valores -->
```

Valores por defecto

- **valor concreto/valor por defecto:** si al final de la declaración de un atributo aparece un valor concreto, se entiende que ese será el valor por defecto. Ejemplo:

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
  <!ATTLIST ejemplo color CDATA "verde">
]>
```

Atributos correctos:

```
<ejemplo></ejemplo>
<ejemplo color=""></ejemplo>
<ejemplo color="amarillo"></ejemplo>
<ejemplo color="verde"></ejemplo>
```

- **#REQUIRED:** el atributo es obligatorio, es decir, hay que darle obligatoriamente algún valor en el documento XML. Obviamente usando #REQUIRED no se puede indicar un valor por defecto (al no poder dejarse sin especificar el atributo).

Por ejemplo:

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
  <!ATTLIST ejemplo color CDATA #REQUIRED>
]>
```

Atributos correctos:

```
<ejemplo color=""></ejemplo>
<ejemplo color="amarillo"></ejemplo>
<ejemplo color="azul marino #000080"></ejemplo>
```

Atributo incorrecto:


```
<ejemplo></ejemplo>    <!-- No es correcto porque falta el atributo
"color" -->
```

- **#FIXED valor:** el atributo tiene un valor fijo. En ese caso en ningún documento XML se podrá modificar dicho atributo. Su uso está limitado. Por ejemplo:

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
  <!ATTLIST ejemplo color CDATA #FIXED "verde">
]>
```

Atributos correctos:

```
<ejemplo></ejemplo>
<ejemplo color="verde"></ejemplo>
```

Atributos incorrectos:

```
<ejemplo color=""></ejemplo>    <!-- No es correcto porque el atributo
"color" no tiene el valor "verde" -->
<ejemplo color="amarillo"></ejemplo>    <!-- No es correcto porque el
atributo "color" no tiene el valor "verde" -->
```