



DTD



¿Qué es un DTD?

Un DTD es una definición de tipo de documento.

DTD (Document Type Definition, Definición de Tipo de Documento) sirve para definir la estructura de un documento XML, permitiendo su **validación**.

Un documento XML es válido (valid) cuando, además de estar **bien formado**, no incumple ninguna de las normas establecidas en su estructura.

Existen otros métodos que también permiten validar documentos XML, como por ejemplo XML Schema .



Declaración Interna:

Una DTD se puede escribir tanto interna como externamente a un archivo XML:

Declaración Interna:

El DTD se declara dentro del archivo XML, debe estar envuelta dentro de la definición <!DOCTYPE>:

```
<!DOCTYPE elemento_raíz [
```

Definición de Tipo de Documento (DTD):

```
]>
```

Ejemplo:

Diseña un documento XML **bien formado y validado** que permita describir la información necesaria para un catálogo de artistas, donde para cada artista debe aparecer: nombre, apellido, grupo al que pertenece y nacionalidad:



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

<!DOCTYPE catalogo [

<!ELEMENT catalogo (artista)*>

<!ELEMENT artista (nombre,apellidos,grupo,nacionalidad)>

<!ELEMENT nombre (#PCDATA)>

<!ELEMENT apellidos (#PCDATA)>

<!ELEMENT grupo (#PCDATA)>

<!ELEMENT nacionalidad (#PCDATA)>

]>

<catalogo>

<artista>

<nombre>Marie</nombre>

<apellidos>Fredriksson</apellidos>

<grupo>Roxette</grupo>

<nacionalidad>Suecia</nacionalidad>

</artista>

<artista>

<nombre>Mark</nombre>

<apellidos>Knopfler</apellidos>

<grupo>Dire Straits</grupo>

<nacionalidad>Escocia</nacionalidad>

</artista>

</catalogo>

EXPLICACIÓN:

- Obsérvese que, en la DTD se ha indicado que **catalogo** es el elemento raíz del documento XML, el cual puede contener cero o más artista.: (artista)*
- Escribiendo artista(nombre, apellidos, grupo, nacionalidad) se especifica que, cada elemento "artista" tiene que contener cuatro elementos (hijos): "nombre", "apellidos", "grupo" y "nacionalidad".
- Con #PCDATA (Parsed Character Data) escrito entre paréntesis "()" se indica que los elementos "nombre", "apellidos", "grupo" y "nacionalidad" pueden contener datos (cadenas de caracteres).



Declaración Externa

Declaración Externa:

El DTD se declara en un archivo externo, la definición <!DOCTYPE> debe contener una referencia al archivo DTD:

```
<!DOCTYPE elemento-raíz SYSTEM "URI">
```



```
<?xml version="1.0" encoding="UTF-8"
standalone="no"?>
<!DOCTYPE catalogo SYSTEM "DTD/catalogo.dtd">
<catalogo>
```

```
  <artista>
    <nombre>Marie</nombre>
    <apellidos>Fredriksson</apellidos>
    <grupo>Roxette</grupo>
    <nacionalidad>Suecia</nacionalidad>
  </artista>
```

```
  <artista>
    <nombre>Mark</nombre>
    <apellidos>Knopfler</apellidos>
    <grupo>Dire Straits</grupo>
    <nacionalidad>Escocia</nacionalidad>
  </artista>
```

```
</catalogo>
```

```
<!ELEMENT catalogo (artista)*>
  <!ELEMENT artista (nombre, apellidos, grupo,
nacionalidad)>
    <!ELEMENT nombre (#PCDATA)>
    <!ELEMENT apellidos (#PCDATA)>
    <!ELEMENT grupo (#PCDATA)>
    <!ELEMENT nacionalidad (#PCDATA)>
```


Estructura

Estructura de un documento XML

En una DTD se pueden declarar:

- Elementos
- Atributos
- Entidades
- Notaciones

Por tanto, un documento XML será válido si –además de no tener errores de sintaxis– cumple lo indicado en las declaraciones de elementos, atributos, entidades y notaciones, de la DTD a la que esté asociado.



Elementos



Elementos:

Para declarar un elemento en una DTD se utiliza la siguiente sintaxis:

<!ELEMENT nombre-del-elemento tipo-de-contenido>

En el **tipo de contenido** se especifica el contenido permitido en el elemento, pudiendo ser:

- Texto (#PCDATA).
- Otros elementos (hijo1, hijo2). En un documento XML, los elementos (hijos) de un elemento (padre), deben escribirse en el mismo orden en el que han sido declarados en la DTD.
- Estar vacío, EMPTY.
- pueden contener cualquier combinación de datos analizables (texto y otros elementos), ANY

Elementos con contenido de tipo texto (#PCDATA): Estos elementos podrían estar vacíos y el documento XML seguiría siendo válido:

Un elemento vacío (EMPTY) puede tener atributos: Aunque un elemento se declare vacío, no pudiendo contener texto ni otros elementos, sí puede tener atributos.

Elementos con cualquier contenido ANY: estos elementos pueden contener :

- Texto
- Otros elementos
- Mezcla de elementos y texto
- Estar vacíos

Secuencias de elementos: Un elemento (padre) puede ser declarado para contener a otro u otros elementos (hijos). En la sintaxis, los hijos tienen que escribirse entre paréntesis "(" y separados por comas ",".

A su vez, los hijos también pueden tener sus propios hijos.

En un documento XML, los elementos (hijos) de un elemento (padre), deben escribirse en el mismo orden en el que han sido declarados en la DTD.

Ejercicio 1

Diseña un documento XML bien formado y validado que permita describir la información necesaria para una lista de alumnos, donde para cada alumno debe aparecer: nombre, apellido, edad, dirección y teléfono.

- El elemento edad puede contener texto y otros elementos.
- El elemento direccion tiene 3 elementos hijos: calle, número y ciudad
- El elemento teléfono debe declararse como vacío.

Cardinalidad de los elementos en una DTD

En una DTD, para definir el número de veces que pueden aparecer los elementos de un documento XML, se pueden utilizar los operadores de cardinalidad mostrados en la siguiente tabla:

Los elementos sobre los que no actúe **ningún operador** de cardinalidad, tendrán que aparecer **obligatoriamente una única vez**, en el o los documentos XML a los que se asocie.

| Operadores de cardinalidad en DTD | | |
|-----------------------------------|--------------|--|
| Operador | Cardinalidad | Significado |
| ? (interrogación) | 0-1 | El elemento es opcional, pudiendo aparecer una sola vez o ninguna. |
| * (asterisco) | 0-n | El elemento puede aparecer cero, una o más veces. |
| + (signo más) | 1-n | El elemento tiene que aparecer, obligatoriamente, una o más veces. |

Ejemplo:

Diseña un documento XML **bien formado y validado** que permita describir la información necesaria para un recetario de cocina, donde para cada receta debe aparecer: nombre, ingrediente, preparación, calorías , dificultad y tipo.

El elemento ingrediente tiene que aparecer 1 o más veces

El elemento **preparación** está formado por el elemento hijo **paso** que se puede repetir 1 o más veces.

El elemento tipo puede estar vacío, contener texto u otros elementos.

El elemento tipo es opcional puede aparecer 1 vez o no aparecer



Elementos opcionales en una DTD

En la DTD asociada a un documento XML, se pueden declarar elementos que contengan elementos opcionales. Para ello, se utiliza el operador de elección, representado por una barra vertical (|).

EJEMPLO En el siguiente documento XML el elemento "articulo" puede contener un elemento "codigo" o un elemento "id"; obligatoriamente uno de ellos, pero no ambos:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE articulo [
    <!ELEMENT articulo (codigo | id)>
    <!ELEMENT codigo (#PCDATA)>
    <!ELEMENT id (#PCDATA)>
]>

<articulo>
    <codigo>AF-33</codigo>
</articulo>
```


Ejercicio 2

Diseña un documento XML **bien formado y validado** que permita almacenar la información sobre los alumnos de un colegio, donde para cada **alumno** debe aparecer: **nombre, apellido, curso al que pertenece, edad y datos de contacto**.

El elemento **apellido** aparece 1 o más veces

El elemento **edad** tiene que estar vacío.

El elemento **datos de contacto** está formado por el elemento :direccion, telefono y email.

- El elemento direccion tiene 3 elementos hijos: calle, numero y ciudad.
- El elemento telefono aparece 0 o mas veces
- El elemento email es opcional, aparece 1 vez o ninguna.



Atributos

Declaración de atributos

La sintaxis básica para declarar un atributo en una DTD es:

<!ATTLIST nombre-del-elemento nombre-del-atributo tipo-de-atributo valor-del-atributo>



Tipos de atributos en DTD

| Tipo | Descripción |
|--------------|---|
| CDATA | (<i>Character DATA</i>) El valor son datos de tipo carácter, es decir, texto. |
| (en1 en2 ..) | El valor puede ser uno de los pertenecientes a una lista de valores escritos entre paréntesis "()" y separados por el carácter " ". |
| ID | El valor es un identificador único. Debe ser un nombre XML válido (El primer carácter tiene que ser una letra o un <i>guion bajo</i> "_") |
| IDREF | El valor es un identificador que tiene que existir en otro atributo ID del documento XML. |

Valor del atributo en DTD

| <i>Valor</i> | <i>Significado</i> |
|---|---|
| Valor entre comillas dobles (") o simples ('). | El atributo tiene un valor por defecto. |
| #REQUIRED | El atributo es obligatorio escribirlo. |
| #IMPLIED | El atributo es opcional escribirlo. |
| #FIXED valor entre comillas dobles (") o simples ('). | El valor del atributo es fijo. |

Ejemplo:

Diseña un documento XML **bien formado y validado** que permita describir la información necesaria para un catálogo de artistas, donde para cada artista debe aparecer: nombre, apellido, grupo al que pertenece y nacionalidad.

El elemento **artista** debe tener **obligatoriamente** un **atributo** tipo **identificador** único.

El elemento **grupo** puede tener un atributo con **nombre 'instrumento'** de tipo **opcional** cuyo valor puede ser: guitarra, bajo o batería.



Ejemplo:

Diseña un documento XML **bien formado y validado** que permita describir la información necesaria para un recetario de repostería, donde para cada receta debe aparecer: nombre, ingrediente, preparación.

El elemento ingrediente tiene que aparecer 1 o más veces.

El elemento preparación está formado por el elemento hijo paso que se puede repetir 1 o más veces.

El elemento **receta** debe tener **obligatoriamente** un **atributo** con nombre **'dificultad'** cuyo valor puede ser: baja, media, alta.

El elemento **nombre** tiene un atributo con nombre **'tipo'** cuyo valor por defecto será **postre**.



Nota: El elemento **nombre** tiene un atributo con nombre **'tipo'** cuyo valor por defecto será **postre**.

Al tener este atributo un valor por defecto, en los elementos 'nombre' que no definas un valor para este atributo tendrán el valor 'postre' como se observa al cargar el documento XML en un navegador:

```
▼<recetario>
  ▼<receta dificultad="baja">
    <nombre tipo="pincho">tortilla</nombre>
    <ingrediente>huevo</ingrediente>
    ▼<preparacion>
      <paso>batir huevo</paso>
      <paso>freir</paso>
    </preparacion>
  </receta>
  ▼<receta dificultad="alta">
    <nombre tipo="postre">Tiramisú</nombre>
    <ingrediente>Mascarpone</ingrediente>
    <ingrediente>café</ingrediente>
    <ingrediente>bizcocho</ingrediente>
    <ingrediente>cacao</ingrediente>
    ▼<preparacion>
      <paso>empapar los bizcochos en café</paso>
      <paso>intercalar capa de bizcochos con mascarpone</paso>
      <paso>espolvorear cacao</paso>
    </preparacion>
  </receta>
```


Ejercicio 3

Diseña un documento XML bien formado y validado utilizando DTD que permita almacenar la información sobre **jugadores de fútbol**, donde para cada **jugador** debe aparecer: nombre, apellido, el equipo al que pertenece, división, sueldo.

ELEMENTOS:

- tipo de contenido

El elemento sueldo debe declararse vacío.

- cardinalidad:

El elemento apellidos puede aparecer: 1-n veces.

El elemento equipo es opcional puede aparecer: 0-1

ATRIBUTOS:

El elemento **equipo** tiene un **atributo** para almacenar el **país** al que pertenece ese equipo, que por defecto será 'España'.

El elemento **sueldo** tiene un **atributo** opcional con la moneda utilizada.

Ejercicio 4

Diseña un documento XML bien formado y validado utilizando DTD que permita almacenar la información sobre la **plantilla de Zara**, donde para cada **empleado** debe aparecer: nombre, apellido, el departamento al que pertenece, edad, nacionalidad e información..

ELEMENTOS:

- tipo de contenido:

El elemento información puede contener texto, otros elementos o estar vacío.

El elemento edad debe declararse vacío.

El elemento nacionalidad está formado únicamente por uno de los siguientes elementos hijo 'europa', 'america', 'asia', o 'africa'. Ejemplo:

```
<nacionalidad>
```

```
  <europa>España</europa>
```

```
</nacionalidad>
```

- Cardinalidad:

El elemento apellidos puede aparecer: 1-n veces.

El elemento info es opcional puede aparecer: 0-1

ATRIBUTOS:

El elemento **departamento** debe tener obligatoriamente el atributo '**puesto**' cuyo valor puede ser: dependiente o encargado.

El elemento **empleado** tiene obligatoriamente el atributo 'id' con un identificador único para cada empleado.

