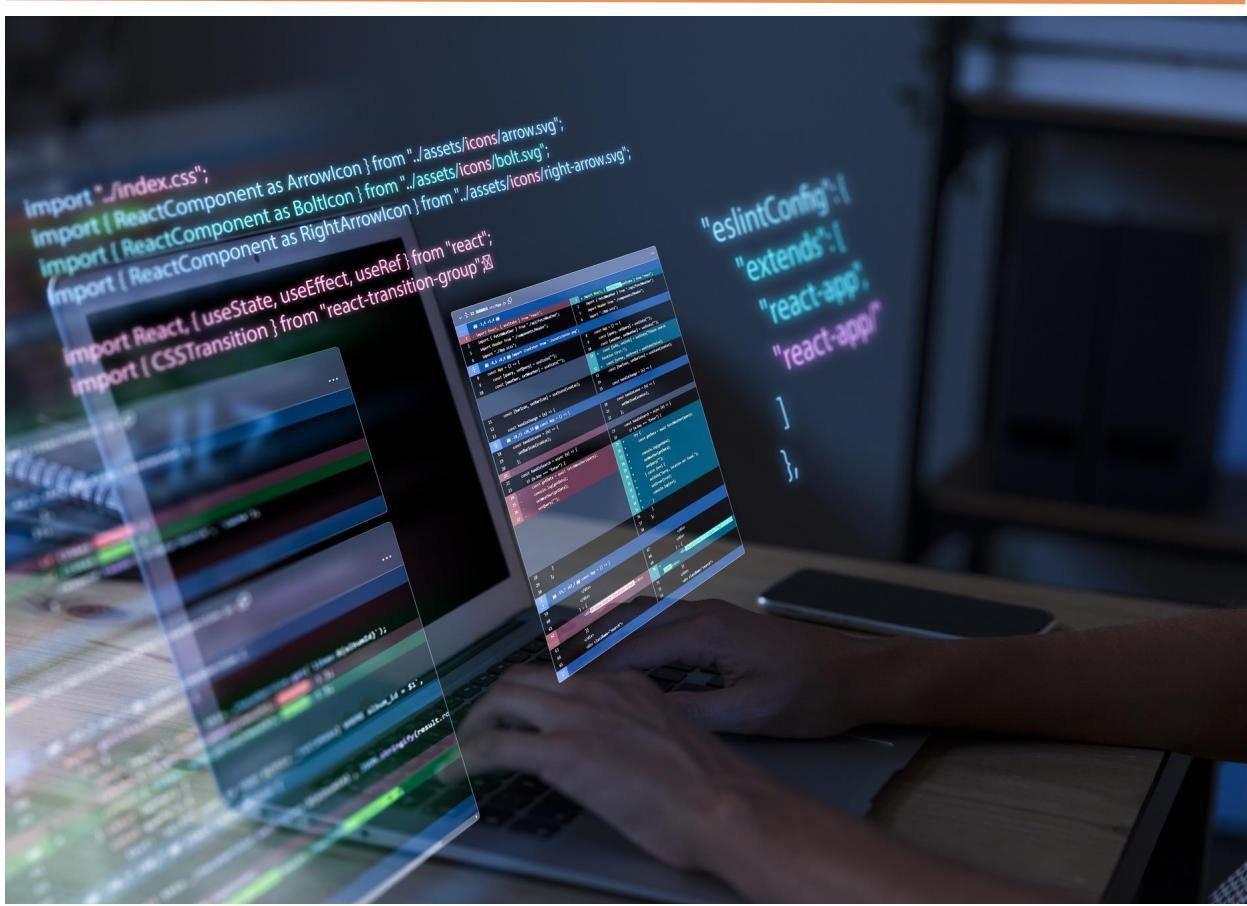


XML, DTD y Schemas

3.2 DTD



Autor: Lisa ERIKSEN

Fecha: 2025 / 2026

1. ¿Qué es DTD y por qué es útil?

DTD (Document Type Definition) es una especificación que define la estructura y las reglas que debe seguir un documento XML para considerarse válido. Al usar un DTD, aseguramos que los datos en un archivo XML cumplen con una estructura predefinida, lo que facilita la interoperabilidad y consistencia en el intercambio de datos.

1. Explica con tus palabras para qué sirve un DTD.
2. ¿Por qué es importante cuando varias empresas intercambian archivos XML?
3. Da un ejemplo de un documento XML cotidiano que necesite validación.

2. DTD Interno o Externo

2.1. DTD interno

Un DTD interno se define dentro del documento XML, entre las etiquetas `<!DOCTYPE>`. Aquí tienes un ejemplo de un archivo XML con DTD interno:

```
<?xml version="1.0"?>
<!DOCTYPE libro [
    <!ELEMENT libro (titulo, autor)>
    <!ELEMENT titulo (#PCDATA)>
    <!ELEMENT autor (#PCDATA)>
]>
<libro>
    <titulo>Programación en XML</titulo>
    <autor>María Pérez</autor>
</libro>
```

Actividad:

1. Crea un archivo **libro.xml** y copia el código de arriba
2. Según el DTD, qué elementos tienen que estar presentes dentro del XML.
3. Crea una nueva etiqueta `<descripcion>` en la parte XML
4. Añade el elemento correspondiente en la parte DTD para que la validación sea correcta.

2.2. DTD externo

Un DTD externo se define en un archivo separado y se referencia desde el XML.

1. Crea el archivo **libro.dtd**:

```
<!ELEMENT libro (titulo, autor)>
<!ELEMENT titulo (#PCDATA)>
<!ELEMENT autor (#PCDATA)>
```

2. Suprime la parte DTD en la etiqueta `<!DOCTYPE>` de tu archivo **libro.xml** y reemplaza la por la importación del archivo **libro.dtd**:

```
<!DOCTYPE libro SYSTEM "libro.dtd">
```

5. Valida tu archivo XML. Añade la extensión XML (Red Hat) para ver los errores en rojo.

3. Sintaxis de DTD

3.1. Los elementos

Define los elementos permitidos en el XML y su contenido.

Elemento	Descripción
<code><!ELEMENT libro (titulo, autor) ></code>	El elemento <code><libro></code> contiene <code><titulo></code> y <code><autor></code> en ese orden.
<code><!ELEMENT titulo (#PCDATA) ></code>	El contenido de <code><titulo></code> será texto .
<code><!ELEMENT imagen EMPTY ></code>	El elemento <code><imagen></code> no tendrá contenido.
<code><!ELEMENT comentario (#PCDATA negrita cursiva)* ></code>	<code><comentario></code> puede contener texto o una mezcla de <code><negrita></code> y <code><cursiva></code>

3.2. Crea el DTD de una tienda online

Actividad: Imagina que trabajas en una empresa que recibe miles de archivos XML de diferentes proveedores. Estás definiendo un catálogo de productos para una tienda online.

1. Crea un nuevo archivo **techshop.xml**
2. Escribe el código siguiente:

```
<producto>
  <nOMBRE>Nike Air Force 1</nOMBRE>
  <IMAGEN></IMAGEN>
  <PRECIO moneda="EUR">100</PRECIO>
  <CATEGORÍA>Zapatillas</CATEGORÍA>
</PRODUCTO>
```

2. Crea un archivo llamado **techshop.dtd** y ayúdate de la tabla de arriba para escribir un DTD que defina:
 - <producto> → (nombre, precio)
 - <categoría> → texto
 - <imagen> → elemento vacío
3. Inserta tu DTD en tu archivo **techshop.xml** con la línea:

```
<!DOCTYPE libro SYSTEM " techshop.dtd">
```

4. Abre el XML y observa los errores.
5. Corrige el XML para que sea válido.

3.3. Los atributos

El DTD permite añadir una línea que define los **atributos** que puede tener un elemento. Se escribe de la siguiente forma:

```
<!ATTLIST elemento atributo tipo-del-atributo obligación >
```

Tipos de atributos	Descripción
CDATA	Cualquier texto.
ID	Identificador único.
ENUMERATED	Lista de valores permitidos.

Observa un ejemplo de código XML

```
<libro id="libro01" genero="poesia">
</libro>
```

Observa el código DTT

```
<!ATTLIST libro id ID #REQUIRED>
<!ATTLIST libro genero (ficción | no_ficción | poesía) "ficción">
```

Pregúntate:

- Cuales son los dos atributos que tiene la etiqueta <libro>
- ¿Que significa **REQUIRED** y a que atributo se refiere?
- ¿Mira la tabla de arriba, de que **tipo** es el atributo id? ¿Y el atributo genero?
- ¿Cual es el valor por defecto del atributo genero?

Actividad: En el archivo **techshop.dtd**, que se utiliza para validar los productos, realiza las siguientes modificaciones:

1. Añade un atributo **id** obligatorio, de tipo **ID** (debe ser único) para el producto.
2. Añade un atributo **stock** opcional con valores posibles: **alto | medio | bajo** con valor **por defecto: medio**
3. Actualiza tu archivo **techshop.xml** para que cumpla correctamente con el DTD modificado y sea **válido**.

3.4. Las Entidades

Las entidades permiten definir **valores reutilizables** o caracteres especiales. En el ejemplo siguiente, definimos la entidad **autor** para poder usarla varias veces, sin escribir el nombre del autor.

```
<!ENTITY autor "Paco Po">
```

En el XML, se usa la entidad con el carácter &.

```
<nOMBRE>&autor;</nOMBRE>
```

1. En tu archivo **techshop.dtd**, define:
 - Una entidad con el nombre de la tienda: "TechShop"
2. En tu archivo **techshop.xml** añade:
 - Una línea con la entidad
3. Valida el XML para verificar que todo este correcto.

3.5. Comentarios

Los comentarios se añaden con:

```
<!-- Este es un comentario -->
```

4. Validación de DTD

Para validar un archivo XML contra un DTD usa herramientas como:

- **Oxygen XML Editor**.
- **VS Code** (con extensiones).
- **xmllint**: Línea de comandos para validar XML.



```
Terminal
xmllint --valid --noout errores.xml
```

Actividad: Identifica **errores habituales en la validación de documentos XML** y aprender a corregirlos mediante el uso de DTD.

1. Descarga los archivos **errores.dtd** y **errores.xml**.
2. Intenta validar el XML con el DTD con la línea de comando de arriba.
3. Analiza los mensajes de error y **corrige el DTD y/o el XML** según corresponda.
4. Repite la validación hasta que el documento sea **totalmente válido**.
5. Introduce un nuevo error en el XML y reta a un compañero a encontrarlo.