

Explicación Detallada: Conversión de la Tabla 'Coche' a XML y DTD

1. Contexto y Objetivo

- Se dispone de una tabla SQL llamada 'Coche' con campos como ID_coche, marca, modelo, matrícula, kilometraje, año de fabricación, precio, cv y color.
- El objetivo es transformar esta estructura relacional a un documento XML para facilitar intercambio de datos y validación mediante DTD.

2. Estructura del Documento XML

a) Prolog:

- Declaración XML: `<?xml version="1.0" encoding="UTF-8"?>`
- DOCTYPE: Referencia al archivo 'coches.dtd' para validación.

b) Elemento Raíz <Coches>: Envoltorio que agrupa múltiples elementos <Coche>.

c) Elemento <Coche>: Cada instancia representa un registro de la tabla original.

- Sub-elementos en orden secuencial según definición DTD.

3. Mapeo de Columnas a Elementos

- <ID_coche>: Identificador único del coche (tipo entero).
- <marca>, <modelo>, <matricula>, <color>: Datos alfanuméricos (cadena).
- <kilometraje>, <precio>: Valores numéricos (flotantes).
- <ano_fabricacion>, <cv>: Números enteros.

4. Definición del DTD

- <!ELEMENT Coches (Coche*)>: <Coches> puede contener cero o más <Coche>.
- <!ELEMENT Coche (ID_coche, marca, modelo, matricula, kilometraje, ano_fabricacion, precio, cv, color)> Secuencia obligatoria y ordenada de elementos.
- Cada sub-elemento definido como (#PCDATA) para textos simples.

5. Limitaciones y Buenas Prácticas

- DTD valida estructura y orden, pero no valida tipos de datos (se usan #PCDATA).
- Para validación de tipos más estricta, se recomienda usar XML Schema (XSD).
- Utilizar herramientas como 'xmllint --noout --dtdvalid coches.dtd coches.xml' para validar.

6. Ejemplo de Validación y Uso

- Compartir el XML con sistemas externos que acepten DTD.
- Transformar con XSLT para generar HTML o integrar con otros XML.

Con esta documentación se asegura comprensión completa de la estructura, mapeo de datos y proceso de validación de los archivos XML y DTD generados.