

XML. Ejercicios DTD.

1. Ejercicio “RUTAS”

Se desea generar una estructura de archivo, en formato XML, para validar cada ruta de turismo (montaña, monumentos, etc.) del sistema.

(El presente ejercicio desarrolla, paso a paso, el documento DTD para la validación de las rutas, conforme al documento XML que se indica).

► Se pide: generar el documento XML “ruta01.xml” y el documento DTD “rutas.dtd”, siguiendo las instrucciones que se detallan a continuación. Comprobar que está bien formado y validar.

(Nota: texto con fondo verde se corresponde con el enunciado. Texto con fondo azul se corresponde con la solución).

▫ Documento XML: “ruta01.xml”

```
<?xml version="1.0" encoding="UTF-8"?>
<ruta cod="R01" tiporuta="Montaña">
  <denominacion>Ruta de las Cascadas</denominacion>
  <distancia>8 Km</distancia>
  <dificultad>Baja</dificultad>
  <puntosinteres>
    <punto pos="P01">
      <nombre>Cascada 1</nombre>
      <puntokilometrico>1,5</puntokilometrico>
    </punto>
    <punto pos="P02">
      <nombre>Cascada 2</nombre>
      <puntokilometrico>2,3</puntokilometrico>
    </punto>
    <punto pos="P03">
      <nombre>Cascada 3</nombre>
      <coordenadas>259;367</coordenadas>
    </punto>
  </puntosinteres>
  <plano tipo="JPG">MapaR01.jpg</plano>
</ruta>
```

La primera línea del documento XML nos indica el tipo de documento, la versión utilizada y la codificación.

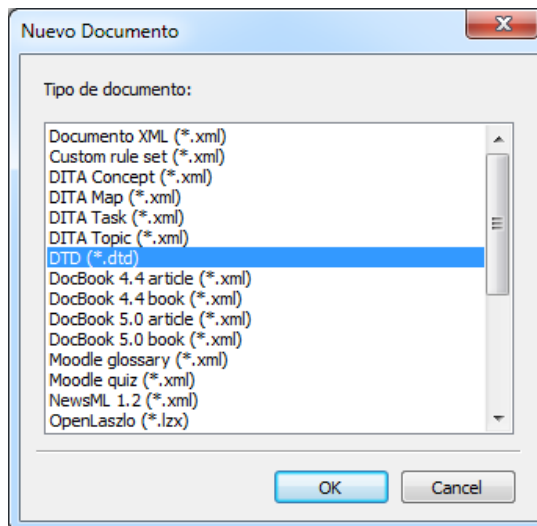
Como paso previo identificamos el nodo raíz en el documento XML, en este caso “ruta”.

```
<?xml version="1.0" encoding="UTF-8"?>
<ruta cod="R01">
  ...
</ruta>
```

1.1. Crear archivo DTD.

El primer paso será crear el documento DTD, al que denominaremos “rutas.dtd”.

- Documento DTD: “rutas.dtd”



1.2. Asociar documento XML a documento DTD.

A continuación, procedemos a insertar en el documento XML su asociación con el documento DTD creado.

```
<!DOCTYPE ruta SYSTEM "rutas.dtd">
```

- Documento XML: “ruta01.xml”

De esta forma la cabecera del documento XML quedará de la siguiente manera:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ruta SYSTEM "rutas.dtd">
<ruta cod="R01">
....
```

1.3. Construcción del nodo principal en el documento DTD.

A partir de este momento comenzaremos a desarrollar el documento DTD, desgranando el árbol de nodos poco a poco, comenzando por el nodo raíz, “ruta”.

```
<ruta cod="R01" tiporuta="Montaña">
  <denominacion>Ruta de las Cascadas</denominacion>
  <distancia>8 Km</distancia>
  <dificultad>Baja</dificultad>
  <puntosinteres>
    ...
  </puntosinteres>
  <plano tipo="JPG">MapaR01.jpg</plano>
</ruta>
```

Elemento raíz “ruta”, composición:

- Se compone de los siguientes elementos: “denominación”, “distancia”, “dificultad”, “puntos de interés” y “plano”.

```
<!ELEMENT ruta (denominacion,distancia,dificultad,puntosinteres,plano)>
```

Todos ellos van a aparecer 1 vez, por lo que no se indica cardinalidad especial.

- Presenta dos atributos: “cod” y “tiporuta”
 - o El atributo “cod” es identificador único.
 - o El atributo “tiporuta” adoptará uno de los siguientes valores: “Montaña, Ciclista, Monumental”, no pudiendo quedar vacío.

```
<!ATTLIST ruta
    cod ID #REQUIRED
    tiporuta (Montaña | Ciclista | Monumental) #REQUIRED>
```

▫ Documento DTD: “rutas.dtd”

```
<!ELEMENT ruta (denominacion, distancia, dificultad, puntosinteres, plano)>
<!ATTLIST ruta
    cod ID #REQUIRED
    tiporuta (Montaña | Ciclista | Monumental) #REQUIRED>
```

En este instante hemos definido el elemento raíz “ruta” de forma completa (composición y atributos).

1.4. Construcción de elementos simples.

Dentro de la composición del elemento raíz “ruta” existen 3 elementos simples, “denominación”, “distancia”, “dificultad”, que no contienen nada más que datos simples (no tienen elementos hijos ni datos complejos).

```
<denominacion>Ruta de las Cascadas</denominacion>
<distancia>8 Km</distancia>
<dificultad>Baja</dificultad>
```

En el documento DTD procedemos a declarar los 3 elementos:

```
<!ELEMENT denominacion (#PCDATA)>
<!ELEMENT distancia (#PCDATA)>
<!ELEMENT dificultad (#PCDATA)>
```

(Nota: en los DTD no se puede restringir el contenido de los elementos, por lo que si quisiéramos que “dificultad” solo pudiese contener los valores “Alta, Media, Baja” deberíamos convertirlo en atributo de su elemento superior, en este caso el elemento raíz “ruta”. Este paso lo realizaremos al final del ejercicio, modificando, así mismo, el documento XML).

▫ Documento DTD: “rutas.dtd”

```
<!ELEMENT ruta (denominacion, distancia, dificultad, puntosinteres, plano)>
<!ATTLIST ruta
```

```
cod ID #REQUIRED
tiporuta (Montaña | Ciclista | Monumental) #REQUIRED>
```

```
<!ELEMENT denominacion (#PCDATA)>
<!ELEMENT distancia (#PCDATA)>
<!ELEMENT dificultad (#PCDATA)>
```

1.5. Construcción de elementos complejos (nodos hijos)

El siguiente nodo, “puntosinteres”, está compuesto de elementos hijo “punto”, pudiendo aparecer 1 o varias veces.

```
<puntosinteres>
  <punto pos="P01">
    ...
  </punto>
  <punto pos="P02">
    ...
  </punto>
  <punto pos="P03">
    ...
  </punto>
</puntosinteres>
```

En el documento DTD procedemos a declarar el elemento “puntosinteres”, que está compuesto por el elemento “punto”, pudiendo suceder 1 o varias veces:

```
<!ELEMENT puntosinteres (punto+)>
```

▫ Documento DTD: “rutas.dtd”

```
<!ELEMENT ruta (denominacion, distancia, dificultad, puntosinteres, plano)>
<!ATTLIST ruta
  cod ID #REQUIRED
  tiporuta (Montaña | Ciclista | Monumental) #REQUIRED>

<!ELEMENT denominacion (#PCDATA)>
<!ELEMENT distancia (#PCDATA)>
<!ELEMENT dificultad (#PCDATA)>

<!ELEMENT puntosinteres (punto+)>
```

1.6. Construcción de elementos hijo simples:

Siguiendo la estructura del documento XML, dentro de la composición del elemento “puntosinteres”, nos encontramos con el elemento “punto”.

```
<punto pos="P01">
  <nombre>Cascada 1</nombre>
  <puntokilometrico>1,5</puntokilometrico>
</punto>
```

```
<punto pos="P02">
  <nombre>Cascada 2</nombre>
  <puntokilometrico>2,3</puntokilometrico>
</punto>
<punto pos="P03">
  <nombre>Cascada 3</nombre>
  <coordenadas>259;367</coordenadas>
</punto>
```

Elemento “punto”, composición:

- Se compone de los siguientes elementos: “nombre”, “puntokilometrico” o “coordenadas” (el elemento “nombre” siempre aparece, pero los elementos “puntokilometrico” y “coordenadas” o aparece uno o aparece el otro, siendo excluyentes). Todos ellos de tipo texto y que solo suceden una vez.

```
<!ELEMENT punto (nombre, (puntokilometrico | coordenadas))>
```

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

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

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

Todos ellos van a aparecer 1 vez, por lo que no se indica cardinalidad especial.

- Presenta el atributo “pos”, que es identificador único.

```
<!ATTLIST punto
  pos ID #REQUIRED>
```

▫ Documento DTD: “rutas.dtd”

```
<!ELEMENT ruta (denominacion, distancia, dificultad, puntosinteres, plano)>
```

```
<!ATTLIST ruta
```

```
  cod ID #REQUIRED
```

```
  tiporuta (Montaña | Ciclista | Monumental) #REQUIRED>
```

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

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

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

```
<!ELEMENT puntosinteres (punto+)>
```

```
<!ELEMENT punto (nombre, (puntokilometrico | coordenadas))>
```

```
<!ATTLIST punto
```

```
  pos ID #REQUIRED>
```

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

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

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

1.7. Construcción elementos con datos complejos:

Siguiendo desarrollando el documento DTD a medida que se lee el documento XML en sentido vertical, el siguiente elemento que nos encontramos es “plano”, que depende del nodo raíz “ruta”. Este elemento solo aparece una vez y contiene un atributo.

```
<plano tipo="JPG">MapaR01.jpg</plano>
```

Este nodo contiene información referente a un archivo fotográfico cuyo contenido no debe ser analizado, por lo que haremos uso de notaciones.

Elemento “plano”, composición:

- Contenido del elemento:

```
<!ELEMENT plano (#PCDATA)>
```
- Presenta un atributo: “tipo”
 - o El atributo “tipo” indica el tipo de archivo MIME de la imagen.
 - o Dado que hacemos referencia a códigos MIME habrá que indicar las correspondientes notaciones (aunque el archivo que se indicará en el documento XML es .jpg, añadiremos también la notación al formato .gif).

```
<!NOTATION GIF SYSTEM "image/gif">
<!NOTATION JPG SYSTEM "image/jpeg">
<!ATTLIST plano
    tipo NOTATION (GIF | JPG) #REQUIRED>
```

Nota: si los ficheros fuesen conocidos se podrían generar entidades con la ruta de los mismos, indicando el tipo NDATA para evitar su procesamiento. En este caso suponemos que la persona que remite el documento XML adjuntará la imagen del plano de la ruta, no pudiendo ser conocido por el sistema anteriormente.

▫ Documento DTD: “rutas.dtd”

```
<!ELEMENT ruta (denominacion, distancia, dificultad, puntosinteres, plano)>
<!ATTLIST ruta
    cod ID #REQUIRED
    tiporuta (Montaña | Ciclista | Monumental) #REQUIRED>

<!ELEMENT denominacion (#PCDATA)>
<!ELEMENT distancia (#PCDATA)>
<!ELEMENT dificultad (#PCDATA)>

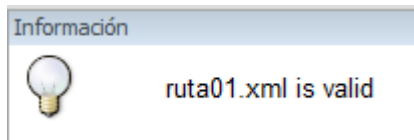
<!ELEMENT puntosinteres (punto+)>

<!ELEMENT punto (nombre, (puntokilometrico | coordenadas))>
<!ATTLIST punto
    pos ID #REQUIRED>

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

```
<!ELEMENT plano (#PCDATA)>
<!NOTATION GIF SYSTEM "image/gif">
<!NOTATION JPG SYSTEM "image/jpeg">
<!ATTLIST plano
    tipo NOTATION (GIF | JPG) #REQUIRED>
```

En este momento se puede validar el documento XML con el documento DTD asociado, siendo totalmente válido.



1.8. Depuración del fichero. Uso de Entidades.

Para mejorar la lectura del documento DTD, así como facilitar ampliaciones futuras, vamos a hacer uso de entidades parámetro en la descripción del contenido de elementos. En este caso vamos a aplicarlo sobre el elemento raíz “ruta” y sobre el elemento “punto”.

También vamos a aplicarlo sobre la descripción del contenido del atributo “tiporuta” del elemento “ruta”.

Declaración de entidades parámetro:

```
<!ENTITY % datosRuta "denominacion, distancia, dificultad, puntosinteres,
plano">
<!ENTITY % datosPuntos "nombre, (puntokilometrico | coordenadas)">
<!ENTITY % tiposRutas "Montaña | Ciclista | Monumental">
```

Llamada a entidades parámetro (sustitución del contenido declarado en los elementos “ruta” y “punto”, así como el atributo “tiporuta”):

```
<!ELEMENT ruta (%datosRuta;)>
<!ATTLIST ruta
    ...
    tiporuta (%tiposRutas;) #REQUIRED>
...
<!ELEMENT punto (%datosPuntos;)>
```

También reordenamos el fichero, declarando las entidades y las notaciones al comienzo del fichero.

▫ Documento DTD: “rutas.dtd”

```
<!ENTITY % datosRuta "denominacion, distancia, dificultad, puntosinteres,
plano">
<!ENTITY % datosPuntos "nombre, (puntokilometrico | coordenadas)">
<!ENTITY % tiposRutas "Montaña | Ciclista | Monumental">
```

```
<!NOTATION GIF SYSTEM "image/gif">
<!NOTATION JPG SYSTEM "image/jpeg">

<!ELEMENT ruta (%datosRuta;)>
<!ATTLIST ruta
    cod ID #REQUIRED
    tiporuta (%tiposRutas;) #REQUIRED>

<!ELEMENT denominacion (#PCDATA)>
<!ELEMENT distancia (#PCDATA)>
<!ELEMENT dificultad (#PCDATA)>

<!ELEMENT puntosinteres (punto+)>

<!ELEMENT punto (%datosPuntos;)>
<!ATTLIST punto
    pos ID #REQUIRED>

<!ELEMENT nombre (#PCDATA)>
<!ELEMENT puntokilometrico (#PCDATA)>
<!ELEMENT coordenadas (#PCDATA)>

<!ELEMENT plano (#PCDATA)>
<!ATTLIST plano
    tipo NOTATION (GIF | JPG) #REQUIRED>
```

Lógicamente, el documento XML continúa siendo válido.

1.9. Modificación información “dificultad”.

Como se indicó inicialmente, en los DTD no se puede restringir el contenido de los elementos, por lo que si quisiéramos que “dificultad” solo pudiese contener los valores “Alta, Media, Baja” deberíamos convertirlo en atributo de su elemento superior, en este caso el elemento raíz “ruta”.

Esta variación va a suponer la modificación del DTD (eliminar el elemento “dificultad” e indicarlo como atributo de “ruta”), así como la adecuación del documento XML a la nueva estructura.

Modificación documento DTD:

- Eliminación elemento “dificultad”.
~~<!ELEMENT dificultad (#PCDATA)>~~
- Eliminación de la declaración del elemento “dificultad” dentro de la declaración de composición del elemento “ruta” (recordar que dicha declaración la hemos especificado como entidad).
~~<!ENTITY % datosRuta "denominacion, distancia, dificultad, puntosinteres, plano">~~
- Creación de nuevo atributo “dificultad” dentro del elemento “ruta”.
<!ATTLIST ruta


```
...
    dificultad (Alta | Media | Baja) #REQUIRED>
```

Así mismo, para depurar el documento XML vamos a crear una entidad que agrupe los posibles valores que puede adoptar “dificultad”.

```
<!ENTITY % tiposDificultad "Alta | Media | Baja">
```

Resultando, por tanto, la definición del atributo “dificultad”:

```
<!ATTLIST ruta
```

```
...
    dificultad (%tiposDificultad;) #REQUIRED>
```

Modificación documento XML:

- Eliminación elemento “dificultad”.

```
<difficulty>Baja</difficulty>
```

- Indicación de la información como atributo del elemento “ruta”.

```
<ruta cod="R01" tiporuta="Montaña" dificultad="Baja">
```

▫ Documento XML: “ruta01.xml”

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ruta SYSTEM "rutas.dtd">
<ruta cod="R01" tiporuta="Montaña" dificultad="Baja">
  <denominacion>Ruta de las Cascadas</denominacion>
  <distancia>8 Km</distancia>
  <puntosinteres>
    <punto pos="P01">
      <nombre>Cascada 1</nombre>
      <puntokilometrico>1,5</puntokilometrico>
    </punto>
    <punto pos="P02">
      <nombre>Cascada 2</nombre>
      <puntokilometrico>2,3</puntokilometrico>
    </punto>
    <punto pos="P03">
      <nombre>Cascada 3</nombre>
      <coordenadas>259;367</coordenadas>
    </punto>
  </puntosinteres>
  <plano tipo="JPG">MapaR01.jpg</plano>
</ruta>
```

▫ Documento DTD: “rutas.dtd”

```
<!ENTITY % datosRuta "denominacion, distancia, puntosinteres, plano">
<!ENTITY % datosPuntos "nombre, (puntokilometrico | coordenadas)">

<!ENTITY % tiposRutas "Montaña | Ciclista | Monumental">
<!ENTITY % tiposDificultad "Alta | Media | Baja">

<!NOTATION GIF SYSTEM "image/gif">
```

```
<!NOTATION JPG SYSTEM "image/jpeg">

<!ELEMENT ruta (%datosRuta;)>
<!ATTLIST ruta
    cod ID #REQUIRED
    tiporuta (%tiposRutas;) #REQUIRED
    dificultad (%tiposDificultad;) #REQUIRED>

<!ELEMENT denominacion (#PCDATA)>
<!ELEMENT distancia (#PCDATA)>

<!ELEMENT puntosinteres (punto+)>

<!ELEMENT punto (%datosPuntos;)>
<!ATTLIST punto
    pos ID #REQUIRED>

<!ELEMENT nombre (#PCDATA)>
<!ELEMENT puntokilometrico (#PCDATA)>
<!ELEMENT coordenadas (#PCDATA)>

<!ELEMENT plano (#PCDATA)>
<!ATTLIST plano
    tipo NOTATION (GIF | JPG) #REQUIRED>
```

El documento XML “ruta01.xml” está bien formado y es válido, conforme a las especificaciones contenidas en el documento DTD “rutas.dtd”.

2. Ejercicio “EQUIPO”

Se desea generar una estructura de archivo, en formato XML, para que cada equipo que participa en alguna liga comunique a la Federación sus integrantes.

(El presente ejercicio comienza desde una estructura simple y, en cada apartado, se van añadiendo componentes DTD con objeto de facilitar su comprensión y elaboración).

► Se pide: generar el documento XML “equipo_partido.xml” y el documento DTD “equipo.dtd”, siguiendo las instrucciones que se detallan a continuación. Comprobar que está bien formado y validar.

2.1. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<equipo>
  <jugadores>Yo</jugadores>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, asociar al documento XML y validar.

El elemento “jugadores” tiene que aparecer 1 vez.

▫ Asociación documento XML con DTD:

En el propio documento XML insertamos la línea que lo asocia al DTD que se utilizará para su validación.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo>
...
```

▫ Documento DTD:

El documento XML solo consta de dos elementos: equipo, que es el nodo raíz, y jugadores (en este caso una única ocurrencia, y contiene texto).

```
<!ELEMENT equipo (jugadores)>

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

2.2. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo">
  <jugadores>Yo</jugadores>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar.

Se ha añadido al elemento “equipo” (nodo raíz) el atributo “denominación”, que es obligatorio.

▫ Documento DTD:

```
<!ELEMENT equipo (jugadores)>
<!ATTLIST equipo denominacion CDATA #REQUIRED>

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

2.3. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo" categoria="Liga1">
  <jugadores>Yo</jugadores>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar. También se ha añadido al elemento “equipo” (nodo raíz) el atributo “categoría”, que es obligatorio.

▫ Documento DTD:

```
<!ELEMENT equipo (jugadores)>
<!ATTLIST equipo
  denominacion CDATA #REQUIRED
  categoria CDATA #REQUIRED>

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

2.4. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo" categoria="Liga1">
  <jugadores>Yo</jugadores>
  <tecnicos>Tú</tecnicos>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar. Se añade el elemento “técnicos” al elemento “equipo”. Ahora el elemento “equipo” está compuesto de los elementos “jugadores” y “técnicos”. El elemento “tecnicos” tiene que aparecer 1 vez, a continuación del elemento “jugadores”.

▫ Documento DTD:

```
<!ELEMENT equipo (jugadores,tecnicos)>
<!ATTLIST equipo
  denominacion CDATA #REQUIRED>
```

```
categoria CDATA #REQUIRED>
```

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

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

2.5. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo" categoria="Liga1">
  <jugadores>
    <jugador>Él</jugador>
    <jugador>Ella</jugador>
  </jugadores>
  <tecnicos>Tú</tecnicos>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar. El elemento “jugadores” está compuesto del elemento “jugador”, que puede aparecer 1 o varias veces.

▫ Documento DTD:

```
<!ELEMENT equipo (jugadores,tecnicos)>
<!ATTLIST equipo
  denominacion CDATA #REQUIRED
  categoria CDATA #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (#PCDATA)>

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

2.6. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo" categoria="Liga1">
  <jugadores>
    <jugador dorsal="D01" posicion="base">Él</jugador>
    <jugador dorsal="D02">Ella</jugador>
  </jugadores>
  <tecnicos>Tú</tecnicos>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar.

El elemento “jugador” tiene dos atributos:

- El atributo “dorsal” será obligatorio, correspondiendo con el identificador único del jugador en el equipo.
- El atributo “posicion” será opcional y, en caso de ser indicado, adoptará un solo valor de la siguiente relación: base, escolta, alero, alapivot, pivot. (si no se especifica este atributo quedará vacío).

▫ Documento DTD:

```
<!ELEMENT equipo (jugadores,tecnicos)>
<!ATTLIST equipo
    denominacion CDATA #REQUIRED
    categoria CDATA #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (#PCDATA)>
<!ATTLIST jugador
    dorsal ID #REQUIRED
    posicion (base | escolta | alero | alapivot | pivot) #IMPLIED>

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

2.7. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo" categoria="Liga1">
  <jugadores>
    <jugador dorsal="D01" posicion="base">Él</jugador>
    <jugador dorsal="D02">Ella</jugador>
  </jugadores>
  <tecnicos>
    <tecnico codigo="T01" cargo="entrenador">Tú</tecnico>
    <tecnico codigo="T02" cargo="fisio">Tuya</tecnico>
  </tecnicos>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar.

El elemento “tecnicos” está compuesto del elemento “tecnico”, que puede aparecer 1 o varias veces.

El elemento “tecnico” tiene dos atributos:

- El atributo “codigo” será obligatorio, correspondiendo con el identificador único del técnico en el equipo.
- El atributo “cargo” será obligatorio y adoptará un solo valor sin espacios ni símbolos especiales.

▫ Documento DTD:

```
<!ELEMENT equipo (jugadores,tecnicos)>
<!ATTLIST equipo
    denominacion CDATA #REQUIRED
    categoria CDATA #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (#PCDATA)>
<!ATTLIST jugador
    dorsal ID #REQUIRED
    posicion (base | escolta | alero | alapivot | pivot) #IMPLIED>

<!ELEMENT tecnicos (tecnico+)>
<!ELEMENT tecnico (#PCDATA)>
<!ATTLIST tecnico
    codigo ID #REQUIRED
    cargo NMTOKEN #IMPLIED>
```

2.8. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo" categoria="Liga1">
  <jugadores>
    <jugador dorsal="D01" posicion="base">
      <nombre>Jugador1</nombre>
      <apellidos>Primero</apellidos>
      <nacionalidad>Estadounidense</nacionalidad>
    </jugador>
    <jugador dorsal="D02">
      <nombre>Jugador2</nombre>
      <apellidos>Segundo</apellidos>
    </jugador>
  </jugadores>
  <tecnicos>
    <tecnico codigo="T01" cargo="entrenador">
      <nombre>Técnico1</nombre>
      <apellidos>Ape1</apellidos>
      <nacionalidad>Española</nacionalidad>
    </tecnico>
    <tecnico codigo="T02" cargo="fisio">
      <nombre>Técnico2</nombre>
      <apellidos>Ape2</apellidos>
    </tecnico>
  </tecnicos>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar.
Los elementos “jugador” y “tecnico” están compuestos de los elementos:

- “nombre”: tiene que aparecer 1 vez, contenido texto.
- “apellidos”: tiene que aparecer 1 vez, contenido texto.
- “nacionalidad”: puede aparecer 1 o 0 veces, contenido texto.

(Nota: debido a que la estructura para los dos elementos “jugador” y “técnico” es la misma, podemos aprovechar a crear una entidad parámetro interna “datosIntegrantes” que contenga la estructura y hacer referencia a ella evitando la redundancia. La definición de la entidad debe realizarse antes de hacer referencia a ella, en caso contrario no la identificaría).

En el documento DTD:

```
<!ENTITY % datosIntegrantes "nombre,apellidos,nacionalidad?">
```

Y para hacer referencia a ella:

```
<!ELEMENT jugador (%datosIntegrantes;)>
```

▫ Documento DTD:

```
<!ENTITY % datosIntegrantes "nombre,apellidos,nacionalidad?">

<!ELEMENT equipo (jugadores,tecnicos)>
<!ATTLIST equipo
    denominacion CDATA #REQUIRED
    categoria CDATA #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (%datosIntegrantes;)>
<!ATTLIST jugador
    dorsal ID #REQUIRED
    posicion (base | escolta | alero | alapivot | pivot) #IMPLIED>

<!ELEMENT tecnicos (tecnico+)>
<!ELEMENT tecnico (%datosIntegrantes;)>
<!ATTLIST tecnico
    codigo ID #REQUIRED
    cargo NMTOKEN #IMPLIED>

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

2.9. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo" categoria="Liga1">
  <jugadores>
    <jugador dorsal="D01" posicion="base">
      <nombre>Jugador1</nombre>
      <apellidos>Primero</apellidos>
```



```
<nacionalidad>Estadounidense</nacionalidad>
</jugador>
<jugador dorsal="D02">
  <nombre>Jugador2</nombre>
  <apellidos>Segundo</apellidos>
  <edad>21</edad>
</jugador>
</jugadores>
<tecnicos>
  <tecnico codigo="T01" cargo="entrenador">
    <nombre>Técnico1</nombre>
    <apellidos>Ape1</apellidos>
    <nacionalidad>Española</nacionalidad>
  </tecnico>
  <tecnico codigo="T02" cargo="fisio">
    <nombre>Técnico2</nombre>
    <apellidos>Ape2</apellidos>
  </tecnico>
</tecnicos>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar.
Se ha añadido al elemento “jugador”, dentro de su composición, el elemento “edad”, que puede aparecer 1 o 0 veces, contenido texto.

▫ Documento DTD:

```
<!ENTITY % datosIntegrantes "nombre,apellidos,nacionalidad?">

<!ELEMENT equipo (jugadores,tecnicos)>
<!ATTLIST equipo
  denominacion CDATA #REQUIRED
  categoria CDATA #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (%datosIntegrantes;;edad?)>
<!ATTLIST jugador
  dorsal ID #REQUIRED
  posicion (base | escolta | alero | alapivot | pivot) #IMPLIED>

<!ELEMENT tecnicos (tecnico+)>
<!ELEMENT tecnico (%datosIntegrantes;)>
<!ATTLIST tecnico
  codigo ID #REQUIRED
  cargo NMTOKEN #IMPLIED>

<!ELEMENT nombre (#PCDATA)>
<!ELEMENT apellidos (#PCDATA)>
<!ELEMENT nacionalidad (#PCDATA)>
<!ELEMENT edad (#PCDATA)>
```

2.10. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo" categoria="Liga1">
  <jugadores>
    <jugador dorsal="D01" posicion="base">
      <nombre>Jugador1</nombre>
      <apellidos>Primero</apellidos>
      <nacionalidad>Estadounidense</nacionalidad>
      <foto formato="JPG">FotoD01.jpg</foto>
    </jugador>
    <jugador dorsal="D02">
      <nombre>Jugador2</nombre>
      <apellidos>Segundo</apellidos>
      <edad>21</edad>
      <foto formato="GIF">FotoD02.jpg</foto>
    </jugador>
  </jugadores>
  <tecnicos>
    <tecnico codigo="T01" cargo="entrenador">
      <nombre>Técnico1</nombre>
      <apellidos>Ape1</apellidos>
      <nacionalidad>Española</nacionalidad>
    </tecnico>
    <tecnico codigo="T02" cargo="fisio">
      <nombre>Técnico2</nombre>
      <apellidos>Ape2</apellidos>
    </tecnico>
  </tecnicos>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar.

Se le ha añadido al elemento “jugador” el elemento foto, que aparecerá una vez por cada jugador.

El elemento foto contendrá la ruta de la imagen del jugador y el atributo “formato”, que indicará el formato de la imagen, pudiendo ser “gif” o “jpg”. El atributo será obligatorio.

(Nota: para la referenciación al formato de imagen se hace uso de notaciones, para relacionar el código MIME).

En el documento DTD:

```
<!NOTATION GIF SYSTEM "image/gif">
<!NOTATION JPG SYSTEM "image/jpeg">
```

Y para hacer referencia a ella:

```
<!ATTLIST foto
```

formato NOTATION (GIF / JPG) #REQUIRED>

▫ Documento DTD:

```
<!ENTITY % datosIntegrantes "nombre,apellidos,nacionalidad?">
<!NOTATION GIF SYSTEM "image/gif">
<!NOTATION JPG SYSTEM "image/jpeg">

<!ELEMENT equipo (jugadores,tecnicos)>
<!ATTLIST equipo
    denominacion CDATA #REQUIRED
    categoria CDATA #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (%datosIntegrantes;,edad?, foto)>
<!ATTLIST jugador
    dorsal ID #REQUIRED
    posicion (base | escolta | alero | alapivot | pivot) #IMPLIED>

<!ELEMENT tecnicos (tecnico+)>
<!ELEMENT tecnico (%datosIntegrantes;)>
<!ATTLIST tecnico
    codigo ID #REQUIRED
    cargo NMTOKEN #IMPLIED>

<!ELEMENT nombre (#PCDATA)>
<!ELEMENT apellidos (#PCDATA)>
<!ELEMENT nacionalidad (#PCDATA)>
<!ELEMENT edad (#PCDATA)>
<!ELEMENT foto (#PCDATA)>
<!ATTLIST foto
    formato NOTATION (GIF | JPG) #REQUIRED>
```

2.11. Crear otras entidades para facilitar el desarrollo de la estructura DTD

(esto permitirá facilitar el seguimiento del código, la reutilización, la modificación, etc.): uso de entidades parámetro internas

- Entidad que agrupe “datosIntegrantes” con los elementos adicionales que componen al elemento “jugador”, denominándola “datosJugadores”.

```
<!ENTITY % datosJugadores "%datosIntegrantes;,edad?, foto">
```

- Entidad que agrupe las opciones de las posiciones de los jugadores, denominándola “posicionesJugadores”.

```
<!ENTITY % posicionesJugadores "base | escolta | alero | alapivot | pivot">
```

- Entidad que agrupe los elementos que componen el nodo raíz (jugadores, técnicos), denominándola “datosEquipo”.

```
<!ENTITY % datosEquipo "jugadores,tecnicos">
```

▫ Documento DTD:

```
<!ENTITY % datosEquipo "jugadores,tecnicos">

<!ENTITY % datosIntegrantes "nombre,apellidos,nacionalidad?">
<!ENTITY % datosJugadores "%datosIntegrantes;,edad?, foto">

<!ENTITY % posicionesJugadores "base | escolta | alero | alapivot | pivot">

<!NOTATION GIF SYSTEM "image/gif">
<!NOTATION JPG SYSTEM "image/jpeg">

<!ELEMENT equipo (%datosEquipo;)>
<!ATTLIST equipo
    denominacion CDATA #REQUIRED
    categoria CDATA #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (%datosJugadores;)>
<!ATTLIST jugador
    dorsal ID #REQUIRED
    posicion (%posicionesJugadores;) #IMPLIED>

<!ELEMENT tecnicos (tecnico+)>
<!ELEMENT tecnico (%datosIntegrantes;)>
<!ATTLIST tecnico
    codigo ID #REQUIRED
    cargo NMTOKEN #IMPLIED>

<!ELEMENT nombre (#PCDATA)>
<!ELEMENT apellidos (#PCDATA)>
<!ELEMENT nacionalidad (#PCDATA)>
<!ELEMENT edad (#PCDATA)>
<!ELEMENT foto (#PCDATA)>
<!ATTLIST foto
    formato NOTATION (GIF | JPG) #REQUIRED>
```

2.12. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd">
<equipo denominacion="MiEquipo" categoria="Liga1" federacion="&FEB;"
flogo="fedlogo">
  <jugadores>
    <jugador dorsal="D01" posicion="base">
      <nombre>Jugador1</nombre>
      <apellidos>Primero</apellidos>
      <nacionalidad>Estadounidense</nacionalidad>
      <foto formato="JPG">FotoD01.jpg</foto>
    </jugador>
```

```
<jugador dorsal="D02">
  <nombre>Jugador2</nombre>
  <apellidos>Segundo</apellidos>
  <edad>21</edad>
  <foto formato="GIF">FotoD02.jpg</foto>
</jugador>
</jugadores>
<tecnicos>
  <tecnico codigo="T01" cargo="entrenador">
    <nombre>Técnico1</nombre>
    <apellidos>Ape1</apellidos>
    <nacionalidad>Española</nacionalidad>
  </tecnico>
  <tecnico codigo="T02" cargo="fisio">
    <nombre>Técnico2</nombre>
    <apellidos>Ape2</apellidos>
  </tecnico>
</tecnicos>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar.

Se le han añadido al elemento “equipo” (nodo raíz) los siguientes atributos:

- “federacion”: atributo obligatorio, tipo texto. Se reflejará, de forma obligatoria, la denominación de la Federación (por lo tanto, será un valor fijo). Se reflejará mediante una Entidad, para poder modificar la ruta de forma más asequible (todos los documentos XML harán referencia a la misma entidad. Modificando la entidad en la DTD modificará todos los documentos asociados).
- “flogo”: atributo obligatorio, tipo texto. Se reflejará, de forma obligatoria, la ruta al logotipo de la Federación (por lo tanto, será un valor fijo). Se reflejará mediante una Entidad (ver explicación punto anterior).

(Debido a que se usará la llamada a entidades desde el propio documento XML se hará uso de entidades generales (las entidades parámetro solo pueden ser usadas dentro del documento DTD):

- Para “federacion”: se hace uso de entidad general interna (declarada en el propio documento XML).

En el documento XML:

```
<!DOCTYPE equipo SYSTEM "equipo.dtd" [
  <!ENTITY FEB "Federación de Baloncesto">
]>
```

- Para “flogo”: se hace uso de entidad no procesada (entidad general externa)).

En el documento DTD:

```
<!ENTITY fedlogo SYSTEM "LogotipoFederacion.gif" NDATA GIF>
```

- Documento DTD:

```
<!ENTITY % datosEquipo "jugadores,tecnicos">
```

```
<!ENTITY % datosIntegrantes "nombre,apellidos,nacionalidad?">
<!ENTITY % datosJugadores "%datosIntegrantes;,edad?, foto">

<!ENTITY % posicionesJugadores "base | escolta | alero | alapivot | pivot">

<!NOTATION GIF SYSTEM "image/gif">
<!NOTATION JPG SYSTEM "image/jpeg">

<!ENTITY fedlogo SYSTEM "LogotipoFederacion.gif" NDATA GIF>

<!ELEMENT equipo (%datosEquipo;)>
<!ATTLIST equipo
    denominacion CDATA #REQUIRED
    categoria CDATA #REQUIRED
    federacion CDATA #FIXED "Federación de Baloncesto"
    flogo ENTITY #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (%datosJugadores;)>
<!ATTLIST jugador
    dorsal ID #REQUIRED
    posicion (%posicionesJugadores;) #IMPLIED>

<!ELEMENT tecnicos (tecnico+)>
<!ELEMENT tecnico (%datosIntegrantes;)>
<!ATTLIST tecnico
    codigo ID #REQUIRED
    cargo NMTOKEN #IMPLIED>

<!ELEMENT nombre (#PCDATA)>
<!ELEMENT apellidos (#PCDATA)>
<!ELEMENT nacionalidad (#PCDATA)>
<!ELEMENT edad (#PCDATA)>
<!ELEMENT foto (#PCDATA)>
<!ATTLIST foto
    formato NOTATION (GIF | JPG) #REQUIRED>
```

2.13. Dado el siguiente documento XML, bien formado

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE equipo SYSTEM "equipo.dtd" [
  <!ENTITY FEB "Federación de Baloncesto">
]>
<equipo denominacion="MiEquipo" categoria="Liga1" federacion="&FEB;"
flogo="fedlogo">
  <jugadores>
    <jugador dorsal="D01" posicion="base">
      <nombre>Jugador1</nombre>
      <apellidos>Primero</apellidos>
```

```
<nacionalidad>Estadounidense</nacionalidad>
<foto formato="JPG">FotoD01.jpg</foto>
</jugador>
<jugador dorsal="D02">
  <nombre>Jugador2</nombre>
  <apellidos>Segundo</apellidos>
  <edad>21</edad>
  <foto formato="GIF">FotoD02.jpg</foto>
</jugador>
</jugadores>
<tecnicos>
  <tecnico codigo="T01" cargo="entrenador">
    <nombre>Técnico1</nombre>
    <apellidos>Ape1</apellidos>
    <nacionalidad>Española</nacionalidad>
  </tecnico>
  <tecnico codigo="T02" cargo="fisio">
    <nombre>Técnico2</nombre>
    <apellidos>Ape2</apellidos>
  </tecnico>
</tecnicos>
<quinteto integra="D01 D02"/>
</equipo>
```

Construir su correspondiente DTD, en documento externo privado, y validar.
Se le han añadido al elemento “equipo” (nodo raíz) el elemento “quinteto”, que presenta la siguiente composición:

- Es un elemento vacío, no contendrá información ni elementos hijo.

```
<!ELEMENT quinteto EMPTY>
```
- Atributo “integra”: atributo obligatorio, que hará referencia a los dorsales de los jugadores que integrarán el quinteto inicial en el partido.

```
<!ATTLIST quinteto
  integra IDREFS #REQUIRED>
```

(Nota: dado que hemos creado una entidad para indicar la composición de los elementos principales, añadimos a % datosEquipo el nuevo elemento)

```
<!ENTITY % datosEquipo "jugadores,tecnicos,quinteto">
```

(Nota: Gracias al uso de la entidades, añadir o eliminar componentes de los distintos elementos se hace de forma más sencilla)

▫ Documento DTD:

```
<!ENTITY % datosEquipo "jugadores,tecnicos,quinteto">

<!ENTITY % datosIntegrantes "nombre,apellidos,nacionalidad?">
<!ENTITY % datosJugadores "%datosIntegrantes;,edad?, foto">

<!ENTITY % posicionesJugadores "base | escolta | alero | alapivot | pivot">

<!NOTATION GIF SYSTEM "image/gif">
<!NOTATION JPG SYSTEM "image/jpeg">
```

```
<!ENTITY fedlogo SYSTEM "LogotipoFederacion.gif" NDATA GIF>

<!ELEMENT equipo (%datosEquipo;)>
<!ATTLIST equipo
    denominacion CDATA #REQUIRED
    categoria CDATA #REQUIRED
    federacion CDATA #FIXED "Federación de Baloncesto"
    flogo ENTITY #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (%datosJugadores;)>
<!ATTLIST jugador
    dorsal ID #REQUIRED
    posicion (%posicionesJugadores;) #IMPLIED>

<!ELEMENT tecnicos (tecnico+)>
<!ELEMENT tecnico (%datosIntegrantes;)>
<!ATTLIST tecnico
    codigo ID #REQUIRED
    cargo NMTOKEN #IMPLIED>

<!ELEMENT nombre (#PCDATA)>
<!ELEMENT apellidos (#PCDATA)>
<!ELEMENT nacionalidad (#PCDATA)>
<!ELEMENT edad (#PCDATA)>
<!ELEMENT foto (#PCDATA)>
<!ATTLIST foto
    formato NOTATION (GIF | JPG) #REQUIRED>

<!ELEMENT quinteto EMPTY>
<!ATTLIST quinteto
    integra IDREFS #REQUIRED>
```

2.14. Incorporar al documento DTD secciones condicionales.

Modificar el documento DTD para incorporar la posibilidad que los documentos XML sean utilizados para enviar a la Federación la composición de la plantilla al inicio de la temporada (nodo raíz compuesto de jugadores y técnicos) y para enviar a la Federación el equipo en cada partido (nodo raíz compuesto de jugadores, técnicos y quinteto).

(Nota: Gracias a que la composición de los elementos se ha efectuado haciendo uso de entidades, incorporar secciones condicionales se realizará de forma más sencilla.)

Renombramos la entidad “% datosEquipo” a “% datosEquipoPartido”, estableciendo que la composición será “jugadores,tecnicos,quinteto”.

```
<!ENTITY % datosEquipoPartido "jugadores,tecnicos,quinteto">
```

Incorporamos la entidad “% datosEquipoTemporada”, estableciendo que la composición será “jugadores,tecnicos”.


```
<!ENTITY % datosEquipoTemporada "jugadores,tecnicos">
```

Modificamos la declaración del elemento “equipo” añadiendo las secciones condicionales, a las que denominaremos “partido” (la composición del elemento “equipo” hará referencia a “datosEquipoPartido”) y “temporada” (la composición del elemento “equipo” hará referencia a “datosEquipoTemporada”):

```
<![ %partido; [  
  <!ELEMENT equipo (%datosEquipoPartido;)>  
]]>  
  
<![ %temporada; [  
  <!ELEMENT equipo (%datosEquipoTemporada;)>  
]]>
```

En la cabecera del documento DTD haremos referencia a la sección incluida (“partido”) y a la sección ignorada (“temporada”) para este fichero (en este fichero se comunican los datos del “quinteto”, por lo que se refiere a partido. Si suprimimos dicha comunicación nos referiríamos a temporada, por lo que modificaríamos la declaración de incluido e ignorado).

```
<!DOCTYPE equipo SYSTEM "equipo.dtd" [  
  <!ENTITY FEB "Federación de Baloncesto">  
  <!ENTITY % partido "INCLUDE">  
  <!ENTITY % temporada "IGNORE">  
>
```

▫ Documento XML:

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE equipo SYSTEM "equipo.dtd" [  
  <!ENTITY FEB "Federación de Baloncesto">  
  <!ENTITY % partido "INCLUDE">  
  <!ENTITY % temporada "IGNORE">  
>  
<equipo denominacion="MiEquipo" categoria="Liga1" federacion="&FEB;"  
flogo="fedlogo">  
  <jugadores>  
    <jugador dorsal="D01" posicion="base">  
      <nombre>Jugador1</nombre>  
      <apellidos>Primero</apellidos>  
      <nacionalidad>Estadounidense</nacionalidad>  
      <foto formato="JPG">FotoD01.jpg</foto>  
    </jugador>  
    <jugador dorsal="D02">  
      <nombre>Jugador2</nombre>  
      <apellidos>Segundo</apellidos>  
      <edad>21</edad>  
      <foto formato="GIF">FotoD02.jpg</foto>  
    </jugador>  
  </jugadores>
```

```
<tecnicos>
  <tecnico codigo="T01" cargo="entrenador">
    <nombre>Técnico1</nombre>
    <apellidos>Ape1</apellidos>
    <nacionalidad>Española</nacionalidad>
  </tecnico>
  <tecnico codigo="T02" cargo="fisio">
    <nombre>Técnico2</nombre>
    <apellidos>Ape2</apellidos>
  </tecnico>
</tecnicos>
<quinteto integra="D01 D02"/>
</equipo>
```

▫ Documento DTD:

```
<!ENTITY % datosEquipoTemporada "jugadores,tecnicos">
<!ENTITY % datosEquipoPartido "jugadores,tecnicos,quinteto">

<!ENTITY % datosIntegrantes "nombre,apellidos,nacionalidad?">
<!ENTITY % datosJugadores "%datosIntegrantes;,edad?, foto">

<!ENTITY % posicionesJugadores "base | escolta | alero | alapivot | pivot">

<!NOTATION GIF SYSTEM "image/gif">
<!NOTATION JPG SYSTEM "image/jpeg">

<!ENTITY fedlogo SYSTEM "LogotipoFederacion.gif" NDATA GIF>

<![ %partido; [
  <!ELEMENT equipo (%datosEquipoPartido;)>
]]>

<![ %temporada; [
  <!ELEMENT equipo (%datosEquipoTemporada;)>
]]>

<!ATTLIST equipo
  denominacion CDATA #REQUIRED
  categoria CDATA #REQUIRED
  federacion CDATA #FIXED "Federación de Baloncesto"
  flogo ENTITY #REQUIRED>

<!ELEMENT jugadores (jugador+)>
<!ELEMENT jugador (%datosJugadores;)>
<!ATTLIST jugador
  dorsal ID #REQUIRED
  posicion (%posicionesJugadores;) #IMPLIED>

<!ELEMENT tecnicos (tecnico+)>
```

```
<!ELEMENT tecnico (%datosIntegrantes;)>
<!ATTLIST tecnico
    codigo ID #REQUIRED
    cargo NMTOKEN #IMPLIED>

<!ELEMENT nombre (#PCDATA)>
<!ELEMENT apellidos (#PCDATA)>
<!ELEMENT nacionalidad (#PCDATA)>
<!ELEMENT edad (#PCDATA)>
<!ELEMENT foto (#PCDATA)>
<!ATTLIST foto
    formato NOTATION (GIF | JPG) #REQUIRED>

<!ELEMENT quinteto EMPTY>
<!ATTLIST quinteto
    integra IDREFS #REQUIRED>
```

3. Ejercicio "PERSONA"

Crea un DTD que permita validar el siguiente documento XML y añade la correspondiente declaración del DTD:

```
<?xml version="1.0" encoding="UTF-8"?>
<persona dni="12345678-L" estadoCivil="casado">
  <nombre>María Pilar</nombre>
  <apellido>Sánchez</apellido>
  <edad>25</edad>
  <enActivo/>
</persona>
```

Restricciones:

- Una persona puede estar en activo o no estarlo.
- El DNI es un identificador obligatorio.
- El estado civil puede ser soltero, casado o divorciado, y por defecto es soltero.

Valida el documento XML del ejercicio anterior utilizando "XML Copy Editor".

¿Se produce algún error? ¿A qué se debe? ¿Cómo podrías corregirlo?

4. Ejercicio "TICKETS"

Completa el DTD del siguiente documento XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE listatickets [
    ...
]>
<listatickets>
  <ticket>
    <numero>0001</numero>
    <fecha>01/02/2016</fecha>
    <hora>22:00</hora>
    <producto>
      <nombre>Agua</nombre>
      <precio moneda="euro">1.00</precio>
    </producto>
    <producto>
      <nombre>Refresco</nombre>
      <precio moneda="euro">1.50</precio>
    </producto>
    <total moneda="euro">2.50</total>
  </ticket>
  <ticket>
    <numero>0002</numero>
    <fecha>01/02/2016</fecha>
    <hora>22:05</hora>
    <producto>
      <nombre>Refresco</nombre>
      <precio moneda="euro">1.50</precio>
    </producto>
    <producto>
      <nombre>Refresco</nombre>
      <precio moneda="euro">1.50</precio>
    </producto>
    <total moneda="euro">3.00</total>
  </ticket>
</listatickets>
```

5. Ejercicio “ALMACEN”

El siguiente documento XML incluye las declaraciones del DTD. Valida el documento y corrige los errores detectados en el XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE almacen [
  <!ELEMENT almacen (producto+)>
  <!ELEMENT producto (referencia, seccion, pasillo, descripcion)>
  <!ELEMENT referencia (#PCDATA)>
  <!ELEMENT seccion (#PCDATA)>
  <!ELEMENT pasillo (#PCDATA)>
  <!ELEMENT descripcion (#PCDATA)>
]>
<almacen>
  <producto>
    <referencia>A123</referencia>
    <seccion>20</seccion>
    <pasillo>5</pasillo>
    <descripcion>Silla LEON</descripcion>
  </producto>
  <producto>
    <referencia>A124</referencia>
    <seccion>09</seccion>
    <seccion>2</seccion>
    <descripcion>Mesa IBIZA</descripcion>
  </producto>
</almacen>
```

6. Ejercicio "LIBRO"

El siguiente documento XML incluye las declaraciones del DTD. Corrige el DTD para que el documento sea válido.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE libro [
  <!ELEMENT libro (capitulo)>
  <!ELEMENT capitulo (titulo, paginas, autor)>
  <!ELEMENT titulo (#PCDATA)>
  <!ELEMENT autor (#PCDATA)>
]>
<libro>
  <capitulo>
    <titulo>HTML</titulo>
    <paginas>44</paginas>
    <autor>Oscar</autor>
  </capitulo>
  <capitulo>
    <titulo>CSS</titulo>
    <paginas>32</paginas>
    <autor>Oscar</autor>
    <autor>Ruben</autor>
  </capitulo>
  <capitulo>
    <titulo>XML</titulo>
    <paginas>36</paginas>
    <autor>Oscar</autor>
  </capitulo>
</libro>
```