

## **EXERCICIS XSD 2ª PART**

### **de Ruth Vacas**

Ajudeu-se dels exemples i de les diapositives per crear els esquemes i els diferents tipus de dades.

#### **Exercici 1:** CREA ELS SEGÜENTS ESQUEMES PER A VALIDAR ELS ELEMENTS CORRESPONENTS.

A) PRODUCTE AMB ATRIBUTS. Es desitja crear un esquema que permeti validar un element anomenat producte de tipus xsd:string. El producte té dos atributs:

- Un atribut es dirà quantitat i és obligatori. Ha d'acceptar sol enters positius.
- També hi haurà un atribut anomenat unitat que només accepta els xsd:string «caixes» i «palets».

```
<fichas xmlns:xsi="http://www.w3.org/2001/XMLSchema">
<xsd:element name="producto" type="tipoProducto">
<xsd:complexType name="tipoProducto">
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="cantidad" type="xsd:unsignedInt" use="">
<xsd:attribute name="unidad" type="tipoUnidad">
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
<xsd:complexType>
<xsd:simpleType name="tipoUnidad">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="caja"/>
<xsd:enumeration value="pale"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:complexType>
```

B) CLIENTS. Es desitja crear un esquema XML que permeti validar un element anomenat client que pot emmagatzemar un xsd:string. El client conté:

- Un atribut obligatori anomenat codi que conté el codi del client, que sempre consta de sis números.
- Un atribut optatiu anomenat habitual que s'usarà per a saber si és un client habitual o no. Accepta valors «true» i «false».
- Un atribut optatiu anomenat quantitat que indica la seua compra. És un decimal positiu amb valors d'entre 0 i 1000.

Exemple: <client codi="123456" habitual="true" quantitat="968.56"> Pepe Pérez </client>

```
<fichas xmlns:xsi="http://www.w3.org/2001/XMLSchema">
<xsd:element name="producto" type="tipoClient">
<xsd:complexType name="tipoClient">
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="codi" type="xsd:unsignedInt" use="6">
<xsd:attribute name="habitual" type="xsd:unsignedBoolean">
<xsd:attribute name="quantitat" type="xsd:unsignedDouble">
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
```

## **Exercici 2:** A PARTIR DELS DOCUMENTS XML, CREA EL XSD.

Nota: No oblidés validar els documents XML i el XSD i la relació entre els dos.

A) Utilitza xs:extension i complexContent (serveix per definir restriccions o extensions a un tipus complex) per crear tipus complejos personalitzats en cas de que ho necessites.

```
<fichas xmlns:xsi="http://www.w3.org/2001/XMLSchema">
<xsd:element name="fichas.xsd" type="tipoFicha">
<xsd:complexType name="tipoFicha">
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="edad" type "xsd:unsignedInt" >
<xsd:attribute name="nombre" type "xsd:unsignedString">
<xsd:attribute name="ciudad" type "xsd:unsignedString">
<xsd:attribute name="pais" type "xsd:unsignedString">

</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<fichas xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="fichas.xsd">
  <ficha numero="1">
    <nombre>Eva</nombre>
    <edad>25</edad>
    <ciudad>París</ciudad>
    <pais>Francia</pais>
  </ficha>
  <ficha numero="2">
    <nombre>Giovanni</nombre>
    <edad>26</edad>
    <ciudad>Firenze</ciudad>
    <pais>Italia</pais>
  </ficha>
</fichas>
```

B) Escriure un XML Schema per al següent document XML, i incloure els canvis necessaris en el mateix per a referenciar a l'esquema creat. S'ha de complir també el següent:

- Tots els elements que apareixen són obligatoris i han d'aparèixer sempre en el mateix ordre.
- Els elements “ordinador” i “modificacio” han d'aparèixer mínim una vegada i el màxim no està limitat. La resta dels elements han d'aparèixer una vegada.
- Els elements que contenen informació de data són de tipus “date”.
- L'element “tipus” només podrà ser “Sobretaula” o “Portàtil”.

```
<?xml version="1.0" encoding="UTF-8"?>
<ordinadors>
<ordinador>
  <nom>PC Gaming</nom>
  <tipus>Sobretaula</tipus>
  <fabricacio>
```

```

        <inici>01/01/2022</inici>
        <fi>15/02/2022</fi>
    </fabricacio>
    <modificacions>
        <modificacio>Ampliació memòria RAM</modificacio>
        <modificacio>Canvi placa base</modificacio>
    </modificacions>
</ordinador>
</ordinadors>

```

C) Escriure un XML Schema per al següent document XML, i incloure els canvis necessaris en el mateix per a referenciar a l'esquema creat. S'ha de complir també el següent:

- Tots els elements que apareixen són obligatoris i han d'aparèixer sempre en el mateix ordre.
- S'han de definir dos tipus de dades, “tipoPersona” (els elements “persona” són d'aquest tipus) i “info” (els elements “dades” són d'aquest tipus).
- L'atribut “naixement” és de tipus “date”.
- L'element “comentari” no és obligatori; si apareix ho fa només 1 vegada.

```

<?xml version="1.0" encoding="UTF-8"?>
<persona naixement="10-12-1815">
    <dades>
        <nom>Ada</nom>
        <cognoms>Lovelace</cognoms>
        <dni>12345678</dni>
    </dades>
    <comentari>Primer any en l'IES</comentari>
</persona>

```

D) Escriure un XML Schema per al següent document XML, i incloure els canvis necessaris en el mateix per a referenciar a l'esquema creat. S'ha de complir també el següent:

- Tots els elements que apareixen són obligatoris i han d'aparèixer sempre en el mateix ordre.
- L'element “domicili” ha d'aparèixer mínim una vegada i màxim dues vegades.
- S'han de definir tres tipus de dades, “tMatricula” (l'element “matricula” és d'aquest tipus), “tPersonal” (l'element “personal” és d'aquest tipus) i “tPagament” (l'element “pagament” és d'aquest tipus).
- L'element “dni” tindrà una longitud de 9 caràcters.

```

<?xml version="1.0" encoding="UTF-8"?>
<matricula>
    <personal>
        <dni>12345678A</dni>
        <nom>Ada Lovelace</nom>
        <titulacio>Batxillerat</titulacio>
        <curs_academic>2021/2022</curs_academic>
        <domicilis>

```

```

        <domicili tipus="familiar">
        <nom>C/ Principal nº1</nom>
        </domicili>
    </domicilis>
</personal>
<pagament>
    <tipus_matricula>Matrícula Ordinària</tipus_matricula>
</pagament>
</matricula>

```

**Exercici 3:** Convertir la DTD que es mostra a continuació en un XML Schema, tenint en compte el següent:

- S'han de definir dos tipus de dades simples per al mes (“tMes”) i l'any (“tAny”), com a restriccions del tipus “positiveInteger”. El mes ha de ser un valor comprès entre 1 i 12, i l'any menor o igual a 2023.

- També es defineix el tipus “tNom”, que serà l'utilitzat per als elements “titol”, “autor” i “editor”. Es tracta d'una restricció del tipus “string” en la qual els valors tenen entre 0 i 100 caràcters.

- Finalment, es defineix el tipus “tISBN” per a l'element “ISBN”, com un tipus string que té entre 5 i 8 caràcters.

```

<!ELEMENT biblioteca (llibre)*>
<!ELEMENT llibre (titol, autor+, data, ISBN, editor)>
<!ATTLIST llibre
categoria (autobiografia | noficcio | ficcio) #REQUIRED
enstock (true | false) "false"
traductor CDATA "">
<!ELEMENT titol (#PCDATA)>
<!ELEMENT autor (#PCDATA)>
<!ELEMENT data (mes?, any)>
<!ELEMENT ISBN (#PCDATA)>
<!ELEMENT editor (#PCDATA)>
<!ELEMENT mes (#PCDATA)>
<!ELEMENT any (#PCDATA)>

```

- Escriure un document XML que complisca l'esquema amb les següents dades:

TITOL	AUTOR/S	DATA	ISBN	EDITOR	CATEG.	STOCK	TRADUC.
Alfagann es Flanagan	Jaume Ribera Andreu Marti	Octubre 2015	9926223	Debate	NoFiccio	SI	Joan Ros
Sapiens	Yuval Harari	2009	66784726	Anaya	Ficcio	NO	

Nota: No oblidis validar els documents XML i el XSD i la relació entre els dos.

- Prova també a inserir un llibre que no complisca el XSD per diferents motius.