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

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

Esta es la cabecera de cualquier elemento XSD.

<xs:element name="biblioteca">

Aquí se establece el nombre del documento XML, biblioteca.

<xs:complexType>

<xs:sequence>

<xs:element ref="libro" minOccurs="2" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

</xs:element>

En este fragmento, vemos que el elemento biblioteca es de tipo complejo. Incluye una secuencia con un elemento llamado libro, que tiene como mínimo dos ocurrencias. Es decir, es una biblioteca con al menos dos libros.

En el siguiente fragmento se describe el elemento libro. El elemento libro también es un elemento completo, y contiene 6 elementos y 2 atributos:

<xs:element name="libro">

<xs:complexType>

<xs:sequence>

<xs:element name="autor" type="xs:string" maxOccurs="3"/>

Elemento autor, de tipo string y con un máximo de tres ocurrencias. Es decir, que cada libro tiene máximo tres autores.

<xs:element name="editorial" type="xs:string" minOccurs="0" maxOccurs="1"/>

El elemento editorial, de tipo string, que tiene mínimo una ocurrencia y máximo una. Esto lo podemos interpretar como un elemento opcional, ya que puede aparecer una vez, o ninguna.

<xs:element name="edicion" type="xs:string"/>

El elemento edición, de tipo cadena, sin más restricciones.

<xs:element name="isbn" type="xs:string"/>

El elemento isbn, de tipo cadena, sin más restricciones.

<xs:element name="donacion" type="xs:string" minOccurs="0" maxOccurs="1"/>

El elemento donación, de tipo cadena, con una ocurrencia de máximo y cero de mínimo. Al igual que el elemento editorial, podemos considerar que es un elemento opcional.

<xs:element name="ubicacion" type="tipoUbicacion"/>

El elemento ubicación, de tipo tipoUbicacion, que se describe más adelante.

Y sus dos atributos:

</xs:sequence>

<xs:attribute name="codigo" type="xs:string" use="required"/>

El atributo codigo, de tipo string y obligatorio.

<xs:attribute name="titulo" type="xs:string" use="required"/>

El atributo titulo, de tipo string y obligatorio.

</xs:complexType>

</xs:element>

Se cierra el elemento libro y se pasa a describir el tipo “tipoUbicacion”, el cual es un tipo complejo.

<xs:complexType name="tipoUbicacion">

<xs:attribute name="sede" use="required">

Con esta descripción del tipoUbicacion, se nos indica que el elemento “ubicación” tiene un atributo llamado sede, que es obligatorio. Además, ese atributo tiene una serie de restricciones. A continuación, se abre una etiqueta de “sympleType”, ya que ello es obligatorio para establecer restricciones:

<xs:simpleType>

<xs:restriction base="xs:string">

<xs:enumeration value="norte"/>

<xs:enumeration value="sur"/>

</xs:restriction>

</xs:simpleType>

</xs:attribute>

</xs:complexType>

</xs:schema>

Una vez se cierran simpleType, attribute y complexType, queda establecido que el elemento ubicación tiene un atributo llamado sede, la cual es de tipo string y solo puede tener dos valores: norte o sur. Al tratarse de un tipo y de ser un elemento con atributo con restricciones, la ubicación es un elemento vacío ya que no se declara que sea MIXED.