

5.3 Ejercicio EMPLEADO

Diseña un documento ***XML Schema*** (crea un nuevo archivo con ***extensión xsd***) que permita validar el siguiente documento, teniendo en cuenta las siguientes restricciones:

- El atributo código debe contener tres letras mayúsculas seguidas de dos números. Recuerda que los atributos son de tipo `simpleType`.
- El elemento extensión será un valor numérico con un total de 3 dígitos.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <empleado>
3   <nombre código="AXL43">Sergio</nombre>
4   <puesto>Atención al cliente</puesto>
5   <extensión>345</extensión>
6 </empleado>
```

EJEMPLOS DE SOLUCIONES

En ambas soluciones, observa:

Tipos de elementos

- El elemento *simpleType* define un tipo simple y especifica las restricciones y la información sobre los valores de los atributos o elementos de solo texto.
- El elemento *complexType* define un tipo complejo. Un elemento de tipo complejo es un elemento XML que contiene otros elementos y/o atributos. A su vez, dentro de un tipo *complexType*, distinguimos:
 - Contenido *simpleContent* contiene extensiones o restricciones en un único texto-tipo complejo.
 - Contenido *complexContent* define extensiones o restricciones en un tipo complejo que contiene solo elementos o contenido mixto.

SOLUCIÓN 1

En esta solución, observa que dentro de la definición de cada elemento o atributo se incluye la información del tipo (simple o complejo).

```

2  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
3  <xs:element name="empleado">
4  <xs:complexType>
5  <xs:sequence>
6  <xs:element name="nombre" minOccurs="1" maxOccurs="1">
7  <xs:complexType>
8  <xs:simpleContent>
9  <xs:extension base="xs:string">
10 <xs:attribute name="código">
11 <xs:simpleType>
12 <xs:restriction base="xs:string">
13 | <xs:pattern value="[A-Z][A-Z][A-Z][0-9][0-9]" />
14 | </xs:restriction>
15 | </xs:simpleType>
16 | </xs:attribute>
17 | </xs:extension>
18 | </xs:simpleContent>
19 | </xs:complexType>
20 | </xs:element>
21 <xs:element name="puesto" type="xs:string" minOccurs="1" maxOccurs="1" />
22 <xs:element name="extensión" minOccurs="1" maxOccurs="1">
23 <xs:simpleType>
24 <xs:restriction base="xs:integer">
25 | <xs:totalDigits value="3" />
26 | </xs:restriction>
27 | </xs:simpleType>
28 | </xs:element>
29 | </xs:sequence>
30 | </xs:complexType>
31 | </xs:element>
32 </xs:schema>

```

SOLUCIÓN 2

En esta solución, observa que está separada la definición de elementos y atributos de los tipos a los que pertenecen.

```
1 <?xml:version="1.0" encoding="UTF-8"?>
2 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
3 |   <xs:element name="empleado" type="tipoEmpleado"/>
4 |   <xs:complexType name="tipoEmpleado">
5 |     <xs:sequence>
6 |       <xs:element name="nombre" type="tipoNombre" minOccurs="1" maxOccurs="1"/>
7 |       <xs:element name="puesto" type="xs:string" minOccurs="1" maxOccurs="1"/>
8 |       <xs:element name="extensión" type="tipoExtension" minOccurs="1" maxOccurs="1"/>
9 |     </xs:sequence>
10 |   </xs:complexType>
11 |
12 |   <xs:complexType name="tipoNombre">
13 |     <xs:simpleContent>
14 |       <xs:extension base="xs:string">
15 |         <xs:attribute name="código" type="tipoCodigo"/>
16 |       </xs:extension>
17 |     </xs:simpleContent>
18 |   </xs:complexType>
19 |
20 |   <xs:simpleType name="tipoCodigo">
21 |     <xs:restriction base="xs:string">
22 |       <xs:pattern value="[A-Z][A-Z][A-Z][0-9][0-9]"/>
23 |     </xs:restriction>
24 |   </xs:simpleType>
25 |
26 |   <xs:simpleType name="tipoExtension">
27 |     <xs:restriction base="xs:integer">
28 |       <xs:totalDigits value="3"/>
29 |     </xs:restriction>
30 |   </xs:simpleType>
31 |
32 </xs:schema>
```