

RELACIÓN 2 - SCHEMAS

1. Define un elemento llamado "edad" con una restricción. El valor de la edad no puede ser menor que 0 o mayor que 120.

```
<xs:element name="edad">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>

      <xs:maxInclusive value="120"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

2. Define un elemento llamado "coche" con una restricción. Los únicos valores aceptables son: Audi, Golf y BMW.

```
<xs:element name="coche">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Audi"/>

      <xs:enumeration value="Golf"/>

      <xs:enumeration value="BMW"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

3. Define un elemento llamado "letra" restricción. El único valor aceptable es de tres de las letras mayúsculas de la A a la Z.

```
<xs:element name="letra">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[A-Z]{3}"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

4. Define un elemento llamado "prodid" con una restricción. El único valor aceptable es de cinco dígitos en una secuencia, y cada dígito debe estar en un rango de 0 a 9.

```
<xs:element name="prodid">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:pattern value="[0-9]{5}"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

5. Define un elemento llamado "letra" con una restricción. El único valor aceptable es cero o más ocurrencias de letras minúsculas de la a a la z.

```
<xs:element name="letra">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[a-z]*"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

6. Define un elemento llamado "contraseña" con una restricción. Debe haber exactamente ocho caracteres en una fila y los caracteres deben estar en minúsculas o en mayúsculas (desde la a/A a la z/Z) o ser números del 0 al 9.

```
<xs:element name="contraseña">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[a-zA-Z0-9]{8}"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

7. Define un elemento llamado "dirección" con una restricción. La restricción de espacio en blanco en ajustado a "preservar". No eliminará los caracteres de espacio en blanco.

```
<xs:element name="dirección">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:whiteSpace value="preserve"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

8. Define otro elemento llamado "contraseña" con una restricción. El valor debe ser como mínimo de cinco caracteres y un máximo de ocho caracteres.

```
<xs:element name="contraseña">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:minLength value="5"/>

      <xs:maxLength value="8"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

9. Rellena los huecos que faltan en la siguiente parte del esquema:

```
<xs:element name="bola">
  <xs:complexType>
    <xs:attribute name="numero" type="____"/>
  </xs:complexType>
</xs:element>

<xs:simpleType name="numeroDeBola">
  <xs:____ base="xs:positiveInteger">
    <xs:____ value="1"/>

    <xs:____ value="90"/>
  </xs:restriction>
</xs:____>
```

Respuesta:

```
<xs:element name="bola">
  <xs:complexType>
    <xs:attribute name="numero" type="numeroDeBola"/>
  </xs:complexType>
</xs:element>

<xs:simpleType name="numeroDeBola">
  <xs:restriction base="xs:positiveInteger">
    <xs:minInclusive value="1"/>

    <xs:maxInclusive value="90"/>
  </xs:restriction>
</xs:simpleType>
```

10. Dado el siguiente esquema, responde a las siguientes preguntas:

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="agenda">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="contacto" maxoccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="nombre" type="xs:string"/>

              <xs:element name="telefono" type="xs:string" minOccurs="0"
maxoccurs="2"/>

              <xs:element name="telefono" type="unsignedInt"/>

              <xs:element name="edad" type="xs:unsignedByte"/>

              <xs:element name="email" type="xs:string" minOccurs="1"
maxoccurs="3"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

- a. ¿Podrías añadir 7 contactos más a la agenda?
- b. Si un usuario deja el campo de teléfono sin rellenar, estaría bien?
- c. Si el usuario Andrés dispone de dos teléfonos móviles y uno fijo, ¿podrían añadirse todos en XML?
- d. ¿Cuántos correos electrónicos, como máximo y mínimo, hay que introducir?
- e. ¿De qué tipo es el campo edad? ¿Qué significa? ¿Qué valores podrá tomar según el tipo?