

EJERCICIOS PARA CREAR XSD QUE VALIDE XMLS

PÉREZ SELLERS, ALEJANDRO
LENGUAJE DE MARCAS 1º de Daw

Ejercicio: peso de productos

```
1  <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
2    <xsd:element name="peso" type="tipoPeso"/>
3    <xsd:simpleType name="tipoPeso">
4      <xsd:restriction base="xsd:decimal">
5        <xsd:minInclusive value="0"/>
6        <xsd:maxInclusive value="1000"/>
7      </xsd:restriction>
8    </xsd:simpleType>
9  </xsd:schema>
10 |
```

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼ <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="peso" type="tipoPeso"/>
  ▼ <xsd:simpleType name="tipoPeso">
    ▼ <xsd:restriction base="xsd:decimal">
      <xsd:minInclusive value="0"/>
      <xsd:maxInclusive value="1000"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

<xsd:element name="peso" type="tipoPeso"> mediante este elemento llamado peso se validaran ciertas reglas.

<xsd:simpleType name="tipopeso"> indica que el elemento peso es simple, no tiene ni subelementos ni atributos.

<xsd:restriction base="xsd:decimal"> indica que peso debe ser numero decimal.

<xsd:minInclusive value="0"/> el valor mínimo es 0.

<xsd:maxInclusive value="1000"/> el valor máximo es 1000.

Ejercicio: pagos de productos.

```
C: > Users > alexp > Desktop > practica xsd validacion > pago.xml > ...
1  <?xml version="1.0" encoding="UTF-8"?>
2  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
3      <xs:element name="pago">
4          <xs:simpleType>
5              <xs:restriction base="xs:integer">
6                  <xs:minInclusive value="0"/>
7                  <xs:maxInclusive value="3000"/>
8              </xs:restriction>
9          </xs:simpleType>
10     </xs:element>
11 </xs:schema>
12
```

```
▼ <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  ▼ <xs:element name="pago">
    ▼ <xs:simpleType>
      ▼ <xs:restriction base="xs:integer">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="3000"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
</xs:schema>
```

<xs:element name="pago"> Se declara el elemento pago.

<xs:simpleType> Se indica un elemento simple dentro de pago

<xs:restriction base="xs:integer"> se indica que pago es un número entero mediante xs:integer y con xs:restriction se indica que no se permiten valores decimales.

Ejercicio: Dni

```
1 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
2   <xs:element name="dni">
3     <xs:simpleType>
4       <xs:restriction base="xs:string">
5         <xs:pattern value="[0-9]{7,8}[A-Za-z]"/>
6       </xs:restriction>
7     </xs:simpleType>
8   </xs:element>
9 </xs:schema>
10
```

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="dni">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="[0-9]{7,8}[A-Za-z]"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
</xs:schema>
```

`<xs:element name="dni">` Se declara un elemento llamado DNI

`<xs:simpleType>` Se indica un elemento simple dentro de DNI

`<xs:restriction base="xs:string">` Mediante la restricción se dicta que es un elemento de cadena de texto

`<xs:pattern value="[0-9]{7,8}[A-Za-z]"/>` Se indica que tiene que tener entre 7 u 8 números y con un rango de 0 hasta 9, también con un número de la A a la Z.