

SCHEMA

- ELEMENTOS SIMPLES:

Los elementos simples solamente pueden contener texto (caracteres). Dicho de otro modo, los elementos simples no pueden contener a otro u otros elementos (hijos), ni tampoco pueden tener atributos.

```
<xs:element name="nombre_del_elemento" type="tipo_de_dato"/>
```

```
<xs:element name="nombre" type="xs:string"/>
```

```
<xs:element name="edad" type="xs:integer"/>
```

Si se quiere indicar que un valor es fijo (*fixed*), se puede escribir, por ejemplo:

```
<xs:element name="mes" type="xs:string" fixed="agosto"/>
```

También, se puede especificar un valor por defecto (*default*), por ejemplo, tecleando:

```
<xs:element name="mes" type="xs:string" default="agosto"/>
```

- ATRIBUTOS:

Para definir un atributo se puede emplear la siguiente sintaxis:

```
<xs:attribute name="nombre_del_atributo" type="tipo_de_dato"/>
```

```
<xs:element name="curso" type="xs:integer"/>
```

```
<xs:attribute name="grupo" type="xs:string"/>
```

Para indicar que el valor de un atributo es fijo (*fixed*), es posible escribir, por ejemplo:

```
<xs:attribute name="grupo" type="xs:string" fixed="B"/>
```

Para especificar el valor por defecto (*default*) de un atributo, se puede escribir:

```
<xs:attribute name="grupo" type="xs:string" default="B"/>
```

Para indicar que un atributo es obligatorio (*required*) escribirlo, se puede teclear:

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

Por defecto, si no se indica nada, el atributo será opcional (*optional*).

- RESTRICCIONES:

Las facetas permiten definir restricciones sobre los posibles valores de atributos o elementos. Las facetas que pueden utilizarse son:

xs:length	Especifica una longitud fija.
xs:minLength	Especifica una longitud mínima.
xs:maxLength	Especifica una longitud máxima.
xs:pattern	Especifica un patrón de caracteres admitidos.
xs:enumeration	Especifica una lista de valores admitidos.
xs:whiteSpace	Especifica cómo se debe tratar a los posibles espacios en blanco, las tabulaciones, los saltos de línea y los retornos de carro que puedan aparecer.
xs:maxInclusive	Especifica que el valor debe ser menor o igual que el indicado.
xs:maxExclusive	Especifica que el valor debe ser menor que el indicado.
xs:minExclusive	Especifica que el valor debe ser mayor que el indicado.
xs:minInclusive	Especifica que el valor debe ser mayor o igual que el indicado.
xs:totalDigits	Especifica el número máximo de dígitos que puede tener un número.
xs:fractionDigits	Especifica el número máximo de decimales que puede tener un número.

Ejemplos:

```
<xs:element name="mes">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="12"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
—
<xs:element name="color">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="verde"/>
      <xs:enumeration value="amarillo"/>
      <xs:enumeration value="rojo"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
—
<xs:element name="letra">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[a-z]"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

```

<xs:element name="clave">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:length value="12"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

```

```

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

```

- preserve es la restricción para cuando los espacios en blanco, las tabulaciones, los saltos de línea y los retornos de carro que aparezcan en él deben mantenerse.
- replace para sustituir todas las tabulaciones, los saltos de línea y los retornos de carro por espacios en blanco.
- collapse para, después de reemplazar todas las tabulaciones, los saltos de línea y los retornos de carro por espacios en blanco, eliminar todos los espacios en blanco únicos y sustituir varios espacios en blanco seguidos por un único espacio en blanco.

- EXTENSIONES:

xs:extension sirve para extender un elemento simpleType o 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>Florencia</ciudad>
    <pais>Italia</pais>
  </ficha>
</fichas>

```

- Obsérvese que, infoPersonaAmpliada se basa en infoPersona, añadiendo dos elementos: "ciudad" y "país".
- En cuanto a **xs:complexContent**, sirve para definir restricciones o extensiones a un tipo complejo (**complexType**).

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="fichas">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="ficha" type="infoPersonaAmpliada"
          maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:complexType name="infoPersonaAmpliada">
    <xs:complexContent>
      <xs:extension base="infoPersona">
        <xs:sequence>
          <xs:element name="ciudad" type="xs:string"/>
          <xs:element name="pais" type="xs:string"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="infoPersona">
    <xs:sequence>
      <xs:element name="nombre" type="xs:string"/>
      <xs:element name="edad" type="edadPersona"/>
    </xs:sequence>
    <xs:attribute name="numero" type="xs:integer"/>
  </xs:complexType>

  <xs:simpleType name="edadPersona">
    <xs:restriction base="xs:integer">
      <xs:minExclusive value="-1"/>
      <xs:maxExclusive value="131"/>
    </xs:restriction>
  </xs:simpleType>

</xs:schema>

```

```
<?xml version="1.0" encoding="UTF-8"?>
<precios xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="precios.xsd">
  <precio moneda="Euro">5</precio>
  <precio moneda="Dólar">6.2</precio>
  <precio moneda="Libra esterlina">4.3</precio>
</precios>
```

- **xs:simpleContent** permite definir restricciones o extensiones a elementos que solo contienen datos, es decir, no contienen a otros elementos.
- Nótese que, utilizando **xs:extension**, al elemento "precio" se le ha incorporado el atributo moneda.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="precios">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="precio" maxOccurs="unbounded">
          <xs:complexType>
            <xs:simpleContent>
              <xs:extension base="xs:decimal">
                <xs:attribute name="moneda" type="xs:string"/>
              </xs:extension>
            </xs:simpleContent>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

- ELEMENTOS COMPLEJOS:

Un elemento es complejo (**complexType**) cuando contiene uno o más elementos y/o atributos. De entre las posibles combinaciones de elementos y/o atributos que puede contener un elemento complejo (1 elemento y 0 atributos, 1 elemento y 1 atributo, 1 elemento y varios atributos, 0 elementos y 1 atributo...) cabe destacar las siguientes:

- Un elemento complejo puede estar vacío, es decir, no contener elementos ni texto, pero sí tener al menos un atributo.
- Un elemento complejo puede contener contenido mixto, es decir, contener uno o más elementos, además de texto. Por otra parte, podría tener atributos, o no.

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

```
<xs:element name="persona">
  <xs:complexType mixed="true">
    <xs:sequence>
      <xs:element name="nombre" type="xs:string"/>
      <xs:element name="ciudad" type="xs:string"/>
      <xs:element name="edad" type="xs:positiveInteger"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

- **INDICADORES:**

Indicadores de orden: secuencia (**sequence**), cualquier orden (**all**) y elección (**choice**).

Indicadores de ocurrencia: el máximo ocurrencias que pueden aparecer (**maxOccurs**) y el mínimo de ocurrencias que pueden aparecer(**minOccurs**).

Indicadores de grupo: de elementos (**group**) y de atributos (**attributeGroup**).