
ADMINISTRACIÓN DE SISTEMAS INFORMÁTICOS EN RED

Lenguajes de Marcas y SGI

ACTIVIDADES UNIDAD 6.2 SCHEMA

ELEMENTOS SIMPLES

Simple1)

Crear un documento simple1.xml que contenga los datos personales de un alumno ficticio como el que se muestra

```
<?xml version="1.0" encoding="UTF-8"?>
<alumno xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="simple.xsd">
  <nombre>juan</nombre>
  <dni>28542345</dni>
  <direccion>San José 2</direccion>
  <edad>22</edad>
  <telefono>a959434523</telefono>
</alumno>
```

- Crear el documento simple1.xsd que aparece a continuación y complétalo para validar simple1.xml.

```
<?.....?>
<xs:.....">
  <xs:element name="alumno">

  <xs:complexType>
    <xs:sequence>
      <xs:element name="nombre" type="xs:string" />
      .....
    </xs:sequence>
  </xs:complexType>
  </xs:element>
</xs:schema>
```

Simple2)

Crear simple2.xsd para validar el siguiente documento simple2.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Libro xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="simple2.xsd">

  <Titulo>Genetic Programming</Titulo>
  <Autor>John Koza</Autor>
  <Fecha>1994</Fecha>
  <ISBN>0-202-111170-5</ISBN>
  <Editorial>The MIT Press</Editorial>

</Libro>
```

RESTRICCIONES

Restricción1)

- Modificar simple1.xml. *escribiendo como contenido de edad 20 y guardarlo como restricción1.xml.*
- Guardar simple1.xsd como restricción1.xsd
- Modificar restricción1.xsd aplicando una restricción sobre un valor numérico (por ejemplo: la edad, que debe estar comprendida entre 16 y 25).
- Probar la validez

Restricción2)

- Modificar restricción1.xml. *añadiendo el elemento sexo con el contenido M y guardarlo como restricción2.xml.*
- Modificar restricción1.xsd aplicando una restricción sobre un conjunto de valores. Por ejemplo el campo sexo puede ser M o H. y guardarlo como restricción2.xsd.
- Probar la validez.
- Modificar restricción2.xsd creando el tipo personalizado tipoSexo, es decir que el elemento debe aparecer en la secuencia como `<xs:element name="sexo" type="tipoSexo"/>` y salvarlo como restricción2A.xsd
- Probar la validez de restricción2.xml.

Restricción3)

- Modificar restricción2.xsd aplicando una restricción sobre series de valores y guardarlo como restricción3.xsd. Por ejemplo el teléfono debe ser del formato c-nnnnnnnn.
- Validar el documento xml escribiendo números de teléfonos que no se adapten al patrón.

Restricción4)

- Modificar restricción3.xsd aplicando restricciones sobre tamaño y guardarlo como restricción4.xsd. Por ejemplo el DNI o el teléfono.
- Validar el documento xml escribiendo números de DNI o el teléfono con mayor y menor longitud.

ELEMENTOS COMPLEJOS

Complejo1)

Dado el siguiente documento xml complejo1.xml crear el correspondiente esquema complejo1.xsd que lo valide.

```
<?xml version="1.0" encoding="UTF-8"?>
<alumno dni="111" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="alumno.xsd">
  <nombre>Antonio Pérez</nombre>
  <direccion>
    <calle>Calle Fuente</calle>
    <numero>4</numero>
    <ciudad>Jabugo</ciudad>
    <provincia>Huelva</provincia>
  </direccion>
  <telefono>959222324</telefono>
</alumno>
```

Complejo2)

Añadir a complejo1.xsd la restricción de tamaño 9 al número de teléfono y guardarlo como complejo2.xsd

Complejo3)

Modificar complejo3.xsd creando el tipo dirección y haciendo uso del mismo, salvándolo como complejo3.xsd.

INDICADORES

Indicadores1)

- Modificar complejo1.xsd donde sea necesario usando las cláusulas all. Y guardarlo como indicador1.xsd.

Indicadores2)

- Añadir otro elemento complejo, por ejemplo el elemento Teléfonos, donde puede aparecer el fijo o el móvil. Guardarlo como indicador2.xml y crear indicador2.xsd para validarlo.

Indicadores3)

- a) Modificar indicador2.xml de forma que tenga como elemento raíz alumnos y añadir varios alumnos, guardarlo como indicador3.xml.
- b) Crear indicador3.xsd para validar utilizando indicadores de ocurrencia.

Indicadores4)

- Documentar *personal.xsd* añadiendo usando anotaciones

SCHEMA

Schema1)

Definir el esquema para el siguiente documento xml y probar su validez

```
<?xml version="1.0" encoding="UTF-8"?>
<pizzas xsi:NoNameSchemaLocation="pizza.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <pizza nombre="Barbacoa" precio="8">
    <ingrediente nombre="Salsa Barbacoa"/>
    <ingrediente nombre="Mozzarella"/>
    <ingrediente nombre="Pollo"/>
    <ingrediente nombre="Bacon"/>
    <ingrediente nombre="Ternera"/>
  </pizza>
  <pizza nombre="Margarita" precio="6">
    <ingrediente nombre="Tomate"/>
    <ingrediente nombre="Jamón"/>
    <ingrediente nombre="Queso"/>
  </pizza>
  <pizza nombre="Tres Quesos" precio="10">
    <ingrediente nombre="Cabrales"/>
    <ingrediente nombre="Mozzarella"/>
    <ingrediente nombre="Manchego"/>
    <ingrediente nombre="Jamón"/>
    <ingrediente nombre="Bacon"/>
  </pizza>
</pizzas>
```