

Ejercicios XML schema

- 1) Definir un esquema para que los datos de alumno contenidos en el siguiente fichero sean validados correctamente.

Todos los elementos son obligatorios, deben aparecer en dicho orden y sólo una vez.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<alumno xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="schemaEjercicio1Schema.xsd">
  <dni>11111111</dni>
  <nombre>Iraide Etxeberria</nombre>
  <direccion>
    <calle>Kalenagusia</calle>
    <numero>5</numero>
    <ciudad>Bilbao</ciudad>
    <provincia>Bizkaia</provincia>
  </direccion>
  <telefono>944000000</telefono>
</alumno>
```

- 2) Vamos a ampliar el ejercicio anterior haciendo que permita una relación de alumnos.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<alumnos xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="schemaEjercicio2Schema.xsd">
  <alumno>
    <dni>11111111</dni>
    <nombre>Iraide Etxeberria</nombre>
    <direccion>
      <calle>Kalenagusia</calle>
      <numero>5</numero>
      <ciudad>Bilbao</ciudad>
      <provincia>Bizkaia</provincia>
    </direccion>
    <telefono>944000000</telefono>
  </alumno>
  <alumno>
    <dni>22222222</dni>
    <nombre>Joseba Mikel Zarraga</nombre>
    <direccion>
      <calle>Gorbeia</calle>
      <numero>75</numero>
      <ciudad>Arteaga</ciudad>
      <provincia>Bizkaia</provincia>
    </direccion>
    <telefono>944999999</telefono>
  </alumno>
</alumnos>
```

- 3) Modifica el ejercicio anterior teniendo en cuenta:

- El alumno tiene que tener obligatoriamente un teléfono y opcionalmente otro más. Opcionalmente podrá tener un email (el email irá a continuación del teléfono).
- El alumno debe tener teléfono o móvil, pero sólo uno de ellos. Si tiene móvil, puede tener hasta 3. Obligatoriamente tiene que tener un email.

c) Suponiendo que es indiferente el orden de los elementos, ¿se podría cambiar esto

```
<xs:element name="alumno" maxOccurs="unbounded">
  <xs:complexType>
    <xs:sequence>
```

por esto otro?

```
<xs:element name="alumno" maxOccurs="unbounded">
  <xs:complexType>
    <xs:all>
```

¿En qué ejercicio sí y en cuál no?

- 1
- 2
- 3 a
- 3 b

- 4) Crea un tipo nuevo con el elemento direccion.
- 5) Añade los siguientes atributos:
 - a) Cambia el elemento dni y ponlo como atributo al elemento alumno, es obligatorio.
 - b) Añade al elemento alumno un atributo repetidor que por defecto es false.
 - c) Añade al elemento alumno un atributo fechaNacimiento, es obligatorio.
- 6) Añade las siguientes restricciones:
 - a) El número de la dirección debe ser como mínimo 1 y como máximo 1000, ambos inclusive.
 - b) La calle debe contener como máximo 100 caracteres.
 - c) Ciudad y provincia ambas máximo 50 caracteres.
 - d) El teléfono y el móvil deben contener exactamente 9 caracteres numéricos.
 - e) El dni debe contener exactamente 8 dígitos y una letra en mayúsculas.
- 7) Crea un tipo nuevo para el dni, otro para ambos teléfonos y otro para ciudad y provincia.
- 8) Añade al elemento ciudad el atributo codPostal. Añade a la calle el atributo tipo. Ambos son opcionales. Añade al final del elemento direccion un elemento vacío opcional llamado verificado.
- 9) Añade al final del elemento alumno un elemento opcional vacío minusvalia que contiene un atributo obligatorio llamado grado.
- 10) Añade las siguientes restricciones:
 - a) El tipo de calle sólo puede ser uno de los siguientes: calle, carretera, camino, plaza, avenida, travesía.
 - b) El grado de la minusvalía tiene que contener un número de máximo 2 cifras y el símbolo porcentaje.
- 11) A continuación de la dirección, añade el elemento direccionTrabajo, es opcional. Contiene exactamente lo mismo que el elemento direccion excepto porque no tiene el elemento verificado.
- 12) A continuación del email, el alumno puede tener padre y madre. Podrán aparecer como máximo 2 veces (puede tener sólo un padre, sólo una madre, madre y padre, 2 padres, 2 madres o ninguno de ellos). Padre y madre serán de tipo "progenitor" que contendrá el nombre y un teléfono de contacto (en ese orden).

13) Define un tipo de datos de esquemas XML que valide el siguiente fragmento de un documento XML:

```
<temperatura escala="Celsius">37</temperatura>
```

El tipo de datos del elemento *temperatura* es integer, y el del atributo *escala* uno derivado de string que solamente permita los valores *Celsius*, *Fahrenheit* o *Kelvin*.

14) Construye los tipos de datos de esquemas XML más adecuados para representar:

- a. Una matrícula de coche actual española, como 1234-BCD (no se permiten vocales)
- b. Un DNI o un NIE, que sean alternativa uno de otro

DNI: 7 dígitos + carácter de control

NIE: 1 letra (X, Y o Z) + 7 dígitos + carácter de control

Carácter de control: cualquier letra de la A a la Z exceptuando 3: I, O y U