

EJERCICIOS XSD

1. Crea un archivo XML que incluya el siguiente contenido y un XSD capaz de validarlo. El orden de los elementos debe ser tal cual aparecen en el XML:

```
<cancion>
  <banda>Europe</banda>
  <titulo>The final countdown</titulo>
  <letra>We're leavin' together...</letra>
  <fecha-lanzamiento>1986</fecha-lanzamiento>
</cancion>
```

2. Crea un XSD capaz de validar los tres posibles contenidos de un documento XML:

```
<premio>
  <medalla>Oro</medalla>
</premio>
-----
<premio>
  <metalico>1000.00</metalico>
</premio>
-----
<premio>
  <vale>Fin de semana para dos personas en Paymogo</vale>
</premio>
```

3. Crea un archivo XSD capaz de validar el siguiente contenido XML:

```
<periodico>
  <articulo>
    <titulo>Asalto al Capitolio</titulo>
    <subtitulo>Cuernos en la democracia estadounidense</subtitulo>
    <contenido>El pasado día 6 de enero...</contenido>
    <autor>Juan de la Olla</autor>
  </articulo>
  <articulo>
    <titulo>Cumbres borrascosas en Madrid</titulo>
    <contenido>La borrasca Filomena ha dejado...</contenido>
    <autor>Pedro Flores</autor>
  </articulo>
</periodico>
```

4. Crea un archivo XSD capaz de validar el siguiente contenido XML:

```
<catalogo>
  <producto>
    <nombre>Teclado inalámbrico Zeus 23Q</nombre>
    <vendedor>Todo Teclados</vendedor>
    <precio>23.90</precio>
    <detalle>Opera en la frecuencia de 2.4 GHz</detalle>
    <detalle>Hasta 20 metros de distancia</detalle>
  </producto>
  <producto>
    <nombre>Smartwatch Pulserita Wear</nombre>
    <precio>120.90</precio>
    <opiniones>
      <opinion>Mi abuela me toma mejor el pulso</opinion>
      <opinion>No es resistente al agua</opinion>
    </opiniones>
  </producto>
</catalogo>
```

5. Crea un archivo XSD capaz de validar el siguiente contenido XML sabiendo que las tallas se pueden ofertar hasta en 3 posibles tamaños: *small*, *medium* y *large*. Además, la etiqueta “opiniones” puede aparecer una vez o ninguna, pero si aparece tendrá al menos 1 “opinión”:

```
<catalogo>
  <modelo>
    <id_modelo>QWZ5671</id_modelo>
    <precio>39.95</precio>
    <tallas>
      <medium>
        <color>Red</color>
        <color>Blue</color>
      </medium>
      <large>
        <color>Red</color>
        <color>Burgundy</color>
      </large>
    </tallas>
  </modelo>
  <opiniones>
    <opinion>Muy ponible en cualquier situación</opinion>
  </opiniones>
</catalogo>
```

6. Crea un archivo XSD capaz de validar el siguiente contenido. Podrán existir tantas matrículas como sean necesarias, pero también es posible que no haya ninguna. Además, se sabe que, al menos, un domicilio debe aparecer obligatoriamente. Y que el tipo de pago solo puede contener uno y solo uno de estos elementos: “efectivo” o “transferencia” o “tarjeta”.

```
<matriculacion>
  <curso_academico>2020/2021</curso_academico>
  <matricula>
    <datos_personales>
      <dni>99223366M</dni>
      <nombre>Juan Pardo Martín</nombre>
      <titulacion>Ingeniería Informática</titulacion>
      <domicilios>
        <domicilio>
          <nombre>C/ Principal nº1</nombre>
        </domicilio>
        <domicilio>
          <nombre>C/ Secundaria nº2</nombre>
        </domicilio>
      </domicilios>
    </datos_personales>
    <pago>
      <tipo_matricula>Matrícula Ordinaria</tipo_matricula>
      <fecha_pago>2021-01-24</fecha_pago>
      <tipo_pago>
        <efectivo>648.12</efectivo>
      </tipo_pago>
    </pago>
  </matricula>
</matriculacion>
```

7. Crea un archivo XSD capaz de validar el siguiente contenido. Piensa que el *head*, el *title* y el *body* son opcionales. Que al menos debe haber un contenido en el *body*... Sigue mentalmente las reglas que conoces de HTML.

```
<html>
  <head><title>Canal Marcas</title></head>
  <body>
    <h1>¿Es ético falsificar conversaciones de Whatsapp?</h1>
    <h2>...solo para gastar una broma inofensiva</h2>
    <p>Se dice que el superpoder de la invisibilidad solo puede usarse para hacer el mal. Esconderse no tiene una utilidad heroica y solo sirve para espiar y engañar. Engañar es precisamente lo que hace la aplicación que hoy traemos a este espacio. Con ella puedes falsificar conversaciones de WhatsApp. <br/> ¿Hasta qué punto es ética? <br/> ¿Cómo se justifica su existencia?</p>
  </body>
</html>
```

8. Tomando la siguiente tabla de información, escribe un documento XML bien formado que refleje la relación entre vehículos y sus dueños. Además, debe validarse contra un XSD que debe tener una anotación/documentación indicando el autor, la fecha actual y el nombre del documento.

VENTA VEHICULOS				
MATRÍCULA	MARCA	MODELO	PROPIETARIO	DNI
4534-FRG	Seat	Ibiza	Juan Luis Guerra	12654123T
8333-ABC	Volvo	S40	Luis Miguel López	62854854Z
1234-ZAP	Seat Fiat	Toledo Punto		

Debes tener en cuenta que:

- La columna DNI debe entenderse como un metadato del propietario y que además es obligatorio.
 - La columna MATRÍCULA debe entenderse como un metadato del vehículo y, en caso de que un vehículo aún no tenga matrícula se le debe asignar el valor por defecto "0000-AAA".
9. Una empresa de reparaciones informáticas quiere almacenar información sobre los clientes y sus dispositivos mediante un documento XML bien formado teniendo en cuenta que:
- De los clientes queremos guardar: nombre, apellidos, DNI, teléfono, email. El DNI y el teléfono deben ser tratados atributos obligatorios. El email será un atributo opcional.
 - Un cliente puede tener más de un dispositivo.
 - De los dispositivos queremos guardar: tipo (ordenador, Tablet...), marca, modelo y las reparaciones que se le han realizado.
 - De cada reparación se debe almacenar: fecha, descripción del problema según el cliente, solución aplicada por el personal de la tienda y número de horas dedicadas. El número de horas dedicadas será un atributo que, en caso de no escribirse, deberá valer 1.

Debes escribir un documento con al menos 2 clientes. Un cliente tendrá 1 dispositivo con una reparación. El otro cliente tendrá 2 dispositivos sin reparaciones. Además, debes realizar un XSD que valide el documento anterior.

10. Retoca el XSD del ejercicio anterior para que cree y utilice los siguientes tipos definidos por el usuario:
- Tipo "horas-trabajo" que será un entero de 0 a 100.
 - Tipo "id-cliente" que será un entero mayor o igual que 1000.

Modifica el XML para que el cliente lleve un nuevo elemento obligatorio "id" de tipo "id-cliente"

11. Retoca el ejercicio anterior para que el XSD defina y utilice un tipo complejo para los siguientes elementos del XML: dispositivo, reparaciones y reparación.

12. Vamos a preparar el camino para el ejercicio 13 fabricando algunos tipos de datos simples. Para ello crea un documento XSD que defina los siguientes tipos:
- a) *tipo_codigo* será un entero comprendido entre 1 y 999999
 - b) *tipo_cod_postal* será un entero de 5 cifras
 - c) *tipo_importe* será un número real con hasta 2 cifras decimales y hasta 8 cifras en total.
 - d) *tipo_cif* será una cadena de texto que cumpla con el patrón: 1 letra mayúscula seguida de 8 dígitos.
 - e) *tipo_cantidad* será un entero cuyo valor mínimo será 1
 - f) *tipo_forma_pago* será una cadena que solo puede tomar uno de estos tres valores: efectivo o tarjeta o transferencia
 - g) *tipo_moneda* será una cadena que solo podrá tomar uno de estos dos valores: euro o dólar.
13. Crea un XSD que valide el siguiente XML. Para ello debes partir de los tipos simples que has declarado en el ejercicio anterior y además debes declarar y utilizar los siguientes tipos complejos:
- a) *tipo_datos_empresa* que contendrá la estructura del elemento `<datos_empresa>` del documento XML.
 - b) *tipo_datos_cliente* que contendrá la estructura del elemento `<datos_cliente>` del documento XML.
 - c) *tipo_linea_factura* que contendrá la estructura del elemento `<linea>` del documento XML.
 - d) *tipo_datos_factura* que contendrá la estructura del elemento `<datos_factura>` del documento XML.

De este modo el nodo raíz del XSD quedará definida de la siguiente forma:

```
<xs:element name="factura">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="datos_empresa" type="tipo_datos_empresa"/>
      <xs:element name="datos_cliente" type="tipo_datos_cliente"/>
      <xs:element name="datos_factura" type="tipo_datos_factura"/>
    </xs:sequence>
    <xs:attribute name="cod_fac" type="tipo_codigo" use="required"/>
  </xs:complexType>
</xs:element>
```

Una última consideración: todos los atributos son obligatorios.

```
<?xml version="1.0" encoding="UTF-8"?>
<factura cod_fac="999">
  <datos_empresa>
    <nombre>Equipos Digitales S.L.</nombre>
    <dir>Av. Valladolid</dir>
    <cod_postal>28043</cod_postal>
    <poblacion>Madrid</poblacion>
    <provincia>Madrid</provincia>
    <cif>Q98765433</cif>
    <telefono/>
  </datos_empresa>
  <datos_cliente cod_cli="879">
    <nombre>Darío Bueno Gutiérrez</nombre>
    <dir_env>Av. Oporto nº7 4ºd</dir_env>
    <cod_postal>28043</cod_postal>
    <poblacion>Madrid</poblacion>
    <provincia>Madrid</provincia>
  </datos_cliente>
  <datos_factura cod_ped="731" iva="21" forma_pago="efectivo" moneda="euro">
    <fecha>2005-12-01</fecha>
    <linea>
      <ref>MII93000F/8</ref>
      <desc>MICRO PENTIUM IV 3000MHZ FB800</desc>
      <cant>1</cant>
      <precio>230</precio>
      <subtotal>230</subtotal>
    </linea>
    <linea>
      <ref>MB8QDIP4</ref>
      <desc>PLACA BASE QDI P4</desc>
      <cant>1</cant>
      <precio>208.80</precio>
      <subtotal>208.80</subtotal>
    </linea>
    <linea>
      <ref>MEDD512M32</ref>
      <desc>DIMM DDR 512MB 3200</desc>
      <cant>2</cant>
      <precio>40</precio>
      <subtotal>80</subtotal>
    </linea>
    <linea>
      <ref>HD250GSA7</ref>
      <desc>DISCO DURO 250GB S-ATA 7200</desc>
      <cant>4</cant>
      <precio>60.10</precio>
      <subtotal>240.40</subtotal>
    </linea>
    <base>759.2</base>
    <cuota_iva>159.43</cuota_iva>
    <total>918.63</total>
  </datos_factura>
</factura>
```

EJERCICIOS OPCIONALES**EJERCICIOS DEL ANEXO "ESTRATEGIAS DE DISEÑO DE UN XSD"**

14. Crea un XSD que valide el siguiente XML utilizando para ello el modelo de referencias a elementos. Para ello debes definir y referenciar los elementos <personal>, <domicilios>, <domicilio> y <pago>. Realiza las suposiciones que crea necesarias pero anótalas usando comentarios <!--esta es la notación de los comentarios -->

```
<matricula>
  <personal>
    <dni>99223366M</dni>
    <nombre>Juan Pardo Martín</nombre>
    <titulacion>Ingeniería Informática</titulacion>
    <curso_academico>2020/2021</curso_academico>
    <domicilios>
      <domicilio>
        <nombre>C/ Principal nº1</nombre>
      </domicilio>
      <domicilio>
        <nombre>C/ Secundaria nº2</nombre>
      </domicilio>
    </domicilios>
  </personal>
  <pago fecha="2021-02-01">
    <tipo_matricula>Matrícula Ordinaria</tipo_matricula>
    <forma_pago>Efectivo</forma_pago>
  </pago>
</matricula>
```

15. Parte el XSD del ejercicio 13 en dos archivos. Uno debe llamarse "tipos.xsd" y contener las definiciones de tipos del modelo. El otro documento debe llamarse "factura.xsd" y debe incluir al documento anterior, así como la definición del nodo raíz.

Utiliza el documento XML del ejercicio 13 para validar contra el archivo "factura.xsd".