## **Los archivos necesarios están en la carpeta BOX / DRIVE compartida.**

## Ejercicio 1

Partiendo de las siguientes etiquetas, crea el correspondiente documento XML y el Schema XML asociado empleando el generador de XML Copy Editor.

<curso>

<denominacion>Historia de los videojuegos</denominacion>

<numAlumn>12</numAlumn>

<fechaIni>2022-03-12</fechaIni>

<online>true</online>

</curso>

Cuando lo hayas realizado compáralos con 02ElementosXSD.xml y 02ElementosXSD.xsd

¿Cuáles son las diferencias?

-----------------------------------------------------------------------

## Ejercicio 2

Partiendo de la estructura de datos Ejercicio02Schema.xsd crear un documento XML válido.

-----------------------------------------------------------------------

## Ejercicio 3

Dado el documento XML Ejercicio03Schema.xml crear el archivo Ejercicio03Schema.xsd con la estructura de datos correspondiente. El documento XML debe ser validado bajo las siguientes circunstancias:

* La "edad" no pueda ser menor que 0 ni mayor que 100
* Debes incluir la restricción de forma que pueda ser empleada por otros elementos además de la edad.

-----------------------------------------------------------------------

## Ejercicio 4

Partiendo del documento XML Ejercicio04Schema.xml crear el archivo Ejercicio04Schema.xsd con la estructura de datos correspondiente. El documento XML debe ser validado teniendo en cuenta que el elemento "precio" puede tomar por valor un número que contenga tres dígitos como máximo y, de ellos, solamente dos pueden ser decimales. Para ello, escribir una restricción que no podrá ser utilizada por otros elementos y, por otra parte, haga uso de:

* *xs:totalDigits* que sirve para especificar el número máximo de dígitos que puede tener un número, incluyendo a los decimales.
* *xs:fractionDigits* que sirve para especificar el número máximo de decimales que puede tener un número.

----------------------------------------------------------------------

## Ejercicio 5

Partiendo del esquema Ejercicio05Schema.xsd generar un documento XML que lo valide correctamente.

----------------------------------------------------------------------

## Ejercicio 6

Convertir el DTD Ejercicio06Schema.dtd dado en un Schema XML teniendo en cuenta lo siguiente:

* El esquema se guarda en un archivo de nombre “catalogo\_libros.xsd”
* Para los tipos complejos se deben utilizar referencias a elementos.
* Se deben definir dos tipos de datos simples para el mes (“tipoMes”) y el año (“tipoAño”), como restricciones del tipo “positiveInteger”. El mes tiene que ser un valor comprendido entre 1 y 12, y el año entre 1900 y 2021.
* También se define el tipo “tipoNombre”, que será el utilizado para los elementos “titulo”, “autor” y “editor”. Se trata de una restricción del tipo “string” en la que los valores tienen entre 0 y 100 caracteres.
* Del mismo modo se define el tipo “tipoISBN”, para el elemento ISBN, como un tipo string que tiene entre 13 y 17 caracteres.

Además debes crear un documento XML que valide la estructura XSD con los siguientes datos,

**Título: Los hombres que no amaban a las mujeres.**

Autores: Stieg Larsson

Fecha: 2008

ISBN: 978-84-233-4044-6

Editor: Destino

Categoría: Ficción

Stock: Sí

**Título: Java a través de ejemplos.**

Autores: Jesús Bobadillo, Adela Sancho

Fecha: 2003

ISBN: 84-7897-549-7

Editor: Ra-ma

Categoría: No Ficción

Stock: No

Revisor: Carlos Montes

## Ejercicio7

Dado el documento Ejercicio07Schema.xsd explica, en un documento de texto, línea por línea el código empleado y genera un documento XML válido sin modificar la estructura de datos.

## Ejercicio8

Partiendo del XSD del ejercicio anterior crea otro Ejercicio08Schema.xsd que obligue a un nuevo documento XML Ejercicio08Schema.xml a cumplir las siguientes limitaciones:

1. El contenido de ISBN debe ser *nn-nn-nnnn* donde n es un dígito entre 0 y 9.
2. El atributo código debe ser único y estar formado por tres caracteres.
3. Debemos incluir la fecha de publicación de libro.
4. La edición no es un dato obligatorio pero sí lo es la editorial que sólo podrá tomar los valores “Lengua de trapo” y “Libros del KO”.
5. Hay que incluir un nuevo elemento (CARACTERISTICAS) que tendrá dos elementos hijos (NUMPAGINAS y CALIDAD) sobre los que hay las siguientes restricciones:
   1. NUMPAGINAS, es el número de páginas, siempre un valor entero entre 160 y 430, inclusive.
   2. CALIDAD puede tomar ser RÚSTICA o BOLSILLO

---------------------------------------------------------------

## Ejercicio9

1. Definir un elemento simple que no pueda contener nodo texto y que pueda aparecer una vez o ninguna. <matriculaParcial/>
2. Definir un elemento complejo (con atributo sin restricciones) que no pueda contener nodo texto y que pueda aparecer una vez o ninguna. <matricula activado=”parcial”/>
3. Definir un elemento complejo (con atributo que puede tomar los valores “parcial” o “total”) que no pueda contener nodo texto y que pueda aparecer una vez o ninguna.

Crear Ejercicio09Schema.xsd y Ejercicio09Schema.XML que recojan estos casos planteados.

## Ejercicio10

Partiendo del documento XML Ejercicio10Schema.xml y las condiciones indicadas, crear el correspondiente Schema.

\* El nombre del aeropuerto, los vuelos y la fecha pueden aparecer en distinto orden.

\* Se tiene que indicar que el código ha de ser único y obligatorio para cada vuelo.

\* Haciendo uso *pattern* indicar que los posibles estados de un vuelo son C (Cancelado), E (En hora), R (Retrasado). El valor por defecto debe ser E.

\* Debe permitirse aparecer desde cero hasta ilimitados elementos vuelo y, para cada uno de ellos, se tiene que guardar la información en el mismo orden en el que aparece en el documento XML.

\* Para indicar si un vuelo es diario, se debe utilizar un elemento vacío que, respecto a cada vuelo, podrá aparecer (en el caso de sí ser diario) o no aparecer (en el caso contrario)

\* Respecto a los elementos nombre, origen, destino, hora-llegada, hora-salida y fecha, cada uno de ellos debe definirse del tipo que se considere más apropiado, de entre los proporcionados por XML Schema.

## Ejercicio11

Dado el fichero Ejercicio11Schema.xsd crear un documento XML que lo valida y explicar porque sólo hay un tipo de elemento complejo que pueda cumplirlo.