

## Bloque 1 - Práctica 5 - Ejercicios XML, JSON, DTD, VSCode, ...

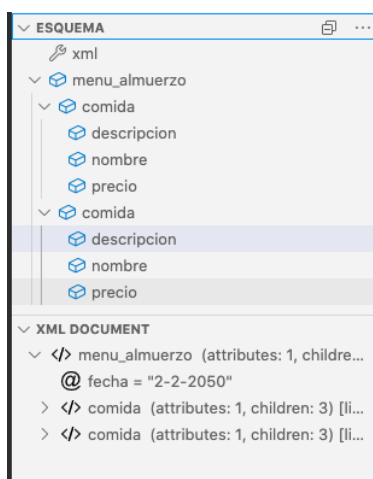
Recuerda el [protocolo de realización y entrega de prácticas](#)

1. Pulsa en el enlace [b1p5](#) para crear un nuevo repositorio en GitHub en el que realizar los ejercicios de esta práctica.
2. Instala las siguientes extensiones:
  - a. [XML Language Support by Red Hat](#) → ver [diapositiva](#). En caso de error instala JAVA
  - b. [XML Tools](#)
3. ¿Está bien formado el siguiente fichero xml? Haz las correcciones necesarias para que lo esté.

```
<menu_almuerzo>
  <comida calorías="650">
    <nombre>Waffles</nombre>
    <precio>2.00</Precio>
    <descripcion>Waffles baratos de McDonalds</descripcion>
  </comida>
  <comida calorías=1500>
    <nombre>Hamburguesa</nombre>
    <precio>5.00
    <descripcion>La hamburguesa mas común de McDonalds</descripcion>
  </comida>
</menu_amuerzo>
```
4. Utiliza VSCode para probar si el fichero anterior está bien formado. Observa en el panel "Explorador" los apartados Esquema y XML Document. Nombre: menu.xml
5. Busca un fichero XML en Internet y comprueba si está bien formado en VSCode, añádelo en un subdirectorío de nombre ejemplos dentro del proyecto.
6. Crea un documento XML de nombre cds.xml para publicar el catálogo de CDs de una tienda online de música, los datos son: título, artista, precio y año.
7. Crea un fichero XML de nombre productos.xml para almacenar e intercambiar los datos que aparecen en la siguiente hoja de cálculo.

	A	B	C	D	E
1	Producto	Cantidad	Precio	Maximo Ganancia	
2	Coca Cola	100	2	200	
3	Pepsi	50	1,5	75	
4	Fanta	26	2	52	
5	Mirinda	56	3	168	
6	Sprite	203	2	406	
7					

8. Necesitamos intercambiar datos entre dos aplicaciones de dos empresas distintas. Se ha decidido que se seguirá la recomendación XML 1.0 de W3C. Los datos a transferir son sobre el nivel de existencias de una lista de productos, de forma que una empresa comunica a la otra su nivel de stock. Cuando los niveles son inferiores a una cantidad la empresa suministra automáticamente los productos. La información a intercambiar comprende: sku del producto (código de identificación), nombre, stock actual, stock mínimo y stock máximo. También se envía información de la fecha y hora en la que los datos han sido obtenidos. Crea el documento XML **sin utilizar atributos**. Nombre del fichero: existenciasV1.xml
9. Modifica el documento XML creado en el ejercicio anterior de forma que parte de la información esté en atributos. Nombre del fichero: existenciasV2.xml
10. **Opcional** - Utiliza <http://codebeautify.org/xmlviewer/sampleData> para ver en formato árbol la página web <https://www.w3.org/> comprueba si la web valida utilizando una herramienta online, como <https://validator.w3.org>
11. Descarga datos en formato XML, JSON y CSV del portal de [OpenData de Cáceres](#) y almacénalos en un subdirectorio dentro del proyecto de nombre opendata. Explica cada uno de los elementos que incluye el formato XML. Compara los tres formatos de forma que se pueda ver cómo se delimitan e identifican los datos en cada uno de ellos. Observa la licencia con la que se ofrecen los datos. Muestra en forma de árbol el fichero XML, utiliza <http://codebeautify.org/xmlviewer/sampleData>
12. Utiliza VSCode para crear el DTD de los ficheros XML que has realizado, incluyendo el obtenido del portal de OpenData de Cáceres (*hay que eliminar los atributos del elemento raíz, sparql, para que se pueda generar el DTD de forma automática*). ¿Cuál es el elemento raíz (root) de cada uno de los ficheros? ¿puede haber más de un elemento raíz? Comprueba que validan.
13. Modifica el catálogo de CDs para que incluya información del número de canciones que incluye cada disco. Vuelve a validar el fichero, ¿qué ocurre? ¿cómo lo podrías solucionar?
14. Observa en el panel “Explorer → Esquema” y en “Explorer → XML Document” los documentos XML creados.



15. ¿Qué lenguaje se utiliza en los datos de la imagen? conviértelo a XML con DTD y comprueba si valida

```
{
  "employees": {
    "employee": [
      {
        "id": "1",
        "firstName": "Tom",
        "lastName": "Cruise",
        "photo": "http://cdn2.gossipcenter.com/sites/default/files/imagecache/story_header/photos/tom-cruise-020514sp.jpg"
      },
      {
        "id": "2",
        "firstName": "Maria",
        "lastName": "Sharapova",
        "photo": "http://thewallmachine.com/files/1363603040.jpg"
      },
      {
        "id": "3",
        "firstName": "James",
        "lastName": "Bond",
        "photo": "http://georgesjournal.files.wordpress.com/2012/02/007_at_50_ge_pierece_brosnan.jpg"
      }
    ]
  }
}
```

16. Convierte el fichero XML del catálogo de CDs y del menú a formato JSON.
17. Realiza el ejercicio anterior, el de empleados.xml y empleados.dtd, utilizando la herramienta online <https://jsonformatter.org/json-to-xml>
18. Crea un fichero XML de nombre plantas.xml según la gramática del siguiente DTD, con al menos 4 registros, comprueba que valida.

```
<?xml version='1.0' encoding='UTF-8'?>
<!--
    <?xml version="1.0"?>

    <!DOCTYPE CATALOGO SYSTEM "newXMLDocument.dtd">

    <CATALOGO>
        * * *
    </CATALOGO>
-->

<!--ELEMENT CATALOGO (PLANTA)*-->
<!--ELEMENT PLANTA (NOMBRE_COMUN|NOMBRE_CIENTÍFICO|ZONA|LIGHT|PRECIO|DISPONIBILIDAD)*-->

<!--ELEMENT NOMBRE_COMUN (#PCDATA)-->
<!--ELEMENT NOMBRE_CIENTÍFICO (#PCDATA)-->
<!--ELEMENT ZONA (#PCDATA)-->
<!--ELEMENT LIGHT (#PCDATA)-->
<!--ELEMENT PRECIO (#PCDATA)-->
<!--ELEMENT DISPONIBILIDAD (#PCDATA)-->
```

19. Observa los siguientes conjuntos de datos en formato CSV

- [Occupancy Detection · master · Data Science Dojo / datasets](#)
- [datasets/titanic.csv at master · datasciencedojo/datasets · GitHub](#)

20. Un centro educativo quiere intercambiar información de horarios entre distintas aplicaciones utilizando XML y JSON. La información general que tiene que recoger es: **nombre** y **dirección web del centro educativo** y **curso escolar**. Por cada grupo recogerá la información completa del horario, incluida horas, abreviaturas de asignaturas/módulos, tutor/a, **horario de atención de tutoría**, profesoras, etc. Inventa los datos que no aparezcan

*Nota: este ejercicio se ampliará en el bloque de XML y en JS.*

- Utiliza los siguientes datos para crear un fichero XML de nombre `horarios.xml` con los datos de este curso.
- Asocia al fichero XML un DTD de nombre `horarios.dtd` de forma que el fichero pueda validarse.

TUTOR: Fátima Hontiveros Almaraz					
Horario 1º ASIR					
Hora	Lunes	Martes	Miércoles	Jueves	Viernes
1ª [ 15:20 – 16:15 ]	ISO	GBD	PARE	ISO	FHW
2ª [ 16:15 – 17:10 ]	LMSGI	FHW	GBD	ISO	FHW
3ª [ 17:10 – 18:05 ]	FOL	PARE	ISO	LMSGI	PARE
RECREO – 18:05 – 18:25					
4ª [ 18:25 – 19:20 ]	GBD	LMSGI Tutoría	LMSGI	GBD	PARE
5ª [ 19:20 – 20:15 ]	GBD	ISO	FOL	PARE	ISO
6ª [ 20:15 – 21:10 ]	PARE	ISO	FOL	PARE	GBD
1º GS – Administración de Sistemas Informáticos en Red					
ISO	Implantación de Sistemas Operativos (224 h) - Antonio Berrocal				
PARE	Planificación y Administración de Redes (224h) - Marian Valdera				
FHW	Fundamentos de Hardware (96 h) - Fernando Torrecilla				
GBD	Gestión de Bases de Datos (192 h) - Marcos Fernández				
LMSGI	Lenguajes de Marca y Sistemas Gestión de Inf.(128 h) - Antonio Berrocal				
FOL	Formación y Orientación Laboral (96 h) - Fátima Hontiveros				

TUTORA: Puerto Cruz Mateos					
Horario 1º DAM					
Hora	Lunes	Martes	Miércoles	Jueves	Viernes
1ª [ 15:20 – 16:15 ]	FOL	BBDD	ENTDL	ENTDL	SSINF
2ª [ 16:15 – 17:10 ]	FOL	BBDD	LMSGI	BBDD	SSINF
3ª [ 17:10 – 18:05 ]	ENTDL	LMSGI	BBDD	SSINF	ENTDL
RECREO [ 18:05 – 18:25 ]					
4ª [ 18:25 – 19:20 ]	PGRMC	PGRMC	BBDD	SSINF	LMSGI
5ª [ 19:20 – 20:15 ]	BBDD	PGRMC	PGRMC	FOL	PGRMC
6ª [ 20:15 – 21:10 ]	SSINF	SSINF	PGRMC	LMSGI	PGRMC

1º GS – Desarrollo de Aplicaciones Multiplataforma	
SSINF	Sistemas Informáticos (192 horas) - Julio Barbero
BBDD	Bases de Datos - (192 horas) - Puerto Cruz
PGRMC	Programación - (224 horas) - Quique Pineda
ENTDL	Entornos de Desarrollo (128 horas) - Marcos Fernández
LMSGI	Lenguajes de Marca y Sistemas Gestión de Inf.(128 h) - Antonio Berrocal
FOL	Formación y Orientación Laboral (96 h) - Fátima Hontiveros

TUTOR: Marcos Fernández Sellers					
Horario 1º DAW					
Hora	Lunes	Martes	Miércoles	Jueves	Viernes
1ª [15:20 – 16:15 ]	FOL	FOL	LMSGI	BBDD	BBDD
2ª [16:15 – 17:10 ]	SSINF	FOL	PGRMC	ENTDL	PGRMC
3ª [17:10 – 18:05 ]	SSINF	BBDD	PGRMC	ENTDL	PGRMC
RECREO – 18:05 – 18:25					
4ª [18:25 – 19:20 ]	LMSGI	BBDD	SSINF	LMSGI	ENTDL
5ª [19:20 – 20:15 ]	PGRMC	SSINF	BBDD	SSINF	ENTDL
6ª [20:15 – 21:10 ]	PGRMC	PGRMC	BBDD	SSINF	LMSGI
1º GS – Desarrollo de Aplicaciones Web					
SSINF	Sistemas Informáticos (192 horas) - Julio Barbero				
BBDD	Bases de Datos - (192 horas) - Puerto Cruz				
PGRMC	Programación - (224 horas) - Quique Pineda				
ENTDL	Entornos de Desarrollo (128 horas) - Marcos Fernández				
LMSGI	Lenguajes de Marca y Sistemas Gestión de Inf.(128 h) - Antonio Berrocal				
FOL	Formación y Orientación Laboral (96 h) - Marta Pérez				

- c. Crea un fichero JSON de nombre `horarios.json` con los datos de los horarios. Puedes utilizar una herramienta como

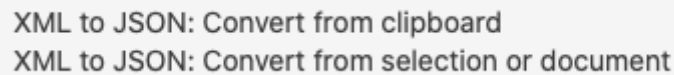
<https://jsonformatter.org/xml-to-json>, pero no olvides revisar el fichero generado.

21. **Opcional** - Instala estas extensiones en VSCode para facilitar el trabajo con JSON:

- a. [Json for Visual Studio Code](#)
- b. [Json Editor](#)

22. **Opcional** - Muestra en VSCode un fichero JSON en formato árbol utilizando la extensión [JSON Editor](#) instalada anteriormente.

23. **Opcional** - Utiliza la extensión de VSCode ["XML to JSON"](#) para convertir XML a JSON. Se accede con el botón derecho del ratón sobre un documento XML.



XML to JSON: Convert from clipboard  
XML to JSON: Convert from selection or document

24. **Opcional** - Utiliza el panel JSON para consultar los datos del fichero JSON del fichero `horarios.json`



25. **Opcional** -¿Cuándo necesitamos utilizar espacios de nombre-namespaces?

26. **Opcional** - Busca el DTD de XHTML 1.0 Strict, XHTML 1.0 Transitional, HTML 4.01 Strict y HTML 4.01 Transitional.