

Software y estándares para la Web

P2. TECNOLOGÍAS XML

Tabla de contenidos

Objetivos	2
Ejercicio 1: Documento XML, validación con DTD y con XML Schema	2
Tarea 1.....	2
Guía para resolver la tarea 1 del ejercicio 1.....	3
Tarea 2.....	3
Guía para resolver la tarea 2 del ejercicio 1.....	3
Tarea 3.....	3
Guía para resolver la tarea 3 del ejercicio 1.....	3
Tarea 4.....	4
Guía para resolver la tarea 4 del ejercicio 1.....	4
Tarea 5.....	4
Guía para resolver la tarea 5 del ejercicio 1.....	4
Tarea 6.....	4
Guía para resolver la tarea 6 del ejercicio 1.....	5
Tarea 7.....	5
Guía para resolver la tarea 7 del ejercicio 1.....	5
Ejercicio 2: Procesar y generar documentos XML.....	6
Tarea 1.....	6
Guía para resolver la tarea 1 del ejercicio 2.....	6
Tarea 2.....	7
Guía para resolver la tarea 2 del ejercicio 2.....	7
Tarea 3.....	7
Guía para resolver la tarea 3 del ejercicio 2.....	7
Ejercicio 3: Diseñar y construir una aplicación.....	8
Guía para resolver el ejercicio 3.....	8
Entrega de la práctica.....	9
Criterios de evaluación.....	9

Objetivos

En esta práctica se va a realizar:

- La creación de un documento XML (Teoría “Introducción a XML”)
- La validación del documento XML con un DTD (Teoría “DTD”)
- La validación del documento XML con un XML Schema (Teoría “Schemas”)
- La transformación de los documentos XML usando XPATH y XSLT (Teoría “XPath” y Teoría “XSLT”)
- El procesamiento del documento XML y la generación de nuevos documentos XML (Teoría “Procesamiento y generación de XML”)

Ejercicio 1: Documento XML, validación con DTD y con XML Schema

En este ejercicio se va a realizar la creación de un documento XML (Teoría “Introducción a XML”), la validación del documento XML con DTD (Teoría “DTD”) y la validación del documento XML con XML Schema (Teoría “Schemas”).

Tarea 1

Construir un documento XML bien formado para contener la red social de cada estudiante. Cada estudiante enlazará a tres amigos y cada amigo a otros tres amigos. El número total de personas en la red social será de 13 personas (el estudiante, los tres amigos iniciales y los nueve amigos de sus amigos iniciales), es decir la red social tiene 13 nodos.

Los nodos de la red social deberán contener la siguiente información:

- Nombre de la persona (por ejemplo “Juan Manuel”)
- Apellidos de la persona (por ejemplo “Cueva Lovelle”)
- Fecha de nacimiento
- Lugar de nacimiento (por ejemplo “Oviedo”)
- Coordenadas geográficas del lugar de nacimiento, especificando longitud, latitud y altitud
- Lugar de residencia (por ejemplo “Colloto”)
- Coordenadas geográficas del lugar de residencia, especificando longitud, latitud y altitud
- Fotografías de la persona (mínimo 1). Poner enlaces a archivos externos.
- Vídeos de la persona (mínimo 0 y máximo 3). Poner enlaces a archivos externos
- Comentarios sobre la persona

Se debe comprobar que el documento XML creado está bien formados en los 4 navegadores de referencia de la asignatura: Chrome, Firefox, Microsoft Edge y Opera.

Guía para resolver la tarea 1 del ejercicio 1

Se debe utilizar un editor de texto que permita crear archivos “UTF-8” sin BOM. Por ejemplo Visual Studio Code o NotePad++.

Se aconseja realizar simultáneamente la tarea 1 y la tarea 2.

Se aconseja consultar los ejercicios resueltos situados al final de los temas 1 y 2 de teoría de XML.

El primer paso es comprobar si el archivo XML está bien formado usando un agente de usuario o navegador.

Tarea 2

Construir un DTD que permita validar el documento XML bien formado.

Comprobar la validez con alguna de las herramientas enumeradas en las clases de teoría u otras herramientas encontradas por Internet.

Guía para resolver la tarea 2 del ejercicio 1

Se debe utilizar un editor de texto que permita crear archivos “UTF-8” sin BOM. Por ejemplo Visual Studio Code o NotePad++.

Se aconseja realizar simultáneamente la tarea 1 y la tarea 2.

Los DTD’s tienen limitaciones que les impide comprobar rangos y tipos. Estas comprobaciones se realizarán con los XML Schema.

Las reglas de los DTD pueden ser recursivas y en este ejercicio se aconseja utilizar reglas recursivas.

Se aconseja consultar los ejercicios resueltos situados al final de los temas 1 y 2 de teoría de XML.

Se aconseja comprobar la validez del archivo XML creado con más de una herramienta. En caso de producirse errores se tendrá más información sobre los avisos y errores.

Tarea 3

Representar en SVG el árbol DOM n-ario del archivo XML utilizando el programa **xml2svg.exe** que se encuentra en el Campus Virtual.

Guardar el gráfico en un archivo PDF que deberá ser entregado.

Guía para resolver la tarea 3 del ejercicio 1

El programa **xml2svg.exe** se utiliza en modo consola y los archivos SVG se visualizan con los navegadores. La forma de uso se explica en el tema 1 de teoría “Introducción a XML”.

El objetivo de esta tarea es que el estudiante visualice el árbol n-ario de cualquier archivo XML. Este es el árbol DOM de XML. Es una herramienta didáctica.

Se aconseja tener los archivos *.xml, *.dtd y **xml2svg.exe** en la misma carpeta.

Los archivos “.svg” se pueden visualizar con cualquier agente de usuario o navegador. El navegador Opera tiene la opción de “guardar como PDF”.

Tarea 4

Utilizando Visual Studio 2022 Community Edition generar automáticamente un XML Schema del documento XML bien formado a partir del DTD. Comprobar que el documento XML bien formado con el XSD generado es válido.

Guía para resolver la tarea 4 del ejercicio 1

Visual Studio 2022 Community Edition ya está instalado en los ordenadores del laboratorio 32. Se puede obtener en el siguiente sitio Web:

<https://visualstudio.microsoft.com/es/vs/community/>

Abrir el archivo .dtd con Visual Studio 2022 y se activa el menú XML en Visual Studio. Consultar el tema 3 de teoría “Schemas”. Guardar el archivo .xsd

Tarea 5

Modificar el archivo .xsd generado automáticamente para ajustar los tipos de datos a la riqueza de tipos de XML Schema y comprobar los rangos.

También debe comprobarse que la codificación es “UTF-8”.

Además, pueden hacerse otros tipos de ajuste aprovechando la mayor potencia del XML Schema sobre los DTD.

Comprobar que el documento XML bien formado con el XSD modificado es válido.

Guía para resolver la tarea 5 del ejercicio 1

Las comprobaciones de validez del archivo XML con el XSD pueden hacerse con las herramientas indicadas en el tema 3 de teoría “Schemas”.

Se aconseja comprobar la validez del archivo XML con más de una herramienta. En caso de producirse errores se tendrá más información sobre los avisos y errores.

Tarea 6

Representar en un archivo SVG el árbol n-ario del archivo .xsd utilizando el programa **xml2svg.exe** que se encuentra en el Campus Virtual.

Guardar el gráfico en un archivo PDF que deberá ser entregado.

Guía para resolver la tarea 6 del ejercicio 1

Se recuerda que **xml2svg.exe** se utiliza en modo consola y los archivos SVG se visualizan con los navegadores. La forma de uso está en el tema 1 de teoría “Introducción a XML”.

El objetivo de esta tarea es que el estudiante visualice el árbol n-ario de cualquier archivo XML. Este es el árbol DOM de XML. Es una herramienta didáctica.

Los archivos “.svg” se pueden visualizar con cualquier agente de usuario o navegador. El navegador Opera tiene la opción de “guardar como PDF”.

Tarea 7

Escribir un informe en un documento PDF indicando las principales diferencias entre el XML Schema modificado manualmente y el generado automáticamente.

Guía para resolver la tarea 7 del ejercicio 1

Se deben describir los tipos *string* que se han cambiado a otro tipo de datos. También se deben describir los rangos que se han añadido. También se pueden comentar otros aspectos sobre el Schema desarrollado.

Ejercicio 2: Procesar y generar documentos XML

En este ejercicio se va a realizar el procesamiento y generación de documentos XML, puede ser realizado en cualquier lenguaje de programación (véase Teoría “XPath”, Teoría “XSLT” y “Procesamiento y generación de XML”).

El lenguaje de programación para realizar el ejercicio se deja a elección del estudiante.

Tarea 1

Precondición: tener el documento XML bien formado y válido con la red social del estudiante realizado en el ejercicio 1.

Se debe realizar una transformación automática del archivo XML a un archivo HTML5 utilizando un lenguaje de programación (se deja a elección del estudiante el lenguaje de programación).

El archivo HTML5 debe referenciar a un archivo CSS (construido manualmente).

El archivo HTML5 generado debe mostrar toda la información del documento XML.

El formato de presentación es de libre diseño por parte del estudiante.

El código generado HTML5 y las hojas de estilo CSS deben ser válidas y sin advertencias utilizando los validadores del W3C.

Se debe entregar:

- Archivo XML
- Código fuente del programa que realiza la transformación
- Archivo PDF de documentación indicando:
 - Forma de uso del programa
 - Versión del compilador o intérprete utilizado
 - Si se usa XSLT debe indicarse la herramienta que se ha utilizado.
- Archivo HTML5 generado
- Archivo CSS creado

Guía para resolver la tarea 1 del ejercicio 2

Consultar los ejercicios resueltos de los temas: Teoría “XPath”, Teoría “XSLT” y “Procesamiento y generación de XML”

Tarea 2

Se debe realizar una transformación del archivo XML a un **archivo KML** que debe mostrar los lugares de nacimiento y residencia de las personas de la red social del estudiante cuando se abre en Google Earth.

La transformación puede hacerse con cualquier lenguaje de programación se deja a elección del estudiante.

Se debe entregar:

- Archivo XML
- Código fuente y ejecutable (si lo hubiera) del programa que realiza la transformación.
- Archivo PDF de documentación indicando:
 - Forma de uso del programa
 - Versión del compilador o intérprete utilizado
 - Si se usa XSLT debe indicarse la herramienta que se ha utilizado.
- Archivo KML generado
- Archivo PDF con la captura de la imagen de Google Earth

Guía para resolver la tarea 2 del ejercicio 2

Consultar los ejercicios resueltos de los temas: Teoría “XPath”, Teoría “XSLT” y “Procesamiento y generación de XML”

Tarea 3

Se debe realizar una transformación del archivo XML a un archivo SVG que muestre la red social del estudiante. El diseño se deja libre a cada estudiante.

La transformación puede hacerse con cualquier lenguaje de programación se deja a elección del estudiante.

Se debe entregar:

- Archivo XML
- Código fuente del programa y ejecutable (si lo hubiera) que realiza la transformación.
- Archivo PDF de documentación indicando:
 - Forma de uso del programa
 - Debe indicarse la versión del compilador o intérprete utilizado
 - Si se usa XSLT debe indicarse la herramienta que se ha utilizado.
- Archivo SVG generado
- Archivo PDF con la captura de la imagen del árbol genealógico

Guía para resolver la tarea 3 del ejercicio 2

Consultar los ejercicios resueltos de los temas: Teoría “XPath”, Teoría “XSLT” y “Procesamiento y generación de XML”

Ejercicio 3: Diseñar y construir una aplicación

Diseñar y construir una aplicación cuya entrada es un archivo de un lenguaje derivado de XML. El lenguaje derivado de XML para cada estudiante se asigna en función del último dígito de su identificador de la Universidad de Oviedo (UOXXXXX~~X~~) de la siguiente lista:

- UOXXXXXX terminados en 0: **GPX**
- UOXXXXXX terminados en 1: **TTML**
- UOXXXXXX terminados en 2: **Extensible Application Markup Language (XAML)**
- UOXXXXXX terminados en 3: **ePUB**
- UOXXXXXX terminados en 4: **Ecological Metadata Language (EML)**
- UOXXXXXX terminados en 5: **WaterML**
- UOXXXXXX terminados en 6: **DocBook**
- UOXXXXXX terminados en 7: **XBRL (eXtensible Business Reporting Language)**
- UOXXXXXX terminados en 8: **GXL**
- UOXXXXXX terminados en 9: **X3D**

Características de la aplicación:

- El diseño de la aplicación se deja libre al estudiante
- El lenguaje de programación a utilizar se deja a libre elección del estudiante
- La entrada al programa debe ser obligatoriamente el archivo del lenguaje derivado de XML que le ha correspondido al estudiante
- Se valorará la originalidad y la utilidad de la aplicación desarrollada

Se debe entregar:

- Código fuente y ejecutable (si lo hubiera)
- Archivo PDF indicando el lenguaje de programación utilizado, la versión del compilador o intérprete usada y los archivos de prueba utilizados. También se incluirán las instrucciones de uso.
- Archivos XML de prueba utilizados
- Archivos generados (si los hubiera)

Guía para resolver el ejercicio 3

Consultar los ejercicios resueltos de los temas: Teoría “XPath”, Teoría “XSLT” y “Procesamiento y generación de XML”

Lista de lenguajes derivados de XML:

https://en.wikipedia.org/wiki/List_of_XML_markup_languages

Entrega de la práctica

Se deben entregar **1 archivo** comprimido denominado “**UOXXXXX-practica2.zip**” que contenga **TODOS** los archivos de la práctica (el tamaño máximo de la entrega en el campus virtual es de **90MB**)

Donde el UOXXXXXX es el identificador del estudiante en Uniovi

El archivo empaquetado tendrá una carpeta por ejercicio y dentro de cada ejercicio una carpeta por cada tarea donde se encontrarán todos los archivos solicitados.

- Ejercicio-1
 - Tarea-1 (archivos de la tarea 1)
 - Tarea-2 (archivos de la tarea 2)
 - ...
- Ejercicio-2
 - Tarea-1 (archivos de la tarea 1)
 - Tarea-2 (archivos de la tarea 2)
 - ...
- Ejercicio-3 (archivos del ejercicio 3)

El plazo de entrega se encuentra publicado en el Campus Virtual y las tareas de entrega se habilitarán con anterioridad suficiente.

Se considerarán como “no entregadas” las prácticas que no cumplan el formato de entrega y el nombrado de los archivos y el contenido.

Criterios de evaluación

Los archivos entregados deben cumplir estrictamente las siguientes normas, **en caso contrario la calificación será cero puntos**:

- Es **OBLIGATORIO** presentar **TODOS** los ejercicios y tareas con los nombres y la estructura de directorios y carpetas especificada en el apartado “Entrega de la práctica”
- **TODOS** los archivos XML deben estar bien formados y válidos
- **TODOS** los documentos HTML deben ser HTML5 válidos y sin advertencias utilizando el validador de lenguajes de marcado del W3C
- **TODAS** las hojas de estilo CSS deben ser válidas y sin advertencias utilizando el validador CSS del W3C

IMPORTANTE: Recuerda las pautas de trabajo establecidas en la primera sesión de prácticas (P0. Pautas de trabajo) sobre la validación de los documentos HTML (escritos o generados) y las hojas de estilo CSS.