

Software y estándares para la Web

P2. TECNOLOGÍAS XML

Tabla de contenidos

Objetivos generales.....	2
Ejercicio 1	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.....	4
Tarea 7.....	5
Guía para resolver la tarea 7 del ejercicio 1.....	5
Ejercicio 2	6
Tarea 1.....	6
Guía para resolver la tarea 1 del ejercicio 2.....	6
Tarea 2.....	6
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	8
Guía para resolver el ejercicio 3.....	8
Entrega de la práctica.....	9
Criterios de evaluación.....	9

Objetivos generales

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

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 **el árbol genealógico de cada estudiante con una profundidad que llegue hasta sus bisabuelos**.

Los nodos del árbol genealógico deberán tener los siguientes requisitos mínimos:

- 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
- Fecha de fallecimiento
- Lugar de fallecimiento
- Coordenadas geográficas del lugar de fallecimiento, 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 “Notepad++”, Visual Studio Code (<https://code.visualstudio.com/>), Brackets, ...

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 “Notepad++”, Visual Studio Code (<https://code.visualstudio.com/>), Brackets, ...

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.

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 se entregará.

Guía para resolver la tarea 3 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 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.

Tarea 4

Utilizando Visual Studio 2019 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 2019 Community Edition se puede obtener en el siguiente sitio Web:

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

Abrir el archivo .dtd con Visual Studio 2019 y se activa el menú XML en Visual Studio. Consultar el tema 3 de teoría. 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 XML.

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 se entregará.

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 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

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.

Ejercicio 2

En este ejercicio el objetivo es procesar y generar documentos XML.

El procesamiento y generación de documentos XML se puede hacer en cualquier lenguaje de programación (Teoría “XPath”, Teoría “XSLT” y “Procesamiento y generación de XML”). Se deja la libertad al estudiante la elección del lenguaje de programación.

Tarea 1

Se supone que se tiene el documento XML bien formado y válido con el árbol genealógico del estudiante obtenido en el ejercicio 1.

Se debe realizar una **transformación automática del archivo XML a un archivo HTML5** utilizando un lenguaje de programación (se deja libre al estudiante la elección del 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 debe comprobar la validez de los documentos generados HTML5 y CSS con los validadores del W3C.

Se deben entregar:

- Archivo XML
- Código fuente del programa 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 HTML5 generado
- Archivo CSS

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 defunción del árbol genealógico del estudiante cuando se abre en Google Earth.

La transformación puede hacerse con cualquier lenguaje de programación.

Se deben 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
 - Debe indicarse la 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 muestren el árbol genealógico del estudiante**. El diseño se deja libre a cada estudiante.

La transformación puede hacerse con cualquier lenguaje de programación.

Se deben 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 cuya **entrada es un archivo de un lenguaje derivado de XML**

- El lenguaje derivado de XML será elegido por el estudiante según la terminación de su UOXXXXXX
 - UOXXXXXX terminados en 0: **COLLADA**
 - UOXXXXXX terminados en 1: **CityGML**
 - UOXXXXXX terminados en 2: **Music Markup Language (MML)**
 - UOXXXXXX terminados en 3: **Office Open XML**
 - UOXXXXXX terminados en 4: **Shareable Content Object Reference Model (SCORM)**
 - UOXXXXXX terminados en 5: **VoiceXML (VXML)**
 - UOXXXXXX terminados en 6: **WaterML**
 - UOXXXXXX terminados en 7: **WITSML**
 - UOXXXXXX terminados en 8: **Extensible Application Markup Language (XAML)**
 - UOXXXXXX terminados en 9: **XBRL (eXtensible Business Reporting Language)**
- 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 presentar:

- 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 en el Campus Virtual en **un único archivo comprimido** denominado “UOXXXXX-practica2.zip” que contengan TODOS los archivos de la práctica

- El tamaño máximo de la entrega en el campus virtual es de 90MB
- UOXXXXX es el identificador de cada estudiante en la Universidad de Oviedo.
- El archivo empaquetado tendrá una estructura de directorios denominados Ejercicio-1, Ejercicio-2, etc.
- En su interior deben estar las carpetas Tarea-1, Tarea-2, etc.
- En el interior de estas carpetas deberán estar los archivos solicitados.
- El plazo de entrega se encuentra publicado en el Campus Virtual y las tareas de entrega se habilitarán con anterioridad suficiente.

Criterios de evaluación

Los archivos entregados 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