



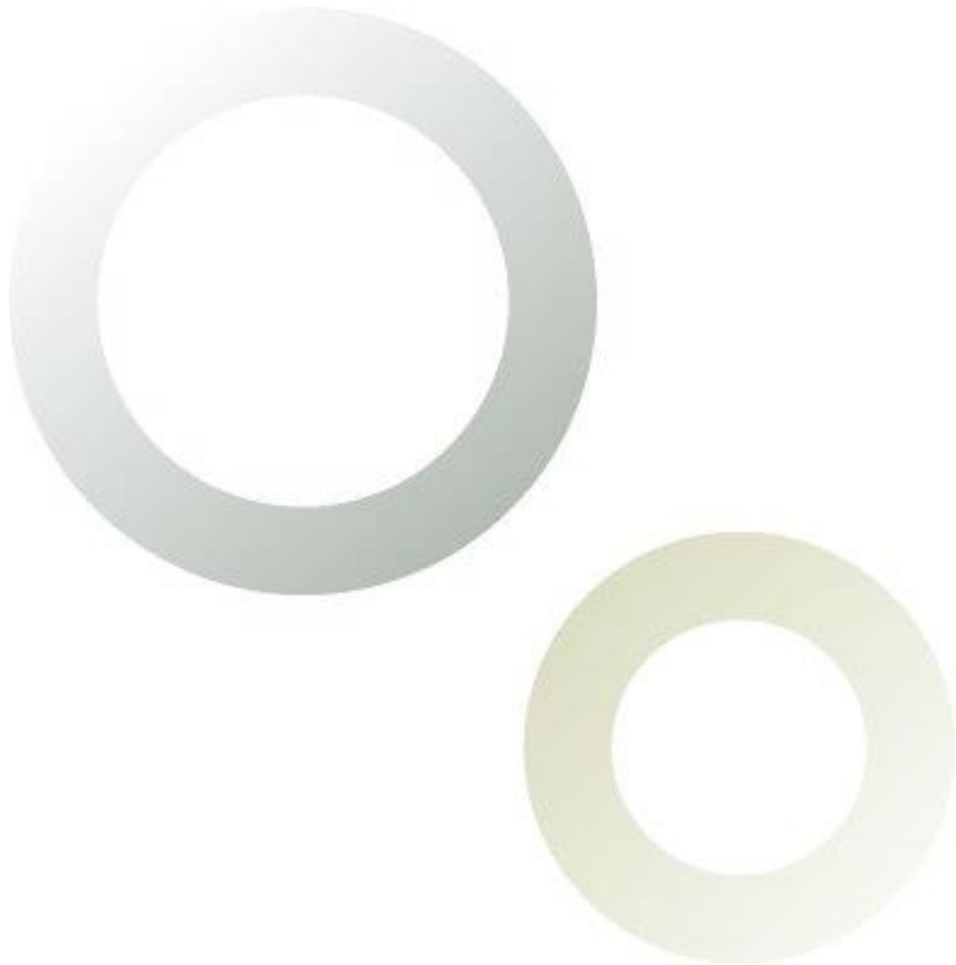
LENGUAJES DE MARCAS

UD6. Tratamiento y creación de archivos XML

Desarrollo de Aplicaciones Web

Profesores: Diana Bautista / Óscar Villar

Índice de contenido



1. Introducción a XML
2. Estructura XML
3. Documentos bien formados
4. Espacios de nombres

1.Introducción a XML

Introducción a XML

XML es un lenguaje de marcado **Extensible Markup Language** o **lenguaje de marcas extensible** basado en SGML (Lenguaje de Marcado Generalizado Standard)

Los archivos XML nos van a permitir **intercambiar información entre distintas aplicaciones, como formato en algunos sistemas de bases de datos, como uso en archivos de configuración y para documentación.**

Las características principales del XML es que es **similar al HTML**, es **extensible** (podemos añadir tantas etiquetas como queramos), su estructura se basa en unas **reglas concisas de construcción**, fácil de implantar en cualquier sistema (**multiplataforma**), **fácil de leer para los humanos y de procesar por los ordenadores.**

Tecnologías basadas en XML: **RSS** (producción de noticias), **ePUB** (formato de libro digital), **SOAP** (comunicación con servicios web), **XHTML** (versión de lenguaje de programación para páginas web)

```
<note>
  <to>pedro</to>
  <from>Manuel</from>
  <heading>Recordatorio</heading>
  <body>No olvides traer las
facturas del mes de abril</body>
</note>
```

Introducción a XML

Tecnologías relacionadas con XML:

- **DTD** y **XML Schema** crean documentos de validación para XML
- **XPath** es lenguaje de consulta para acceder a partes de un XML
- **CSS** y **XSLT** dan estilo a los XML
- **XQuery** permite acceder a los datos del XML como si fuera un base de datos
- **DOM** permite acceder a la estructura jerárquica del documento para utilizarla dentro de un lenguaje de programación

<note>

<to>pedro</to>

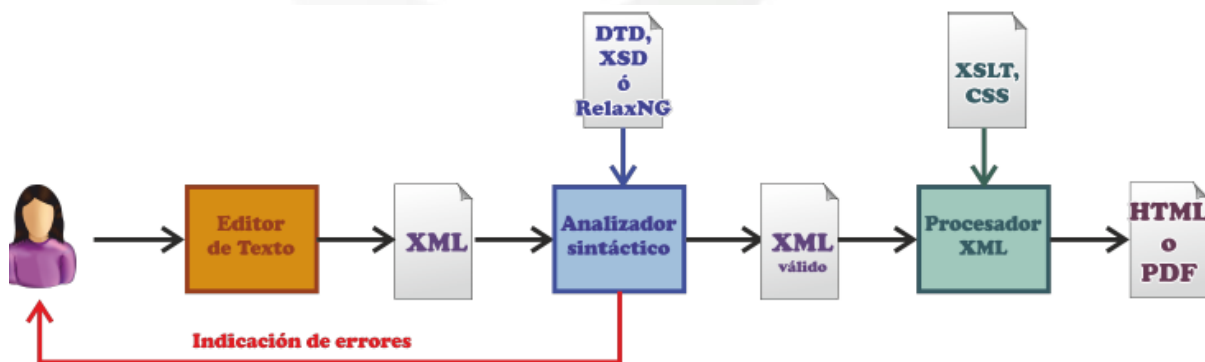
<from>Manuel</from>

<heading>Recordatorio</heading>

<body>No olvides traer las

facturas del mes de abril</body>

</note>



2. Estructura de un XML

```
<?xml version="1.0" encoding="utf-8"?>
<display_options>
  <platform name="*">
    <option name="specified-fonts">true</option>
  </platform>
</display_options>
```

Estructura de un XML

PRÓLOGO

- Primera zona del documento y describe el tipo de documento.
- Declaración del documento
- Instrucciones
- Comentarios
- Indicación del documento DTD,...

```
<?xml version="1.0" encoding="utf-8"?>
<display_options>
  <platform name="*">
    <option name="specified-fonts">true</option>
  </platform>
</display_options>
```

ELEMENTO RAÍZ

- Más elementos
- Atributos
- Texto normal
- Entidades
- Comentarios
- ...

Estructura de un XML

Normas de creación de un XML:

- Nombre de la etiqueta en **minúsculas** entre <>
- Las etiquetas se cierran indicando </ seguido del nombre de la etiqueta que se está cerrando y >
- Los comentarios en el código se inician con los símbolos <!-- terminan con - >
- Es importante la **tabulación** de las etiquetas para una mejor lectura del documento
- Dentro de estos archivos XML tendremos una o varias etiquetas, que a su vez tendrán otras etiquetas, que **podrán estar o no anidadas**.

```
<note>
  <to>pedro</to>
  <from>Manuel</from>
  <heading>Recordatorio</heading>
  <body>No olvides traer las
facturas del mes de abril</body>
</note>
```

Estructura de un XML

- **Declaración del XML:** indica el tipo de documento XML, es opcional pero recomendable, se pone en la primera línea del documento.

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

- **Instrucciones de procesamiento:** sirven para validar el documento, darle estilo u otras funciones:

```
<?xml-stylesheet type="text/xsl" href="stylesheet.xsl"?>  
<!DOCTYPE saludo SYSTEM "saludo.dtd">
```

- **Comentarios:** Como se ha indicado antes comienzan con el símbolo <!-- y terminan con -->. Dentro puede haber cualquier texto que se utiliza con fines explicativos o de documentación del código

- **Elementos:** dan significado al texto. Pueden ser un solo carácter o un carácter, un espacio u otros elementos anidados.

Etiqueta	<nombre></nombre>
Contenido	Juan
Elemento	<nombre>Juan</nombre>

Estructura de un XML

- **Elementos:** dan significado al texto o a otros elementos. Puede contener texto, estar vacío u otros elementos anidados.

SOLO TEXTO

`<descripción>`

Producto con precio rebajado debido a su escasa demanda

`</descripción>`

ELEMENTOS VACÍOS

`<persona>`

`<nombre></nombre>`

`<apellido></apellido>`

`</persona>`

ELEMENTOS ANIDADOS

`<persona>`

`<nombre>pedro</nombre>`

`<apellido>Sánchez</apellido>`

`</persona>`

Estructura de un XML

- **Atributos:**
 - Se indica su nombre seguido del signo = y del valor (entre comillas) que se le da al atributo.
 - Un elemento puede contener varios atributos.
 - Un atributo sirve para **proporcionar información extra** sobre el elemento que lo contiene.
- **Texto:** se puede introducir cualquier texto menos los símbolos <>, &, "'", para poder introducirlos se hace uso de CDATA
- **CDATA:** existe la posibilidad de marcar texto para que no sea procesado como parte de XML.

<![CDATA [texto no procesable...]]>

EJEMPLO 1. ATRIBUTOS

```
<jardin>  
    <plantas categoria="flores" />  
    <plantas categoria="perenne" />  
</jardin>
```

EJEMPLO 2. ATRIBUTOS

```
<producto codigo="G45">  
    <nombre color="negro" precio="12.56">Gorro de lana</nombre>  
</producto>
```

EJEMPLO CDATA

```
<documento>  
    <título>Prueba</título>  
    <ejemplo>  
        <![CDATA[En HTML la negrita se escribe: <strong>]]>  
    </ejemplo>  
</documento>
```

Estructura de un XML

- Atributos vs Elemento

EJEMPLO 1.

```
<persona id="1">  
  <nombre>Adolfo</nombre>  
  <apellidos>Sanz de Diego</apellidos>  
  <direccion>  
    <calle>mi casa</calle>  
    <numero>1</numero>  
    <ciudad>Madrid</ciudad>  
  </direccion>  
</persona>
```

EJEMPLO 2.

```
<persona id="1">  
  <nombre>Adolfo</nombre>  
  <apellidos>Sanz de Diego</apellidos>  
  <direccion calle="mi casa" numero="1"  
ciudad="Madrid"></direccion>  
</persona>
```

Estructura de un XML

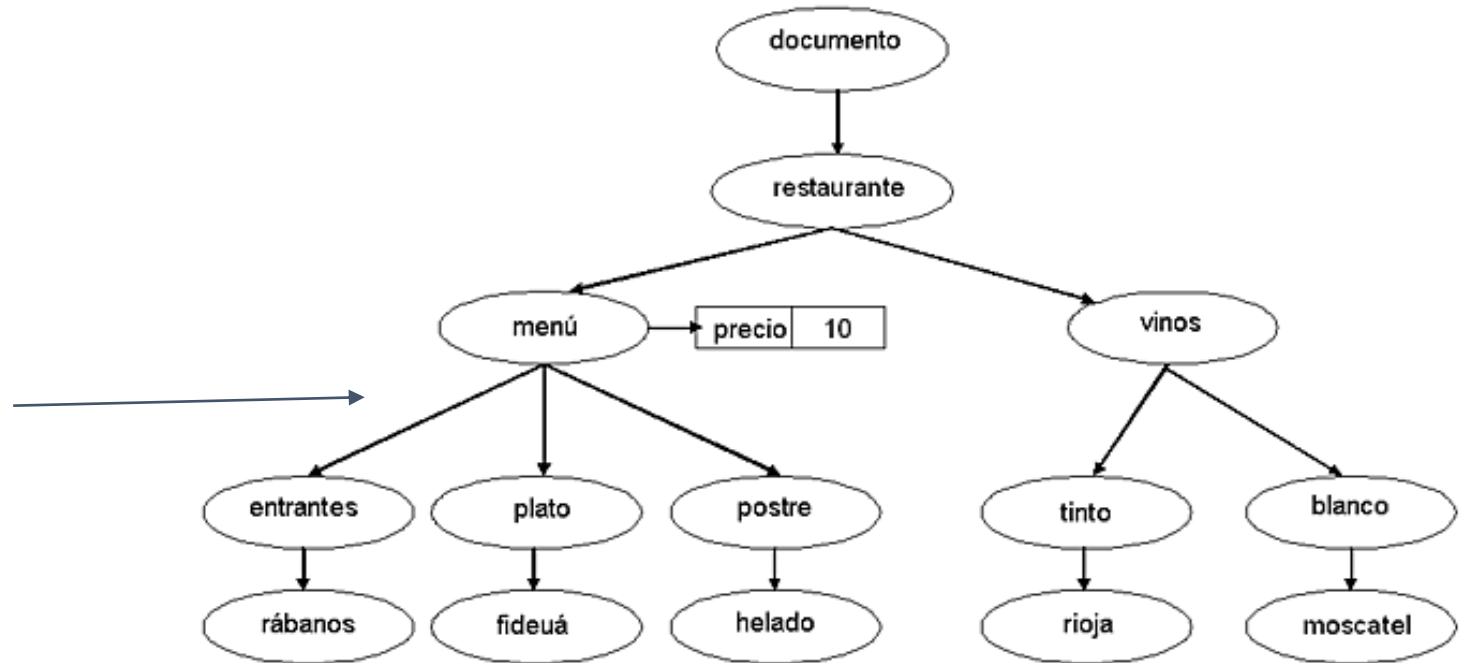
- Jerarquías en XML

- Los elementos de nuestros documentos se ordenan por medio de relación **padre/hijo** o **hermano/hermano**
- A la estructura jerárquica también la podemos llamar **árbol**. Las partes del árbol que contienen hijos se denominan **ramas**, mientras que las partes que no contienen hijos se denominan **hojas**.



Estructura de un XML

```
<?xml version="1.0"?>
<restaurante>
  <menu precio="10">
    <entrantes>rábanos</entrantes>
    <plato>fideuá</plato>
    <postre>helado</postre>
  </menu>
  <vinos>
    <tinto>rioja</tinto>
    <blanco>moscatel</blanco>
  </vinos>
</restaurante>
```



3. Documentos bien formados

Documentos bien formados

- La **validación** de los documentos XML se hace a través del **DTD** o **XSD**
- Un documento bien formado debe cumplir las siguientes normas:
 - Dentro del texto, símbolos **prohibidos**: `<>`, `&`, `"`, `'`. Uso de **CDATA** para solventar el problema
 - Espacios en blanco, tabuladores y saltos de línea no se tienen en cuenta, es decir si el texto tiene 25 espacios seguidos sólo se contará como uno.
 - Elementos anidados correctamente
 - Valores de atributos encerrado en comillas dobles o simples
 - Todo elemento debe contener etiqueta de fin o de elemento vacío
 - El documento sólo debe contener un elemento raíz
 - Todo texto debe de estar incluido en un elemento
 - Los nombres de los elementos no pueden empezar por la palabra reservada XML.
 - Los nombres empiezan por letra, seguidos de más letras, números, guiones o puntos

EJEMPLO BIEN FORMADO

```
<?xml version="1.0" encoding="UTF-8"?>
<identificacion>
  <nombre_completo>
    <nombre> Pepe </nombre>
    <apellido1> González </apellido1>
    <apellido2> Ribera </apellido2>
  </nombre_completo>
  <apodo> Pepito Grillo </apodo>
</identificacion>
```

Documentos bien formados

EN EL NAVEGADOR

Este fichero XML no parece tener ninguna información de estilo asociada. Se muestra debajo el árbol del documento.

```
<identificacion>  
  <nombre_completo>  
    <nombre> Pepe </nombre>  
    <apellido1> González </apellido1>  
    <apellido2> Ribera </apellido2>  
  </nombre_completo>  
  <apodo> Pepito Grillo </apodo>  
</identificacion>
```

EJEMPLO BIEN FORMADO

```
<?xml version="1.0" encoding="UTF-8"?>  
<identificacion>  
  <nombre_completo>  
    <nombre> Pepe </nombre>  
    <apellido1> González </apellido1>  
    <apellido2> Ribera </apellido2>  
  </nombre_completo>  
  <apodo> Pepito Grillo </apodo>  
</identificacion>
```

Documentos bien formados

EJEMPLO **MAL** FORMADO

```
<?xml version="1.0" encoding="UTF-8"?>
<identificacion>
  <nombre_completo>
    <nombre> Pepe </nombre>
    <apellido1> González </apellido1>
    <apellido2> Ribera
  </nombre_completo>
  <apodo> Pepito Grillo </apodo>
</identificacion>
```

FALTA EL CIERRE DE LA ETIQUETA APELLIDO2

EN EL NAVEGADOR

Error de lectura XML: etiqueta sin pareja. Se esperaba: </apellido2>.

Ubicación: file:///D:/CURSO%202020-21/LM/UNIDAD%206.%20TRATAMIENTO%20Y%20CREACI%C3%93N%20DE%20ARCHIVOS%20XML/EJERCICIOS/documento2.xml

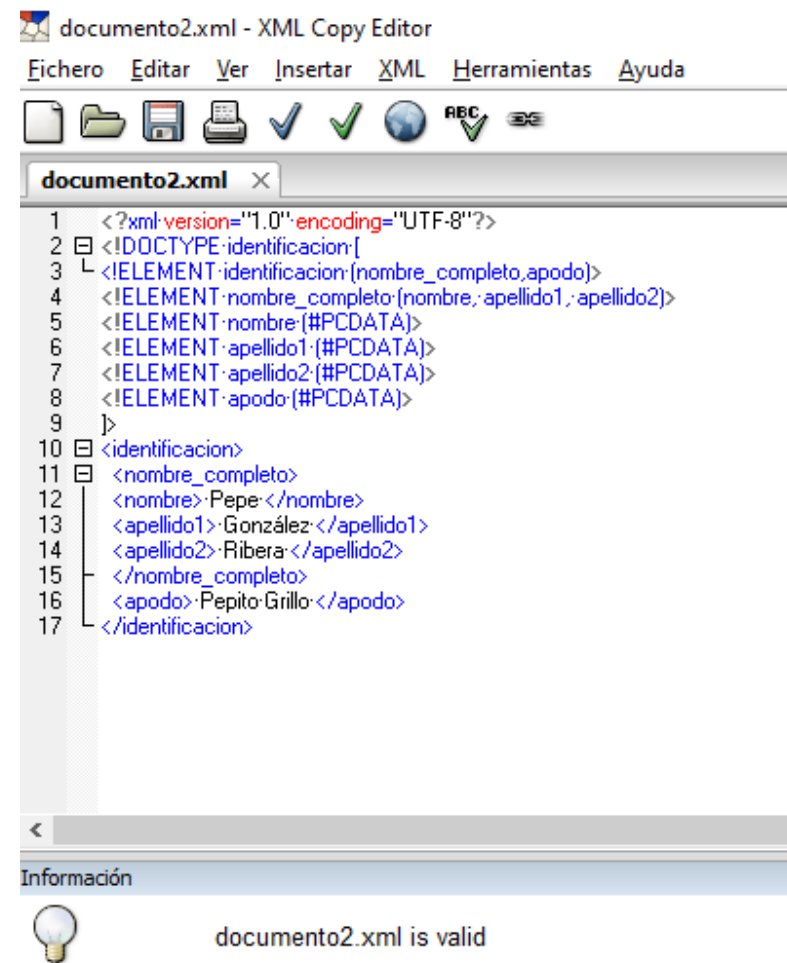
Número de línea 7, columna 7:

```
    </nombre_completo>
    -----^
```

Documentos bien formados

VALIDACIÓN DTD PARA EL EJEMPLO ANTERIOR

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE identificacion [
  <!ELEMENT identificacion (nombre_completo,apodo)>
  <!ELEMENT nombre_completo (nombre, apellido1, apellido2)>
  <!ELEMENT nombre (#PCDATA)>
  <!ELEMENT apellido1 (#PCDATA)>
  <!ELEMENT apellido2 (#PCDATA)>
  <!ELEMENT apodo (#PCDATA)>
]>
<identificacion>
  <nombre_completo>
    <nombre> Pepe </nombre>
    <apellido1> González </apellido1>
    <apellido2> Ribera </apellido2>
  </nombre_completo>
  <apodo> Pepito Grillo </apodo>
</identificacion>
```



Ejercicios

WFD - Ejercicio 1 - Deportistas

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

<deportistas>

  <deportista>

    <deporte Atletismo />

    <nombre>Jesse
Owens</nombre>

  <deportista>

    <deporte Natación />

    <nombre>Mark
Spitz</nombre>

  </deportista>

</deportistas>
```

WFD - Ejercicio 2 - Películas

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

<pelicula>

  <titulo>Con faldas y a lo
loco</titulo>

  <director>Billy Wilder</director>

</pelicula>

<pelicula>

  <director>Leo McCarey</director>

  <titulo>Sopa de ganso</titulo>

</pelicula>

<autor />barto</autor>
```

WFD - Ejercicio 3 - Texto

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

<texto>

  <Titulo>XML explicado a los
niños</titulo>

  <párrafo>El
<abreviatura>XML</abreviatura>define
cómo crear

  lenguajes de marcas.</párrafo>

  <párrafo>Las marcas se añaden a un
documento de texto

  para añadir información.</párrafo>

  <http://>www.example.org</http://>

</texto>
```

WFD - Ejercicio 4 - Información geográfica

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

<geografia mundial>

    <pais>

        <pais>España</pais>

        <continente>Europa</continente>

        <capital></capital nombre="Madrid">

    </pais>

</geografia mundial>
```

WFD - Ejercicio 5 - Programas

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

<programas>

    <programa nombre="Firefox" licencia="GPL" licencia="MPL" />

    <programa nombre="LibreOffice" licencia="LGPL" />

    <programa nombre="Inkscape" licencia="GPL" />

</programas>
```

WFD - Ejercicio 6 - Mundiales de fútbol

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

<mundiales-de-futbol>

    <mundial>

        <pais="España" />

        <1982 />

    </mundial>

</mundiales-de-futbol>
```

WFD - Ejercicio 7 - Medios de transporte

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

<mediosDeTransporte>

    <bicicleta velocidad="v<100km/h" />

    <patinete velocidad maxima="50 km/h" />

</mediosDeTransporte>
```

4. Espacios de nombres

Espacios de nombres

Cuando se manejan diversos documentos XML se puede dar el caso que varios de ellos utilicen las mismas etiquetas, aunque el contexto sería distinto.

PROBLEMA: El analizador, no sabría cómo manejar ambas etiquetas iguales.

Los espacios de nombres (*namespacing* en inglés) evitan el problema indicando en cada elemento una indicación que sirve para indicar el contexto de cada etiqueta y así diferenciar las que son iguales.

```
<?xml version="1.0" encoding="UTF-8"?>
<document>
  <content>
    <html>
      <head>
      </head>
      <body>
        Texto del documento
      </body>
    </html>
  </content>
  <author>Pedro</author>
</document>
  <title>Documento de prueba</title>
  <title>Titulo HTML</title>
```


Espacios de nombres. Diferenciar elementos

```
<?xml version="1.0" encoding="UTF-8"?>
<document>    <content>
    <html>
        <head>
        </head>
        <body>
            Texto del documento
        </body>
    </html>
</content>
    <author>pedro</author>
</document>
<pedro.title>Documento de prueba</pedro.title>
<html.title>Titulo HTML</html.title>
```

Espacios de nombres. Diferenciar elementos

```
<?xml version="1.0" encoding="UTF-8"?>
<www.pedro.net.document>
  <www.pedro.net.title>
    Documento de prueba
  </www.pedro.net.title>
  <www.pedro.net.content>
    <www.w3c.org.html>
      <www.w3c.org.head>
        <www.w3c.org.title>
          Título HTML
        </www.w3c.org.title>
      </www.w3c.org.head>
      <www.w3c.org.body>
        Texto del documento
      </www.w3c.org.body>
    </www.w3c.org.html>
  </www.pedro.net.content>
  <www.pedro.net.author>
    Pedro
  </www.pedro.net.author>
</document>
```

Espacios de nombres. Atributo xmlns

```
<?xml version="1.0" encoding="UTF-8"?>
<document
  xmlns:pedro="http://www.pedro.net/document"
  xmlns:html="http://www.w3c.org/html">
  <pedro:title>
    Documento de prueba
  </pedro:title>
  <pedro:content>
    <html:html>
      <html:head>
        <html:title>Titulo HTML</html:title>
      </html:head>
      <html:body>
        Texto del documento
      </html:body>
    </html:html>
  </pedro:content>
  <pedro:author>
    pedro
  </pedro:author>
</document>
```

Espacios de nombres. Espacios de nombres por defecto

```
<?xml version="1.0" encoding="UTF-8"?>
<document
  xmlns="http://www.pedro.net/document"
  xmlns:html="http://www.w3c.org/html">
  <title>
    Documento de prueba
  </title>
  <content>
    <html:html>
      <html:head>
        <html:title>Titulo HTML</html:title>
      </html:head>
      <html:body>
        Texto del documento
      </html:body>
    </html:html>
  </content>
  <author>
    pedro
  </author>
</document>
```

Espacios de nombres. Espacios de nombres en etiquetas interiores

```
?xml version="1.0" encoding="UTF-8"?>
<document
  xmlns="http://www.pedro.net/document">
  <title>
    Documento de prueba
  </title>
  <content>
    <html:html xmlns:html="http://www.w3c.org/html">
      <html:head>
        <html:title>
          Titulo HTML
        </html:title>
      </html:head>
      <html:body>
        Texto del documento
      </html:body>
    </html:html>
  </content>
  <author>
    pedro
  </author>
</document>
```

Espacios de nombres. Varios espacios de nombres en etiquetas interiores

```
<?xml version="1.0" encoding="UTF-8"?>
<document xmlns="http://www.pedros.net/document">
  <!-- comienza el espacio de nombres de pedro.net -->
  <title>
    Documento de prueba
  </title>
  <content>
    <html xmlns="http://www.w3c.org/html">
      <!-- desde aquí el espacio ahora es el de html -->
      <head>
        <title>
          Titulo HTML
        </title>
      </head>
      <body>
        Texto del documento
      </body>
    </html>
  <!-- fin del espacio html, regresa el espacio pedro.net -->
  </content>
  <author>
    Pedro
  </author>
</document>
```

Enlaces de interés

1. [W3 Schools - DTD Tutorial](#)
2. [W3Schools XML Tutorial](#)
3. [XML Copy Editor](#)

Bibliografía

Todo el documento ha sido generado a partir de la información contenida en este enlace: <http://pedro.net/#lmsgi>

