XML

Contents

[1 Introduction 1](#_Toc125805778)

[2 Syntax & Properties 2](#_Toc125805779)

[3 Programming Support 3](#_Toc125805780)

[4 MS Word, Excel, etc are Stored in XML 3](#_Toc125805781)

[5 Expectations 4](#_Toc125805782)

[Appendix 1 W3Schools Tutorial 5](#_Toc125805783)

[Appendix 2 Perspective 5](#_Toc125805784)

# Introduction

XML (Extensible Markup Language):

* A text-based markup language.
* You define your own tags.
* Is self-descriptive
* Designed to transport, organize, and store data.
* Stored in plain text so reduces the complexity of exchanging data between incompatible systems
* Currently, mostly used for B2B transactions and enterprise level applications. For smaller things, has been replaced with JSON.
* Most browsers will display an XML file. Example: <https://www.w3schools.com/xml/cd_catalog.xml>
* A pubic standard maintained by the World Wide Web Consortium (W3C)

Example XML document:

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

<bookstore>

<book category="children">

<title>Harry Potter</title>

<author>J K. Rowling</author>

<year>2005</year>

<price>29.99</price>

</book>

<book category="web">

<title>Learning XML</title>

<author>Erik T. Ray</author>

<year>2003</year>

<price>39.95</price>

</book>

</bookstore>

XML is used to create new languages:

* [Android XML](https://developer.android.com/guide/topics/ui/declaring-layout) – User interface design
* [FXML](https://en.wikipedia.org/wiki/FXML) – User interface design
* [LegalXML](http://www.legalxml.org/), [wikipedia](https://en.wikipedia.org/wiki/Legal_XML)
* [MathML](https://en.wikipedia.org/wiki/MathML)
* [Others](http://en.wikipedia.org/wiki/List_of_XML_markup_languages)

# Syntax & Properties

|  |  |
| --- | --- |
| 1. A *well-formed* XML document:
2. Must have a root element
3. Elements must have a closing tag
4. Tags are case sensitive
5. Elements must be properly nested
6. Attribute values must be quoted
7. XML Element
8. Can contain other elements
9. Can contain text
10. Can contain attributes – attributes should be meta-data
 | <root>  <child>    <subchild>.....</subchild>  </child>  <child>    <subchild>.....</subchild>  </child>...</root> |

1. A *valid* XML document
2. Must be well-formed
3. Conforms to a DTD (Document Type Definition) or Schema
4. Example:

<?xml version="1.0"?>
<!DOCTYPE note [
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
]>

<note>

<to>Tove</to>

<from>Jani</from>

<heading>Reminder</heading>

<body>Don't forget me this weekend</body>

</note>

1. DTD
2. With a DTD, independent groups of people can agree to use a standard DTD for interchanging data.
3. With a DTD, you can verify that the data you receive from the outside world is valid.

Resources:

* <https://www.w3schools.com/xml/xml_dtd.asp>
* <https://www.tutorialspoint.com/xml/xml_dtds.htm>

# Programming Support

1. XML DOM - defines a standard way for accessing and manipulating XML documents. It presents an XML document as a tree-structure. Also: XPath, XQuery, XSLT.
2. All major programming languages have support for XML and XML DOM including Java, C++, Python, C#. Two resources for Java:
* <https://docs.oracle.com/javase/tutorial/jaxp/index.html>
* <https://www.baeldung.com/java-xml>
1. XSLT (Extensible Stylesheet Language Transformations) – used to display XML in HTML. Can be done on the client or server. Resources:
* <https://www.tutorialspoint.com/xslt/index.htm>
* <https://www.w3schools.com/xml/xsl_intro.asp>

|  |  |
| --- | --- |
| XML | XSLT |
| <?xml version="1.0" encoding="UTF-8"?><catalog> <cd> <title>Empire Burlesque</title> <artist>Bob Dylan</artist> <country>USA</country> <company>Columbia</company> <price>10.90</price> <year>1985</year> </cd> <cd> <title>Hide your heart</title> <artist>Bonnie Tyler</artist> <country>UK</country> <company>CBS Records</company> <price>9.90</price> <year>1988</year> </cd></catalog> | <?xml version="1.0" encoding="UTF-8"?><xsl:stylesheet version="1.0"xmlns:xsl="http://www.w3.org/1999/XSL/Transform"><xsl:template match="/"> <html> <body> <h2>My CD Collection</h2> <table border="1"> <tr bgcolor="#9acd32"> <th>Title</th> <th>Artist</th> <th>Price</th>  </tr> <xsl:for-each select="catalog/cd"> <xsl:if test="price>10"> <tr> <td><xsl:value-of select="title"/></td> <td><xsl:value-of select="artist"/></td> <td><xsl:value-of select="price"/></td>  </tr> </xsl:if> </xsl:for-each> </table> </body> </html></xsl:template></xsl:stylesheet> |

# MS Word, Excel, etc are Stored in XML

1. To see the XML: Change the extension of a word file from .docx to .zip and then upzip.
* Open the *word* folder.
* Open *document.xml.*
1. An example is a file I created with the contents: “Hello World in XML!”. This is the file:

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

<w:document xmlns:wpc="http://schemas.microsoft.com/office/word/2010/wordprocessingCanvas" xmlns:cx="http://schemas.microsoft.com/office/drawing/2014/chartex" … >

<w:body>

 <w:p w14:paraId="6E88ED03" w14:textId="554D7B7D" w:rsidR="008B20DB" w:rsidRDefault="00D40AB7">

 <w:r>

 <w:t>Hello World in XML!</w:t>

 </w:r>

 <w:bookmarkStart w:id="0" w:name="\_GoBack"/><w:bookmarkEnd w:id="0"/>

 </w:p>

 <w:sectPr w:rsidR="008B20DB"><w:pgSz w:w="12240" w:h="15840"/>

 <w:pgMar w:top="1440" w:right="1440" w:bottom="1440" w:left="1440" w:header="720" w:footer="720" w:gutter="0"/>

 <w:cols w:space="720"/>

 <w:docGrid w:linePitch="360"/>

 </w:sectPr>

</w:body>

</w:document>

# Expectations

These are expectations I have of you for testing:

1. What does the acronym XML stand for?
2. What is XML used for?
3. Describe what a well-formed XML document is and what a valid xml document is.
4. What is a DTD used for?
5. Write a well-formed document

 Appendix

1. W3Schools Tutorial

I have listed the chapters and highlighted in blue that are useful (all of them).

|  |
| --- |
| [XML Tutorial](http://www.w3schools.com/xml/default.asp)1. XML HOME
2. XML Introduction
3. XML How to use
4. XML Tree
5. XML Syntax
6. XML Elements
7. XML Attributes
8. XML Display

…1. XML XSLT (this is short, links to a full tutorial). You will need this for Requirement 10

…1. XML Validator
2. XML Schema
3. XML DTD
4. XML CSS
 |

1. Perspective

[A brief history of XML - From hype to useful data format](https://sonra.io/2016/10/18/brief-history-xml-hype-useful-data-format/)

[A Library for XML Data Standards](https://sonra.io/2017/11/03/library-xml-data-standards/)

[In Defense of XML](https://blog.frankel.ch/defense-xml/)