

XML for Developers teaches developers how to create Extensible Markup Language (XML) data structures. It also teaches the student how to design and code both Document Type Declarations (DTDs) and XML Schemas. This course consists of a printed set of Microsoft PowerPoints.

Audience

- Application developers and anyone else who needs a working knowledge of XML.

Prerequisites

- It would be helpful if the student has some knowledge of data files and parsing techniques

Course Length

- Three Days

Teaching Methods

- Lecture
- Hands-on examples and exercises

Learning Objectives

- What is XML
- History of XML
- XML Rules, Details and Examples
- Valid vs. Well-Formed
- Document Type Definitions (DTD)
- XML Schema
- Namespaces
- Schemas Part 1
- Schemas Part 2
- Schemas Part 3

Course Outline

HL9

What is XML?

- Meta-language
- eXtensible
- Human-readable, Application specific
- Good for Structured Data, and “narratives”

XML Is Not

- A Programming Language (quasi-exceptions)
- A Single Language
- Presentation-oriented
- Most Efficient (usually)
- Good for Large Volumes of “bitstream” Data (e.g. multimedia)

History of XML

- Derived from SGML
- Related, but Different Than HTML
- HTML is NOT Legal XML
- XHTML
- Custom *ML’s:
- MathML, CML, WML, SVG, etc.

XML Rules, Details And Examples

- XML Declarations
- XML Prolog
- XML Elements
 - Matching Case-sensitive Tags
 - One Root
 - Nested, but Not Overlapping
 - Markup vs. Character
 - White Space, Entity References
- Attributes
- Processing Instructions
- DTD Specification
 - Embedded DTDs
 - Referenced DTDs
- Handling Character Sets

Valid Versus Well-Formed

- Well-Formed Documents
- Valid Documents

DTD

- “SYSTEM” versus “PUBLIC” DTDs
- Elements

- Data Type
- Sequence, Order, Number of Children
- Attributes
 - #REQUIRED, #IMPLIED, #FIXED,
 - ID, IDREF, ENTITY, NMTOKEN
 - Conditionals

XML Schema

- DTDs are Completely Different Syntax Than XML
- XML Schema Is an XML Replacement for DTD

Namespaces

- Why Namespaces?
 - Multiple XML Vocabularies Conflict
 - XSLT
 - Xlinks
- Qualified Name vs. Raw Name
 - Prefix
 - Local Part
- Default Namespace

Schemas Part 1

- Problems With DTDs
- Schema Requirements
- Schemas Versus DTDs
- Schema Primary Components
- Schema Secondary Components
- Schema Helper Components
- Simple Versus Complex Types
- Primitive Versus Derived Types
- Simple Types
- Primitive Types
- Derived Types
- Schema Namespace
- Element Declarations
- A Simple Schema
- Complex Types

- Complex Type Example
- Using A Complex Type

Schemas Part 2

- Complex Type Example
- Complex Types
- Sequence
- MINOCCURS And MAXOCCURS
- Sequence Example
- Choice
- Choice Example
- All
- All Example
- Mixed Content
- Mixed Content Example
- Attributes
- Attribute Example

Schemas Part 3

- Derived Types
- Deriving Simple Types
- Facets
- Integer Range Example
- Enumeration Example
- Pattern Facet
- Pattern Regular Expressions
- Regular Expression Qualifier
- Regular Expression Escape Sequences
- Regular Expression Examples
- Pattern
- Pattern Example
- Deriving Complex Types
- Restricting Simple Content
- Extending Simple Content
- Restricting Complex Content
- Extending Complex Content
- Namespaces
- Namespaces Example