

**Mastering XSLT**

**Course Number:** XML-120  
**Duration:** 3 days

**Overview**

XSLT (Extensible Stylesheet Language Transformations) enables you to transform XML data from one document type into other XML document types, and even into other formats such as HTML, XHTML, WML, and XQL.

**Prerequisites**

Prior knowledge of XML, as taught in Accelebrate's Core XML training class, is required.

**Materials**

All XSLT training students receive comprehensive courseware.

**Software Needed on Each Student PC**

* Appropriate software for performing XSLT - we will work with you to select this based on your available tools and developers' experience.

**Objectives**

* Master how to use XSLT and XPath to transform XML documents into XHTML, other XML document formats, and plain text
* Learn how to automate XSLT

**Outline**

* XSLT Basics
  + eXtensible Stylesheet Language
  + An XSLT Stylesheet
    - xsl:template
    - xsl:value-of
    - Whitespace and xsl:text
  + Output Types
    - Text
    - XML
    - HTML
  + XSLT Elements and Attributes
    - xsl:element
    - xsl:attribute
    - Attributes and Curly Brackets
* XPath
  + XPath in XSLT
  + XPath Expression
  + XPath Terminology
    - Context Node
    - Current Node
    - Context Size
    - Proximity Position
  + Location Paths
  + Axis
  + Node Test
  + Predicate
  + Accessing Nodes
  + Abbreviated Syntax
  + XPath Functions
  + XPath Operators
* Flow Control in XSLT
  + Looping in XSLT
  + Sorting with XSLT
  + Looping and Sorting
  + Conditions with XSLT
* XSLT Templates, Parameters and Variables
  + xsl:apply-templates
  + xsl:call-template
  + Passing Parameters
  + Removing Content
  + Template Modes
  + Template Priority
    - Default Priorities
    - Assigning Priorities
  + XSLT Variables
* Multiple XML and XSLT Documents
  + Including XSLTs
  + Importing XSLTs
  + Conflict resolution
  + The document() Function
* Working with Keys
  + Key Basics
    - The key() Function
    - Improving Performance with Keys
    - Grouping
* Advanced XSLT Techniques
  + Working with Numbered Lists
    - The position() function
    - xsl:number
  + Outputting Processing Instructions
  + Copying Nodes
    - xsl:copy
    - xsl:copy-of
* Conclusion