

**Mastering XSLT 2.0 and XPath 2.0**

**Course Number:** XML-122
**Duration:** 3 days

**Overview**

Accelebrate's XSLT 2.0 training class teaches developers how to use XPath 2.0 and XSLT 2.0 to transform XML documents into other XML document formats, XHTML, and plain text.

**Prerequisites**

Prior knowledge of core XML tagging syntax, as taught on the first day of Accelebrate's Core XML training class or acquired through experience, is required. Prior knowledge of XML Schema is helpful, but we can adapt the class to any level of prior XML Schema familiarity.

**Materials**

All XSLT training students receive comprehensive courseware.

**Software Needed on Each Student PC**

* The XML/XSLT editing and transformation environment that your developers will be using after class. If none has been selected, we default to using XMLSpy, for which we can provide an academic license for the duration of the training.

**Objectives**

* Use XPath 2.0 to access XML elements and attributes
* Loop through XML elements
* Use conditionals in XSLT 2.0
* Create and call templates
* Work with multiple XSLTs
* Reference external XML documents via document()
* Master grouping results with xsl:group (similar to grouping in SQL queries)
* Build user-defined functions with xsl:function
* Specify data relationships with keys
* Work with namespaces
* Apply advanced XSLT

**Outline**

* XSLT Basics
	+ eXtensible Stylesheet Language
	+ An XSLT Stylesheet
		- xsl:template
		- xsl:value-of
		- Whitespace and xsl:text
	+ Output Types
		- Text
		- XML
		- HTML and XHTML
	+ XSLT Elements and Attributes
		- xsl:element
		- xsl:attribute
		- Attributes and Curly Brackets
* XPath Path Expressions
	+ 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 Language Basics
	+ XPath Functions
	+ XPath Operators
	+ XPath Data Types
	+ XPath Expressions
		- for Expressions
		- if-then-else Expressions
* 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
	+ The collection() Function
	+ Multiple Result Documents
* Grouping
	+ Grouping by Value
	+ Grouping Adjacent Elements
	+ Grouping by Starting and Ending Elements
* Regular Expressions
	+ Regular Expression Syntax in XSLT
	+ The xsl:analyze-string Element
	+ Built-in Functions that Use Regular Expressions
* User-Defined Functions
	+ The xsl:function Element
	+ Function Names and Parameters
	+ Calling User-Defined Functions
* Advanced XSLT Techniques
	+ Working with Namespaces
		- What Are Namespaces?
		- Declaring Namespaces in XSLT
		- Namespace Declarations and XPath Expressions
	+ Working with Keys
		- Key Basics
		- The generate-id() Function
	+ Working with Numbered Lists
		- The position() function
		- xsl:number
	+ Outputting Processing Instructions
	+ Copying Nodes
		- xsl:copy
		- xsl:copy-of
* Tying It All Together
	+ Workshop: Sharing data and transforming it for the Web
* Conclusion