**New Perspectives XML Comprehensive, 3e**

**Tutorial 6: Functional Programming with XSLT and XPath 1.0**

**Key Terms**

**Base case** The stopping condition used with recursion. (XML 371)

**Change of state**  The change that is applied each time a function calls itself recursively. (XML 371)

**Concat()** An XPath function used to combine two or more substrings into a single text string. (XML 356)

**Contains()** An XPath function used to determine whether a text string contains a specified substring. (XML 355)

**Count()** An XPath function used to calculate the count of numeric values. (XML 342)

**Current()** An XPath function to return the current node being processed. (XML 342)

**Deep copy** A copy that includes the active node and any descendant nodes and attributes. (XML 332)

**Div** An XPath keyword used to divide one quantity by another. (XML 342)

**Doc()** An XPath 2.0 function used to access the contents of a source document. (XML 335)

**Doc-available()** An XPath 2.0 function used to determine whether an external document is available. (XML 338)

**Document()** An XSLT function used to access the contents of an XML source document. (XML 324)

**Double precision floating point** A data format in which values are stored using eight bytes of storage in order to achieve greater precision with calculations. (XML 351)

**Format-number()** An XPath function used to format the appearance of a numeric value. (XML 342)

**Functional programming** A programming technique that relies on the evaluation of functions and expression. (XML 369)

**Global parameter** A parameter with global scope that can be accessed anywhere within the style sheet. (XML 364)

**Global scope** Scope that encompasses the entire style sheet. (XML 327)

**Global variable** A variable with global scope. (XML 328)

**Identity template** A template that matches any kind of node in the source document and copies them to the result document. (XML 332)

**Local parameter** A parameter with local scope, limited to the template in which it is defined. *See* *also* template parameter. (XML 364)

**Local scope** Scope that is limited to the template or the element in which the variable is declared. (XML 328)

**Local variable** A variable with local scope. (XML 329)

**Lookup table** A collection of data values that can be searched in order to return data that matches a supplied key value. (XML 344)

**Named template** A template that is not matched to a node set but instead acts like a function to calculate a value or perform an operation. (XML 362)

**Normalize-space()** An XPath function used to trim leading and trailing white space characters from a text string. (XML 360)

**Parameter** A variable whose value can be passed to it from outside of its scope. (XML 362)

**Position()** An XPath function that returns the position of individual nodes within the node set. (XML 348)

**Preserve-space** An XSLT element used to ensure that white space nodes are not deleted in the result document. (XML 360)

**Recursion** A process by which a function calls itself until a stopping condition is met. (XML 371)

**Recursive template** A template that continually calls itself until a specified stopping condition or base case is reached. (XML 362)

**Result tree fragment** A well-formed code that is written to a result document. (XML 326)

**Scope** The locations within a style sheet from which a variable can be referenced. (XML 327)

**Shallow copy** A copy limited only to the active node and does not include any child nodes. (XML 332)

**Substring** A section of characters from a text string. (XML 355)

**Substring()** An XPath function used to extract a substring from a larger text string. (XML 355)

**Sum()** An XPath function used to calculate the sum of numeric values. (XML 342)

**Template parameter** A parameter defined and referenced from within a template. *See* *also* local parameter. (XML 362)

**Unparsed-text()** An XPath 2.0 function used access the contents of non-XML documents. (XML 335)

**Unparsed-text-available()** An XPath 2.0 function used to determine whether an external non-XML document is available. (XML 338)

**Variable** A user-defined named that stores a value or an object. (XML 324)

**White space node** Nodes that contain only white space characters such as blanks, line returns, or tabs. (XML 360)

**Xsl:call-template** An XSLT element used to call and execute the commands contained within a named template. (XML 362)

**Xsl:copy** XSLT element used to shallow copy a node from the source document. (XML 332)

**Xsl:copy-of** An XSLT element used to deep copy a node from the source document. (XML 325)

**Xsl:decimal-format** An XSLT format used to define numbering schemes for use with international formats. (XML 352)

**Xsl:import** An XSLT element used to import the contents of a style sheet. (XML 339)

**Xsl:include** An XSLT element that includes the contents of an external style sheet in the current style sheet. (XML 324)

**Xsl:number** An XSLT element that returns the position of nodes as they appear in the source document. (XML 349)

**Xsl:strip-space** An XSLT element used to remove extraneous white space nodes from a source document. (XML 360)

**Xsl:text** An XSLT element used to insert a text node into the result document. (XML 362)

**Xsl:variable** An XSLT element that creates a variable in the style sheet. (XML 324)

**Xsl:with-param** An XSLT element that sets the value of a template parameter defined within a named template. (XML 362)