

# Index

COPYRIGHTED MATERIAL



# Index

- (double-dash string), 46
- (minus sign), 10
- # (hash mark), 133
- ##any value, 164, 181
- ##local value, 164, 181
- ##other value, 164, 181
- ##targetNamespace value, 164, 181
- #FIXED keyword, 126
- #IMPLIED keyword, 120
- #PCDATA keyword, 108, 138
- #REQUIRED keyword, 122, 127, 884
- \$ variableName notation, 321
- \$a variable, 369
- \$b variable, 369
- \$book variable, 368
- \$i variable, 363, 365
- \$input variable, 370
- \$lower variable, 278
- \$p variable, 423
- \$ParameterName syntax, 320
- \$searchLetter param, 321
- \$text variable, 278
- \$VariableName syntax, 320
- % (percent sign), 139, 140
- & (ampersand) character, 132, 135, 140, 599, 905
- & character, 59, 60, 444, 756
- &amp; character, 444
- &amp; entity, 131
- &apos; entity, 132
- &gt; entity, 131
- &lt; entity, 131
- &quot; entity, 132
- \* (asterisk), 891
- \* cardinality indicator, 109, 217
- \* indicator, 114, 116
- \* node test, 924
- /\* /text ( ) method, 658
- \/\)/ argument, 796
- // Implementation comment, 855
- //element ( ) function, 284
- //element(\*, xs:token) function, 284
- @ character, 262
- @ symbol, 406
- @BasketPointer variable, 412
- @BasketXml variable, 412
- @number location path, 890
- @type attribute, 827
- @XmlAttribute variable, 410
- [ character, 101
- <![CDATA[ ]> declaration, 866
- [none] indicator, 114
- ] character, 101
- ^\d+\$ expression, 282
- \_ (underscore), 122, 135
- \_\_str\_\_ method, Python, 547
- \_query parameter, 390
- \_xsl parameter, 393
- { } (curly braces), 72, 332, 343, 359, 540

- | (vertical bar) character, 108, 125
- + cardinality indicator, 217
- + indicator, 114
- + symbol, 729
- < (opening) character, 34
- < character, 24, 59, 756, 880
- <!-- string, 46
- = (equals sign) references, 34, 228
- = assign method, 230
- = character, 355
- > character, 756
- > string, 46
- : (colon) character, 34, 490
- (dash) character, 34
- ! (exclamation mark), 99
- (hyphens), 135
- ; (semicolon), 132, 133, 135, 140
- <? ?> notation, 56
- ? cardinality indicator, 217
- / character, 347
- ? indicator, 114

## A

- A command, 778
- <a> tags, 721
- <abbr> element, 764
- abort( ) method, 672
- absolute location path, 299
- absolute positioning, 706, 714–716, 718
- absolute units, 774
- abstract modules, XHTML, 760
- Accept header, 580
- Accept-Encoding header, 580
- Accept-Language header, 580
- accesskey attribute
  - xforms:input element, 818
  - xforms:range element, 819
  - xforms:secret element, 819
  - xforms:select element, 822
  - xforms:select1 element, 822
  - xforms:trigger element, 820
  - xforms:upload element, 819
- AchieveForms, 809
- Acrobat Reader, 837
- <acronym> element, 764
- action attribute, 814, 845
- Action Module, 835–836
- Active Server Pages (ASP), 295
- ActiveX component, 450
- ActiveXObject( ) function, 588, 647
- actualText public member, 672
- actualText variable, 669
- actuate attribute, 724, 725
- add( ) method, 474
- addNode( ) method, 465
- <address> element, 107, 116, 134, 226
- address-unknown entity, 138
- AddToCart procedure, 614, 624
- Adjacent sibling selector, 701
- Adobe Acrobat Reader, 837
- Adobe GoLive, 770
- Adobe Illustrator, 770
- Adobe LiveCycle, 837
- Adobe SVG Viewer, 448
- Adobe SVG Zone, 800
- ADO.NET, 380
- AEIfred2 parser, 485
- after clause, 419
- :after pseudo-element, 726, 729, 731
- aggregators
  - implementation
    - address list reader, 541–542
    - application controller, 543–546
    - extending, 555–557
    - handler helpers, 552–555
    - model, 546–548
    - XML markup handler, 548–552
  - modeling feeds
    - different syntaxes, 538–539
    - Simple API for XML (SAX), 539
  - program flow, 540
  - transforming RSS with XSLT
    - browser processing, 564–565
    - generating a feed from existing data, 558–559
    - pre-processing feed data, 565
    - processing feed data for display, 561–563
    - reviewing the different formats, 565–567
- Ajax (Asynchronous JavaScript and XML)
  - asynchronous updates
    - cross-browser solutions, 647–649
    - Microsoft versus Mozilla, 647
  - communicating with proxy web service using, 858–864

- posting techniques, 649–652
- processing on server
  - AutoSuggest box, 658–681
  - JavaScript Object Notation (JSON), 652–653
  - payment card validator, 653–658
  - server-side proxies, 681–686
  - transport on server
    - AutoSuggest box, 658–681
    - JavaScript Object Notation (JSON), 652–653
    - payment card validator, 653–658
- Ajax technologies, 808**
- alert( ) **function, 15, 450, 919**
- alien **command, Debian, 427**
- align **attribute, 913, 914**
- <all> **declarations, 168, 171–174**
- Alternate **attribute, 700**
- American Standard Code for Information Interchange (ASCII), 52**
- <amount> **element, 218**
- ampersand (&) character, 132, 135, 140, 599, 905**
- ancestor **axis, 260, 264–265, 271, 348, 925**
- ancestor **element node, 265**
- ancestor-or-self **axis, 260, 265, 348, 925**
- <annotation> **declaration, 208, 209**
- annotations**
  - Scalable Vector Graphics (SVG), 782–784
  - XML Schemas, 208–209
- annual percentage rate (APR), 841**
- anonymous boxes, 705–706**
- anonymous complex types, 165**
- <AnotherTag> **element, 38**
- any content, 110**
- ANY **keyword, 110**
- any **namespace, 208**
- <any> **declaration, 163**
- <anyAttribute> **declaration, 180**
- anyElement **children, 240**
- AnyName **name-class feature, 240–241**
- anyURI **type, 189**
- Apache Batik, 770, 771, 800**
- Apache Xerces, 24**
- API (application programming interface), 483, 576**
- APP (Atom Publishing Protocol) interface, 394, 525**
- App\_Code **folder file, 683**
- appearance **attribute**
  - xforms:input element, 817
  - xforms:range element, 819
  - xforms:secret element, 819
  - xforms:select1 element, 822
  - xforms:trigger element, 820
  - xforms:upload element, 819
- appendChild( ) **method, 444, 461, 462, 478**
- <appinfo> **element, 208, 209**
- Applet module, XHTML, 760**
- application programming interface (API), 483, 576**
- applications, Scalable Vector Graphics (SVG), 769**
- application/xhtml+xml **mime type, 746, 747, 756, 757, 758**
- apply-templates instruction, 253**
- APR (annual percentage rate), 841**
- Arial typeface, 699**
- <article> **element, 699**
- as **attribute, 333**
- ASaying.xquery **query, 369**
- ASCII (American Standard Code for Information Interchange), 52**
- ASCII 7-bit character encoding, 52**
- ASP (Active Server Pages), 295**
- ASP.NET page, 609**
- asterisk (\*), 891**
- Asynchronous JavaScript and XML (Ajax)**
  - asynchronous updates
    - cross-browser solutions, 647–649
    - Microsoft versus Mozilla, 647
  - communicating with proxy web service using, 858–864
  - posting techniques, 649–652
  - processing on server
    - AutoSuggest box, 658–681
    - JavaScript Object Notation (JSON), 652–653
    - payment card validator, 653–658
    - server-side proxies, 681–686
    - transport on server
      - AutoSuggest box, 658–681
      - JavaScript Object Notation (JSON), 652–653
      - payment card validator, 653–658
- asynchronous updates**
  - Asynchronous JavaScript and XML (Ajax), 645–686
  - early attempts
    - cross-browser solutions, 647–649
    - Microsoft and Mozilla, 647
- Atom, 533–535**
- atom prefix, 538**
- Atom Publishing Protocol (APP) interface, 394, 525**
- Atom site, 567**

## Atom Syndication Format (RFC), 533

Atom Wiki site, 567

atomic value, 361

ATTLIST declaration, 120, 121, 127, 130, 134, 140, 141

Attr node, 480

attr property, 910

attribute axis, 259, 260, 261, 262–264, 890, 925

### attribute declarations

attribute names, 120–121

attribute types

CDATA, 122

ENTITIES, 123

ENTITY, 123

enumerated, 124–125

ID, 122–123

IDREF, 122–123

IDREFS, 122–123

NMTOKEN, 124

NMTOKENS, 124

attribute value declarations

default values, 125–126

fixed values, 126

implied values, 127

required values, 126–127

specifying multiple attributes, 127

attribute minimization, 742

attribute nodes, 255, 256, 259, 262, 266, 890

attribute pattern, 213, 233, 237

attribute selectors, 701

attribute value, 40, 122

attribute value template, 343, 941

attribute values, XHTML 1.x, 742

attribute::\*, 928

<attribute> declarations

attribute qualified form, 179

attribute use, 179–180

attribute wildcards, 180–182

default values, 180

fixed values, 180

global, 177–178

local, 177

naming attributes, 179

referring to existing global attributes, 179–183

attributeDecl event, DeclHandler interface, 514

attributeFormDefault attribute, 154, 179

<attributeGroup> declaration, 179, 180, 183–185, 186, 216

attributeName attribute, 730

attribute::number location path, 890

### attributes

@type attribute, 827

accesskey attribute

xforms:input element, 818

xforms:range element, 819

xforms:secret element, 819

xforms:select element, 822

xforms:select1 element, 822

xforms:trigger element, 820

xforms:upload element, 819

action attribute, 814, 845

actuate attribute, 724, 725

align attribute, 913, 914

Alternate attribute, 700

<anyAttribute> declaration, 180

appearance attribute

xforms:input element, 817

xforms:range element, 819

xforms:secret element, 819

xforms:select1 element, 822

xforms:trigger element, 820

xforms:upload element, 819

as attribute, 333

from attribute, xsl:number, 952

attribute axis, 259, 260, 261, 262–264, 890, 925

attribute minimization, 742

attribute nodes, 255, 256, 259, 262, 266, 890

attribute pattern, 213, 233, 237

attribute selectors, 701

attribute value, 40, 122

attribute value template, 343, 941

attribute values, XHTML 1.x, 742

attribute::\*, 928

<attribute> declarations

attribute names, 120–121

attribute qualified form, 179

attribute types, 122–126

attribute use, 179–180

attribute value declarations, 125–127

attribute wildcards, 180–182

default values, 180

fixed values, 180

- global, 177–178
- local, 177
- naming attributes, 179
- referring to existing global attributes, 179–183
- specifying multiple attributes, 127
- attributeDecl event, DeclHandler interface, 514
- attributeFormDefault attribute, 154, 179
- <attributeGroup> declaration, 179, 180, 183–185, 186, 216
- attributeName attribute, 730
- attribute:number location path, 890
- Attributes interface, 492, 494
- attributes property, Node object, 458, 459, 471
- attributes' setNamedItem( ) method, 475
- attributes.item(j) method, 920
- base attribute, 195
- bgcolor attribute, 694, 736, 913, 914
- bind attribute
  - xforms:output element, 819
  - xforms:range element, 819
  - xforms:select1 element, 822
  - xforms:submission element, 815
- case-order attribute, xsl:sort, 958
- CDATA attribute, 121, 122
- cdata-section-elements attribute
  - xforms:submission element, 815
  - xsl:output, 955
- Charset attribute, 700
- class attribute, 701, 764, 818
- comments attribute, 207
- compared to elements, 43–45
  - annotations, 45
  - complexity, 44–45
  - order, 45
  - separating information, 43–44
  - space, 44
  - visual preferences, 45
- constraint attribute, 828, 833
- contacts attribute, 223
- ContentType attribute, 673
- count attribute, xsl:number, 952
- createAttribute( ) method, Document interface, 475, 479, 480
- createAttributeNS( ) method, Document interface, 479, 480
- cx attribute, 774
- cy attribute, 774
- d attribute, 777
- data-type attribute, xsl:sort, 957
- decimal-separator attribute, xsl:decimal-format element, 945
- default attribute, 162, 180
- default values, 180
- digit attribute, xsl:decimal-format element, 946
- disable-output-escaping attribute
  - xsl:text, 960
  - xsl:value-of, 961
- DOB attribute node, 264
- doctype-public attribute, xsl:output, 955
- doctype-system attribute, xsl:output, 955
- duplicate attribute names, 120–121
- elementFormDefault attribute, 151, 154, 161, 175
- elements attribute
  - xsl:preserve-space, 956
  - xsl:strip-space, 958
- encoding attribute, 50–51, 52, 740, 750, 815, 912, 954
- encodingStyle attribute, 615
- ENTITIES attribute, 121
- ENTITIES attribute, 123
- ENTITY attribute, 121
- ENTITY attribute, 123
- Enumerated List attribute, 121
- env:encodingStyle attribute, rdf:RDF element, 616
- event-handler attributes, 763
- exclude-result-prefixes attribute
  - xsl:stylesheet, 958
  - xsl:transform, 960
- extension-element-prefixes attribute
  - xsl:stylesheet, 958
  - xsl:transform, 960
- fill attribute, 771, 775, 782, 785, 795, 799
- fill-opacity attribute, 790
- FirstName attribute, 300, 301
- fixed attribute, 162, 163, 180, 183
- fixed values, 180
- font-size attribute, 782
- form attribute, 161, 179
- format attribute, xsl:number, 952
- functions attribute, 813
- gender attribute declaration, 884, 886
- getTransformBlock attribute, 795

## attributes (continued)

---

### attributes (continued)

- global attributes, 84, 177–178, 179
- group-by attribute, 326
- grouping-separator attribute
  - xsl:decimal-format, 945
  - xsl:number, 953
- grouping-size attribute, xsl:number, 953
- height attribute, 774, 787, 789
- href attribute, 700, 722, 724, 764, 819, 949, 950
- http-equiv attribute, 746
- id attribute, 41, 83, 85, 121, 122–123, 126, 739, 740, 744, 747, 813, 814, 815, 820, 825, 913, 914, 958, 960
- ID attribute nodes, 323
- IDREF attribute, 121, 122–123
- IDREFS attribute, 121, 122–123
- implied attributes, 127
- includenamespacesprefixes attribute, xforms: submission element, 815
- incremental attribute
  - xforms:input element, 818
  - xforms:range element, 820
  - xforms:select element, 822
  - xforms:select1 element, 822
- indent attribute
  - xforms:submission element, 815
  - xsl:output, 955
- infinity attribute, xsl:decimal-format element, 945
- isPermalink attribute, 532
- itemType attribute, 196
- kind attribute, 122, 124, 125, 126, 128, 130, 178, 180, 186, 192, 193, 195, 196
- lang attribute, 739, 745, 747, 953, 957
- LastName attribute, 300
- letter-value attribute, xsl:number, 953
- level attribute, xsl:number, 952
- local attributes, 177
- margin attribute, 910
- match attribute, 293, 296, 315, 319, 950, 959
- maxOccurs attribute, 161, 163, 168, 170
- Media attribute, 700
- media-type attribute
  - xforms:submission, 815
  - xsl:output, 955
- memberTypes attribute, 197
- method attribute
  - xforms:submission element, 814
  - xsl:output element, 306, 954
- minOccurs attribute, 161, 163, 164, 168, 170
- minus-sign attribute, xsl:decimal-format element, 946
- mixed attribute, 176
- mode attribute
  - xsl:apply-templates element, 315, 941
  - xsl:template element, 959
- model attribute
  - xforms:itemset element, 827
  - xforms:select1 element, 826
- mustUnderstand, 621–622
- mustUnderstand attribute, 621–622, 623, 906
- myAttr attribute, 40
- myElement[myAttribute] selector, 729
- myElement[myAttribute~="myValue"] selector, 729
- myElement[myAttribute="myValue"] selector, 729
- name attribute
  - xsl:attribute element, 941
  - xsl:attribute-set element, 942
  - xsl:call-template element, 943
  - xsl:decimal-format element, 945
  - xsl:element element, 303, 305, 947
  - xsl:key element, 950
  - xsl:param element, 956
  - xsl:processing-instruction element, 957
  - xsl:template element, 322, 959
  - xsl:variable element, 961
- namespace attribute, 164, 180, 181, 204, 914, 941, 947
- namespace pseudo-attribute, 807
- namespaces, 83–86
- naming, 179
- NaN attribute, xsl:decimal-format element, 946
- navigationindex attribute, xforms:select1 element, 822
- navindex attribute, 818, 819, 820
- ncname attribute, xsl:sort, 957
- NMTOKEN attribute, 121, 124
- NMTOKENS attribute, 121, 124
- nodeset attribute
  - xforms:bind element, 828
  - xforms:itemset element, 827

- noNamespaceSchemaLocation attribute, 152
- NSAttributes object, 550
- number attribute, 252, 265, 890
- omit-xml-declaration attribute
  - xforms:submission element, 815
  - xsl:output element, 955
- onclick attribute, 452, 785, 845, 864
- onload attribute, 789, 793
- onLoad value, actuate attribute, 725
- optional attribute, 180
- order attribute, xsl:sort, 957
- orderID attribute, 910
- from other namespaces, 207
- padding attribute, 910
- path data, d attribute, 777
- pattern-separator attribute, xsl:decimal-format element, 947
- percent attribute, xsl:decimal-format element, 946
- per-mille attribute, xsl:decimal-format element, 946
- person attribute, 128
- phone attribute declaration, 140
- points attribute, 918
- portrait attribute, 123
- priority attribute, xsl:template, 959
- processContents attribute, 164, 181, 887
- qualification, 154–155
- qualified form, 179
- r attribute, 774
- rdf:about attribute, 531, 535
- ref attribute, 170, 178, 184, 815, 817, 819, 822, 826, 836
- RELAX NG patterns, 216–223
  - cardinality, 216–217
  - compared to elements, 216
  - connector patterns and grouping, 217–220
  - co-occurrence constraints, 221
  - empty patterns, 222–223
  - enumerated values, 220–221
  - mixed content patterns, 222
- relay, 622
- relay attribute, 622
- replace attribute, xforms:submission element, 815
- required attribute, 180
- result-prefix attribute, xsl:namespace-alias, 951
- role, 622
- role attribute, 622, 764
- rotate attribute, 791
- rotate expression, transform attribute, 776
- scale expression, transform attribute, 777
- schema attribute, 834
- schemaLocation attribute, 152, 200, 204, 381
- security attribute, 278, 280
- select attribute
  - xsl:apply-templates element, 313, 318, 321, 941
  - xsl:copy-of element, 945
  - xsl:for-each element, 948
  - xsl:param element, 956
  - xsl:sort element, 314, 957
  - xsl:value-of element, 299, 961
  - xsl:variable element, 321, 962
  - xsl:with-param element, 963
- selected attribute, 742
- selection attribute, xforms:select element, 821
- selection attribute, xforms:select1 element, 822
- show attribute, 724, 725
- source attribute, 120, 128, 130, 183, 185, 208, 209
- specifying multiple, 127
- src attribute, 764, 813, 825
- standalone attribute, 50–54, 815, 955
- status attribute, 353
- step attribute, xforms:range element, 820
- stroke attribute, 773, 776
- stroke-width attribute, 773, 776
- style attribute, 84, 86, 636, 736
- stylesheet-prefix attribute, xsl:namespace-alias, 951
- submission attribute, xforms:submit element, 820
- tags attribute, 130, 192, 243
- target attribute, 785
- targetNamespace attribute, 151, 152, 175, 206
- terminate attribute, xsl:message, 951
- test attribute
  - xsl:if, 949
  - xsl:when, 962
  - xsl:with-param, 962
- text-anchor attribute, 792
- title attribute, 152, 177, 204, 216, 217, 226, 231, 234, 235, 700, 724
- transform attribute, 776, 777, 793, 795, 798, 799, 871, 918, 919

## attributes (continued)

---

### attributes (continued)

- translate attribute, 791
- translate expression, transform attribute, 776
- transport attribute, 636
- type attribute, 157, 159, 196, 221, 494, 535, 700, 789
- type="html" attribute, 535
- type="text" attribute, 535
- types, 122–125
  - CDATA, 122
  - ENTITIES, 123
  - ENTITY, 123
  - enumerated, 124–125
  - ID, 122–123
  - IDREF, 122–123
  - IDREFS, 122–123
  - NMTOKEN, 124
  - NMTOKENS, 124
- use, 179–180
- use attribute, 179, 637, 950
- use-attribute-sets attribute
  - xsl:attribute-set element, 942
  - xsl:copy template, 944
  - xsl:element, 947
- value attribute, xsl:number, 952
- value attributes, 845
- version attribute, 50–51, 127, 130, 152, 182, 183, 192, 815, 954, 959, 960
- viewBox attribute, 774
- width attribute, 774, 787, 789
- wildcards, 180–182
- x attribute, 777
- xhtml:body attribute, 533
- xlink:actuate attribute, 722
- xlink:href attribute, 722
- xlink:show attribute, 722, 725
- xlink:title attribute, 722
- xlink:type attribute, 722
- xml:lang attribute, 270, 739, 745, 747
- xmlns attribute, 74, 76, 77, 80, 912
- xmlns="" attribute, 882
- xmlns:pers attribute, 72
- xml:space attribute, 270, 469
- xs:boolean value, incremental attribute, 818
- xsi:nil attribute, 400
- xsi:schemaLocation attribute, 185
- xsl: attribute element, 941–942
- xsl: attribute-set element, 942
- xsl:attribute element, 301
- y attribute, 777
- zero-digit attribute, xsl:decimal-format element, 946
- Attributes **interface**, 492, 494
- attributes **property**, Node object, 458, 459, 471
- attributes' setNamedItem( ) **method**, 475
- attributes.item (j) **method**, 920
- <authentication> **header entry**, 621
- author **element**, 899
- author **elements**, 368
- authoring tools**, 754
- AUTO **option**, 405
- AutoSuggest box, class design**
  - adding suggestions, 666–667
  - coping with speed typists, 670–671
  - creating suggestions list, 663–665
  - handling control key, 662–663
  - handling input key, 662–663
  - handling keyboard selection, 667–670
  - handling mouse actions, 665–666
  - implementing suggest( ) method, 662
  - implementing SuggestionProvider, 671–673
  - implementing web service, 673–680
  - showing a suggestion, 660–662
- AutoSuggest **textbox**, 653
- AutoSuggestControl class**, 659
- AutoSuggestControl **constructor**, 663, 671, 678
- AutoSuggestControl.suggest( ) method**, 673
- AutoSuggest.css **file**, 659, 678
- AutoSuggestDemo.html file**, 659, 678
- AutoSuggest.js **file**, 659, 678
- axes**
  - ancestor, 925
  - ancestor-or-self, 925
  - attribute, 925
  - child, 925
  - descendant, 925
  - descendant-or-self, 926
  - following, 926
  - following-sibling, 926
  - namespace, 926
  - parent, 927
  - preceding, 927
  - preceding-sibling, 927
  - self, 927
  - in XQuery, 361

**B**

- `<b>` element, 694, 701, 913
- B2B (business-to-business) e-commerce**, 21
- B2B (business-to-business) transactions**, 342
- B2C (business-to-consumer) transactions**, 21
- background property, CSS, 695
- Background property, CSS, 695
- background-attachment property, CSS, 695
- background-color property, 694, 695
- background-image property, 695, 725
- background-position property, 695
- background-repeat property, 695
- balancePath variable, 870
- bandwidth**, 8
- base attribute, 195
- Base module**, 761, 762
- base64Binary type, 187
- base-uri declaration, XQuery, 356
- Basic forms module**
  - XHTML, 760
  - XHTML Basic, 762
- Basic tables module**
  - XHTML, 760
  - XHTML Basic, 762
- BasicRetrievalDemo.aspx** page, 650
- BasicRetrievalDemo.html**, 657
- Batik, Apache**, 770, 771, 800
- before clause, 419
- :before pseudo-element, 726, 729
- BegXML.xslt stylesheet, 318
- bgcolor attribute, 694, 736, 913, 914
- BibAdapted.xml file, 349, 352, 367
- BibQuery1Out.xml file, 351
- BibQuery3Out.xml file, 353
- Bidirectional text module**, XHTML, 760
- bin directory, 291, 382
- binary file formats**, 6
- binary files**, 4
- bind attribute
  - xforms:output element, 819
  - xforms:range element, 819
  - xforms:select1 element, 822
  - xforms:submission element, 815
- bind elements, compared to XML Schema, 834
- `<binding>` element, 635
- bits, 4
- blank node, 527
- bLoaded variable, 453
- block boxes, 703–704
- blog collection, 385
- Blogger blogging service**, 524
- blogging tools**, 523
- blogitem1.xml file, 385
- blogitem2.xml file, 385
- Bloglines**, 536
- body blocks**, 614
- `<body>` element, 593, 614–615, 696, 846, 913, 914, 915
- BOL (Books Online)**, 404
- `<bold>` element, 880
- book element, 359
- book-indented.xml file, 469
- Books element node**, 252, 257, 279, 347, 351, 370
- Books Online (BOL)**, 404
- book.xml file, 469
- book-xml-space.xml file, 469
- boolean( ) function, 274, 964
- boolean flags**, 513
- boolean function-available function, 964
- boolean object, 929
- boolean type, 188, 966
- Booleans**
  - functions, 274–275
  - XPath 1.0, 258
- border**, 702
- border-collapse property, CSS, 695
- border-spacing caption-side property, CSS, 695
- Both value, clear property, 713
- BottomFeeder**, 536
- boundary whitespace**, 358
- boundary-space declaration, XQuery, 357–358
- box model**, 702
- boxes**
  - anonymous, 705–706
  - block, 703–704
  - inline, 703–704
- BP4WS (Business Process Execution Language for Web Services)**, 603
- br elements, 743
- `<br>` element, 32, 50, 109, 110, 222, 223, 236, 549, 553, 743, 888, 890

## branches, 15

### browsers

- plug-ins, 804
- processing, 564–565
- XML Document Object Model (DOM), 450

### Browsing Collection, eXist, 384

### building, proxy web services, 849–853

### built-in datatypes, 187–192

### built-in entities, 131–132

### <bulletPoint> element, 728

### Business Process Execution Language for Web Services (BPEL4WS), 603

### business protocols, 603

### business-to-business (B2B) e-commerce, 21

### business-to-business (B2B) transactions, 342

### business-to-consumer (B2C) transactions, 21

### byte type, 187

## C

### c command, 778

### C++ language, 517

### Calculate function, 857, 858

### calculate property, XForms model items, 828

### CalculatePassThrough function, 852, 858

### Candidate Recommendation, 633

### cardinality

- Document Type Definitions (DTDs), 114–119
- RELAX NG patterns, 216–217
- XML Schemas, 161–162

### cardinality indicator, 109

### carriage return characters, 38

### Cascading Style Sheets (CSS)

- attaching to an XML document, 699–700
- attribute content
  - attribute selectors, 729
  - using attribute values in documents, 729–731
- images in XML documents, 725–726
- inheritance, 695–699
- laying out tabular data, 719–721
- layout of XML documents
  - box model, 702–706
  - positioning, 706–719
- links in XML documents
  - forcing links using the XHTML namespace, 725
  - XLink support in Firefox, 721–724

### overview, 693–701

### properties, 694–695

### purpose, 692–693

### selectors, 700–701

### using to add content to documents, 726–729

### case sensitivity, 35, 742

### case-order attribute, xsl:sort, 958

### CASTing, 414

### Castor, 380

### CDATA (character data) sections, 61–62

### CDATA attribute, 121, 122

### CDATA keyword, 120

### CDATA section, 394, 785

### CDATA type, 182

### CDATASection node, 449

- cdata-section-elements attribute
  - xforms:submission element, 815
  - xsl:output, 955

### CDF (Content Definition Format), Microsoft, 525–526

### ceiling( ) function, 275

### center object, 796

### Chameleon components, 206

### channel description, 528

### channel resource, 538

### channel start-tag, 560

### channel template, 564

### <channel> element, 563, 903

### Chapter element nodes, 252, 265, 279, 280

### Chapter elements, 254, 256, 266, 280, 353

### Chapter nodes, 318, 480

### <Chapter> element, 253, 259

### character code, 51

### character content, ContentHandler interface, 494–497

### character data (CDATA) sections, 61–62

### Character element nodes, 309

### Character elements, 307

### character entities, 132–133

### character entity reference, 133

### Character Map application, 874

### character references, 61

### character type, 966

### Characteristic element nodes, 312, 314

### characters

- /, 347
- &, 59, 60, 444, 756
- &amp;, 444

- @, 262
- [, 101
- ], 101
- <, 24, 59, 756, 880
- =, 355
- >, 756
  - type, 966
- ampersand (&), 132, 135, 140, 599, 905
- ASCII 7-bit encoding, 52
- carriage return s, 38
- closing >, 34
- colon (:), 34, 490
- dash (-), 34
- encoding
  - specifying, 52–53, 746
  - Unicode, 52
- end <...>, 563
- French, 874
- illegal, 60–61, 133
- linefeed, 38
- newline, 38, 549, 553, 902, 920
- opening (<), 34
- parsed character data (PCDATA)
  - illegal characters, 60–62
  - whitespace, 38–39
- raw, 60
- un-escaped, 60
- vertical bar (|), 108, 125
- characters **callbacks, 495**
- characters **event, 487, 494**
- characters **function, 495, 539, 551, 896**
- Characters.xslt **stylesheet, 307**
- charCode **property, 919**
- Charset **attribute, 700**
- Chemical Markup Language (CML), 13, 104**
- Chiba, 808**
- child **axis, 252, 258, 259, 260–261, 279, 280, 925**
- child **element nodes, 279**
- child elements, 27, 44, 239, 876**
- child node, 347**
- Child selector, 701**
- child::comment( ) **function, 928**
- childNodes **property, 458, 459, 475, 798**
- child::text( ) **function, 928**
- choice (sequence) connector, 217–219**
- choice **declaration, 176**
- choice pattern, 217**
- <choice> **declarations, 169–170**
- choices**
  - combining with sequences using groups, 108
  - name-classes, 238–239
- Choreography Definition Language, 603**
- CHTML (Compact HTML), 759**
- Chumpalogica, 525**
- <circle> **element, 771, 776, 783, 916**
- class **attribute, 701, 764**
  - xforms:input element, 818
- class selector, 701**
- classes**
  - AutoSuggestControl, 659
  - ContentHandler class, 504
  - current class, 664
  - DefaultHandler class, 504, 510, 515
  - design
    - adding suggestions, 666–667
    - coping with speed typists, 670–671
    - creating suggestions list, 663–665
    - handling control key, 662–663
    - handling input key, 662–663
    - handling keyboard selection, 667–670
    - handling mouse actions, 665–666
    - implementing suggest( ) method, 662
    - implementing SuggestionProvider, 671–673
    - implementing web service, 673–680
    - showing a suggestion, 660–662
  - dtstart class, 764
  - ErrorHandler class, 504–509
  - feed , 546–548
  - Feed class, 546–548
  - FeedHandler , 548–552
  - FeedHandler class, 548–552, 553
  - FeedReader, 543–546
  - FeedReader class, 543–546
  - item, 546–548
  - item class, 546–548
  - ListReader, 541–542
  - ListReader class, 541–542, 545
  - MSXML2.XMLHTTP class, 588
  - MyClass class, 486
  - SuggestionProvider class, 659, 671–673
  - System.Xml class, 674
  - TextRange class, 660

## classes (continued)

---

### classes (continued)

- TrainReader class, 515, 897
- XmlHttpRequest class, 592, 646
- XMLHttpRequest class, 462, 592, 650
- XMLSerializer class, 457
- XslCompiledTransform class, 675
- zXPath class, 595, 658
- CLASSPATH **environment variable, 290, 486, 490**
- clauses**
  - OPENXML, 409–413
  - order by, 366–367
  - where, 365–366
  - FOR XML, 396–425
  - FOR XML AUTO, 401–404
  - FOR XML EXPLICIT, 405
  - FOR XML PATH, 405–409
  - FOR XML RAW, 396–401
- clear **property, 713**
- clearTimeout( ) **method, 670**
- client.bat **script, 386**
- client-consumer syndication systems, 523**
- client-producer syndication systems, 523–524**
- clients, creating, 684–686**
- client/server **implementations, 805, 808**
- client.sh **script, 386**
- Client-side image map module, XHTML, 760**
- client-side scripting, 804**
- client-side validation, 804**
- cloneNode( ) **method, Node object, 461, 462**
- closepath Z **command, 779**
- closing > character, 34**
- cm **absolute unit, SVG, 774**
- CML (Chemical Markup Language), 13, 104**
- Cocoon, 808**
- <Code> **element, 622**
- Codeplot editor, 96, 213**
- collection( ) **function, 346, 348**
- collections, 384**
- colon (:): character, 34, 490**
- color **property, 695, 699**
- <color> **element, 498, 896**
- <column1> **element, 716**
- <column2> **element, 716**
- COM (Component Object Model), 573**
- combining**
  - grammars, 227–236
  - patterns, 227–236
  - sequences and choices using groups, 108
- command-line **user interface, 537**
- commands**
  - A, 778
  - alien, 427
  - C, 778
  - closepath Z, 779
  - dpkg -i, 427
  - elliptical arc, 779
  - H, 778
  - horizontal lineto, 778
  - L, 778
  - line, 870
  - M, 778
  - path commands, 779
  - Q, 778
  - rotate, 793
  - rotate(degrees), 871
  - S, 778
  - special, 499
  - T, 778
  - translate(x,y), 871
  - V, 778
  - vertical lineto, 778
  - Z, 778
- comment( ) **function, 928**
- comment **event, LexicalHandler interface, 515**
- Comment **nodes, 255, 257, 449, 459**
- Comment **objects, 448**
- comments**
  - RELAX NG, 244
  - Scalable Vector Graphics (SVG), 782–784
  - XML Schemas, 206–207
  - XQuery, 360
- comments **attribute, 207**
- Common Object Request Broker Architecture (CORBA), 574**
- communicating with proxy web service using Ajax, 858–864**
- Compact HTML (CHTML), 759**
- <comparison> **element, 60**
- com/pers **namespace, 83**
- <complexType> **element, 151, 157, 158, 160, 165–166, 168, 170, 175, 178, 180, 182, 183–184, 186, 192, 216, 886**
- Compliance Studio, 842, 849, 852, 857**
- Component Object Model (COM), 573**
- component-based software, 573**
- composition, 234**

- computing, distributed, 21**
- `concat( )` **function, 276, 369**
- conceptual resource, 87**
- conditional expressions, 367–368**
- conditional processing**
  - `<xsl:choose>` **element, 308–311**
  - `<xsl:if>` **element, 306–308**
- `<conductors>` **element, 899**
- Configure Database Sources button, Database Explorer window, 389**
- connector patterns, and grouping, 217–220**
- connectors**
  - interleave, 219–220
  - sequence (choice), 217–219
- constraining XForms instances**
  - choosing between schema and bind elements, 834
  - W3C XML Schema in XForms, 833–834
  - `xforms:bind` **element, 828–833**
- `constraint` **attribute, 828, 833**
- `constraint` **property, XForms model items, 828**
- constructors**
  - computed, 358–359
  - element, 351–355
- consumers, Simple API for XML (SAX), 516**
- `<contact>` **element, 106, 107, 115, 124, 192, 203, 884, 885, 886**
- `contacts` **attribute, 223**
- `contacts` **element, 175**
- `contacts` **namespace, 241**
- `contacts` **prefix, 178, 237**
- contacts schema, 242**
- `contacts` **vocabulary, 234**
- `<contacts>` **element, 120, 127, 130, 181, 182, 185, 226, 235**
- `contacts19.rnc` **schema file, 888**
- `contacts4.xml` **document, 139**
- `contacts-main.rnc` **declaration, 888**
- `contacts-names.rnc` **declaration, 888**
- `ContactTagsListType` **type, 197**
- containing block, 705**
- containing box, 705**
- containing element, 705**
- `contains( )` **function, 276**
- content**
  - any, 110
  - element
    - choices, 107–108
    - combining sequences and choices using groups, 108
    - sequences, 106–107
  - empty, 110
  - incorporating from other people, 757
  - mixed, 108–109
- Content Definition Format (CDF), Microsoft, 525–526**
- content model, 105**
- content model inheritance, 147, 533**
- content models**
  - `<all>` **declarations, 171–174**
  - `<choice>` **declarations, 169–170**
  - `<group>` **references, 170–171**
  - XML Schemas, 147
- `content` **prefix, 538**
- `content` **property, 730, 731, 910**
- content syndication**
  - aggregators
    - implementation, 540–557
    - modeling feeds, 537–539
    - program flow, 540
    - transforming RSS with XSLT, 557–567
  - meta data
    - evolution of systems, 525–535
    - syndication systems, 522–525
  - news feeds
    - data quality, 536–537
    - newsreaders, 536
    - resources, 567–568
- `<content>` **element, 534, 535**
- `ContentHandler` **class, 504**
- `ContentHandler` **interface**
  - character content, 494–497
  - element events, 491–494
  - handling special commands with processing instructions, 499
  - ignorablewhitespace events, 498
  - namespace prefixes, 499
  - providing the location of the error, 502–504
  - skipped entities, 498–499
  - stopping process, 500–502
- Content-Length header, 581**
- `ContentType` **attribute, 673**
- `Content-Type` **header, 581**
- `Content-Type` **HTTP header, 747**
- context, XPath, 252–254**
- context node, 252, 264, 266, 271, 278, 923**

## controllers, application, 543–546

### controls, XForms

- xforms:input element, 817–818
- xforms:output element, 818–819
- xforms:range element, 819–820
- xforms:secret element, 818
- xforms:select element, 821–822
- xforms:select1 element, 822–824
- xforms:submit element, 820–821
- xforms:textarea element, 818
- xforms:trigger element, 820
- xforms:upload element, 819

### conventional databases

- data binding, 380–381
- moving XML to relational databases, 380
- producing XML from relational databases, 380

ConversionRate( ) **method**, 684

### CONVERTing, 414

### co-occurrence constraints, 221

### coordination, web services, 603–604

### copyright symbol ((c)), 61

### CORBA (Common Object Request Broker Architecture), 574

count( ) **function**, 258, 275, 298, 368, 369–370

count **attribute**, xsl:number, 952

### count node-set, 930

<countries> **element**, 676

### CountriesSuggestionProvider2.js file, 681

### CountriesSuggestionProvider.js file, 659, 678

countries.xml **file**, 659, 673, 674, 676, 678

countries.xslt **file**, 673, 677, 678

<country> **element**, 673, 675, 677, 681

Count.xquery **query**, 369

### Cover Pages, 104

### CPAN, 771

### Create Collection button, eXist, 385

### create table SQL instruction, 428

create\_item **method**, 547

create\_parser **method**, 544, 545

createAttribute( ) **method**, Document **interface**, 475, 479, 480

createAttributeNS( ) **method**, Document **interface**, 479, 480

createCDATASection( ) **method**, Document **interface**, 479, 480

createComment( ) **method**, Document **interface**, 479, 480

createDocumentFragment( ) **method**, Document **interface**, 479

createElement( ) **method**, Document **interface**, 444, 465, 478, 479, 480, 894

createElementNS( ) **method**, Document **interface**, 480, 894

createElementNS **method**, Document **interface**, 479

createEntityReference **method**, Document **interface**, 479

createProcessingInstruction( ) **method**, Document **interface**, 479, 480

createSuggestionsBox( ) **method**, 664, 670

createTextNode **method**, Document **interface**, 479

createTextNode **property**, 465

createTextRange **method**, 662

**Creative Commons license**, 533

<credit> **element**, 218

**Crimson parser**, 485, 502

**Croczilla SVG**, 800

**cross-domain posting**, 595

**Cross-Domain XML HTTP Requests**, 849

**Cross-Site Scripting (XSS)**, 849

**CSR (customer service representative)**, 26

### CSS (Cascading Style Sheets)

attaching to an XML document, 699–700

attribute content

attribute selectors, 729

using attribute values in documents, 729–731

images in XML documents, 725–726

inheritance, 695–699

laying out tabular data, 719–721

layout of XML documents

box model, 702–706

positioning, 706–719

links in XML documents

forcing links using the XHTML namespace, 725

XLink support in Firefox, 721–724

overview, 693–701

properties, 694–695

purpose, 692–693

selectors, 700–701

using to add content to documents, 726–729

### Curl language, 517

**curly braces ({ })**, 72, 332, 343, 359, 540

### currency converter proxy

creating clients, 684–686

creating web services, 682–683

implementing GetRate( ) **method**, 683–684

**CurrencyProxyService** folder, 684

**currencyProxyServiceDemo.html**, 684

current **class**, 664

current-date( ) **function**, 283

current-DateTime( ) **function**, 283

current-group( ) **function**, 324

current-grouping-key( ) **function**, 324, 326, 332

currentSuggestionIndex **variable**, 668

current-time( ) **function**, 283

cust\_id **parameter**, 905

custom \_\_str\_\_ **method**, 548

**customer service representative (CSR)**, 26

cx **attribute**, 774

cy **attribute**, 774

## D

d **attribute**, 777

danny-description **entity value**, 139

dash (-) **character**, 34

**data binding, conventional databases**, 380–381

**data files**

binary files, 4

history of markup languages, 6–7

text files, 5

**data models**

XPath 2.0, 360

XQuery

axes in, 361

shared with XPath 2.0 and XSLT 2.0, 360

types in, 361

XSLT 2.0, 323, 360

**Data Modification Language (DML)**, 416, 417

**data quality, news feeds**, 536–537

**Database Explorer**, 389

**databases**

choosing a database to store XML, 438

commercial relational database management systems (RDBMSs)

Web service support, 426

XML functionality in SQL Server 2000, 395

efficient data stores

XML growth, 376–377

XML-based data and relational data compared, 377

future of, 438

native databases

eXist, 382–394

interacting with eXist, 384–394

open source relational database management systems (RDBMSs)

adding information in MySQL, 427–430

client-side XML support, 437

installing MySQL, 426–427

querying MySQL, 430–435

updating XML in MySQL, 435–436

usability of XML in MySQL, 436

storing XML

conventional databases, 379–381

file systems, 378–379

native databases, 381–382

DataEncodingUnknown **identifier**, 623

data-type **attribute**, xsl:sort, 957

datatype **value**, 220

**datatypes**

built-in, 187–192

document type definitions (DTDs), 143

RELAX NG, 241–243

user-defined, 193

validation, 221

xml, 413–416

XML Schemas, 147

**datatyping, control**, 924

Date **element nodes**, 261, 265

date **properties**, 545

date **type**, 188

date **values**, 547

date\_labels **container**, 870

<date> **element**, 539

dateTime **type**, 188

david-description **entity**, 138

day-from-date( ) **function**, 283

/db/blog **collection**, 385, 391

dc **prefix**, 538

**DCOM (Distributed Component Object Model)**, 573–574, 609

dc:source **extension**, 900

<dc:title> **element**, 550

.deb **packages**, 427

**Debian**, 427

**Debian alien package**, 427

**debugging**

SVG in Firefox, 770

XForms, 836

# decimal type

---

decimal **type**, **187, 193**

**decimal-separator attribute**, **xsl:decimal-format**  
**element**, **945**

## declarations

- <all>, 171–174
  - attribute value
    - default values, 125–126
    - fixed values, 126
    - implied values, 127
    - required values, 126–127
  - <attribute>
    - attribute qualified form, 179
    - attribute use, 179–180
    - attribute wildcards, 180–182
    - default values, 180
    - fixed values, 180
    - global, 177–178
    - local, 177
    - naming attributes, 179
    - referring to existing global attribute, 179
  - <attributeGroup>, 183–185
  - attributes
    - names, 120–121
    - specifying multiple, 127
    - types, 121–125
    - value declarations, 125–127
  - base-uri, 356
  - boundary-space, 357–358
  - character encoding
    - specifying, 52–53
    - Unicode, 52
  - <choice>, 169–170
  - <complexType>, 165–166
  - DOCTYPE, 741
  - <element>
    - any content, 110
    - cardinality, 114–119, 161–162
    - default values, 162–163
    - element content, 106–108
    - element qualified form, 161
    - element wildcards, 163–165
    - empty content, 110
    - fixed values, 162–163
    - global, 156, 157–158
    - local, 156–157
    - mixed content, 108–109
    - naming elements, 160–161
    - referring to existing global element, 159–160
  - <group>, 167–168
  - <import>, 200–204
  - <include>, 204–206
  - <list>, 196–197
  - namespace, 356
  - <restriction>, 194–196
  - <schema>
    - element and attribute qualification, 154–155
    - target namespaces, 153–154
    - XML Schema namespaces, 153
  - <simpleType>, 193
  - standalone attribute, 53–54
  - <union>, 197–199
  - validation, 357
  - variable, 357
  - version, 355
  - version attribute, 51
  - XML, 740
- DeclHandler **events**, **897**
- DeclHandler **extension interface**, **514**
- DeclHandler **property**, **513**
- default **attribute**, **162, 180**
- default namespaces**, **75–81, 204, 236, 237, 881**
- default **values**, **125–126, 162–163, 180**
- default XPath behavior**, **432**
- DefaultHandler **class**, **504, 510, 515**
- DefaultHandler2 **events**, **897**
- DefaultPhoneKind **parameter**, **140**
- definitions, combining named pattern**, **230–231**
- <definitions> **element**, **633**
- delete **keyword**, **416, 417**
- deleteNode( ) **method**, **464, 465**
- demilitarized zone (DMZ)**, **583**
- derived types**, **193**
- desc **element**, **784**
- <desc> **element**, **783, 784, 788**
- descendant **axis**, **260, 265–266, 890, 925**
- Descendant selector**, **701**
- descendant text nodes**, **256**
- descendant-or-self **axis**, **260, 266, 890, 926**
- Description **element node**, **298**
- <description> **element**, **109, 110, 132, 138, 139, 166, 222, 236, 239, 533, 561, 887**
- descriptionContents **pattern**, **890**

- DescriptionType** <complexType> **definition, 176**
- <Detail> **element, 623**
- details **parameter, 578**
- DHTML (Dynamic HTML), 14**
- digit **attribute, xsl:decimal-format element, 946**
- direction **property, CSS, 695**
- disable-output-escaping **attribute**
  - xsl:text, 960
  - xsl:value-of, 961
- display **property, 703, 704, 719, 721, 911**
- display:table; **value, 719**
- display:table-caption; **value, 719**
- display:table-cell; **value, 719**
- display:table-row; **value, 719**
- Distributed Component Object Model (DCOM), 573–574, 609**
- distributed systems, 571**
- <div> **element, 535, 758, 843, 845, 847, 913**
- divisions, RELAX NG, 244**
- </DL> **tags, 543**
- DLL hell, 574**
- DML (Data Modification Language), 416, 417**
- DMZ (demilitarized zone), 583**
- DOB attribute node, 264**
- doc( ) **function, 346, 347–348**
- doc **directory, Saxon, 345**
- doc expression, 352**
- DocBook documents, 376, 697**
- Docs table, 425**
- DOCTYPE (document type declaration), 100–104, 251, 741**
- DOCTYPE declaration, 99, 103, 105, 117, 136, 251, 563, 740, 747, 750, 789, 843, 912, 914**
- doctype **property, Document interface, 479**
- doctype-public **attribute, xsl:output, 955**
- doctype-system **attribute, xsl:output, 955**
- document( ) **function, XSLT, 323, 347**
- document **element, 462**
- Document **interface, 446, 478–480**
- Document **node, 444, 446, 447, 448, 449, 894**
- document **object, 15**
- Document **object, 448, 459**
- Document Object Model (DOM), 14, 17, 20, 83, 147, 250, 251, 341–342, 376, 448, 483, 588, 893**
- Document Object Model Level 1 Recommendation, 446**
- document start-tag, 881**
- document type, 17, 70**
- document type declaration (DOCTYPE), 100–104, 251, 741**
- Document Type Definitions (DTDs)**
  - compared to XML Schemas, 147–148
  - developing, 141–142
  - example
    - document type declaration (DOCTYPE), 100–104
    - preparing to create, 96–100
  - limitations
    - data types, 143
    - limited content model descriptions, 143
    - namespaces, 143
    - syntax, 142–143
  - parts of
    - attribute declarations, 120–131
    - element declarations, 105–131
    - entities, 131–141
    - sharing vocabularies, 104
    - XHTML 1.x, 738–739
- document types, 17–18**
- document **value, 636**
- document views, 774**
- <documentation> **element, 208, 209**
- documentElement **property, 446, 454, 464, 479, 862**
- DocumentFragment **node, 449**
- documenting XML Schemas**
  - annotations, 208–209
  - attributes from other namespaces, 207
  - comments, 206–207
- document.rss **element, 564**
- documents**
  - links in XML
    - forcing using XHTML namespace, 725
    - XLink support in Firefox, 721–724
  - order, 361
  - XForms, XML namespaces in, 816–817
  - XHTML
    - attribute values, 742
    - case sensitivity, 742
    - character encoding, 745–747
    - DOCTYPE declaration, 741
    - IDs, 744
    - names, 744
    - specifying language, 745
    - styling, 751–752

### documents (continued)

- validating, 754–757
- well-formedness, 743–744
- XML declaration, 740
- XML
  - attaching Cascading Style Sheets (CSS), 699–700
  - layout using Cascading Style Sheets (CSS), 701–719
  - loading, 462–468
  - modeling, 250–251
  - ways of representing, 250
- DocumentType **node**, 449
- DocumentType **object**, 448
- document.write( ) **method**, 757
- document.xml **element**, 564
- Dojo Toolkit**, 771
- DOM (Document Object Model)**, 14, 17, 20, 83, 147, 250, 251, 341–342, 376, 448, 483, 588, 893
- DOMException.html **file**, 476
- DOMParser **property**, 456, 457
- DOMParser.parseFromStream( ) **method**, 462
- doPost( ) **function**, 587
- double **type**, 188
- double-dash string (-)**, 46
- dpkg -i **command**, Debian, 427
- draconian error handling**, 64
- drawChart **function**, 868
- Dreamweaver, Macromedia**, 753
- DROP XMLSCHEMA **statement**, 425
- DTDHandler **interface**, 509
- DTDs (Document Type Definitions)**
  - compared to XML Schemas, 147–148
  - developing, 141–142
  - example
    - document type declaration (DOCTYPE), 100–104
    - preparing to create, 96–100
  - limitations
    - data types, 143
    - limited content model descriptions, 143
    - namespaces, 143
    - syntax, 142–143
  - parts of
    - attribute declarations, 120–131
    - element declarations, 105–131
    - entities, 131–141
    - sharing vocabularies, 104
  - XHTML 1.x, 738–739

- dtstart **class**, 764
- Dublin Core Metadata Initiative standard**, 530, 556
- duplicate attribute names**, 120–121
- duration **type**, 188
- Dynamic HTML (DHTML)**, 14

## E

- ECMAScript**, 447, 769, 784, 788, 792
- e-commerce**, 21–22
- EDI (Electronic Data Interchange)**, 603
- Edit module**, XHTML, 760
- Edit System Variable window**, 291
- editorial judgment**, 536
- editors, Scalable Vector Graphics (SVG)**, 770
- Electronic Data Interchange (EDI)**, 603
- element constructors**, 351–355, 358
- element content**
  - choices, 107–108
  - combining sequences and choices using groups, 108
  - sequences, 106–107
- element content model**, 105
- ELEMENT declaration**, 105, 106, 127, 134, 139
- element declarations**
  - any content, 110
  - cardinality, 114–119
  - element content
    - choices, 107–108
    - combining sequences and choices using groups, 108
    - sequences, 106–107
  - empty content, 110
  - mixed content, 108–109
- element events, ContentHandler interface**, 491–494
- ELEMENT keyword**, 105
- Element **node**, 449, 468, 471, 480, 817, 890, 894
- Element **object**, 448
- element **pattern**, 213, 237
- element wildcards**, 163, 168
- <element> declarations**
  - cardinality, 161–162
  - default values, 162–163
  - element qualified form, 161
  - element wildcards, 163–165
  - fixed values, 162–163
  - global, 156, 157–158

- local, 156–157
- naming elements, 160–161
- referring to existing global element, 159–160
- elementDecl **event**, DeclHandler **interface**, **514**
- elementDecl **function**, **898**
- elementFormDefault **attribute**, **151, 154, 161, 175**
- elements**
  - bind, compared to XML Schema, 834
  - <binding>, 635
  - <Body>, 614–615
  - compared to attributes
    - annotations, 45
    - complexity, 44–45
    - order, 45
    - separating information, 43–44
    - space, 44
    - visual preferences, 45
  - creating with simple content and attributes, 185–186
  - <definitions>, 633
  - empty, 49–50, 743–744
  - <Envelope>, 614
  - <Fault>, 622–632
  - <Header>
    - mustUnderstand attribute, 621–622
    - relay attribute, 622
    - role attribute, 622
  - <messages>, 634
  - naming, 160–161
  - nesting, 744
  - <portTypes>, 635
  - qualification, 154–155
  - qualified form, 161
  - RELAX NG patterns
    - cardinality, 216–217
    - compared to attributes, 216
    - connector patterns and grouping, 217–220
    - co-occurrence constraints, 221
    - empty patterns, 222–223
    - enumerated values, 220–221
    - mixed content patterns, 222
  - rules
    - case sensitivity, 35
    - matching start- and end-tags, 31–32
    - naming conventions, 34–35
    - properly nested, 32–33
    - root elements, 33–34
    - whitespace, 36–39
  - <service>, 638–641
  - <soap:binding>, 636
  - <soap:body>, 637
  - <soap:operation>, 636–637
  - <types>, 633–634
  - unique root HTML, 743
  - wildcards, 163–165
  - xforms:bind, 828–833
  - xforms:input, 817–818
  - xforms:instance, 813–814
  - xforms:model, 812–813
  - xforms:output, 818–819
  - xforms:range, 819–820
  - xforms:secret, 818
  - xforms:select, 821–822
  - xforms:select1, 822–824
  - xforms:submission, 814–815
  - xforms:submit, 820–821
  - xforms:textarea, 818
  - xforms:trigger, 820
  - xforms:upload, 819
  - xsl: apply-imports, 940
  - xsl: apply-templates, 941
  - xsl: attribute, 941–942
  - xsl: attribute-set, 942
  - xsl: call-template, 943
  - xsl: choose, 943
  - xsl: comment, 944
  - xsl: copy, 944
  - xsl: copy-of, 945
  - xsl: decimal-format, 945–947
  - xsl: element, 947
  - xsl: fallback, 948
  - xsl: for-each, 948–949
  - xsl: if, 949
  - xsl: import, 949
  - xsl: include, 950
  - xsl: key, 950
  - xsl: message, 951
  - xsl: namespace-alias, 951–952
  - xsl: number, 952–953
  - xsl: otherwise, 953
  - xsl: output, 954–955
  - xsl: param, 956
  - xsl: preserve-space, 956

## elements (continued)

---

### elements (continued)

- xsl: processing-instruction, 957
- xsl: sort, 957–958
- xsl: strip-space, 958
- xsl: stylesheet, 958–959
- xsl: template, 959
- xsl: text, 960
- xsl: transform, 960–961
- xsl: value-of, 961
- xsl: variable, 961–962
- xsl: when, 962
- xsl: with-param, 962–963
- <xsl:apply-templates>, 296–297
- <xsl:call-template>, 322
- <xsl:choose>, 308–311
- <xsl:copy>, 299–303
- <xsl:copy-of>, 303–305
- <xsl:for-each>, 311–312
- <xsl:if>, 306–308
- <xsl:sort>, 312–314
- <xsl:stylesheet>, 295–296
- for XSLT 2.0, 323
- <xsl:template>, 296
- <xsl:value-of>, 297–299
- <abbr> element, 764
- <acronym> element, 764
- <address> element, 107, 116, 134, 226
- :after pseudo-element, 726, 729, 731
- <amount> element, 218
- ancestor element node, 265
- <AnotherTag> element, 38
- <appinfo> element, 208, 209
- <article> element, 699
- author elements, 368, 899
- <b> element, 694, 701, 913
- :before pseudo-element, 726, 729
- bind, compared to XML Schema, 834
- bind elements, compared to XML Schema, 834
- binding, 635
- <binding> element, 635
- <Body>, 614–615
- <body> element, 593, 614–615, 696, 846, 913, 914, 915
- <bold> element, 880
- book element, 359
- br elements, 743
- <br> element, 32, 50, 109, 110, 222, 223, 236, 549, 553, 743, 888, 890
- <bulletPoint> element, 728
- <channel> element, 563, 903
- <Chapter> element, 253, 259
- child elements, 27, 44, 239, 876
- <circle> element, 771, 776, 783, 916
- <Code> element, 622
- <color> element, 498, 896
- <column1> element, 716
- <column2> element, 716
- compared to attributes
  - annotations, 45
  - complexity, 44–45
  - order, 45
  - separating information, 43–44
  - space, 44
  - visual preferences, 45
- <comparison> element, 60
- <complexType> element, 151, 157, 158, 160, 165–166, 168, 170, 175, 178, 180, 182, 183–184, 186, 192, 216, 886
- <conductors> element, 899
- <contact> element, 106, 107, 115, 124, 192, 203, 884, 885, 886
- <contacts> element, 120, 127, 130, 175, 181, 182, 185, 226, 235
- <content> element, 534, 535
- <countries> element, 676
- <country> element, 673, 675, 677, 681
- creating with simple content and attributes, 185–186
- <credit> element, 218
- <date> element, 539
- <dc:title> element, 550
- <definitions>, 633
- <definitions> element, 633
- <desc> element, 783, 784, 788
- <description> element, 109, 110, 132, 138, 139, 166, 222, 236, 239, 298, 533, 561, 887
- <Detail> element, 623
- <div> element, 535, 758, 843, 845, 847, 913
- <documentation> element, 208, 209, 462
- document.rss element, 564
- document.xml element, 564
- Element object, 448
- <element> element

- cardinality, 161–162
- default values, 162–163
- element qualified form, 161
- element wildcards, 163–165
- fixed values, 162–163
- global, 156, 157–158
- local, 156–157
- naming elements, 160–161
- referring to existing global element, 159–160
- elements attribute
  - xsl:preserve-space, 956
  - xsl:strip-space, 958
- ELEMENTS keyword, 400
- elementValue variable, 896
- <em> element, 32, 109, 138, 139, 222, 236, 239, 888, 890
- <email> element, 885, 886
- empty, 49–50, 743–744
- empty element syntax, 49–50
- empty elements, 163, 222, 740, 743–744
- <entry> element, 535
- <env:Body> element, 636
- <Envelope>, 614
- <Envelope> element, 612, 614, 621
- evt.target <polygon> element, 799
- <example> element, 62, 64
- <family> element, 232
- <Fault>, 622–632
- <Fault> element, 614, 622–632
- <feed> element, 556
- :first\_letter pseudo-element, 726
- <first> element, 17, 35, 73, 114, 161, 170, 216, 221, 232
- :first-line pseudo-element, 726
- FirstName element, 264, 814, 816
- <font> element, 693, 736, 737, 914
- <footnoteNumber> element, 707
- <form> element, 305, 844, 845, 847
- <g> element, 776, 788, 790
- <generation> element, 234
- getTemperature element, 634
- <given> element, 98, 232
- global elements, 156–160
- <GPS> element, 107
- group elements, 365
- <guid> element, 531
- <h1> element, 564, 693, 752, 843, 913, 914
- <h2> element, 696, 752
- <h3> element, 696
- <head> element, 812, 843, 848, 912, 914
- <Header>
  - mustUnderstand attribute, 621–622
  - relay attribute, 622
  - role attribute, 622
- <Header> element, 614, 620–632
- <heading> element, 718
- <hr> element, 743, 764
- <html> element, 62, 78, 139, 743, 748, 750, 807, 812, 816, 913, 915
- <ID> element, 41
- <id> element, 535
- <iframe> element, 787, 866, 868
- <image> element, 781
- <img> element, 433, 701, 721, 743, 845, 847
- <im:nickname> element, 556
- <info> element, 565
- <input> element, 452, 844, 847
- <interfaces> element, 635
- invalid element names, 161
- Invoice element, 260, 304
- Invoices element, 265
- is\_content\_element method, 552
- Item element, 261, 265
- item element, 333, 365
- items element, 362, 363, 522
- <keyword> element, 704, 707, 708
- <knows> element, 123, 128, 131, 176, 223
- <label> element, 844, 847
- <last> element, 17, 115, 162, 170, 232, 234, 875
- LastName element, 301
- <latitude> element, 108, 116, 189, 197, 199, 226
- <li> element, 751, 753, 913, 915
- library element, 359, 370
- <line> element, 773, 776, 916
- <link> element, 535, 751, 848
- <list> element, 196–197, 728
- <listitem> element, 699
- literal result elements, 296, 351
- local elements, 156–157
- <location> element, 107, 116
- <longitude> element, 108, 116, 189, 197, 199, 226
- <member> element, 589

## elements (continued)

---

### elements (continued)

- <messages>, 634
- <messages> element, 634
- <meta> element, 746, 750, 843, 912, 914
- <metadata> element, 783, 784
- methodName element, 577
- <middle> element, 12, 17, 49, 110, 114, 115, 170, 172, 216, 232
- myNewBib element, 352
- mySubmit xforms:submission element, 827
- <name> element, 12, 16, 17, 78, 80, 83, 96, 106, 107, 151, 155, 160, 162, 164, 165, 168, 172, 177, 200, 203, 216, 231, 882, 883
- naming, 160–161
- nested elements, 406
- nesting, 744
- nesting of, 743
- nom element, 875
- <noscript> element, 763
- <note> element, 45
- object element, 806
- <occupants> element, 899
- <ol> element, 728
- <option> element, 742
- optional elements, 105
- <Order> element, 30, 43
- <Orders> element, 41
- output element, 560
- <p> element, 31, 32, 35, 69, 76, 78, 85, 549, 553, 564, 699, 701, 751, 758, 844, 882, 913
- <page> element, 705, 707, 714
- <pageNumber> element, 705
- <para> element, 699
- <paragraph> element, 250, 280, 704, 705, 707, 712
- <parent> element, 16
- parsererror document element, 457
- Part element node, 271
- <path> element, 777, 780, 917
- <payment> element, 218
- payments\_ table container element, 848
- payments\_table <div> element, 848
- <PAYMENTSTREAM> elements, 862
- <person> element, 68, 72, 73, 76, 78, 83, 264, 267, 296, 299, 300, 303, 814
- <pers:person> element, 72
- <pers:title> element, 69
- <phone> element, 126, 186, 192, 193, 195, 241, 885, 886
- <polygon> element, 776, 785, 789, 790, 791, 793, 794, 797, 798, 799, 916, 919
- polyGroup element, 920
- <portTypes>, 635
- <portTypes> element, 635
- <Product> element, 41
- pseudo-elements, 726
- pubDate element, 552
- publisher element, 366
- <pullQuote> element, 712
- <purchaseOrder> element, 258, 910
- qualification, 154–155
- qualified form, 161
- <rdf:RDF> element, 527, 529, 531, 616
- <rect> element, 771, 774, 776, 916
- <reference> element, 704, 707
- RELAX NG patterns
  - cardinality, 216–217
  - compared to attributes, 216
  - connector patterns and grouping, 217–220
  - co-occurrence constraints, 221
  - empty patterns, 222–223
  - enumerated values, 220–221
  - mixed content patterns, 222
- <r  sum  > element, 68
- <review> element, 712
- <root> element, 25, 33–34, 100, 134, 151, 348, 454, 459, 560, 563, 743, 902
- <row> element, 398
- <rss> element, 538, 563
- rss:channel element, 567
- rules
  - case sensitivity, 35
  - matching start- and end-tags, 31–32
  - naming conventions, 34–35
  - properly nested, 32–33
  - root elements, 33–34
  - whitespace, 36–39
- <schema> element, 151, 153–155, 156, 158, 160, 161, 175, 179, 209
- <script> element, 46, 756, 763, 785, 788, 866
- <section> element, 259, 264, 265, 278, 279, 280, 764
- <separator> element, 764

- <sequence> element, 151, 168, 175, 176
- <service>, 638–641
- <service> element, 638–641
- <simpleContent> element, 185
- <simpleType> element, 157, 178, 193, 194, 195, 197, 199, 216, 886
- <soap:binding>, 636
- <soap:binding> element, 636
- <soap:body>, 637
- <soap:Body> element, 614, 636, 637, 677, 680
- <soap:Envelope> element, 677, 680
- <soap:operation>, 636–637
- <soap:operation> element, 636–637
- <span> element, 843
- <strong> element, 32, 109, 138, 139, 222, 236, 239, 888, 890, 913
- <style> element, 736, 763
- <summary> element, 764
- <svg> element, 771, 789, 793, 866, 867
- <system.web> element, 686
- <table> element, 719
- <Tag> element, 38
- <Text> element, 623
- <text> element, 782, 870
- <title> element, 67–68, 75, 83, 264, 367, 550, 552, 560, 561, 699, 714, 783, 784, 788, 843
- <tr> element, 863
- <types>, 633–634
- <types> element, 633–634
- <ul> element, 701, 728
- unique root HTML, 743
- use element, 917
- <version> element, 163
- <website> element, 885, 886
- weight element, 498
- wildcards, 163–165
- <wrox:countries> element, 677
- <wrox:GetCountriesResponse> element, 677
- xforms:action element, 835
- xforms:alert element, 819, 820
- xforms:bind, 828–833
- xforms:bind element, 813, 828–833, 834
- xforms:dispatch element, 835
- xforms:filename element, 819
- xforms:help element, 820
- xforms:hint element, 820
- xforms:input, 817–818
- xforms:input element, 815, 816, 817–818, 836
- xforms:instance, 813–814
- xforms:instance element, 813–814, 815, 816, 822, 824, 825, 826, 832, 833
- xforms:itemset element, 826, 827
- xforms:label element, 819, 820, 826, 827
- xforms:load element, 835
- xforms:mediatype element, 819
- xforms:message element, 835
- xforms:model element, 812–813, 814, 815, 828
- xforms:output element, 818–819
- xforms:range element, 819–820
- xforms:rebuild element, 835
- xforms:recalculate element, 835
- xforms:refresh element, 835
- xforms:reset element, 836
- xforms:revalidate element, 836
- xforms:secret element, 818, 821–822, 825, 827, 921
- xforms:select, 821–822
- xforms:select1 element, 821, 822–824, 826, 827, 921
- xforms:send element, 836
- xforms:setfocus element, 836
- xforms:setvalue element, 836
- xforms:submission element, 813, 814–815, 820, 824, 832, 921
- xforms:submit element, 814, 820–821, 827, 921
- xforms:textarea element, 818
- xforms:trigger element, 820, 835
- xforms:upload element, 819
- xforms:value element, 821, 827
- xhtml elements, 239
- <xhtml> element, 743
- xhtml:body element, 560
- xhtml:br element, 239
- xhtml:h1 element, 560, 561
- xhtml:html element, 560
- <xhtml:object> element, 239
- xhtml:p element, 561
- <xhtml:script> element, 239
- <xhtml:title> element, 69, 560
- xmml:Book element node, 270
- xmml:SizeAvailable element, 827
- xmml:ToppingAvailable elements, 827

## elements (continued)

---

### elements (continued)

- xs:date element, 834
- xsl: apply-imports element, 940
- xsl: apply-templates element, 941
- xsl: attribute element, 941–942
- xsl: attribute-set element, 942
- xsl: call-template element, 943
- xsl: choose element, 943
- xsl: comment element, 944
- xsl: copy element, 944
- xsl: copy-of element, 945
- xsl: decimal-format element, 945–947
- xsl: element element, 947
- xsl: fallback element, 948
- xsl: for-each element, 948–949
- xsl: if element, 949
- xsl: import element, 949
- xsl: include element, 950
- xsl: key element, 950
- xsl: message element, 951
- xsl: namespace-alias element, 951–952
- xsl: number element, 952–953
- xsl: otherwise element, 953
- xsl: output element, 954–955
- xsl: param element, 956
- xsl: preserve-space element, 956
- xsl: processing-instruction element, 957
- xsl: sort element, 957–958
- xsl: strip-space element, 958
- xsl: stylesheet element, 958–959
- xsl: template element, 959
- xsl: text element, 960
- xsl: transform element, 960–961
- xsl: value-of element, 961
- xsl: variable element, 961–962
- xsl: when element, 962
- xsl: with-param element, 962–963
- xsl: apply-templates element, 253, 296–297, 300, 304, 307, 312, 314, 315, 317, 891, 940
- <xsl:apply-templates>, 296–297
- xsl:attribute element, 301
- <xsl:call-template> element, 322
- <xsl:choose> element, 306, 307, 308–311, 330, 891
- xsl:copy element, 287, 299, 300, 302
- <xsl:copy>, 299–303
- <xsl:copy-of> element, 287, 299–305, 332
- xsl:decimal-format element, 323
- xsl:element element, 303, 941, 942
- <xsl:for-each> element, 267, 268, 272, 311–312, 314, 342, 362, 943
- xsl:if element, 306, 307, 318, 362, 891
- <xsl:if>, 306–308
- <xsl:if> element, 306–308
- <xsl:namespace-alias> element, 951
- xsl:non-matching-substring element, 327, 328, 329
- xsl:otherwise element, 309, 891
- <xsl:output> element, 680
- xsl:param element, 320
- <xsl:Param> element, 677
- xsl:parameter element, 320
- xsl:result-document element, 330, 332
- <xsl:sort> element, 312–314, 327, 362
- <xsl:stylesheet> element, 295–296, 320, 322, 333, 564, 676, 968
- for XSLT 2.0, 323
- <xsl:template> element, 293, 296, 297, 307, 318, 940
- xsl:text element, 318, 563
- xsl:transform element, 295
- <xsl:value-of> element, 254, 259, 264, 273, 280, 287, 297–299, 301, 318, 321, 334–335
- xsl:variable element, 320, 362
- xsl:when element, 309, 891
- xsl:with-param element, 322
- xs:token element, 834
- zero-digit attribute, xsl:decimal-format element, 946
- ZipCode element, 258
- elements **attribute**
  - xsl:preserve-space, 956
  - xsl:strip-space, 958
- ELEMENTS **keyword, 400**
- elementValue **variable, 896**
- elliptical arc **command, 779**
- em **absolute unit, SVG, 774**
- <em> **element, 32, 109, 138, 139, 222, 236, 239, 888, 890**
- <email> **element, 885, 886**
- emp **prefix, 882**
- empty content, 110

- empty **element syntax**, 49–50
- empty **elements**, 163, 222, 740, 743–744
- EMPTY **keyword**, 110
- empty **patterns**, 222–223
- empty-cells **property**, CSS, 695
- encodeURIComponent, 646
- encoding **attribute**, 50–51, 52, 740, 750, 815, 912, 954
- encoding **rules**, 613
- encodingStyle **attribute**, 615
- end <...> **characters**, 563
- endCDATA **event**, *LexicalHandler* **interface**, 515
- endDocument **event**, 487
- endDocument **function**, 491, 502, 894, 895
- endDTD **event**, *LexicalHandler* **interface**, 515
- endElement **event**, 487, 497
- endElement **function**, 539, 896, 900
- endElementNS **method**, 551, 899, 900
- endEntity **event**, *LexicalHandler* **interface**, 515
- end-of-line **logic**, 38
- endPrefixMapping **event**, 488, 499
- end-tags, 23, 24, 31–32, 73, 250, 358
- enhancing, display with Scalable Vector Graphics (SVG), 865–868
- enterprise-level database management system, 414
- entities
  - built-in, 131–132
  - character, 132–133
  - general, 134–138
  - parameter, 139–140
  - skipped, 498–499
- ENTITIES **attribute**, 121
- ENTITIES **attribute**, 123
- ENTITY **attribute**, 121
- ENTITY **attribute**, 123
- ENTITY **declaration**, 134, 141, 147
- ENTITY **functionality**, 147
- ENTITY **keyword**, 134, 139
- entity **reference**, 60–61, 132
- EntityResolver **interface**, 510
- <entry> **element**, 535
- enumerated **list**, 124
- Enumerated List **attribute**, 121
- enumerated **values**, 220–221
- enumeration **facet**, 194
- Enumeration Pattern, 220
- <enumeration> **facet**, 196
- Enum.Parse( ) **method**, 684
- <env:Body> **element**, 636
- <Envelope> **element**, 612, 614, 621
- envelopes, 613
- env:encodingStyle **attribute**, rdf:RDF **element**, 616
- environment **variable**, 357
- Environment Variables **window**, 290
- equals **sign (=)** **references**, 34, 228
- error **event**, 505
- Error **events**, 835
- error **messages**, 341
- ErrorHandler **class**, 504–509
- errors, providing location, 502–504
- escaping **illegal characters**, 60–61
- ev **namespace prefix**, 835
- eval( ) **function**, 653
- event **callbacks**, 497
- Event **object**, 662
- event **object**, 662
- event-handler **attributes**, 763
- events
  - element, 491–494
  - ignorableWhitespace, 498
  - Simple API for XML (SAX)
    - ContentHandler **interface**, 487–504
    - DTDHandler **interface**, 509
    - EntityResolver **interface**, 510
    - ErrorHandler **interface**, 504–509
    - extension **interfaces**, 514–515
    - features, 510–513
    - properties, 513–514
    - XForms, 834–835
- evt **object**, 785, 793
- evt.target <polygon> **element**, 799
- ex **absolute unit**, SVG, 774
- <example> **element**, 62, 64
- exception **mechanism**, 502
- exclamation **mark (!)**, 99
- exclude-result-prefixes **attribute**
  - xsl:stylesheet, 958
  - xsl:transform, 960
- exec **function**, 796
- executable **business processes**, 603
- exercise **solutions**, 873–921

## eXist

- clients, 386–387
- installing, 382–394
- interacting with
  - choosing interfaces, 394
  - eXist client, 386–387
  - other interfaces, 394
  - REpresentational State Transfer (REST) interface, 390–394
  - web interface, 384–386
  - Web-based Distributed Authoring and Versioning (WebDAV), 387–389
  - XML Integrated Development Environment (IDE), 389–390
- obtaining, 382–394
- `exist( )` **method**, 893
- eXist **documentation**, 388
- eXist XQuery engine**, 389
- expand/collapse functionality**, 30
- Expat**, 24
- EXPLICIT **option**, 405
- expression **type**, 966
- expressions**
  - conditional, 367–368
  - FLWOR
    - conditional expressions, 367–368
    - for expressions, 362–365
    - filtering with `where` clause, 365–366
    - sorting with `order by` clause, 366–367
  - for, 362–365
  - structure of XPath, 278–281
  - XQuery
    - FLWOR expressions, 362–367
  - XSLT 2.0, 967–968
- extensibility**, 12–13
- extensible acronym**, 555
- Extensible Markup Language (XML)**
  - benefits of
    - extensibility, 12–13
    - parsers, 11–12
  - compared to HyperText Markup Language (HTML), 13–14
  - data files
    - binary files, 4
    - history of markup languages, 6–7
    - text files, 5

- description, 7–10
- document types, 17–18
- grouping of information, 15–17
- growth of, 376–377
- SQL Server 2000, 395–425
- standards
  - components, 19–20
  - World Wide Web Consortium (W3C), 18–19
- storing
  - conventional databases, 379–381
  - file systems, 378–379
  - native databases, 381–382
- uses
  - distributed computing, 21
  - e-commerce, 21–22
  - reducing server load, 20
  - website content, 20–21
- XML-based data, compared to relational data, 377

## Extensible Stylesheet Language Transformations (XSLT)

- conditional processing
  - `<xsl:choose>` element, 308–311
  - `<xsl:if>` element, 306–308
- declarative programming, 292–293
- defined
  - presenting XML content, 288
  - restructuring XML, 288
- elements
  - `xsl: apply-imports`, 940
  - `xsl: apply-templates`, 941
  - `xsl: attribute`, 941–942
  - `xsl: attribute-set`, 942
  - `xsl: call-template`, 943
  - `xsl: choose`, 943
  - `xsl: comment`, 944
  - `xsl: copy`, 944
  - `xsl: copy-of`, 945
  - `xsl: decimal-format`, 945–947
  - `xsl: element`, 947
  - `xsl: fallback`, 948
  - `xsl: for-each`, 948–949
  - `xsl: if`, 949
  - `xsl: import`, 949
  - `xsl: include`, 950
  - `xsl: key`, 950
  - `xsl: message`, 951
  - `xsl: namespace-alias`, 951–952

- xsl: number, 952–953
  - xsl: otherwise, 953
  - xsl: output, 954–955
  - xsl: param, 956
  - xsl: preserve-space, 956
  - xsl: processing-instruction, 957
  - xsl: sort, 957–958
  - xsl: strip-space, 958
  - xsl: stylesheet, 958–959
  - xsl: template, 959
  - xsl: text, 960
  - xsl: transform, 960–961
  - xsl: value-of, 961
  - xsl: variable, 961–962
  - xsl: when, 962
  - xsl: with-param, 962–963
  - example, 289–292
  - foundational elements
    - <xsl:apply-templates> element, 296–297
    - <xsl:stylesheet> element, 295–296
    - <xsl:template> element, 296
  - functions, 323, 963–967
  - influencing output with <xsl:output> element, 306
  - modes, 314–319
  - named templates, 322
  - parameters, 320–322
  - procedural programming, 292
  - processors, 288–289
  - Saxon XSLT processor, 289–292
  - transforming RDF Site Summary (RSS) with
    - browser processing, 564–565
    - generating feed from existing data, 558–559
    - pre-processing feed data, 565
    - processing feed data for display, 561–563
    - syndication formats, 565–567
  - using information from source tree
    - <xsl:copy> element, 299–303
    - <xsl:copy-of> element, 303–305
    - <xsl:value-of> element, 297–299
  - variables, 320–322
    - <xsl:call-template> element, 322
    - <xsl:for-each> element, 311–312
    - <xsl:sort> element, 312–314
  - XSLT 2.0
    - expressions, 967–968
    - grouping, 324–327
      - Muenchian Method, 968–969
      - multiple outputs, 330–332
      - non-XML input and string handling, 327–330
      - user-defined functions, 332–334
        - using, 968–969
        - and XQuery 1.0, 969
          - xsl:value-of changes, 334–335
  - Extensible Stylesheet Language Transformations (XSLT) 2.0**
    - compared to XQuery, 342–343
    - development of, 341–342
    - expressions, 967–968
    - grouping, 324–327
      - Muenchian Method, 968–969
      - multiple outputs, 330–332
      - non-XML input and string handling, 327–330
      - user-defined functions, 332–334
        - using, 968–969
        - and XQuery 1.0, 969
          - xsl:value-of changes, 334–335
  - Extensible Stylesheet Language (XSL), 19**
  - extension interfaces, Simple API for XML (SAX), 514–515**
    - <extension> **declaration, 186**
    - extension-element-prefixes **attribute**
      - xsl:stylesheet, 958
      - xsl:transform, 960
  - extensions, to XQuery in SQL Server 2005, 416–423**
  - external entity **declaration, 135**
  - external subset **declarations, 100**
  - external **variable, 357**
  - externalEntityDecl **event, DeclHandler interface, 514**
  - external-general-entities feature, 498**
  - external-parameter-entities feature, 498**
  - ExtractValue **function, 431**
  - extraneous whitespace, 38**
- ## F
- facets, 241**
  - FactorialFunction.xquery **query, 372**
  - false( ) **function, 274, 832**
  - <family> **element, 232**
  - fatal errors, 64**

# fatalError event

---

- fatalError **event**, 505
- fault codes, 622
- <Fault> **element**, 614, 622–632
- Feed **class**, 546–548
- feed data
  - generating from existing data, 558–559
  - pre-processing, 565
  - processing for display, 561–563
- Feed **object**, 537
- feed reader.py **file**, 554
- feed\_handler **instance**, 544
- feed\_handler.py **file**, 540, 552
- feed\_reader.py **file**, 540, 543
- <feed> **element**, 556
- FeedHandler **class**, 548–552, 553
- feed.py **file**, 540, 546
- FeedReader **class**, 543–546
- feed-reader **user interface**, 523
- feeds.txt **file**, 543
- Ferna **Distribution**, 26, 30
- fetchData( ) **function**, 651
- fetchSuggestions( ) **method**, 659, 670, 671, 672, 681
- file **protocol**, 814
- file systems
  - building, 379
  - document size, 378
  - indexes, 379
  - updates, 378–379
- files
  - binary, 4
  - data, 3–7
  - text, 5
- fill **attribute**, 771, 775, 782, 785, 795, 799
- fill-opacity **attribute**, 790
- filter **events**, 516
- filtering, with where **clause**, 365–366
- filters, Simple API for XML (SAX), 516
- Firebug **extension**, 770
- Firefox, Mozilla
  - network transports, 591–594
  - XLink support in, 721–724
- firewalls, network transports, 583–584
- first **clause**, 419
- first **element**, 114, 216, 221
- :first\_letter **pseudo-element**, 726
- <first> **element**, 17, 35, 73, 161, 170, 232
- firstChild **method**, 867
- firstChild **property**, 458, 459, 461
- :first-line **pseudo-element**, 726
- FirstName **attribute**, 300, 301
- FirstName **element**, 264, 814, 816
- fixed **attribute**, 162, 163, 180, 183
- fixed **positioning**, 717–718
- fixed **values**, 126, 162–163, 180
- float **positioning**, 706, 709–711, 712
- float **property**, 709
- float **type**, 188, 193
- float **values**, 198
- floatated **positioning**, 706
- floor( ) **function**, 275, 931
- FLWOR **expressions**
  - conditional expressions, 367–368
  - for expressions, 362–365
  - filtering with where clause, 365–366
  - sorting with order by clause, 366–367
- fn: **prefix**, 936
- FOAF (Friend-of-a-Friend) Personal Profile Document, 567
- FOAF **profile**, 903
- foaf:maker **property**, 901
- foaf:mbox\_sha1 **property**, 901
- foaf:name **property**, 901
- foaf:nick **property**, 901
- foaf:Person **property**, 901
- foaf:weblog **property**, 901
- focus( ) **method**, 661
- following **axis**, 260, 266–267, 348, 926
- following-sibling **axis**, 260, 266, 268, 348, 926
- font **property**, CSS, 694
- <font> **element**, 693, 736, 737, 914
- font-family **property**, 694, 696
- font-size **attribute**, 782
- font-size **property**, CSS, 694
- font-style **property**, CSS, 694
- font-variant **property**, 694
- font-weight **property**, 694, 696
- foo **division**, XHTML document, 758
- <footnoteNumber> **element**, 707
- for ... in **expression**, 362
- for ... in ... **to option**, 363
- for **clause**, 362, 423, 892

- for **expressions**, 362–365
- for **statement**, 362, 365, 368, 370
- FOR XML AUTO **clause**, 401–404
- FOR XML **clause**, 395, 396–425
- FOR XML EXPLICIT **clause**, 405
- FOR XML EXPLICIT **option**, 404
- FOR XML PATH **clause**, 405–409
- FOR XML PATH **option**, 404
- FOR XML RAW **clause**, 396–401
- ForIn2.xquery **query**, 363
- ForIn3Out.xml **query**, 364
- ForIn3.xquery **query**, 364
- ForIn4Out.xml **query**, 364
- ForIn4.xquery **query**, 364
- ForIn.xquery **query**, 362
- form **attribute**, 161, 179
- <form> **element**, 844, 845, 847
- Formal Public Identifiers (FPIs)**, 101
- format **attribute**, xsl:number, 952
- format-number( ) **function**, 323
- formats**, syndication, 565–567
- formatted text**, 32
- formatting objects**, 19
- FormFaces, 808
- Forms **module**, XHTML, 760
- formsPlayer plugin**, 806
- FPIs (Formal Public Identifiers)**, 101
- fractionDigits facet**, 194
- frames**, adding to main page, 868–872
- Frames **module**, XHTML, 760
- Frameset documents**, 741
- frameset **flavor**, 90
- French characters**, 874
- Friend-of-a-Friend (FOAF) Personal Profile Document**, 567
- from **attribute**, xsl:number, 952
- From **element nodes**, 305
- FrontPage**, Microsoft, 753
- full-axis feature**, 348
- full-text index**, 416
- function types**, 939
- functional language**, 293
- functions**
  - //element( ) **function**, 284
  - //element(\*, xs:token) **function**, 284
  - ActiveXObject( ) **function**, 588, 647
  - alert( ) **function**, 15, 450, 919
  - Boolean, 274–275
  - boolean( ) **function**, 274, 964
  - boolean **function**-available **function**, 964
  - Calculate **function**, 857, 858
  - CalculatePassThrough **function**, 852, 858
  - ceiling( ) **function**, 275
  - characters **function**, 495, 539, 551, 896
  - child::comment( ) **function**, 928
  - child::text( ) **function**, 928
  - collection( ), 348
  - collection( ) **function**, 346, 348
  - comment( ), 928
  - comment( ) **function**, 928
  - concat( ) **function**, 276, 369
  - contains( ) **function**, 276
  - count( ) **function**, 258, 275, 298, 368, 369–370
  - current-date( ) **function**, 283
  - current-DateTime( ) **function**, 283
  - current-group( ) **function**, 324
  - current-grouping-key( ) **function**, 324, 326, 332
  - current-time( ) **function**, 283
  - day-from-date( ) **function**, 283
  - doc( ), 347–348
  - doc( ) **function**, 346, 347–348
  - document( ) **function**, XSLT, 323, 347
  - doPost( ) **function**, 587
  - drawChart **function**, 868
  - elementDecl **function**, 898
  - endDocument **function**, 491, 502, 894, 895
  - endElement **function**, 539, 896, 900
  - ENTITY **functionality**, 147
  - eval( ) **function**, 653
  - exec **function**, 796
  - expand/collapse **functionality**, 30
  - Extensible Stylesheet Language Transformations (XSLT)
    - types, 966–967
    - XPath, 966
  - ExtractValue **function**, 431
  - false( ) **function**, 274, 832
  - fetchData( ) **function**, 651
  - floor( ) **function**, 275, 931
  - format-number( ) **function**, 323
  - functional language**, 293
  - functions attribute**, 813
  - generate-id( ) **function**, 323

## functions (continued)

---

### functions (continued)

`getCenter`(`polygon`) function, 793, 796, 797  
`getDateValue`( ) function, 862  
`getDom`( ) function, 452  
`getDomFromXml`( ) function, 456  
`getElementById` function, 479, 860  
`getFeature` function, 511, 514  
`getFloatValue`( ) function, 861, 862  
`get-order-total`( ) function, 284  
`getPolygonDetails` function, 919  
`getPrepaidDays`( ) function, 861, 862  
`getProperty` function, 514  
`getRotateAngle` helper function, 797  
`getRotateAngle`(`polygon`) function, 796  
`getSelectValue`( ) function, 861, 862  
`getTransformBlock`(`poly`, `index`) function, 795  
`getXml`( ) function, 453  
`handleClick` function, 785  
`handleKeyUp`( ) function, 667  
`handleReadyState`( ) function, 588  
`handleReadyStateChange`( ) function, 589, 651, 657–658, 685  
`hideProgress` function, 860, 863  
`hour-from-time`( ) function, 283  
`id`( ) function, 256, 275  
`init`(`evt`) function, 793, 794  
`input`, 346–348  
`input` functions, XQuery  
    `collection`( ) function, 348  
    `doc`( ) function, 347–348  
`key`( ) function, 256, 323, 968  
`lang`( ) function, 274  
`last`( ) function, 252, 254, 275, 318, 932  
`local-name`( ) function, 275  
`lower-case`( ) function, 278  
`main`( ) function, 489, 491, 544, 546  
`matches`( ) function, 282  
`mouseout` function, 795  
`mouseout`(`evt`) functions, 793  
`mouseup` function, 795  
`mouseup`(`evt`) functions, 793  
`move`(`evt`) function, 795  
`move`(`evt`) function, 799  
`moveToFront`(`polygon`) function, 793, 797  
`name`( ) function, 257, 275  
`namespace-uri`( ) node-set function, 275  
`node`( ), 928  
`node`( ) function, node tests, 928  
`node-set`, 275  
`node-set` `current`( ) function, 963  
`node-set` `document` function, 963  
`node-set` `key` function, 965  
`normalize-space`( ) function, 276  
`normalize-unicode`( ) function, 282  
`not`( ) function, 275  
`number`( ) function, 275  
`numeric`, 275–276  
`numeric` functions, 275–276  
`object` `system-property` function, 965  
`onreadystatechange` function, 862  
`parse` function, 508, 511  
`parseFloat`(`string`) function, 796  
`parseFromString`( ) function, 456  
`position`( ) function, 252, 254, 275, 318, 933  
`processing-instruction`( ) function, 928  
`processing-instruction`(`Literal`) function, 928  
`receiveLoanInformation` function, 862, 868  
`recursive` function, 371  
`reset`( ) function, 792, 793, 798, 799  
`resolveEntity` function, 510  
`rotatePolygon` function, 797  
`rotateString` function, 796  
`round`( ) `number` function, 275  
`SAXNotRecognizedException` function, 511  
`SAXNotSupportedException` function, 511  
`scramble`( ) function, 792, 793, 798  
`self::node`( ) function, 314  
`setDTDHandler` function, 509  
`setEntityResolver` function, 510  
`setFeature` function, 511, 514  
`showProgress` function, 860  
`showProperties`( ) function, 452, 453, 456  
`sql:column` function, 422, 423  
`sql:function` function, 422  
`sql:variable` function, 423  
`startDocument` function, 491, 497, 895  
`startElement` function, 491, 497, 539, 895  
`starts-with`( ) function, 276, 433  
`str` function, 544  
`string`, 276–278  
`string`( ) function, 276  
`string` `format-number` function, 964

string generate-id function, 965  
 string unparsed-entity-uri function, 965  
 StringBuffer function, 894  
 string-length( ) function, 276  
 submitLoanInformation function, 864  
 substring( ) function, 276  
 substring-after( ) function, 276  
 substring-before( ) function, 276  
 sum( ) function, 276  
 text( ), 928  
 text( ) function, 928  
 tokenize( ) function, 282  
 translate( ) function, 276  
 true( ) function, 275, 832, 936  
 types, 939  
 unparsed-text( ) function, 327, 328, 329  
 UpdateXML function, 431, 435  
 upper-case( ) function, 278  
 user-defined, 332–334, 371–372  
 user-defined functions, 342, 371–372  
 xml\_schema\_namespace intrinsic function, 425  
 XPath, 929–937  
 XPath 2.0, 282, 936–937  
 XQuery  
   concat( ), 369  
   count( ), 369–370  
 XSLT  
   types, 966–967  
   XPath, 966  
 for XSLT 2.0, 323  
 XSLT 2.0, 969  
 years-from-yearMonthDuration( ) function, 283  
 functions **attribute, 813**

## G

<g> element, **776, 788, 790**  
**GAC (global assemble cache), 291**  
**Garrett, Jesse James, 645**  
 gDay type, **188**  
**gender attribute declaration, 884, 886**  
**general entities, 134–138**  
 generate-id( ) function, **323**  
**generating, feed from existing data, 558–559**  
 <generation> element, **234**

**Geographic Information Systems (GIS), 768**  
 GET method, **464, 524, 581**  
 get\_feed\_uris method, **545**  
 get\_formatted\_date method, **548**  
 get\_newest\_items method, **546**  
 get\_uris method, **541, 543, 545**  
 getCenter(polygon) function, **793, 796, 797**  
 getChannels( ) method, **578, 582, 596**  
 getColumnNumber method, **502**  
 GetCountries.aspx file, **659, 673, 678**  
 GetCountries.aspx.cs file, **659, 673, 678**  
 getDateValue( ) function, **862**  
 getDateValue( ) method, **861**  
 getDom( ) function, **452**  
 getDomFromXml( ) function, **456**  
 getElementById function, **479, 860**  
 getElementsByTagName( ) method, **475, 479, 480, 757**  
 getElementsByTagNameNS( ) method, **479, 480**  
 GetFailureXML( ) method, **627**  
 getFeature function, **511, 514**  
 getFloatValue( ) function, **861, 862**  
 getIndex method, **492**  
 getLength method, **492**  
 getLineNumber method, **502**  
 getLocalName method, **493**  
 getNamedItem( ) method, **471, 472**  
 getNamedItemNS method, **NamedNodeMap object, 471**  
 get-order-total( ) function, **284**  
 getOwnerDocument( ) method, **447**  
 getPolygonDetails function, **919**  
 getPrepaidDays( ) function, **861, 862**  
 getProperty function, **514**  
 getPublicId method, **503**  
 getQName method, **493**  
 GetQuantityDiscount( ) method, **612**  
 getRadioOperators( ) method, **906**  
 getRate( ) method, **682, 683–684, 685**  
 getRotateAngle helper function, **797**  
 getRotateAngle(polygon) function, **796**  
 getSelectValue( ) function, **861, 862**  
 getSystemId method, **502**  
 getTemperature element, **634**  
 getTemperature method, **636**  
 getTop( ) method, **665**  
 getTransformBlock attribute, **795**

# getTransformBlock(poly, index) function

---

`getTransformBlock(poly, index)` **function**, 795

`getType` **method**, 493

`getURI` **method**, 493

`getValue` **method**, 493, 494

`getXml( )` **function**, 453

`.gif` **file**, 86

**gigabytes**, 375

**GIS (Geographic Information Systems)**, 768

`<given>` **element**, 98, 232

**Gleaning Resource Descriptions from Dialects of Languages (GRDDL)**, 561

**global assemble cache (GAC)**, 291

**global attributes**, 84, 177–178, 179

**global declarations**, 156

**global elements**, 156–160

`global <first>` **declaration**, 161

**global variables**, 322, 794

`gMonth` **type**, 188

`gMonthDay` **type**, 188

**GNOME**, 388

**GNU Public License (GPL)**, 24

**GoLive**, 770

**Google Reader**, 536

`goToSelection( )` **method**, 669

`goToSuggestion( )` **method**, 668

**GPL (GNU Public License)**, 24

`<GPS>` **element**, 107

`grammar` **declaration**, 235

`grammar` **pattern**, 235

**grammars**

combining, 227–236

extensions, 234

nested, 235–236

restrictions, 234

reusing, 227–236

**graphics, Scalable Vector Graphics (SVG)**, 769

**GRDDL (Gleaning Resource Descriptions from Dialects of Languages)**, 561

**Great Push Revolution**, 526

**green pages**, 602

`group` **elements**, 365

`group` **pattern**, 217

`<group>` **declarations**, 167–168, 175, 183, 216

`<group>` **references**, 170–171

`group-by` **attribute**, 326

`groupedContacts.xslt`, 325

`groupedContacts.xslt` **instruction**, 332

**grouping**

and connector patterns, 217–220

elements, 324

Scalable Vector Graphics (SVG), 776

XSLT 2.0, 324–327

`grouping-separator` **attribute**

`xsl:decimal-format`, 945

`xsl:number`, 953

`grouping-size` **attribute**, `xsl:number`, 953

**groups**

combining sequences and choices using, 108

name-classes, 238–239

`<guid>` **element**, 531

`gYear` **type**, 188

`gYearMonth` **type**, 188

## H

**H command**, 778

`<h1>` **element**, 564, 693, 752, 843, 913, 914

`<h2>` **element**, 696, 752

`<h3>` **element**, 696

`HamSearchPort portType` **interface**, 907

`handleAddToCartResponse( )` **method**, 627

`handleClick` **function**, 785

`handleKeyDown( )` **method**, 669

`handleKeyUp( )` **function**, 667

`handleReadyState( )` **function**, 588

`handleReadyStateChange( )` **function**, 589, 651, 657–658, 685

`handler.feed.items` **instance**, 544

**handlers**

helpers, 552–555

XML markup, 548–552

`hasAttributes( )` **method**, `Node` **object**, 461

`hasChildNodes( )` **method**, `Node` **object**, 461

**hash mark (#)**, 133

`head` **element**, 812

`<head>` **element**, 843, 848, 912, 914

**header blocks**, 621

`<Header>` **element**, 614, 620–632

**headers**

`Accept`, 580

`Accept-Encoding`, 580

`Accept-Language`, 580

`<authentication>`, 621

- Content-Length, 581
- Content-Type, 581
- Content-Type HTTP, 747
- header blocks, 621
- <Header> element, 614, 620–632
- SOAPAction, 636
- <heading> **element, 718**
- height **attribute, 774, 787, 789**
- HelloWorld **method, 851**
- hexadecimal Unicode **value, 133**
- hexBinary **type, 187**
- hidden frames, 646**
- hideProgress **function, 860, 863**
- hideSuggestionsBox( ) **method, 670**
- highlightSuggestion( ) **method, 664, 665, 668**
- horizontal lineto **command, 778**
- hour-from-time( ) **function, 283**
- <hr> **element, 743, 764**
- href **attribute, 700, 722, 724, 764, 819, 949, 950**
- HTML (HyperText Markup Language)**
  - compared to Extensible Markup Language (XML), 13–14
  - forms
    - alternative to XForms, 838
    - compared to XForms, 804
    - grouping of information in, 14–15
    - pages, 288
- html **document element, XHTML, 812**
- html **element, 807, 812, 816**
- HTML markup, 737**
- html **output method, 563**
- html **prefix, 74, 80**
- <html> **element, 62, 78, 139, 743, 748, 750, 913, 915**
- HTTP (Hypertext Transfer Protocol), using for XML-RPC, 584–587**
- http://blogs.it/0100198/rss.xml **root element, 566**
- http-equiv **attribute, 746**
- http://icite.net/blog/?flavor=atom &smm=y: **root element, 567**
- http://localhost/WroxServices/
  - AutoSuggestDemo.html, **678**
- http://purl.org/net/morten/ blog/feed/rdf/
  - root element, 566**
- http://swordfish.rdfweb.org/people/libby/
  - rdfweb/webwho.xrdf **root element, 567**
- http://www.w3.org/2003/05/soap-envelope/
  - role/next, **622**
- http://www.w3.org/2003/05/soap-envelope/
  - role/none, **622**
- http://www.w3.org/2003/05/soap-envelope/
  - role/ultimateReceiver, **622**
- http://www.wiley.com/pers namespace, **883**
- http://www.wiley.com/soap/ordersystem namespace, **614**
- http://www.XMML.com/namespace, **816**
- http://xml.org/sax/features/
  - externalgeneral-entities **feature, 512**
- http://xml.org/sax/features/
  - externalparameter-entities **feature, 512**
- http://xml.org/sax/features/isstandalone
  - feature, 512**
- http://xml.org/sax/features/lexicalhandler/
  - paramaterentities **feature, 512**
- http://xml.org/sax/features/
  - namespaceprefixes **feature, 511**
- http://xml.org/sax/features/namespaces
  - feature, 511**
- http://xml.org/sax/features/resolvedtd-uris
  - feature, 512**
- http://xml.org/sax/features/stringinterning
  - feature, 513**
- http://xml.org/sax/features/
  - unicodenormalizationchecking **feature, 513**
- http://xml.org/sax/features/useattributes2
  - feature, 512**
- http://xml.org/sax/features/
  - useentity-resolver2 **feature, 513**
- http://xml.org/sax/features/uselocator2
  - feature, 512**
- http://xml.org/sax/features/validation
  - feature, 511**
- http://xml.org/sax/features/xml-1.1 **feature, 513**
- http://xml.org/sax/features/xmlns-uris
  - feature, 511**
- http://xml.org/sax/properties/
  - declaration-handler **property, 514**
- http://xml.org/sax/properties/
  - documentxml-version **property, 514**
- http://xml.org/sax/properties/lexicalhandler
  - property, 514**

**hybrid document type, 761**

**hyperlinks, 6**

**HyperText Markup Language (HTML)**

compared to Extensible Markup Language (XML),  
13–14

forms

alternative to XForms, 838

compared to XForms, 804

grouping of information in, 14–15

pages, 288

**Hypertext module, XHTML, 760, 762**

**Hypertext Transfer Protocol (HTTP), using for XML-RPC,  
584–587**

**hyphens (-), 135**

## I

**IBM MQSeries, 579**

**id( ) function, 256, 275**

**id attribute, 41, 83, 85, 121, 122–123, 126, 739, 740,  
744, 747, 813, 814, 815, 820, 825, 913, 914,  
958, 960**

**ID attribute nodes, 323**

**id elements, 220**

**ID mechanism, 122**

**id node-set, 931**

**ID selector, 701**

**id type, 966**

**<ID> element, 41**

**<id> element, 535**

**IDE (Integrated Development Environment), XML,  
389–390**

**identifiers**

public, 101–102

system, 100–101

**ID/IDREF mechanism, 131**

**IDL (Interface Definition Language), 601**

**IDLE (Integrated Development Environment tool), 541**

**IDREF attribute, 121, 122–123**

**IDREFS attribute, 121, 122–123**

**IETF (Internet Engineering Task Force), 86, 387, 533**

**if keyword, 367**

**if/else construct, 283**

**IFrame module, XHTML, 760**

**<iframe> element, 787, 866, 868**

**iframes, 646**

**ignorableWhitespace events, 487, 498**

**IIOIP (Internet Inter-ORB Protocol), 573, 574**

**IIS (Internet Information Services), 608, 842**

**illegal characters, 133**

**Illustrator, Adobe, 770**

**Image module, XHTML, 760, 762**

**Image object, 646**

**<image> element, 781**

**images, Scalable Vector Graphics (SVG), 780–781**

**<img> element, 433, 701, 721, 743, 845, 847**

**im:nickname property, 556**

**<im:nickname> element, 556**

**implementation property, Document interface, 479**

**implementations**

module, 761

network transports, 582

XML Document Object Model (DOM), 447–448

**implementing**

aggregators

address list reader, 541–542

application controller, 543–546

extending, 555–557

handler helpers, 552–555

model, 546–548

XML markup handler, 548–552

**GetRate( ) method, 683–684**

**suggest( ) method, 662**

**SuggestionProvider, 671–673**

**web services, 673–680**

**implied attributes, 127**

**implied values, 127**

**import processing instruction, 807**

**import utilities, 4**

**<import> declarations, 200–204**

**importNode method, Document interface, 480**

**in absolute unit, SVG, 774**

**in keyword, 363**

**IN\_CONTENT constant, 549**

**IN\_ITEM constant, 549**

**IN\_NONE constant, 549**

**include directive, 231–232, 235, 237**

**include pattern, 232**

**include statement, 233**

**<include> declarations, 204–206**

**includenamespaceprefixes attribute, xforms:**

**submission element, 815**

- incomingCall **event**, 763
- incorporating, content from other people, 757
- incremental **attribute**
  - xforms:input element, 818
  - xforms:range element, 820
  - xforms:select element, 822
  - xforms:select1 element, 822
- indent **attribute**
  - xforms:submission element, 815
  - xsl:output, 955
- infinity **attribute**, xsl:decimal-format **element**, 945
- <info> **element**, 565
- InfoPath, Microsoft, 836–837
- InfoPath 2003, 444
- InfoPath 2007, XML DOM used in, 481–482
- information set, 633
- Inherit **value**, clear **property**, 713
- inheritance, Cascading Style Sheets (CSS), 695–699
- iniFile **variable**, 329
- init( ) **method**, 474, 544, 618, 663, 670, 797
- init(evt) **function**, 793, 794
- Initialization **events**, XForms, 834, 835
- inline boxes, 703–704
- in-memory representation, 443
- inner content models, 168
- inner query, 407
- innerHTML **property**, 863
- input **element**, 452
- input functions, XQuery
  - collection( ) **function**, 348
  - doc( ) **function**, 347–348
- input key, handling, 662–663
- <input> **element**, 844, 847
- in-scope **namespaces**, 257
- insert **keyword**, 416, 418
- insert **operation**, 419
- INSERT **query**, 416
- insert **statement**, 428
- insertBefore( ) **method**, Node object, 461, 462
- installing
  - eXist, 382–394
  - MySQL, 426–427
  - Saxon XSLT processor
    - Java version, 290–291
    - .NET version, 291–292
- instance data, 813
- instance document, 150
- instructions, 296
- int **type**, 187
- integer **datatype**, 241
- integer **type**, 187
- Integrated Development Environment (IDE), XML, 389–390
- Integrated Development Environment tool (IDLE), 541
- Interaction **events**
  - XForms, 834
  - XForms Action Module, 835
- InterestOnlyOption loan program, 857
- Interface Definition Language (IDL), 601
- interfaces
  - ContentHandler
    - character content, 494–497
    - element events, 491–494
    - handling special commands with processing instructions, 499
    - ignorableWhitespace events, 498
    - namespace prefixes, 499
    - providing location of errors, 502–504
    - skipped entities, 498–499
    - stopping process, 500–502
  - Document, 478–480
  - DTDHandler, 509
  - EntityResolver, 510
  - ErrorHandler, 504–509
  - eXist
    - choosing, 394
    - other, 394
  - REpresentational State Transfer (REST) interface, 390–394
  - extension, 514–515
  - XML Document Object Model (DOM), 445–446
- <interfaces> **element**, 635
- interleave connector, 219–220
- internal entity **declaration**, 134
- internal parameter entity, 139
- internal subset **declarations**, 100
- internalEntityDecl **event**, DeclHandler **interface**, 514
- Internationalized Resource Identifier (URI), 535
- Internet Engineering Task Force (IETF), 86, 387, 533
- Internet Explorer Parser, 24
- Internet Information Services (IIS), 608, 842
- Internet Inter-ORB Protocol (IIOP), 573, 574

- Internet Protocol (IP), 572**
- Internet Topic Exchange, 576**
- Internet zone, 647**
- interoperability, web services, 603**
- `/intl/en` **directory, 87**
- into clause, 419**
- Intrinsic events module, XHTML, 760**
- invalid element names, 161**
- invalid value, 125**
- `Invoice` **element, 304**
- `Invoice` **element node, 260**
- `Invoices` **element node, 265**
- IP (Internet Protocol), 572**
- `is_content_element` **method, 552**
- ISO 639 language code, 745**
- `isPermalink` **attribute, 532**
- `isSupported` **method, Node object, 462**
- `item( )` **method, 475**
- `item` **class, 546–548**
- `item` **element, 333, 365, 561**
- `Item` **element nodes, 261, 265**
- `item` **method, NamedNodeMap object, 471**
- item start-tag, 363**
- items, XQuery, 361**
- `items` **element, 362, 363, 522**
- `Items` **node, 333**
- `items` **property, 531, 538**
- `itemType` **attribute, 196**
- IzPack, 382**

## J

- Java Development Kit (JDK), 483**
- Java Integrated Development Environment (Java IDE), 490**
- Java Remote Method Invocation (RMI), 575, 609**
- Java Runtime Environment (JRE), 290**
- Java Software Development Kit (SDK), 290**
- Java toolkit, for SVG, 770**
- Java Virtual Machine (JVM), 290**
- JavaScript, 14, 756–757, 805, 838**
- JavaScript Object Notation (JSON), 652–653**
- JavaServer Page (JSP), 295, 382**
- JDK (Java Development Kit), 483**
- `jeff-description` **entity value, 138**

- jetty, 383**
- JRE (Java Runtime Environment), 290**
- JScript, 450**
- JSON (JavaScript Object Notation), 652–653**
- JSP (JavaServer Page), 295, 382**
- JVM (Java Virtual Machine), 290**

## K

- Katmai, 346, 395**
- `key( )` **function, 256, 323, 968**
- keyboard selection, handling, 667–670**
- `keyCode` **property, 662**
- `keydown` **event, 662, 669**
- `keypress` **event, 662**
- keys, 377**
- `keyup` **event, 662**
- `<keyword>` **element, 704, 707, 708**
- keywords**
  - `#FIXED` **keyword, 126**
  - `#IMPLIED` **keyword, 120**
  - `#PCDATA` **keyword, 108, 138**
  - `#REQUIRED` **keyword, 122, 127, 884**
  - `ANY` **keyword, 110**
  - `CDATA` **keyword, 120**
  - `delete` **keyword, 416, 417**
  - `ELEMENT` **keyword, 105**
  - `ELEMENTS` **keyword, 400**
  - `EMPTY` **keyword, 110**
  - `ENTITY` **keyword, 134, 139**
  - `if` **keyword, 367**
  - `insert` **keyword, 416, 418**
  - `in` **keyword, 363**
  - `<keyword>` **element, 704, 707, 708**
  - `list` **keyword, 244**
  - `mixed` **keyword, 222**
  - `OPENXML` **keyword, 410**
  - `PUBLIC` **keyword, 101**
  - `RAW` **keyword, 398**
  - `RNC` **keywords, 227**
  - `ROOT` **keyword, 398**
  - `SYSTEM` **keyword, 100**
  - `text` **keyword, 221**
  - `xquery` **keyword, 355**
  - `XSINIL` **keyword, 400**

kind **attribute**, **122, 124, 125, 126, 128, 130, 178, 180, 186, 192, 193, 195, 196**  
 KindType **global type**, **178**  
 <knows> **element**, **123, 128, 131, 176, 223**  
 KnowsType <complexType> **declaration**, **192**

## L

**L command**, **778**  
 <label> **element**, **844, 847**  
 lang( ) **function**, **274**  
 lang **attribute**, **739, 745, 747, 953, 957**  
 language **type**, **189**  
**language-name type**, **966**  
 last( ) **function**, **252, 254, 275, 318, 932**  
 <last> **element**, **17, 115, 162, 170, 232, 234, 875**  
 lastChild **property**, **458, 459**  
 LastName **attribute**, **300**  
 LastName **element**, **301**  
 latitude **declaration**, **193**  
 latitude **pattern**, **226**  
 <latitude> **element**, **108, 116, 189, 197, 199, 226**  
 Left **value**, **clear property**, **713**  
 Legacy **module**, **XHTML, 761**  
 length **facet**, **194**  
 length **property**, **475**  
 let **clause**, **342, 362, 892**  
 letter-spacing **property**, **CSS, 695**  
 letter-value **attribute**, **xsl:number**, **953**  
 level **attribute**, **xsl:number**, **952**  
 LexicalHandler **extension interface**, **514**  
 LexicalHandler **property**, **513**  
 <li> **element**, **751, 753, 913, 915**  
 library **element**, **359, 370**  
 library **module**, **355, 356**  
 Library.xquery **query**, **358**  
 line **command**, **870**  
 <line> **element**, **773, 776, 916**  
**linefeed characters**, **38**  
**Link module**  
   XHTML, **761**  
   XHTML Basic, **762**  
 <link> **element**, **535, 751, 848**  
**links, in XML documents**  
   forcing using XHTML namespace, **725**  
   XLink support in Firefox, **721–724**

**Linux Documentation Project**, **697**  
 list **keyword**, **244**  
**List module**  
   XHTML, **760**  
   XHTML Basic, **762**  
**list patterns, RELAX NG**, **243–244**  
 List **property**, **CSS, 695**  
 list **types**, **193**  
 list\_reader **available**, **542**  
 list\_reader.py **file**, **540, 541, 543**  
 <list> **element**, **196–197, 728**  
 <listitem> **element**, **699**  
 ListReader **class**, **541–542, 545**  
 list-style **property**, **CSS, 695**  
 list-style-image **property**, **CSS, 695**  
 list-style-position **property**, **CSS, 695**  
 list-style-type **property**, **CSS, 695**  
**literal name node**, **924**  
**literal result elements**, **296, 351**  
**LiveCycle, Adobe**, **837**  
 load( ) **method**, **462, 471**  
 Load( ) **method**, **612**  
**loading, XML documents**, **462–468**  
 loadXML( ) **method**, **453, 471, 588**  
**Loan Origination System Identifier**, **856**  
 loancalculator2. **html**, **859**  
 loancalculator2.js **file**, **869**  
 loancalculator3.html, **868**  
 loancalculator2.html, **864**  
**local attributes**, **177**  
**local declarations**, **156**  
**local elements**, **156–157**  
**local intranet zone**, **647**  
 local-name( ) **function**, **275**  
 localName **parameter**, **492, 494**  
 localName **property**, **458, 459**  
 localname **string**, **900**  
 <location> **element**, **107, 116**  
 LocationType <complexType> **definition**, **176**  
 Locator **interface**, **503**  
 Locator **object**, **502, 504**  
 Locator2 **interface**, **503**  
**long type**, **187**  
 <longitude> **element**, **108, 116, 189, 197, 199, 226**  
**loosely structured data**, **376**  
 lower-case( ) **function**, **278**

lowercaseSearch **variable**, 677

Lucene, 379

Luhn's algorithm, 653

## M

M **command**, 778

machine-readability, 522

Macromedia Dreamweaver, 753

main( ) **function**, 489, 491, 544, 546

margin **attribute**, 910

margins, 702

Marker **property**, CSS, 695

markup languages, history of, 6–7

match **attribute**, 293, 296, 315, 319, 950, 959

matches( ) **function**, 282

MathML, 13

matrix mathematics, 768

maxExclusive **facet**, 194

maxInclusive **facet**, 194

maxLength **facet**, 194

maxOccurs **attribute**, 161, 163, 168, 170

MCF (Meta Content Framework), 525

Media **attribute**, 700

media-type **attribute**

    xforms:submission, 815

    xsl:output, 955

<member> **element**, 589

memberTypes **attribute**, 197

message **property**, 478

<messages> **element**, 634

meta character \*: , node tests, 928

Meta Content Framework (MCF), 525

meta data

    evolution of systems

        Atom, 533–535

        Microsoft's Content Definition Format (CDF), 525–526

        Netscape's RDF Site Summary (RSS) 0.9, 526

        RSS 0.9x, 528–529

        RSS-DEV and RSS 1.0, 529–531

        simplicity, 526–528

        UserLand's RSS 2.0, 531–533

        UserLand's Scripting News format, 526

    Scalable Vector Graphics (SVG), 782–784

    syndication systems

        client-consumer, 523

        client-producer, 523–524

        server-consumer, 524–525

        server-producer, 523

meta information, 621

<meta> **element**, 746, 750, 843, 912, 914

<metadata> **element**, 783, 784

Meta-information module

    XHTML, 760

    XHTML Basic, 762

MetaWeblog API, 524

method **attribute**

    xforms:submission element, 814

    xsl:output element, 306, 954

methodName **element**, 577

methods

    /\*text( ) method, 658

    \_\_str\_\_ method, Python, 547

    = assign method, 230

    abort( ) method, 672

    add( ) method, 474

    addNode( ) method, 465

    appendChild( ) method, 444, 461, 462, 478

    attributes' setNamedItem( ) method, 475

    attributes.item(j) method, 920

    AutoSuggestControl.suggest( ) method, 673

    clearTimeout( ) method, 670

    cloneNode( ) method, Node object, 461, 462

    ConversionRate( ) method, 684

    create\_item method, 547

    create\_parser method, 544, 545

    createAttribute( ) method, Document interface, 475, 479, 480

    createAttributeNS( ) method, Document interface, 479, 480

    createCDATASection( ) method, Document interface, 479, 480

    createComment( ) method, Document interface, 479, 480

    createDocumentFragment( ) method, Document interface, 479

    createElement( ) method, Document interface, 444, 465, 478, 479, 480, 894

    createElementNS( ) method, Document interface, 480, 894

    createElementNS method, Document interface, 479

    createEntityReference method, Document interface, 479

    createProcessing Instruction( ) method, Document interface, 479, 480

createSuggestionsBox( ) method, 664, 670  
 createTextNode method, Document interface, 479  
 createTextRange method, 662  
 custom \_\_str\_\_ method, 548  
 deleteNode( ) method, 464, 465  
 document.write( ) method, 757  
 DOMParser.parseFromStream( ) method, 462  
 endElementNS method, 551, 899, 900  
 Enum.Parse( ) method, 684  
 exist( ) method, 893  
 fetchSuggestions( ) method, 659, 670, 671, 672, 681  
 firstChild method, 867  
 focus( ) method, 661  
 GET method, 464, 524, 581  
 get\_feed\_uris method, 545  
 get\_formatted\_date method, 548  
 get\_newest\_items method, 546  
 get\_uris method, 541, 543, 545  
 getChannels( ) method, 578, 582, 596  
 getColumnNumber method, 502  
 getDateValue( ) method, 861  
 getElementsByTagName( ) method, 475, 479, 480, 757  
 getElementsByTagNameNS( ) method, 479, 480  
 GetFailureXML( ) method, 627  
 getIndex method, 492  
 getLength method, 492  
 getLineNumber method, 502  
 getLocalName method, 493  
 getNamedItem( ) method, 471, 472  
 getNamedItemNS method, NamedNodeMap object, 471  
 getOwnerDocument( ) method, 447  
 getPublicId method, 503  
 getQName method, 493  
 GetQuantityDiscount( ) method, 612  
 getRadioOperators( ) method, 906  
 getRate( ) method, 682, 683–684, 685  
 getSystemId method, 502  
 getTemperature method, 636  
 getTop( ) method, 665  
 getType method, 493  
 getURI method, 493  
 getValue method, 493, 494  
 goToSelection( ) method, 669  
 goToSuggestion( ) method, 668  
 handleAddToCartResponse( ) method, 627  
 handleKeyDown( ) method, 669  
 hasAttributes( ) method, Node object, 461  
 hasChildNodes( ) method, Node object, 461  
 HelloWorld method, 851  
 hideSuggestionsBox( ) method, 670  
 highlightSuggestion( ) method, 664, 665, 668  
 html output method, 563  
 importNode method, Document interface, 480  
 init( ) method, 474, 544, 618, 663, 670, 797  
 insertBefore( ) method, Node object, 461, 462  
 is\_content\_element method, 552  
 isSupported method, Node object, 462  
 item( ) method, 475  
 item method, NamedNodeMap object, 471  
 Java Remote Method Invocation (RMI), 575, 609  
 load( ) method, 462, 471  
 Load( ) method, 612  
 loadXML( ) method, 453, 471, 588  
 modify( ) method, 417, 419, 422, 474, 475, 893  
 mousedown method, 794  
 mouseover(evt) method, 793, 794  
 moveEnd( ) method, 661  
 moveStart( ) method, 661  
 newer\_than method, 546  
 nextSibling method, 867  
 nodes( ) method, 893  
 normalize( ) method, Node object, 462  
 onkeyup( ) method, 663  
 open( ) method, 588, 591  
 output method, 306  
 Page\_Load method, 674, 675  
 PerformLuhnCheck( ) method, 656  
 ping( ) method, 596  
 POST method, 581, 624  
 preventDefault( ) method, 669  
 print method, 543  
 print\_items method, 546  
 process\_tags method, 552, 553  
 put method, xforms:submission element, 824  
 query( ) method, 423–424, 893  
 read( ) method, 489, 544  
 remove( ) method, 474, 475  
 removeChild( ) method, 462, 465  
 removeEntities( ) method, 457  
 removeNamedItem( ) method, 471, 475  
 removeNamedItemNS method, 471  
 replace( ) method, 467

### methods (continued)

replaceChild( ) method, 462, 467  
Response.Write( ) method, 675  
rotatePolygon(evt) method, 794  
select( ) method, 660  
selectNodes( ) method, 471  
selectSingleNode( ) method, 589  
selectTextRange( ) method, 661  
send( ) method, 464, 588, 591  
SendError( ) method, 680  
SendErrorXml( ) method, 676  
SendResponse( ) method, 657  
SendXmlError( ) method, 680  
serializeToString( ) method, 457  
set\_w3cdf\_time method, 547  
setContentHandler method, 505  
setErrorHandler method, 505  
setNamedItem method, NamedNodeMap object, 472  
setNamedItemNS method, NamedNodeMap object, 472  
setPriceAndQuantity( ) method, 618  
setProperty( ) method, 513, 514, 589  
setSelectionRange( ) method, 661, 662  
setTimeout( ) method, 670, 671  
showFullSuggestion( ) method, 661, 667  
showOutput( ) method, 658  
showParseError( ) method, 453  
split method, 543, 795, 796  
startDocument method, 487  
startElementNS method, 550  
struct topicExchange.getChannelInfo  
(string topicName) method, 576  
struct topicExchange.getChannels( ) method,  
576  
struct topicExchange.ping(string  
topicName, struct details) method, 576  
suggest( ) method, 662, 666, 668, 671  
topicExchange.getChannels( ) method, 577  
topicExchange.ping( ) method, 577  
trim method, 553  
tryAddElement( ) method, 477  
unescape method, 552  
using GetBasicTotal( ) method, 612  
validateCardNumber( ) method, 657  
value( ) method, 893  
var xhHTTP = new XmlHttpRequest( ) method,  
591  
xml output method, 563

**Microsoft Core XML Services (MSXML), 923**  
**Microsoft FrontPage, 753**  
**Microsoft Internet Explorer Parser, 24**  
**Microsoft Message Queue (MSMQ), 579**  
<middle> element, **12, 17, 49, 110, 114, 115, 170,**  
**172, 216, 232**  
**MIME type, 757–759, 781**  
**minExclusive facet, 194**  
**minInclusive facet, 194**  
minLength **facet, 194**  
minOccurs **attribute, 161, 163, 164, 168, 170**  
**minus sign (-), 10**  
minus-sign **attribute, xsl:decimal-format element,**  
**946**  
mixed **attribute, 176**  
**mixed content, 108–109, 222**  
mixed **keyword, 222**  
mixed **pattern, 222, 227**  
mm **absolute unit, SVG, 774**  
mode **attribute**  
    xsl:apply-templates element, 315, 941  
    xsl:template element, 959  
model **attribute**  
    xforms:itemset element, 827  
    xforms:select1 element, 826  
model item **properties, 815**  
**modeling feeds**  
    different syntaxes, 538–539  
    Simple API for XML (SAX), 539  
modify( ) **method, 417, 419, 422, 474, 475, 893**  
**modifying**  
    requests, 853–857  
    responses, 853–857  
**modularization, of schemas using include directive,**  
**231–232**  
module **declaration, 355, 356**  
**mouse actions, handling, 665–666**  
mousedown **method, 794**  
mouseout **function, 795**  
mouseout(**evt**) **functions, 793**  
mouseover(**evt**) **method, 793, 794**  
mouseup **function, 795**  
mouseup(**evt**) **functions, 793**  
moveEnd( ) **method, 661**  
move(**evt**) **function, 795**  
move(**evt**) **function, 799**  
moveStart( ) **method, 661**

`moveToFront(polygon)` function, 793, 797

### Mozilla Firefox

network transports, 591–594

XLink support in, 721–724

**MSMQ (Microsoft Message Queue), 579**

**MSXML (Microsoft Core XML Services), 923**

**MSXML component, 445**

**MSXML2 library, 647**

**MSXML2.XMLHTTP class, 588**

Muenchian **Method, 324, 968–969**

MultiAuthor.xquery **query, 367**

**multipane styles, of feed-reader user interface, 523**

**multiple outputs, for XSLT 2.0, 323**

`mustUnderstand` **attribute, 621–622, 623, 906**

`myAttr` **attribute, 40**

`MyClass` **class, 486**

`myDoc` **variable, 417, 419**

`myElement[myAttribute]` **selector, 729**

`myElement[myAttribute~="myValue"]` **selector, 729**

`myElement[myAttribute="myValue"]` **selector, 729**

`myLoggingPage.aspx` **page, 646**

`myNewBib` **element, 352**

### MySQL

adding information in, 427–430

installing, 426–427

querying, 430–435

updating XML in, 435–436

usability of XML in, 436

`mysql` **commandline utility, 437**

`mysql` **prompt, 427**

`mysql.sql` **file, 429**

`mySubmit xforms:submission` **element, 827**

## N

`name( )` **function, 257, 275**

`name` **attribute**

`xsl:attribute` element, 941

`xsl:attribute-set` element, 942

`xsl:call-template` element, 943

`xsl:decimal-format` element, 945

`xsl:element` element, 303, 305, 947

`xsl:key` element, 950

`xsl:param` element, 956

`xsl:processing-instruction` element, 957

`xsl:template` element, 322, 959

`xsl:variable` element, 961

`name` **element, 203, 216**

`Name` **element node, 264, 298, 299**

`Name` **identification module, XHTML, 761**

`name` **pattern, 230, 231, 234**

`name` **type, 188, 967**

`name` **vocabulary, 233**

`<name>` **element, 12, 16, 17, 78, 80, 83, 96, 106, 107, 151, 155, 160, 162, 164, 165, 168, 172, 177, 200, 231, 882, 883**

**name-classes, RELAX NG**

`AnyName` **name-class feature, 240–241**

**basic names, 238**

**choices, 238–239**

**groups, 238–239**

**namespaces with wildcards, 239**

`nameContents` **pattern, 234**

**named node map, 471**

**named pattern choice, 230**

**named pattern interleave, 230**

**named templates, 322**

`NamedNodeMap` **object, 471–475, 893, 894**

`names` **type, 967**

`namespace` **attribute, 164, 180, 181, 204, 914, 941, 947**

`namespace` **axis, 260, 268–270, 348, 361, 926**

`namespace` **declaration, 237, 807**

**Namespace Identifier (NID), 87**

`namespace` **node, 255, 257, 266**

`namespace` **prefix, 72, 154, 170, 204, 238**

`namespace` **pseudo-attribute, 807**

**Namespace Recommendation, 121, 146**

`namespace` **string, 900**

**namespace-prefix type, 967**

**namespaces**

**attributes from other, 207**

**declaration, 356**

**document type definitions (DTDs), 143**

**need for**

**prefixes, 69–70**

**Uniform Resource Identifiers (URIs), 70–72**

**prefixes, 499**

**RELAX NG, 236–237**

**syntax**

**attributes, 83–86**

**default namespaces, 75–81**

**different notations, 81–83**

**target, 153–154**

## namespaces (continued)

---

### namespaces (continued)

- Uniform Resource Identifiers (URIs)
  - meaning, 88–89
  - Resource Directory Description Language (RDDL), 89
  - Uniform Resource Locators (URLs), 86–88
  - Universal Resource Names (URNs), 87–88
- when to use, 89–90
- with wildcards, 239
- in XForms documents, 816–817
- XHTML, 725
- XML Schema, 153
- XML Schemas, 146
- `namespace-uri ( )` **node-set function, 275**
- `namespaceURI` **property, 458, 459**
- `name.xml` **document, 24**
- naming**
  - attributes, 179
  - conventions, 34–35
  - elements, 160–161
- NaN **attribute, `xmlns:decimal-format` element, 946**
- native browser implementations, 804**
- native databases, eXist**
  - installing, 382–383
  - interacting with, 384–394
  - obtaining, 382–383
- native functionality, 426**
- native XML database, 376, 381**
- Nautilus, 388**
- `navigationindex` **attribute, `xforms:select1` element, 822**
- `navindex` **attribute, 818, 819, 820**
- `ncname` **attribute, `xmlns:sort`, 957**
- `NCName` **type, 188**
- `ncname` **type, 967**
- `negativeInteger` **type, 187**
- nested elements, 406**
- nested grammars, 235**
- nested iteration, 342**
- nested patterns, 213, 227**
- nesting, 32–33**
- .NET classes, 853**
- .NET code, 396**
- .NET framework redistributable, 291**
- .NET language, 517**
- Netscape**
  - network transports, 591–594
  - RSS 0.9, 526

### network transports

- firewall-ready, 583–584
- Hypertext Transfer Protocol (HTTP), 579–582
  - posting with Firefox, 591–594
  - posting with Netscape, 591–594
  - request/response paradigm, 582–583
  - security, 584
  - using Hypertext Transfer Protocol (HTTP) for XML-RPC, 584–587
  - wide implementation, 582
- New System **Variable, 291**
- `newer_than` **method, 546**
- newline character, 38, 549, 553, 902, 920**
- news feeds, 536–537**
- newsreaders, 536**
- `nextSibling` **method, 867**
- `nextSibling` **property, 458, 459**
- NID (Namespace Identifier), 87**
- `NMTOKEN` **attribute, 121, 124**
- `NMTOKENS` **attribute, 121, 124**
- No cardinality indicator, 217**
- `node ( )` **function, node tests, 928**
- `Node` **interface, 449, 458**
- `Node` **object**
  - `DOMException` object, 476–478
  - effect of Text nodes, 468–471
  - loading XML documents, 462–468
  - methods, 461–462
  - `NamedNodeMap` object, 471–475
  - `NodeList` object, 475
  - properties, 458–461
- node tests, 927, 928**
- `node` **type, 967**
- `NodeList` **object, 475, 893, 894**
- `nodeName` **property, 454, 458, 459**
- nodes**
  - attribute, 256
  - comment, 257
  - comparing with items, 361
  - element, 255–256
  - kinds, 361
  - namespace, 257
  - processing instruction, 257
  - retrieving, 348–351
  - root, 255
  - Text, 256, 468–471
  - XML Document Object Model (DOM), 448–449

**nodes( ) method, 893**  
**node-set, 252, 343**  
**nodeset attribute**  
    xforms:bind element, 828  
    xforms:itemset element, 827  
**node-set current( ) function, 963**  
**node-set document function, 963**  
**node-set key function, 965**  
**node-set type, 967**  
**node-set-expression type, 967**  
**node-sets**  
    functions, 275  
    sequences of, 361  
    XPath 1.0, 258  
**nodeType property, 458**  
**nodeValue property, 458, 459, 869**  
**nom element, 875**  
**noNamespaceSchemaLocation attribute, 152**  
**nonbreaking space, 36**  
**None value, clear property, 713**  
**nonNegativeInteger type, 187**  
**nonPositiveInteger type, 187**  
**normalize( ) method, Node object, 462**  
**normalizedString type, 187**  
**normalize-space( ) function, 276**  
**normalize-unicode( ) function, 282**  
**<noscript> element, 763**  
**not( ) function, 275**  
**not boolean, 933**  
**notAllowed pattern, 233–234**  
**Notation node, 449**  
**notationDecl event, 509**  
**<note> element, 45**  
**Notification events, 835**  
**NSAttributes object, 550**  
**null locator object, 503**  
**number( ) function, 275**  
**number attribute, 252, 265, 890**  
**number object, 933**  
**number type, 967**  
**numbers, XPath 1.0, 258**  
**numeric expression, 280**  
**numeric functions, 275–276**  
**\_ (copyright symbol), 61**

## O

**object element, 806**  
**Object element nodes, 312, 314**  
**object inheritance, 147**  
**Object Management Group (OMG), 574**  
**object model, 14**  
**Object module**  
    XHTML, 760  
    XHTML Basic, 762  
**Object Request Broker (ORB), 574**  
**object system-property function, 965**  
**object type, 967**  
**<object> tag, 866**  
**objects**  
    DOMException, 476–478  
    NamedNodeMap, 471–475  
    Node  
        DOMException object, 476–478  
        effect of Text nodes, 468–471  
        loading XML documents, 462–468  
        methods, 461–462  
        NamedNodeMap object, 471–475  
        NodeList object, 475  
        properties, 458–461  
        NodeList, 475  
        XML Document Object Model (DOM), 445–446  
    <occupants> element, 899  
**ODF (Open Document Format), 809**  
**oDocumentElement.xml file, 461**  
**oDom variable, 474**  
**offsetLeft property, 664**  
**offsetParent property, 664**  
**offsetTop property, 664**  
**offsetWidth property, 664**  
**<ol> element, 728**  
**OMG (Object Management Group), 574**  
**omit-xml-declaration attribute**  
    xforms:submission element, 815  
    xsl:output element, 955  
**onclick attribute, 452, 785, 845, 864**  
**onkeyup( ) method, 663**  
**online XML IDEs, 754**  
**onload attribute, 789, 793**  
**onload event, 474**

onLoad **value, actuate attribute, 725**  
onmousedown **mouse action, 665**  
onmouseover **mouse action, 665**  
onmouseup **mouse action, 665**  
onreadystatechange **function, 862**  
onreadystatechange **property, 588**  
open( ) **method, 588, 591**  
**Open Document Format (ODF), 809**  
**opening (<) character, 34**  
**open-source graphical editor, 770**  
OPENXML **clause, 409–413**  
OPENXML **keyword, 410**  
OPENXML **rowset provider, 395**  
OPENXML **statement, 412**  
**Opera browser, 783**  
**operators, 923**  
<option> **element, 742**  
optional **attribute, 180**  
**optional elements, 105**  
**Oracle parser, 485, 502**  
**ORB (Object Request Broker), 574**  
**Orbeon Forms, 808, 838**  
order **attribute, xsl:sort, 957**  
order by **clause, 366–367, 892**  
order\_id **parameter, 905**  
<Order> **element, 30, 43**  
order2.xml **file, 46**  
OrderByTitleOut.xml **query, 367**  
OrderByTitle.xquery **query, 366**  
orderID **attribute, 910**  
<Orders> **element, 41**  
orderSummary.xslt **document, 333**  
orders.xml **document, 333**  
order.xml **file, 29, 41**  
**outer query, 407**  
output **element, 560**  
output **method, 306**  
**overlapping**  
    absolutely positioned elements, 718  
    floated boxes, 712  
    relative positioning, 708–709  
    using clear property to prevent, 713  
ownerDocument **property, 447, 458, 459**  
oXmlHttp **variable, 651**  
**oXygen 8.0, 389**

## P

<p> **element, 31, 32, 35, 69, 76, 78, 85, 549, 553, 564, 699, 701, 751, 758, 844, 882, 913**  
p3ptype **property, XForms model items, 828**  
**padding, 702**  
padding **attribute, 910**  
Page\_Load **method, 674, 675**  
<page> **element, 705, 707, 714**  
<pageNumber> **element, 705**  
**painter's model, Scalable Vector Graphics (SVG), 774–776**  
<para> **element, 699**  
**paragraph bar object, 758**  
Paragraph **element, 250, 280**  
<paragraph> **element, 704, 705, 707, 712**  
**parameter entities, 139–141**  
ParameterExample.xquery **query, 370**  
**parameters**  
    \$ParameterName syntax, 320  
    \_query, 390  
    \_xsl, 393  
    cust\_id, 905  
    DefaultPhoneKind, 140  
    details, 578  
    Extensible Stylesheet Language Transformations (XSLT), 320–322  
    external-parameter-entities feature, 498  
    http://xml.org/sax/features/  
        externalparameter-entities feature, 512  
    internal entity, 139  
    localName, 492, 494  
    order\_id, 905  
    parameter entities, 139–141  
    ParameterExample.xquery query, 370  
    person, 321  
    prefix, 499, 873  
    qName, 492, 494  
    uri, 492, 499  
    using with XQuery, 370  
    XML-remote procedure call (RPC), 577  
    xsl:parameter element, 320  
parent **axis, 260, 271, 927**  
parent **grammar, 236**  
parent node, **264, 271**

- parent **pattern, 236**
- <parent> **element, 16**
- parentNode **property, 458, 459, 465**
- parse **function, 508, 511**
- parsed character data (PCDATA)**
  - illegal characters
    - character data (CDATA) sections, 61–62
    - escaping, 60–61
    - whitespace, 38–39
- parseError **object, 453**
- parseFloat(**string**) **function, 796**
- parseFromString( ) **function, 456**
- parsererror **document element, 457**
- parser.parse **instance, 544**
- parsers, 11–12, 24, 40, 74, 135, 878**
- parsing, 444, 484**
- Part **element node, 271**
- Pascal language, 517**
- path commands, 779**
- path **data, d attribute, 777**
- path **element, 917**
- PATH environment **variable, 291, 490**
- PATH **option, 405**
- PATH **query, 405**
- <path> **element, 777, 780**
- paths, Scalable Vector Graphics (SVG), 777–780**
- pattern definitions, 244**
- pattern facet, 194**
- patterns**
  - attribute
    - cardinality, 216–217
    - compared to elements, 216
    - connector patterns and grouping, 217–220
    - co-occurrence constraints, 221
    - empty pattern, 222–223
    - enumerated values, 220–221
    - mixed content pattern, 222
  - combining, 227–236
  - connector, and grouping, 217–220
  - element
    - cardinality, 216–217
    - compared to attributes, 216
    - connector patterns and grouping, 217–220
    - co-occurrence constraints, 221
    - empty pattern, 222–223
    - enumerated values, 220–221
    - mixed content pattern, 222
  - empty, 222–223
  - extensions, 234
  - list, 243–244
  - mixed content, 222
  - named
    - combining definitions, 230–231
    - redefining included, 232–233
  - notAllowed, 233–234
  - restrictions, 234
  - reusing, 227–236
  - text
    - cardinality, 216–217
    - connector patterns and grouping, 217–220
    - co-occurrence constraints, 221
    - elements and attributes, 216
    - empty pattern, 222–223
    - enumerated values, 220–221
    - mixed content pattern, 222
- pattern-separator **attribute, xsl:decimal-format element, 947**
- payloads, 613**
- payment calculator (case study)**
  - list of needs, 842
  - mortgage calculations, 841–842
  - online loan calculator
    - adding frame to main page, 868–872
    - enhancing display with SVG, 865–868
    - integrating calculation web service, 849–865
- payment card validator, 653–658**
- payment schedules, 842**
- <payment> **element, 218**
- payments\_ table **container element, 848**
- payments\_table <div> **element, 848**
- payments\_table **container, 863**
- payments\_table **template, 848**
- <PAYMENTSTREAM> **elements, 862**
- pc **absolute unit, SVG, 774**
- PCDATA (parsed character data)**
  - illegal characters
    - character data (CDATA) sections, 61–62
    - escaping, 60–61
    - whitespace, 38–39
- People.xml **file, 293**
- percent **attribute, xsl:decimal-format element, 946**
- percent sign (%), 139, 140**
- percentages, 774**
- PerformLuhnCheck( ) **method, 656**

## Perl language, 517

per-mille **attribute**, `xsl:decimal-format` **element**, 946

pers **namespace prefix**, 72

pers **prefix**, 69, 70, 74

pers: **prefix**, 80

person **attribute**, 128

Person **element node**, 296, 299, 303

person **parameter**, 321

`<person>` **element**, 68, 72, 73, 76, 78, 83, 267, 300, 814

PersonData **element node**, 264

**Perspective menu**, 389

`<pers:person>` **element**, 72

`<pers:title>` **element**, 69

phone **attribute declaration**, 140

`<phone>` **element**, 126, 186, 192, 193, 195, 241, 885, 886

phoneContents **declaration**, 242

PhoneType `<complexType>` **declaration**, 192

ping( ) **method**, 596

**PIs (processing instructions)**, 56–59

**PITarget**, 56

**Planet Venus aggregator**, 525

**Planète Web Sémantique**, 525

**pluggable datatypes**, 212

points **attribute**, 918

polygon **variable**, 785

`<polygon>` **element**, 776, 785, 789, 790, 791, 793, 794, 797, 798, 799, 916, 919

polyGroup **element**, 920

polyGroup **object**, 919

portrait **attribute**, 123

portType **interface**, 907

`<portTypes>` **element**, 635

position( ) **function**, 252, 254, 275, 318, 933

position **property**, 707, 717, 718

**positioning, in Cascading Style Sheets (CSS)**

absolute, 714–716

fixed, 717–718

float, 709–711

normal flow, 706–707

overlapping absolutely positioned elements, 718

overlapping floated boxes, 712

overlapping relative, 708–709

relative, 707–708

using `clear` property to prevent overlap, 713

vertical margins collapse in normal flow, 707

z-index property, 718

positiveInteger **type**, 187

POST **method**, 581, 624

POST **operation**, 620

**Post Schema Validation Infoset (PSVI)**, 148

**posting techniques**, 649–652

**PostTester-IE.html**, **PostTester-CrossBrowser.html**, 593

**post-to-blog facilities**, 523

`<pre>` **tag**, 37

preceding **axis**, 260, 271–272, 348, 927

preceding-sibling **axis**, 260, 272–273, 348, 927

**predicates**, XPath, 278

prefix **contacts**, 175

prefix **parameter**, 499, 873

prefix **property**, 458, 459

prefix **target**, 175

**prefixes**, 69–70, 74, 499

**Presentation module**, XHTML, 760

preserveWhitespace **property**, 471

preventDefault( ) **method**, 669

previousSibling **property**, Node object, 458

print **method**, 543

print\_items **method**, 546

priority **attribute**, `xsl:template`, 959

**procedural programming**, XSLT, 292

process\_tags **method**, 552, 553

processContents **attribute**, 164, 181, 887

**processing**

conditional

`<xsl:choose>` **element**, 308–311

`<xsl:if>` **element**, 306–308

feed data for display, 561–563

pre-processing feed data, 565

processing instruction **node**, 255, 257

**processing instructions (PIs)**, 56–59

processing-instruction( ) **function**, 928

processingInstruction **event**, 487, 499

ProcessingInstruction **node**, 448, 449

**ProcessingInstruction objects**, 448

processing-instruction(Literal) **function**, node tests, 928

**processors**

Saxon XSLT

installing Java version, 290–291

installing .NET version, 291–292

XSLT, 288–289

**producers, Simple API for XML (SAX), 516****<Product> element, 41****programming**

- declarative, 292–293
- procedural, 292
- tools, 770–771

**ProgramType enumeration, 856****prolog, XQuery**

- base-uri declaration, 356
- boundary-space declaration, 357–358
- default namespace declarations, 356
- modules, 355–356
- namespace declaration, 356
- schema imports, 357
- validation declaration, 357
- variable declarations, 357
- version declaration, 355

**properties**

- attr property, 910
- attributes property, Node object, 458, 459, 471
- Background property, CSS, 695
- background property, CSS, 695
- background-attachment property, CSS, 695
- background-color property, 694, 695
- background-image property, 695, 725
- background-position property, 695
- background-repeat property, 695
- border-collapse property, CSS, 695
- border-spacing caption-side property, CSS, 695
- calculate property, XForms model items, 828
- charCode property, 919
- childNodes property, 458, 459, 475, 798
- clear, 713
- clear property, 713
- color property, 695, 699
- constraint property, XForms model items, 828
- content property, 730, 731, 910
- createTextNode property, 465
- DeclHandler property, 513
- direction property, CSS, 695
- display property, 703, 704, 719, 721, 911
- doctype property, Document interface, 479
- documentElement property, 446, 454, 464, 479, 862
- DOMParser property, 456, 457
- empty-cells property, CSS, 695
- firstChild property, 458, 459, 461
- float property, 709
- foaf:maker property, 901

- foaf:mbox\_sha1 property, 901
- foaf:name property, 901
- foaf:nick property, 901
- foaf:Person property, 901
- foaf:weblog property, 901
- font property, CSS, 694
- font-family property, 694, 696
- font-size property, CSS, 694
- font-style property, CSS, 694
- font-variant property, 694
- font-weight property, 694, 696
- http://xml.org/sax/properties/declaration-handler property, 514
- http://xml.org/sax/properties/documentxml-version property, 514
- http://xml.org/sax/properties/lexicalhandler property, 514
- im:nickname property, 556
- implementation property, Document interface, 479
- Inherit value, clear property, 713
- innerHTML property, 863
- items property, 531, 538
- keyCode property, 662
- lastChild property, 458, 459
- Left value, clear property, 713
- length property, 475
- letter-spacing property, CSS, 695
- LexicalHandler property, 513
- List property, CSS, 695
- list-style property, CSS, 695
- list-style-image property, CSS, 695
- list-style-position property, CSS, 695
- list-style-type property, CSS, 695
- localName property, 458, 459
- Marker property, CSS, 695
- message property, 478
- namespaceURI property, 458, 459
- nextSibling property, 458, 459
- nodeName property, 454, 458, 459
- nodeType property, 458
- nodeValue property, 458, 459, 869
- None value, clear property, 713
- offsetLeft property, 664
- offsetParent property, 664
- offsetTop property, 664
- offsetWidth property, 664
- onreadystatechange property, 588
- ownerDocument property, 447, 458, 459

## properties (continued)

---

### properties (continued)

`p3ptype` property, XForms model items, 828  
`parentNode` property, 458, 459, 465  
`position` property, 707, 717, 718  
`prefix` property, 458, 459  
`preserveWhitespace` property, 471  
`previousSibling` property, Node object, 458  
`readonly` property, XForms model items, 828  
`receiving SAX events`, 513–514  
`referrer` property, 646  
`relevant` property, XForms model items, 829  
`required` property, XForms model items, 829  
`responseXML` property, 464, 652  
`Right value, clear` property, 713  
Simple API for XML (SAX) events, 513–514  
`Table` property, CSS, 695  
`table-layout` property, CSS, 695  
`Text` property, CSS, 694  
`text-align` property, CSS, 695  
`text-decoration` property, CSS, 695  
`text-indent` property, CSS, 695  
`text-transform` property, CSS, 695  
`title` property, 15, 528  
`type` property, XForms model items, 829  
`unicode-bidi` property, CSS, 695  
`white-space` property, CSS, 695  
`width` property, 709, 712, 718  
`word-spacing` property, CSS, 695  
`z-index`, 718  
`z-index` property, 718–719

**Proposal Recommendation status**, 341  
**proxy web service**, 853  
**ProxyCalculationService website**, 850  
**pseudo-elements**, 726  
**PSVI (Post Schema Validation Infoset)**, 148  
`pt` **absolute unit, SVG**, 774  
`pubDate` **element**, 552  
**public identifiers**, 101–102, 140  
**PUBLIC** **keyword**, 101  
`publisher` **element**, 366  
`PublisherOut.xml` **query**, 366  
`Publisher.xquery` **query**, 366  
**publishing Scalable Vector Graphics (SVG) on websites**, 785–786  
**PubSub**, 524  
**pull process**, 484  
`<pullQuote>` **element**, 712

`<purchaseOrder>` **element**, 258, 910  
**push model**, 525  
**push processors**, 484  
`put` **method**, `xforms:submission` **element**, 824  
`px` **absolute unit, SVG**, 774  
**Python**, 517, 541, 771  
**PyXML**, 540, 554

## Q

**Q command**, 778  
`qName` **parameter**, 492, 494  
`qname` **string**, 550  
`QName` **type**, 188  
`qname` **type**, 967  
`qnames` **type**, 967  
**Qualified Names (QNames)**, 72, 255, 813  
`query( )` **method**, 423–424, 893  
**Query dialog**, 387  
**querying MySQL**, 430–435  
**Quirks mode**, 843  
`quot` **entity**, 138  
`quoted string` **value**, 220  
**Qurvi**, 770

## R

`r` **attribute**, 774  
**raw characters**, 60  
**RAW** **keyword**, 398  
**RDBMSs (relational database management systems)**

- commercial
  - Web service support, 426
  - XML functionality in SQL Server 2000, 395
- open source
  - adding information in MySQL, 427–430
  - client-side XML support, 437
  - installing MySQL, 426–427
  - querying MySQL, 430–435
  - updating XML in MySQL, 435–436
  - usability of XML in MySQL, 436

**RDDL (Resource Directory Description Language)**, 89  
**RDF (Resource Description Framework)**, 526, 568, 783  
**RDF Site Summary (RSS)**

- Atom, 533–535
- RSS 0.9, 526

- RSS 0.91, 528–529
- RSS 1.0, 529–531
- RSS-DEV, 529–531
- transforming with XSLT
  - browser processing, 564–565
  - generating feed from existing data, 558–559
  - pre-processing feed data, 565
  - processing feed data for display, 561–563
  - syndication formats, 565–567
- rdf:about **attribute**, **531, 535**
- rdf:RDF **element**, **529, 616**
- <rdf:RDF> **element**, **527, 531**
- read( ) **method**, **489, 544**
- readers, address list**, **541–542**
- readonly **property**, **XForms model items**, **828**
- readystatechange **event**, **651**
- Really Simple Syndication (RSS)**. *See* **RDF Site Summary (RSS)**
- receiveLoanInformation **function**, **862, 868**
- Receiver **identifier**, **623**
- <rect> **element**, **771, 774, 776, 916**
- recursive entity reference**, **134**
- recursive **function**, **371**
- recursive patterns**, **213**
- redefining included named patterns**, **232–233**
- ref **attribute**, **170, 178, 184, 815, 817, 819, 822, 826, 836**
- <reference> **element**, **704, 707**
- references**
  - to built-in entities, 132
  - to character entities, 133
  - to general entities, 135–138
  - <group>, 170–171
  - to parameter entities, 140–141
- referrer **property**, **646**
- regular expressions**, **795**
- Regular Language description for XML (RELAX)**, **211**
- relational data, compared to XML-based data**, **377**
- relational database management systems (RDBMSs)**
  - commercial
    - Web service support, 426
    - XML functionality in SQL Server 2000, 395
  - open source
    - adding information in MySQL, 427–430
    - client-side XML support, 437
    - installing MySQL, 426–427
    - querying MySQL, 430–435
    - updating XML in MySQL, 435–436
    - usability of XML in MySQL, 436
- relational databases, moving XML to/producing XML from**, **380**
- relative location path**, **299**
- relative positioning**, **707–709**
- relative units**, **774**
- RELAX (Regular Language description for XML)**, **211**
- RELAX NG**
  - comments, 244
  - compact (RNC) syntax, 212–213
  - datatypes, 241–243
  - divisions, 244
  - grammars
    - combining, 227–236
    - reusing, 227–236
  - list patterns, 243–244
  - name-classes
    - AnyName name-class feature, 240–241
    - basic names, 238
    - choices, 238–239
    - groups, 238–239
    - namespaces with wildcards, 239
  - namespaces, 236–237
  - patterns
    - attribute patterns, 213–227
    - combining, 227–236
    - element patterns, 213–227
    - reusing, 227–236
    - text patterns, 213–227
  - useful resources, 245
- RELAX NG compact (RNC) syntax**, **212–213**
- relay **attribute**, **622**
- relevant **property**, **XForms model items**, **829**
- remote procedure call (RPC)**
  - Distributed Component Object Model (DCOM), 573–574
  - Internet Inter-ORB Protocol (IIOP), 574
  - Java Remote Method Invocation (RMI), 575
  - network transports
    - firewall-ready, 583–584
    - Hypertext Transfer Protocol (HTTP), 579–582
    - posting with Firefox, 591–594
    - posting with Netscape, 591–594
    - request/response paradigm, 582–583
    - security, 584
    - using Hypertext Transfer Protocol (HTTP) for XML-RPC, 584–587
    - wide implementation, 582

## remote procedure call (RPC) (continued)

---

### remote procedure call (RPC) (continued)

XML-RPC  
  application programming interface (API), 576  
  parameters, 577  
  requests, 577  
  struct value sets, 577–579

`remove()` **method**, 474, 475

`removeChild()` **method**, 462, 465

`removeEntities()` **method**, 457

`removeNamedItem()` **method**, 471, 475

`removeNamedItemNS` **method**, 471

`replace()` **method**, 467

`replace` **attribute**, `xforms:submission` **element**, 815

`replaceChild()` **method**, 462, 467

**replacement text**, 134

**REpresentational State Transfer (REST)**, 390–394, 596–600, 612–613

**requests**  
  modifying, 853–857  
  XML-remote procedure call (RPC), 577

**Requests For Comments (RFCs)**, 387

**required attribute**, 180

**required property**, XForms model items, 829

**required values**, 126–127

`reset()` **function**, 792, 793, 798, 799

`resolveEntity` **event**, `EntityResolver` **interface**, 510

`resolveEntity` **function**, 510

**Resource Description Framework (RDF)**, 526, 568, 783

**Resource Directory Description Language (RDDL)**, 89

**responses, modifying**, 853–857

`Response.Write()` **method**, 675

`responseXML` **property**, 464, 652

**REST (REpresentational State Transfer)**, 390–394, 596–600, 612–613

`rest-query-results.xml` **file**, 392

**restriction types**, 193

`<restriction>` **declarations**, 194–196

**result documents**, 289, 940

**result set**, 924

**result tree**, 289, 342

`result-prefix` **attribute**, `xsl:namespace-alias`, 951

`<r sum ` **element**, 68

**retrieving, nodes**, 348–351

`return` **clause**, 423

`return` **statement**, 363, 368, 796

### returning data, as XML using `FOR XML` clause

extensions to XQuery in SQL Server 2005, 416–423

`OPENXML` **clause**, 409–413

`query()` **method**, 423–424

W3C XML Schema in SQL Server 2005, 424–425

`FOR XML AUTO` **clause**, 401–404

`xml` **datatype**, 413–416

`FOR XML EXPLICIT` **clause**, 405

`FOR XML PATH` **clause**, 405–409

`FOR XML RAW` **clause**, 396–401

XQuery in SQL Server 2005, 416

**reusable references**, 123

**reusing**  
  grammars, 227–236  
  patterns, 227–236

`<review>` **element**, 712

**RFCs (Requests For Comments)**, 387

**Rich Site Summary**. See **RDF Site Summary (RSS)**

**Right value**, `clear` **property**, 713

**RMI (Java Remote Method Invocation)**, 575, 609

**RNC (RELAX NG compact) syntax**, 212–213

**RNC keywords**, 227

`role` **attribute**, 622, 764

`root` **element**, 25, 33–34, 100, 134, 151, 348, 454, 459, 560, 563, 743, 902

`root <html>` **element**, 912

**ROOT keyword**, 398

**root node**, 251, 252, 254, 255, 264, 271, 361

**root start-tag**, 454

`rootElement` **object**, 794

`rotate` **attribute**, 791

`rotate` **command**, 793

`rotate` **expression**, `transform` **attribute**, 776

`rotate` **transform**, 916, 917

`rotate(degrees)` **command**, 871

`rotatePolygon` **event listener**, 919

`rotatePolygon` **function**, 797

`rotatePolygon(evt)` **method**, 794

`rotateString` **function**, 796

`round()` **number function**, 275

**round number**, 934

`<row>` **element**, 398

**RPC (remote procedure call)**  
  Distributed Component Object Model (DCOM), 573–574  
  Internet Inter-ORB Protocol (IIOP), 574  
  Java Remote Method Invocation (RMI), 575

network transports  
 firewall-ready, 583–584  
 Hypertext Transfer Protocol (HTTP), 579–582  
 posting with Firefox, 591–594  
 posting with Netscape, 591–594  
 request/response paradigm, 582–583  
 security, 584  
 using Hypertext Transfer Protocol (HTTP) for XML-RPC, 584–587  
 wide implementation, 582  
 XML-RPC  
 application programming interface (API), 576  
 parameters, 577  
 requests, 577  
 struct value sets, 577–579

`rpc` value, **636**

`.rpm` packages, **427**

### **RSS (RDF Site Summary)**

Atom, 533–535  
 RSS 0.9, 526  
 RSS 0.91, 528–529  
 RSS 1.0, 529–531  
 RSS-DEV, 529–531  
 transforming with XSLT  
 browser processing, 564–565  
 generating feed from existing data, 558–559  
 pre-processing feed data, 565  
 processing feed data for display, 561–563  
 syndication formats, 565–567

### **RSS Bandit, 536**

`rss` prefix, **538**

`rss` start-tag, **560**

`<rss>` element, **538, 563**

`rss:channel` element, **567**

Ruby annotation module, **XHTML, 761**

## **S**

`s` command, **778**

same source origin policy, **645**

### **SAX (Simple API for XML)**

advantages, 515–516  
 consumers, 516  
 description  
 history, 484–485  
 setting up, 486  
 source, 485

drawbacks, 516

filters, 516

modeling feeds, 539

other languages, 516–517

producers, 516

receiving SAX events

ContentHandler interface, 487–504

DTDHandler interface, 509

EntityResolver interface, 510

ErrorHandler interface, 504–509

extension interfaces, 514–515

features, 510–513

properties, 513–514

### **SAX pipeline, 516**

`sax2.jar` file, **486**

`sax2r2.jar` file, **480**

SAXException object, **500, 504**

SAXNotRecognizedException function, **511**

SAXNotSupportedException function, **511**

### **Saxon, 343–345**

Saxon error message, **291**

Saxon XQuery engine, **389**

### **Saxon XSLT processor**

installing Java version, 290–291

installing .NET version, 291–292

SAXParseException object, **506, 507**

### **Scalable Vector Graphics (SVG)**

advantages, 768

annotation, 782–784

comments, 782–784

described, 767–771

enhancing display with, 865–868

example, 771–774

grouping, 776

images, 780–781

metadata, 782–784

painter's model, 774–776

paths, 777–780

publishing on website, 785–786

resources, 799–800

sample application

script, 792–800

SVG shapes, 788–792

XHTML wrapper, 787

scripting, 784–785

text, 781–782

tools, 769–771

transformations, 776–777

## Scalable Vector Graphics (SVG) (continued)

- units, 774
- uses, 769
- views, 774
- scale **expression**, transform **attribute**, 777
- schema **attribute**, 834
- schema **elements**, 888
- schema imports**, XQuery, 357
- schema **validators**, 148, 175, 207
- <schema> **element**, 151, 153–155, 156, 158, 160, 161, 175, 179, 209
- schemaLocation **attribute**, 152, 200, 204, 381
- schemas**, 19, 145
- Schematron**, 209
- scramble( ) **function**, 792, 793, 798
- screen scraping techniques**, 561
- script **file**, 789
- <script> **element**, 46, 756, 763, 785, 788, 866
- Scripting module**, XHTML, 760
- Scripting News format**, 526
- scripts**, Scalable Vector Graphics (SVG), 792–800
- SDD (Standalone Document Declaration)**, 54, 56
- <section> **element**, 259, 264, 265, 278, 279, 280, 764
- Secure Sockets Layer (SSL)**, 582, 584
- security**
  - network transports, 584
  - web services, 604
- security **attribute**, 278, 280
- select( ) **method**, 660
- select **attribute**
  - xsl:apply-templates **element**, 313, 318, 321, 941
  - xsl:copy-of **element**, 945
  - xsl:for-each **element**, 948
  - xsl:param **element**, 956
  - xsl:sort **element**, 314, 957
  - xsl:value-of **element**, 299, 961
  - xsl:variable **element**, 321, 962
  - xsl:with-param **element**, 963
- SELECT **query**, 416
- SELECT **statement**, 395, 417, 419, 423
- selected **attribute**, 742
- selectedPoly **variable**, 794, 795
- selection **attribute**, xforms:select **element**, 821
- selection **attribute**, xforms:select1 **element**, 822
- selectNodes( ) **method**, 471
- selectors**, Cascading Style Sheets (CSS), 700–701
- selectSingleNode( ) **method**, 589
- selectTextRange( ) **method**, 661
- self **axis**, 260, 273–274, 927
- self::node( ) **expression**, 257
- self-closing tag**, 31, 49
- self-contained applications**, Scalable Vector Graphics (SVG), 769
- self::node( ) **function**, 314
- semantic markup**, 692
- Semantic Web initiative**, 526
- semicolon (;)**, 132, 133, 135, 140
- semi-structured data**, 376
- send( ) **method**, 464, 588, 591
- Sender **identifier**, 623
- SendError( ) **method**, 680
- SendErrorXml( ) **method**, 676
- SendResponse( ) **method**, 657
- SendXmlError( ) **method**, 680
- separatedContacts.xslt instruction**, 330
- <separator> **element**, 764
- sequence (choice) connector**, 217–219
- sequence pattern**, 217
- <sequence> **element**, 151, 168, 175, 176
- sequences**, 106–108
- sequencing XPath 2.0**, 924
- serialization**, 444
- serialized XML**, 250
- serializeToString( ) **method**, 457
- sernaDirect **subscription service**, 26
- server **script**, eXist, 382
- server-based applications**, Scalable Vector Graphics (SVG), 769
- server-consumer syndication systems**, 524–525
- server-producer syndication systems**, 523
- servers**
  - AutoSuggest box, 658–681
  - JavaScript Object Notation (JSON), 652–653
  - payment card validator, 653–658
  - processing on, 652–681
  - reducing load, 20
  - transport on, 652–681
- server-side content types**, 747
- Server-side image map module**, XHTML, 760
- server-side proxies**, **currency converter proxy**
  - creating clients, 684–686
  - creating web services, 682–683
  - implementing GetRate( ) **method**, 683–684

- server-side validation, 804**
- <service> **element, 638–641**
- Service.asmx **file, 852, 858**
- Service.cs **file, 851, 854**
- serviceUrl **public member, 672**
- SET **statement, 417**
- set\_w3cdtf\_time **method, 547**
- setContentHandler **method, 505**
- setDocumentLocator **callback, 502, 503**
- setDocumentLocator **event, 488, 504**
- setDTDHandler **function, 509**
- setEntityResolver **function, 510**
- setErrorHandler **method, 505**
- setFeature **function, 511, 514**
- setNamedItem **method, NamedNodeMap object, 472**
- setNamedItemNS **method, NamedNodeMap object, 472**
- setPriceAndQuantity( ) **method, 618**
- setProperty( ) **method, 513, 514, 589**
- setSelectionRange( ) **method, 661, 662**
- setTimeout( ) **method, 670, 671**
- SGML (Standard Generalized Markup Language), 6, 7, 11, 99, 148**
- shapes, Scalable Vector Graphics (SVG), 788–792**
- sharing vocabularies, 104**
- short type, 187**
- show **attribute, 724, 725**
- showFullSuggestion( ) **method, 661, 667**
- showOutput( ) **method, 658**
- showParseError( ) **method, 453**
- showProgress **function, 860**
- showProperties( ) **function, 452, 453, 456**
- shredding, 380, 409, 893**
- Simple API for XML (SAX)**
  - advantages, 515–516
  - consumers, 516
  - description
    - history, 484–485
    - setting up, 486
    - source, 485
  - drawbacks, 516
  - filters, 516
  - modeling feeds, 539
  - other languages, 516–517
  - producers, 516
  - receiving SAX events
    - ContentHandler interface, 487–504
    - DTDHandler interface, 509
    - EntityResolver interface, 510
    - ErrorHandler interface, 504–509
    - extension interfaces, 514–515
    - features, 510–513
    - properties, 513–514
- simple content, 16**
- Simple Extension Element, 555**
- Simple Object Access Protocol (SOAP)**
  - groundwork, 608
  - messages
    - <Body> element, 614–615
    - encoding style, 615–620
    - <Envelope> element, 614
    - <Fault> element, 622–632
    - <Header> element, 620–632
  - Representational State Transfers (RESTs), 612–613
  - Windows 2000, 608
  - Windows 2003, 608
  - Windows XP, 608
- SimpleBooks2WRONGOut.xml output document, 352**
- SimpleBooks2.xquery **file, 351**
- SimpleBooks.xquery **query, 347**
- <simpleContent> **element, 185**
- SimpleFunction.xquery **query, 371**
- <simpleType> **element, 157, 178, 193, 194, 195, 197, 199, 216, 886**
- single pane styles, 523**
- single-element pattern, 219**
- skewX **expression, 777**
- skewY **expression, 777**
- skipped entities, 498–499**
- skippedEntity **event, 487, 498**
- SOAP (Simple Object Access Protocol)**
  - groundwork, 608
  - messages
    - <Body> element, 614–615
    - encoding style, 615–620
    - <Envelope> element, 614
    - <Fault> element, 622–632
    - <Header> element, 620–632
  - Representational State Transfers (RESTs), 612–613
  - Windows 2000, 608
  - Windows 2003, 608
  - Windows XP, 608
- SOAPAction **header, 636**
- <soap:binding> **element, 636**
- <soap:Body> **element, 614, 636, 637, 677, 680**

## <soap:Envelope> element

---

- `<soap:Envelope>` **element**, 677, 680
- `<soap:operation>` **element**, 636–637
- sorting, with** `order by` clause, 366–367
- `source` **attribute**, 120, 128, 130, 183, 185, 208, 209
- `source` **pattern**, 236
- source tree, using information from**
  - `<xsl:copy>` element, 299–303
  - `<xsl:copy-of>` element, 303–305
  - `<xsl:value-of>` element, 297–299
- SourceForge**, 485
- source-text**, 134
- `sp_xml_prepareDocument` **stored procedure**, 410
- `sp_xml_removedocument` **stored procedure**, 410
- `<span>` **element**, 843
- speed typists, coping with**, 670–671
- `split` **method**, 543, 795, 796
- SQL (Structured Query Language)**, 340
- SQL Express**, 395
- SQL Server 2000**
  - returning data as XML using `FOR XML`
    - extensions to XQuery in SQL Server 2005, 416–423
    - `OPENXML` clause, 409–413
    - `query( )` method, 423–424
    - W3C XML Schema in SQL Server 2005, 424–425
    - `FOR XML AUTO` clause, 401–404
    - `xml` datatype, 413–416
    - `FOR XML EXPLICIT` clause, 405
    - `FOR XML PATH` clause, 405–409
    - `FOR XML RAW` clause, 396–401
    - XQuery in SQL Server 2005, 416
  - returning data as XML using `FOR XML` clause, 396–425
- SQL Server 2005**
  - extensions to XQuery in, 416–423
  - XML Schema in, 424–425
  - XQuery in, 416
- `sql:column` **function**, 422, 423
- `sql:function` **function**, 422
- `sql:variable` **function**, 423
- SQLXML**, 395
- square brackets**, 929
- Squiggle**, 770
- `src` **attribute**, 764, 813, 825
- SSL (Secure Sockets Layer)**, 582, 584
- stacking context**, 718
- `standalone` **attribute**, 50–54, 815, 955
- Standalone Document Declaration (SDD)**, 54, 56
- Standard Generalized Markup Language (SGML)**, 6, 7, 11, 99, 148
- standards**
  - components, 19–20
    - World Wide Web Consortium (W3C), 18–19
  - `start` **pattern**, 231, 233, 234
  - `startCDATA` **event**, **LexicalHandler interface**, 515
  - `startDocument` **event**, 487
  - `startDocument` **function**, 491, 497, 895
  - `startDocument` **method**, 487
  - `startDTD` **event**, **LexicalHandler interface**, 515
  - `startElement` **event**, 487, 492
  - `startElement` **function**, 491, 497, 539, 895
  - `startElementNS` **method**, 550
  - `startEntity` **event**, **LexicalHandler interface**, 515
  - `startPrefixMapping` **event**, 488, 499
  - `starts-with( )` **function**, 276, 433
  - start-tags**, 23, 24, 31–32, 73, 250, 358
  - `startup` **script**, eXist, 383
  - `state` **variable**, 549
  - statements**, 526
  - static documents**, 378
  - static graphics**, **Scalable Vector Graphics (SVG)**, 769
  - `status` **attribute**, 353
  - `step` **attribute**, `xforms:range` **element**, 820
  - stored procedures**, 396
  - `str` **function**, 544
  - Strict documents**, 741
  - strict flavor**, 90
  - strict option**, 553
  - Strict XHTML 1.0**, 757, 761
  - `string( )` **function**, 276
  - `string format-number` **function**, 964
  - `string generate-id` **function**, 965
  - string handling for XSLT 2.0**, 323
  - `string` **object**, 934
  - `string` **type**, 175, 187, 189, 967
  - `string unparsed-entity-uri` **function**, 965
  - `string` **value**, 163, 256
  - `StringBuffer` **function**, 894
  - `string-length( )` **function**, 276
  - strings**
    - delimiting, 360
    - functions, 276–278
    - handling, 327–330
    - representation, 444
    - XPath 1.0, 259
  - `stroke` **attribute**, 773, 776
  - `stroke-width` **attribute**, 773, 776

- `<strong>` **element**, [32](#), [109](#), [138](#), [139](#), [222](#), [236](#), [239](#), [888](#), [890](#), [913](#)
- `struct topicExchange.getChannelInfo` (string topicName) **method**, [576](#)
- `struct topicExchange.getChannels( )` **method**, [576](#)
- `struct topicExchange.ping`(string topicName, struct details) **method**, [576](#)
- `struct` **value sets**, **XML-remote procedure call (RPC)**, [577–579](#)
- structural markup, of (X)HTML documents**, [692](#)
- Structure module**
  - XHTML, [760](#)
  - XHTML Basic, [762](#)
- structured data**, [17](#), [376](#)
- Structured Extension Element**, [555](#)
- Structured Query Language (SQL)**, [340](#)
- `style` **attribute**, [84](#), [86](#), [636](#), [736](#)
- `<style>` **element**, [736](#), [763](#)
- Stylesheet module**, **XHTML**, [761](#)
- `stylesheet-prefix` **attribute**, `xsl:namespace-alias`, [951](#)
- styling XHTML documents**, [751–752](#)
- stylistic markup**, [692](#), [736](#)
- `submission` **attribute**, `xforms:submit` **element**, [820](#)
- `submitLoanInformation` **function**, [864](#)
- subquery**, [407](#)
- `substring( )` **function**, [276](#)
- `substring-after( )` **function**, [276](#)
- `substring-before( )` **function**, [276](#)
- Subversion**, [378](#), [379](#)
- `suggest( )` **method**, [662](#), [666](#), [668](#), [671](#)
- `SuggestionProvider` **class**, [659](#), [671–673](#)
- suggestions**
  - adding, [666–667](#)
  - creating list, [663–665](#)
  - showing, [660–662](#)
- `suggestionsBox` **instance variable**, [663](#)
- `sum( )` **function**, [276](#)
- `<summary>` **element**, [764](#)
- surface syntax**, [279](#)
- SVG (Scalable Vector Graphics)**
  - advantages, [768](#)
  - annotation, [782–784](#)
  - comments, [782–784](#)
  - described, [767–771](#)
  - enhancing display with, [865–868](#)
  - example, [771–774](#)
  - grouping, [776](#)
  - images, [780–781](#)
  - metadata, [782–784](#)
  - painter's model, [774–776](#)
  - paths, [777–780](#)
  - publishing on website, [785–786](#)
  - resources, [799–800](#)
  - sample application
    - script, [792–800](#)
    - SVG shapes, [788–792](#)
    - XHTML wrapper, [787](#)
  - scripting, [784–785](#)
  - text, [781–782](#)
  - tools, [769–771](#)
  - transformations, [776–777](#)
  - units, [774](#)
  - uses, [769](#)
  - views, [774](#)
- SVG Validator**, [770](#)
- SVG Viewer**, [448](#), [842](#)
- SVG wiki**, [799](#)
- SVG Zone**, **Adobe**, [800](#)
- SVG#**, [771](#)
- `<svg>` **element**, [771](#), [789](#), [793](#), [866](#), [867](#)
- `svgDoc` **variable**, [793](#)
- SVGDraw**, [771](#)
- SVG.org**, [799](#)
- SVGStudio**, [770](#)
- `svgWindow` **variable**, [866](#), [869](#)
- syndication**, [522](#)
- syndication feed**, [522](#)
- syndication servers**, [524](#)
- syndication systems**
  - client-consumer, [523](#)
  - client-producer, [523–524](#)
  - evolution
    - Atom, [533–535](#)
    - Microsoft's Content Definition Format (CDF), [525–526](#)
    - Netscape's RDF Site Summary (RSS) 0.9, [526](#)
    - RSS 0.91, [528–529](#)
    - RSS-DEV and RSS 1.0, [529–531](#)
    - simplicity, [526–528](#)
    - UserLand's RSS 2.0, [531–533](#)
    - UserLand's Scripting News format, [526](#)
  - server-consumer, [524–525](#)
  - server-producer, [523](#)

## **syntax**

- attributes, 39–45
- comments, 45–49
- declarations
  - character encoding, 51–53
  - standalone attribute, 53–54
  - version attribute, 51
- document type definitions (DTDs), 142–143
- elements, 25–39, 49–50
- errors, 64
- illegal parsed character data (PCDATA) characters
  - character data (CDATA) sections, 61–62
  - escaping characters, 60–61
- modeling feeds, 538–539
- namespaces
  - attributes, 83–86
  - default namespaces, 75–81
  - different notations, 81–83
- parsers, 24
- processing instructions (PIs), 56–59
- RELAX NG compact (RNC), 212–213
- tags, 24–39
- XML Path Language (XPath), 259
- XML Schemas, 146
- XQuery, 359–360

## **system identifiers, 100–101, 102, 140**

### **SYSTEM keyword, 100**

### **System Variables section, 290**

#### **System.IO file, 854**

#### **<system.web> element, 686**

#### **System.Xml class, 674**

#### **System.Xml package, 851**

## **T**

### **T command, 778**

#### **Table module, XHTML, 760**

#### **Table property, CSS, 695**

#### **table variable, 412**

#### **<table> element, 719**

#### **table-layout property, CSS, 695**

#### **tabular data, laying out, 719–721**

#### **<Tag> element, 38**

#### **tagList datatype, 244**

### **tags, rules**

- case sensitivity, 35
- matching start- and end-tags, 31–32

### naming conventions, 34–35

### properly nested, 32–33

### root elements, 33–34

### whitespace, 36–39

### tags **attribute, 130, 192, 243**

### **Tailrank, 524**

### **Tamino database, 346**

#### **tangram.es file, 787, 792**

#### **tangram.html file, 787**

#### **tangram.svg file, 787**

#### **target attribute, 785**

#### **Target module, XHTML, 761**

#### **target prefix, 151, 157**

#### **targetNamespace attribute, 151, 152, 175, 206**

#### **TCP (Transmission Control Protocol), 572**

#### **TechMeme, 524**

#### **temperatureRequestType type, 634**

#### **templates, named, 322**

#### **terminate attribute, xsi:message, 951**

#### **test attribute**

##### **xsi:if, 949**

##### **xsi:when, 962**

##### **xsi:with-param, 962**

### **text**

#### **RELAX NG patterns, 213–227**

##### cardinality, 216–217

##### connector patterns and grouping, 217–220

##### co-occurrence constraints, 221

##### elements and attributes, 216

##### empty patterns, 222–223

##### enumerated values, 220–221

##### mixed content patterns, 222

#### **Scalable Vector Graphics (SVG), 781–782**

#### **text ( ) function, node tests, 928**

#### **text accumulator, 551**

#### **text files, 5**

#### **text instance variable, 549**

#### **text keyword, 221**

### **Text module**

#### **XHTML, 760**

#### **XHTML Basic, 762**

#### **Text nodes, 251, 255, 256, 432, 468–471**

#### **text pattern, 213, 242**

#### **Text property, CSS, 694**

#### **text string, 543**

#### **<Text> element, 623**

#### **<text> element, 782, 870**

#### **text-align property, CSS, 695**

- text-anchor **attribute**, 792
  - text-decoration **property**, CSS, 695
  - text/html **MIME type**, 745, 746, 747, 754, 758, 759
  - text-indent **property**, CSS, 695
  - TextNodes.html **file**, 468
  - TextRange **class**, 660
  - text-transform **property**, CSS, 695
  - throws **Exception declaration**, 489
  - time **type**, 188
  - title **attribute**, 152, 177, 204, 216, 217, 226, 231, 234, 235, 700, 724
  - title **property**, 15, 528
  - <title> **element**, 67–68, 75, 83, 264, 367, 550, 552, 560, 561, 699, 714, 783, 784, 788, 843
  - To **element node**, 304
  - token **type**, 187, 967
  - tokenize( ) **function**, 282
  - tokens **type**, 967
  - tools**
    - Scalable Vector Graphics (SVG)
      - editors, 770
      - programming tools, 770–771
      - viewers, 770
    - XForms, 804–810
    - XHTML, 753–754
    - XML Document Object Model (DOM), 450–458
    - XQuery
      - Microsoft SQL Server 2005, 346
      - Oracle, 346
      - Saxon, 343–345
      - Tamino database, 346
      - X-Hive database, 346
      - X-Hive.com online, 345–346
  - topicExchange.getChannel( ) **method**, 577
  - topicExchange.ping( ) **method**, 577
  - Topologi Schematron Validator**, 209
  - totalDigits **facet**, 194
  - <tr> **element**, 863
  - Trace **tab**, **Results window**, 387
  - track **variable**, 794, 795
  - trainElementValue **buffer**, 895
  - TrainReader **class**, 515, 897
  - Transact-SQL**, 416, 418
  - transform **attribute**, 776, 777, 793, 795, 798, 799, 871, 918, 919
  - transform **string**, 796
  - transformations**, 289, 776–777
  - Transitional DOCTYPE**, 787
  - transitional document type**, 738
  - Transitional documents**, 741
  - transitional flavor**, 90
  - Transitional XHTML 1.0**, 757
  - translate( ) **function**, 276
  - translate **attribute**, 791
  - translate **expression**, **transform attribute**, 776
  - translate **transformation**, 792
  - translate(x,y) **command**, 871
  - translators**, 4
  - Transmission Control Protocol (TCP)**, 572
  - transport **attribute**, 636
  - tree**, 15
  - TREX (Tree Regular Expression for XML)**, 211
  - trim **method**, 553
  - TRIM\_LENGTH **constant**, 549, 553
  - true( ) **function**, 275, 832, 936
  - tryAddElement( ) **method**, 477
  - try/catch **block**, 453, 456, 478, 514, 796, 860, 862
  - try...except **block**, 544
  - try...finally **block**, 502
  - tuple**, 550
  - .txt **extension**, 9
  - txtXml, 459
  - type **attribute**, 157, 159, 196, 221, 494, 535, 700, 789
  - type **channel**, 528
  - TYPE **option**, 407
  - type **property**, **XForms model items**, 829
  - Type selector**, 700
  - type="html" **attribute**, 535
  - type="text" **attribute**, 535
  - typed XML data**, 414
  - types**
    - MIME, 757–759
    - server-side content, 747
    - in XQuery, 361
  - <types> **element**, 633–634
- ## U
- UDDI (Universal Discovery, Description, and Integration)**, 602
  - <ul> **element**, 701, 728
  - ultra-liberal parser**, 537

# UML (Unified Modeling Language)

---

**UML (Unified Modeling Language), 119**  
**unabbreviated syntax, 259**  
**UName, 76**  
**unbounded number, 176**  
**undeclared namespace, 69**  
**underscore ( \_ ), 122, 135**  
**unescape method, 552**  
**un-escaped characters, 60**  
**Unicode, 51, 52, 61, 131, 340**  
**unicode-bidi property, CSS, 695**  
**Unified Modeling Language (UML), 119**  
**Uniform Resource Identifiers (URIs), 70–72, 355, 459, 812**  
**Uniform Resource Locators (URLs), 71, 86–88, 378, 459, 535**  
**union types, 193**  
**<union> declarations, 197–199**  
**unique identifiers, 130**  
**unique root HTML element, 743**  
**United Press International (UPI), 536**  
**units, Scalable Vector Graphics (SVG), 774**  
**universal data format, 6**  
**Universal Discovery, Description, and Integration (UDDI), 602**  
**universal identifier (URI), 526**  
**universal name, 76**  
**Universal Resource Names (URNs), 71, 86, 87–88**  
**Universal selector, 700**  
**Unix paths, 378**  
**UnknownString type, 198**  
**unparsed entity, 123**  
**unparsedEntityDecl event, 509**  
**unparsed-text( ) function, 327, 328, 329**  
**unsignedByte type, 187**  
**unsignedInt type, 187**  
**unsignedLong type, 187**  
**unsignedShort type, 187**  
**untyped XML data, 414**  
**Update specifications, 340**  
**UpdateXML function, 431, 435**  
**updating XML in MySQL, 435–436**  
**UPI (United Press International), 536**  
**upper-case( ) function, 278**  
**uri parameter, 492, 499**  
**uri-reference type, 967**  
**URIs (Uniform Resource Identifiers), 70–72, 355, 459, 812**

**URLs (Uniform Resource Locators), 71, 86–88, 378, 459, 535**  
**urn string, 87**  
**URNs (Universal Resource Names), 71, 86, 87–88**  
**use attribute, 179, 637, 950**  
**use element, 917**  
**use-attribute-sets attribute**  
    **xsl:attribute-set element, 942**  
    **xsl:copy template, 944**  
    **xsl:element, 947**  
**user-defined datatypes, 193**  
**user-defined functions, 342, 371–372**  
**UserLand**  
    **RSS 2.0, 531–533**  
    **Scripting News format, 526**  
**using directives, 851, 854**  
**using GetBasicTotal( ) method, 612**  
**using-xquery.html file, 345**

## V

**v command, 778**  
**valid element names, 160**  
**validate documents, 95**  
**validateCardNumber( ) method, 657**  
**validating XHTML documents, 754–757**  
**validation declaration, XQuery, 357**  
**value( ) method, 893**  
**value attribute, xsl:number, 952**  
**value attributes, 845**  
**value declaration, 120**  
**<value> child, 589**  
**values**  
    **attribute, XHTML 1.x, 742**  
    **default, 125–126, 162–163**  
    **enumerated, 220–221**  
    **fixed, 126, 162–163**  
    **implied, 127**  
    **required, 126–127**  
**var xhHTTP = new XmlHttpRequest( ) method, 591**  
**varchar datatype, 425**  
**variable declaration, 357, 370**  
**Variable Name text box, 291**  
**Variable Value text box, 291**

**variables**

\$book variable, 368  
 \$i variable, 363, 365  
 \$input variable, 370  
 \$lower variable, 278  
 \$p variable, 423  
 \$text variable, 278  
 @BasketPointer variable, 412  
 @BasketXml variable, 412  
 @XmlPointer variable, 410  
 actualText variable, 669  
 balancePath variable, 870  
 bLoaded variable, 453  
 CLASSPATH environment variable, 290, 486, 490  
 currentSuggestionIndex variable, 668  
 elementValue variable, 896  
 environment variable, 357  
 external variable, 357  
 global variables, 322, 794  
 iniFile variable, 329  
 lowercaseSearch variable, 677  
 myDoc variable, 417, 419  
 New System Variable, 291  
 oDom variable, 474  
 oXmlHttp variable, 651  
 PATH environment variable, 291, 490  
 polygon variable, 785  
 selectedPoly variable, 794, 795  
 state variable, 549  
 suggestionsBox instance variable, 663  
 svgDoc variable, 793  
 svgWindow variable, 866, 869  
 table variable, 412  
 text instance variable, 549  
 track variable, 794, 795  
 window variable, 866  
 xhHTTP variable, 587  
 zXml variable, 592, 595  
 zXmlHttp variable, 595

**variables, XSLT, 320–322****vector-based diagrams, 769****Verdana typeface, 699**

version **attribute, 50–51, 127, 130, 152, 182, 183, 192, 815, 954, 959, 960**

**version declaration, XQuery, 355**

<version> **element, 163**

VersionMismatch **identifier, 623**

vertical bar ( | ) **character, 108, 125**

vertical lineto **command, 778**

viewBox **attribute, 774**

**viewers, Scalable Vector Graphics (SVG), 770**

**views, Scalable Vector Graphics (SVG), 774**

**virtual folder, 609**

**Visual Basic language, 517**

**Visual Web Developer 2005, 842**

**vocabularies, sharing, 104**

**W****W3C (World Wide Web Consortium)**

overview, 18–19

XForms page, 804

XML Document Object Model (DOM), 446–448

XML Schema, 281, 323, 804, 807, 833–834, 837

**warning event, 505**

.wav **file, 581**

**Web Applications 1.0, 764****Web Forms 2.0, 764, 838****web interface, eXist, 384–386****Web Service Description Language (WSDL) file, 853****web services**

creating, 682–683

implementing, 673–680

proxy

building, 849–853

communicating with using Ajax, 858–864

remote procedure call (RPC)

Distributed Component Object Model (DCOM),  
573–574

Internet Inter-ORB Protocol (IIOP), 574

Java Remote Method Invocation (RMI), 575

network transports, 579–596

XML-RPC, 576–579

REpresentational State Transfers (RESTs), 596–600

standards

Simple Object Access Protocol (SOAP), 600–601  
specifications, 602–604

Universal Discovery, Description, and Integration  
(UDDI), 602

Web Services Description Language (WSDL),  
601–602

## **Web Services Business Process Execution Language (WS-BPEL), 603**

### **Web Services Description Language (WSDL)**

- <binding> element, 635
- <definitions> element, 633
- groundwork, 608
- <messages> element, 634
- other bindings, 641–644
- <portTypes> element, 635
- <service> element, 638–641
- <soap:binding> element, 636
- <soap:body> element, 637
- <soap:operation> element, 636–637
- <types> element, 633–634
- Windows 2000, 608
- Windows 2003, 608
- Windows XP, 608
- web.config file, **686, 852**
- WebDAV (Web-based Distributed Authoring and Versioning), 378, 387–389**
- weblog tools, 523**
- webserviceex.CurrencyConvertor instance, **684**
- <website> element, **885, 886**
- weight element, **498**
- wget **command-line application, 565, 903**
- wget **download tool, 786**
- where clause, **365–366, 892**
- while **loop, 870**
- white pages, 602**
- whitespace, 36, 123, 132, 152**
- whiteSpace facet, 194**
- white-space **property, CSS, 695**
- width **attribute, 774, 787, 789**
- width **property, 709, 712, 718**
- wildcards**
  - attributes, 180–182
  - declaration, 164
  - elements, 163–165
  - namespaces with, 239
- window **variable, 866**
- Wireless Markup Language (WML), 288, 759**
- WITH clause, 410**
- WML (Wireless Markup Language), 288, 759**
- word processors, 4**

- word-spacing **property, CSS, 695**
- World Wide Web Consortium (W3C)**
  - overview, 18–19
  - XForms page, 804
  - XML Document Object Model (DOM), 446–448
  - XML Schema, 281, 323, 804, 807, 833–834, 837
- wrappers, XHTML, 787**
- writing XHTML**
  - attribute values, 742
  - case sensitivity, 742
  - changes between XHTML and HTML, 747–750
  - character encoding, 745–747
  - DOCTYPE declaration, 741
  - IDs, 744
  - specifying language, 745
  - well-formedness, 743–744
  - XML declaration, 740
- <wrox:countries> **element, 677**
- <wrox:GetCountriesResponse> **element, 677**
- WroxServices virtual directory, 677**
- WS Reliable Messaging, 604**
- WS-Addressing, 603**
- WS-BPEL (Web Services Business Process Execution Language), 603**
- WS-Choreography, 603**
- WSDL (Web Service Description Language) file, 853**
- WSDL (Web Services Description Language)**
  - <binding> element, 635
  - <definitions> element, 633
  - groundwork, 608
  - <messages> element, 634
  - other bindings, 641–644
  - <portTypes> element, 635
  - <service> element, 638–641
  - <soap:binding> element, 636
  - <soap:body> element, 637
  - <soap:operation> element, 636–637
  - <types> element, 633–634
  - Windows 2000, 608
  - Windows 2003, 608
  - Windows XP, 608
- WS-Eventing, 603**
- WS-Policy, 604**
- WS-Security, 604**

**X****x** attribute, 777**XDR** schemas, 395**Xerces** parser, 502, 508**Xerces2** parser, 485

xercesImpl.jar file, 486, 490

**XFormation**, 807, 836**XForms**

Action Module, 835–836

alternatives

Adobe LiveCycle, 837

HTML forms, 838

Microsoft InfoPath, 836–837

compared to HTML forms, 804

constraining instances

choosing between schema and bind elements, 834

W3C XML Schema in XForms, 833–834

xforms:bind element, 828–833

controls

xforms:input element, 817–818

xforms:output element, 818–819

xforms:range element, 819–820

xforms:secret element, 818

xforms:select element, 821–822

xforms:select1 element, 822–824

xforms:submit element, 820–821

xforms:textarea element, 818

xforms:trigger element, 820

xforms:upload element, 819

events, 834–835

example, 810–817

tools, 804–810

xforms:instance element, 813–814

xforms:model element, 812–813

xforms:submission element, 814–815

XML namespaces in XForms documents, 816–817

XML Schema in, 833–834

XPath 1.0 in, 815–816

xforms namespace prefix, 810

xforms:action element, 835

xforms:alert element, 819, 820

xforms:bind element, 813, 828–833, 834

xforms:dispatch element, 835

xforms:filename element, 819

xforms:help element, 820

xforms:hint element, 820

xforms:input element, 815, 816, 817–818, 836

xforms:instance element, 813–814, 815, 816, 822, 824, 825, 826, 832, 833

xforms:itemset element, 826, 827

xforms:label element, 819, 820, 826, 827

xforms:load element, 835

xforms:mediatype element, 819

xforms:message element, 835

xforms:model element, 812–813, 814, 815, 828

XForms:output control, 827

xforms:output element, 818–819

xforms:range element, 819–820

xforms:rebuild element, 835

xforms:recalculate element, 835

xforms:refresh element, 835

xforms:reset element, 836

xforms:revalidate element, 836

xforms:secret element, 818, 821–822, 825, 827, 921

xforms:select1 element, 821, 822–824, 826, 827, 921

xforms:send element, 836

xforms:setfocus element, 836

xforms:setvalue element, 836

xforms:submission element, 813, 814–815, 820, 824, 832, 921

xforms:submit element, 814, 820–821, 827, 921

xforms:textarea element, 818

xforms:trigger element, 820, 835

xforms:upload element, 819

xforms:value element, 821, 827

xforms-value-changed event, 818, 822

xhHTTP variable, 587

**X-Hive** database, 346**X-Hive.com** online, 345–346**XHTML**

future of, 763–765

modularized

module implementations, 761

XHTML 1.1, 761–762

XHTML Basic, 762–763

## XHTML (continued)

- namespaces, 725
- separating style from content, 736–738
- wrappers, 787
- XHTML 1.x
  - constraints, 753
  - Document Type Definitions (DTDs), 738–739
  - MIME types pitfalls, 757–759
  - styling documents, 751–752
  - tools, 753–754
  - validating XHTML documents, 754–756
  - validation pitfalls, 756–757
  - writing, 740–751
- XHTML Basic, 738, 741, 762–763**
- xhtml **elements, 239**
- xhtml: namespace, **560**
- xhtml **prefix, 69, 538, 560**
- XHTML wrappers, Scalable Vector Graphics (SVG), 787**
- <xhtml> **element, 743**
- xhtml:body **attribute, 533**
- xhtml:body **element, 560**
- xhtml:br **element, 239**
- xhtml:h1 **element, 560, 561**
- xhtml:html **element, 560**
- <xhtml:object> **element, 239**
- xhtml:p **element, 561**
- <xhtml:script> **element, 239**
- <xhtml:title> **element, 69, 560**
- XLink, 89, 721–724**
- xlink **prefix, 917**
- xlink:actuate **attribute, 722**
- xlink:href **attribute, 722**
- xlink:show **attribute, 722, 725**
- xlink:title **attribute, 722**
- xlink:type **attribute, 722**
- xlm.sax .ContentHandler **instance, 548**
- XML (Extensible Markup Language)**
  - benefits of
    - extensibility, 12–13
    - parsers, 11–12
  - compared to HyperText Markup Language (HTML), 13–14
  - data files
    - binary files, 4
    - history of markup languages, 6–7
    - text files, 5
  - description, 7–10
  - document types, 17–18
  - grouping of information, 15–17
  - growth of, 376–377
  - SQL Server 2000, 395–425
  - standards
    - components, 19–20
    - World Wide Web Consortium (W3C), 18–19
  - storing
    - conventional databases, 379–381
    - file systems, 378–379
    - native databases, 381–382
  - uses
    - distributed computing, 21
    - e-commerce, 21–22
    - reducing server load, 20
    - website content, 20–21
  - XML-based data, compared to relational data, 377
- xml **datatype, 381, 413–416, 424, 425**
- XML Document Object Model (DOM)**
  - Document interface, 478–480
  - Node object
    - DOMException object, 476–478
    - effect of Text nodes, 468–471
    - loading XML documents, 462–468
    - methods, 461–462
    - NamedNodeMap object, 471–475
    - NodeList object, 475
    - properties, 458–461
  - nodes, 448–449
  - purpose of, 445–446
  - tools, 450–458
  - used in InfoPath 2007, 481–482
  - W3C specifications, 446–448
- XML namespace, 810**
- xml **option, 437**
- xml **output method, 563**
- XML parser, 34, 88, 133, 875**
- XML Path Language (XPath)**
  - axes
    - ancestor, 925
    - ancestor-or-self, 925
    - attribute, 925
    - child, 925
    - descendant, 925
    - descendant-or-self, 926

- following, 926
- following-sibling, 926
- namespace, 926
- parent, 927
- preceding, 927
- preceding-sibling, 927
- self, 927
- functions, 929–937
- node tests
  - comment( ) function, 928
  - examples using meta character \*: , 928
  - node( ) function, 928
  - processing-instruction(Literal) function, 928
  - text( ) function, 928
- predicates, 278
- structure of XPath expressions, 278–281
- syntax, 259
- visualizing
  - context, 252–254
  - nodes, 254–257
  - XPath 1.0 expression types, 257–259
- XML documents
  - modeling, 250–251
  - ways of representing, 250
- XML Pointer (XPointer), 249**
- XML Query Language (XQuery), 281**
- XML Schemas**
  - <attribute> declarations
    - attribute qualified form, 179
    - attribute use, 179–180
    - attribute wildcards, 180–182
    - default values, 180
    - global, 177–178
    - local, 177
    - naming attributes, 179
    - referring to existing global attribute, 179
  - <attributeGroup> declarations, 183–185
- benefits of
  - content models, 147
  - data types, 147
  - namespace support, 146
  - XML syntax, 146
- compared to bind elements, 834
- compared to document type definitions (DTDs), 147–148
- <complexType> declarations, 165–166
- content models
  - <all> declarations, 171–174
  - <choice> declarations, 169–170
  - <group> references, 170–171
- creating elements with simple content and attributes, 185–186
- creating from multiple documents
  - <import> declarations, 200–204
  - <include> declarations, 204–206
- datatypes
  - built-in datatypes, 187–192
  - user-defined datatypes, 193
- documenting
  - annotations, 208–209
  - attributes from other namespaces, 207
  - comments, 206–207
- documents, 148
- <element> declarations
  - cardinality, 161–162
  - default values, 162–163
  - element qualified form, 161
  - element wildcards, 163–165
  - fixed values, 162–163
  - global, 156, 157–158
  - local, 156–157
  - naming elements, 160–161
  - referring to existing global element, 159–160
- example, 148–152
- <group> declarations, 167–168
- <list> declarations, 196–197
- modularization using the include directive, 231–232
- <restriction> declarations, 194–196
- <schema> declarations
  - attribute qualification, 154–155
  - element qualification, 154–155
  - target namespaces, 153–154
  - XML Schema namespaces, 153
- <simpleType> declarations, 193
- in SQL Server 2005, 424–425
- <union> declarations, 197–199
- in XForms, 833–834
- xml\_schema\_ namespace **intrinsic function, 425**
- XML-aware tool, 81**
- XmlDocument **object, 852**
- XML-enabled relational database, 381**
- XMLFilter **interface, 516**

# XmlHttpRequest class

---

XmlHttpRequest **class**, 592, 646

xmlHttpRequest **library**, 859

xmlHttpRequest **object**, 859, 860, 861, 862

xmlHttpRequest.js **library**, 859

XMLHttpRequest **class**, 462, 592, 650

XMLHttpRequest **object**, 592, 909

xml:lang **attribute**, 270, 739, 745, 747

xmlns **attribute**, 74, 76, 77, 80, 912

xmlns="" **attribute**, 882

xmlns:pers **attribute**, 72

XMLReader **object**, 491, 509, 513, 516

## XML-remote procedure call (RPC)

application programming interface (API), 576

parameters, 577

requests, 577

struct value sets, 577–579

using HTTP for, 584–587

## XMLSerializer class, 457

xml:space **attribute**, 270, 469

## XMLSpy, 836

xml:stylesheet **processing instruction**, 806

<?xml-stylesheet?> **processing instruction**, 699, 751

## XMLwriter, 836

xmml **namespace prefix**, 270, 812, 817

xmml:Book **element node**, 270

xmml:SizeAvailable **element**, 827

xmml:ToppingAvailable **elements**, 827

xmml:Toppings **node**, 827

## XP parser, 485

## XPath (XML Path Language)

### axes

ancestor, 925

ancestor-or-self, 925

attribute, 925

child, 925

descendant, 925

descendant-or-self, 926

following, 926

following-sibling, 926

namespace, 926

parent, 927

preceding, 927

preceding-sibling, 927

self, 927

functions, 929–937

node tests

comment( ) function, 928

examples using meta character \*, 928

node( ) function, 928

processing-instruction(Literal) function, 928

text( ) function, 928

predicates, 278

structure of XPath expressions, 278–281

syntax, 259

visualizing

context, 252–254

nodes, 254–257

XPath 1.0 expression types, 257–259

XML documents

modeling, 250–251

ways of representing, 250

## XPath 1.0

### axes

ancestor axis, 264–265

ancestor-or-self axis, 265

attribute axis, 262–264

child axis, 260–261

descendant axis, 265–266

descendant-or-self axis, 266

following axis, 266–267

following-sibling axis, 268

namespace axis, 268–270

parent axis, 271

preceding axis, 271–272

preceding-sibling axis, 272–273

self axis, 273–274

### expression types

Booleans, 258

node-sets, 258

numbers, 258

strings, 259

### functions

Boolean, 274–275

node-set, 275

numeric, 275–276

string, 276–278

in XForms, 815–816

## XPath 2.0

compared to XPath 1.0, 282–285

compared to XQuery, 342–343

datatyping control, 924

development of, 341–342

- features, 282–285
- new sequencing, 924
- revised ata model, 281
- revised data model, 281
- W3C XML Schema data types, 281
- XPointer (XML Pointer), 249**
- XQuery**
  - creation of
    - compared to XSLT and XPath, 342–343
    - current status, 341–343
    - historical factors, 340–343
    - technical factors, 340–341
  - data model
    - axes in, 361
    - comparing items and nodes, 361
    - document order, 361
    - node kinds, 361
    - sequences of node-sets, 361
    - shared with XPath 2.0 and XSLT 2.0, 360
    - types in, 361
  - examples
    - computed constructors, 358–359
    - element constructors, 351–355
    - input functions, 346–348
    - prolog, 355
    - retrieving nodes, 348–351
    - syntax, 359–360
  - expressions
    - FLWOR expressions, 362–368
  - extensions to in SQL Server 2005, 416–423
  - functions, 368–370
  - future of, 372
  - in SQL Server 2005, 416
  - tools
    - Microsoft SQL Server 2005, 346
    - Oracle, 346
    - Saxon, 343–345
    - Tamino database, 346
    - X-Hive database, 346
    - X-Hive.com online, 345–346
    - user-defined functions, 371–372
    - using parameters with, 370
- XQuery (XML Query Language), 281**
- xquery keyword, 355**
- XQuery prolog, 371**
- XQuery Recommendation, 20**
- .xquery suffix, 347**
- xs prefix, 153, 158, 178, 357**
- xs: prefix maps, 924**
- xs:boolean value, incremental attribute, 818**
- xsd datatype prefix, 241**
- xsd:anyURI value, 348**
- xs:date datatype, 824**
- xs:date element, 834**
- xsd:decimal type, 243**
- xsd:float type, 243**
- xsi prefix, 400**
- xsi:nil attribute, 400**
- XSINIL keyword, 400**
- xsi:schemaLocation attribute, 185**
- XSL (Extensible Stylesheet Language), 19**
- xsl: apply-imports element, 940**
- xsl: apply-templates element, 941**
- xsl: attribute element, 941–942**
- xsl: attribute-set element, 942**
- xsl: call-template element, 943**
- xsl: choose element, 943**
- xsl: comment element, 944**
- xsl: copy element, 944**
- xsl: copy-of element, 945**
- xsl: decimal-format element, 945–947**
- xsl: element element, 947**
- xsl: fallback element, 948**
- xsl: for-each element, 948–949**
- xsl: if element, 949**
- xsl: import element, 949**
- xsl: include element, 950**
- xsl: key element, 950**
- xsl: message element, 951**
- xsl: namespace, 560**
- xsl namespace prefix, 940**
- xsl: namespace-alias element, 951–952**
- xsl: number element, 952–953**
- xsl: otherwise element, 953**
- xsl: output element, 954–955**
- xsl: param element, 956**
- xsl: preserve-space element, 956**
- xsl: processing-instruction element, 957**
- xsl: sort element, 957–958**
- xsl: strip-space element, 958**
- xsl: stylesheet element, 958–959**
- xsl: template element, 959**

# xsl: template element

---

- xsl: text **element, 960**
- xsl: transform **element, 960–961**
- xsl: value-of **element, 961**
- xsl: variable **element, 961–962**
- xsl: when **element, 962**
- xsl: with-param **element, 962–963**
- xsl:analyze-string **instruction, 327, 328, 329**
- xsl:apply-templates **element, 300, 304, 307, 312, 314, 315, 317, 891, 940**
- xsl:applytemplates **template, 943**
- <xsl:apply-templates> **element, 253, 296–297**
- xsl:attribute **element, 301**
- <xsl:call-template> **element, 322**
- <xsl:choose> **element, 306, 307, 308–311, 330, 891**
- XslCompiledTransform **class, 675**
- xsl:copy **element, 287, 299, 300, 302**
- <xsl:copy> **element, 299–303**
- <xsl:copy-of> **element, 287, 299, 303–305, 332**
- xsl:decimal-format **element, 323**
- xsl:element **element, 303, 941, 942**
- <xsl:for-each> **element, 267, 268, 272, 311–312, 314, 342, 362, 943**
- xsl:for-each-group **instruction, 324, 326**
- xsl:if **element, 306, 307, 318, 362, 891**
- <xsl:if> **element, 306–308**
- xsl:matching-substring **instruction, 327**
- <xsl:namespace-alias> **element, 951**
- xsl:non-matching-substring **element, 327, 328, 329**
- xsl:otherwise **element, 309, 891**
- <xsl:output> **element, 680**
- xsl:param **element, 320**
- <xsl:Param> **element, 677**
- xsl:parameter **element, 320**
- xsl:result-document **element, 330, 332**
- <xsl:sort> **element, 312–314, 327, 362**
- <xsl:stylesheet> **element, 295–296, 320, 322, 333, 564, 676, 968**
- XSLT (Extensible Stylesheet Language Transformations)**
  - conditional processing
    - <xsl:choose> element, 308–311
    - <xsl:if> element, 306–308
  - declarative programming, 292–293
  - defined
    - presenting XML content, 288
    - restructuring XML, 288
  - elements
    - xsl: apply-imports, 940
    - xsl: apply-templates, 941
    - xsl: attribute, 941–942
    - xsl: attribute-set, 942
    - xsl: call-template, 943
    - xsl: choose, 943
    - xsl: comment, 944
    - xsl: copy, 944
    - xsl: copy-of, 945
    - xsl: decimal-format, 945–947
    - xsl: element, 947
    - xsl: fallback, 948
    - xsl: for-each, 948–949
    - xsl: if, 949
    - xsl: import, 949
    - xsl: include, 950
    - xsl: key, 950
    - xsl: message, 951
    - xsl: namespace-alias, 951–952
    - xsl: number, 952–953
    - xsl: otherwise, 953
    - xsl: output, 954–955
    - xsl: param, 956
    - xsl: preserve-space, 956
    - xsl: processing-instruction, 957
    - xsl: sort, 957–958
    - xsl: strip-space, 958
    - xsl: stylesheet, 958–959
    - xsl: template, 959
    - xsl: text, 960
    - xsl: transform, 960–961
    - xsl: value-of, 961
    - xsl: variable, 961–962
    - xsl: when, 962
    - xsl: with-param, 962–963
- example, 289–292
- foundational elements
  - <xsl:apply-templates> element, 296–297
  - <xsl:stylesheet> element, 295–296
  - <xsl:template> element, 296
- functions, 323, 963–967
- influencing output with <xsl:output> element, 306
- modes, 314–319
- named templates, 322
- parameters, 320–322
- procedural programming, 292
- processors, 288–289

- Saxon XSLT processor, 289–292
  - transforming RDF Site Summary (RSS) with
    - browser processing, 564–565
    - generating feed from existing data, 558–559
    - pre-processing feed data, 565
    - processing feed data for display, 561–563
    - syndication formats, 565–567
  - using information from source tree
    - <xsl:copy> element, 299–303
    - <xsl:copy-of> element, 303–305
    - <xsl:value-of> element, 297–299
  - variables, 320–322
  - <xsl:call-template> element, 322
  - <xsl:for-each> element, 311–312
  - <xsl:sort> element, 312–314
  - XSLT 2.0
    - expressions, 967–968
    - grouping, 324–327
    - Muenchian Method, 968–969
    - multiple outputs, 330–332
    - non-XML input and string handling, 327–330
    - user-defined functions, 332–334
    - using, 968–969
    - and XQuery 1.0, 969
    - xsl:value-of changes, 334–335
  - XSLT (Extensible Stylesheet Language Transformations) 2.0**
    - compared to XQuery, 342–343
    - development of, 341–342
    - expressions, 967–968
    - grouping, 324–327
    - Muenchian Method, 968–969
    - multiple outputs, 330–332
    - non-XML input and string handling, 327–330
    - user-defined functions, 332–334
    - using, 968–969
    - and XQuery 1.0, 969
    - xsl:value-of changes, 334–335
  - X-Smiles browser, 805**
  - XSS (Cross-Site Scripting), 849**
  - xs:token element, 834
- ## Y
- y attribute, 777
  - years-from-yearMonthDuration( ) function, 283
- ## Z
- Z command, 778
  - zero-digit attribute, xsl:decimal-format element, 946
  - z-index property, 718–719
  - ZipCode element, 258
  - zoom, 768
  - zXml library, 461, 651, 685
  - zXml variable, 592, 595
  - zXmlHttp variable, 595
  - zXml.js file, 459, 593, 659, 678
  - zXPath class, 595, 658





