

INDEX

- A**
 - absolute positioning, 288–289
 - add function, 40–41
 - addition function for XML data extraction, 82–83
 - addition operator (+) (JavaScript), 40, 44, 45
 - Ajax (Asynchronous JavaScript + XML). *See also* Ajax frameworks; XMLHttpRequest object
 - advantages, 2, 3, 62
 - attributes, 29, 262–263
 - downloading JavaScript, 94–95
 - event attributes, 29
 - example of data Submit and fetch, 4–5
 - importance of XMLHttpRequest object, 62
 - server-side scripts and PHP used with, 76–77
 - Ajax frameworks
 - ! caution: importance of files and placement, 152
 - ! caution: naming global variable “resultXML”, 171, 175
 - ! caution: on use of global variables, 155, 159
 - advantages, 152–153
 - Ajax tag library, 236–239
 - AJAXLib Ajax framework, 170–175
 - browser-based *versus* server-side, 152
 - downloadable and free, 182
 - installing and allowing access, 152–153
 - IWF (Interactive Website Framework), 200–205
 - JavaScript and server-side, 11, 152
 - LibAjax server-side Ajax framework, 224–229
 - libXmlRequest Ajax framework, 176–181
 - reading text with GET method, 154–157
 - reading text with POST method, 162–163, 164–165
 - reading XML from server, 158–159, 160–161, 166–167, 168–169
 - SACK Ajax framework, 188–193
 - SAJAX server-side Ajax framework, 212–217
 - Sarissa Ajax framework, 194–197
 - sending to server with URL encoding, 155, 157, 159, 161
 - Xajax server-side Ajax framework, 218–223
 - XHConn Ajax framework, 182–187
 - Ajax tag library
 - returning data from server, 239
 - support for autocomplete, 236–237
 - support for callouts, pop-ups, 237
 - updating HTML controls, 238–239
 - updating one field from results of another, 237
 - AJAXLib Ajax framework
 - ! caution: on lack of built-in error handling, 175
 - ! caution: on use of global variable resultXML, 171, 175
 - accessing XML on server, 172–173
 - downloading and installing, 170–171, 176
 - downloading XML data, 174–175
 - stripping out whitespace, 172, 173, 174
 - URL encoding, 171
 - alert boxes
 - display by JavaScript function, 30–31
 - displaying, 28–29
 - displaying text from text fields, 37
 - aligning text in CSS, 278–279
 - AND logical operator (JavaScript), 53
 - anonymous JavaScript functions, 68
 - Apache Tomcat Java-based server, 231
 - Apple Safari, 63
 - arguments, passing to functions, 38–39
 - arithmetic operators (JavaScript), 45
 - assignment operators (JavaScript), 44, 45
 - asynchronous *versus* synchronous requests, 178, 180
 - attributes (Ajax), 29, 262–263
 - autocomplete capability
 - example demonstration site, 8–9
 - support for, in Ajax tag library, 236–237
- B**
 - background color setting in CSS, 284–285
 - background image setting in CSS, 286–287
 - binary objects, 23, 140
 - block elements in HTML, 278
 - boldface, 24, 276–277
 - broadcasts with instant messaging, 21
 - browser events, 28–29, 34–35
 - browsers
 - ! caution: ignoring visibility style property, 292
 - ! caution: on avoiding caching at URL, 177, 200
 - ! caution: on passing JavaScript to, 95
 - accessing different domains, 118–119
 - checking before creating XMLHttpRequest object, 64–65
 - determining type and version, 54–55, 259
 - downloading images, text, or XML, 22–23
 - instant messaging with, 21
 - problems of display refresh, 2, 4, 10, 18
 - viewing JavaScript errors, 26–27

C

- calculator example (JavaScript operators), 45–47
- callback functions, 154
- calling JavaScript functions, 30–31
- callouts and pop-ups, Ajax tag library, 237
- capitalization of JavaScript variables, 43
- centering text in CSS, 278–279
- centimeters as units of measurement, 289
- check box creation and reading, 316–317
- `childNodes` and `siblingNodes` properties (JavaScript), 245, 248–251
- clicks. *See also* mouse events
 - browser events, 28–29, 34–35
- colors
 - background, setting in CSS, 284–285
 - foreground, setting in CSS, 280–281
- commas (,) separating arguments, 39
- commenting
 - HTML, 25
 - JavaScript, 25
- comparison operators (JavaScript), 45, 52–53, 137
- CSS (Cascading Style Sheets)
 - ! caution: use absolute positioning, 147
 - advantages, 10, 270
 - aligning text, 278–279
 - background color settings, 284–285
 - boldfacing text, 276–277
 - creating pop-ups, 294–295
 - creating style rules, 270–271
 - displaying menu systems, 144, 146–147
 - foreground color settings, 280–281
 - hiding menu systems, 148–149
 - moving HTML elements, 296–297
 - overlapping HTML elements, 298–299
 - positioning elements using absolute positioning, 288–289
 - positioning elements using relative positioning, 290–291
 - setting font family and size, 272–273
 - setting HTML element visibility, 292–293
 - specifying styles for HTML elements, 271
 - underlining or italicizing text, 274–275
- curly braces ({})
 - for code handling downloaded data, 68
 - enclosing JavaScript functions, 30

D

- debugging. *See* error handling
- decrement operators (JavaScript), 44, 45

- dialog boxes as alert boxes, 28–29
- Direct Web Remoting (DWR). *See* DWR (Direct Web Remoting)
- directories
 - for data for server-side scripts, 76, 77
 - importance of Ajax framework file placement, 152, 153
 - indicating parent, 73
- `displayText()` function, 36–37, 38–39
- do while loops (JavaScript), 61
- document objects (JavaScript), 24
- `document.getElementById('targetDiv').innerHTML` expression, 33, 42
- `document.getElementById('text1')` expression, 36, 37
- documents
 - analyzing and extracting header information, 108–109
 - checking existence and accessibility, 110–111
- `document.write` method, 24, 32
- domains, accessing, 118–119
- download time penalty, 2
- downloadable code for this book
 - `ajaxframework.js` sample Ajax framework, 160, 164
 - `movie.jpg`, 92
- drag and drop
 - avoiding multiple-page shopping, 12–13
 - handling mouse down and up events, 124–125
 - with RICO JavaScript framework, 206–207
 - support by IWF (Interactive Website Framework), 201
- driving directions from Google Maps, 17
- drop-down menus. *See* menu systems
- DWR (Direct Web Remoting)
 - downloading and installing, 230
 - editing interactive HTML table, 232–233
 - requiring Java-enabled Web server, 230
- dynamic HTML. *See also* HTML
 - display of updated elements, 10
 - downloading images and other binary objects, 22–23, 140–141
 - positioning text and images, 32–33

E

- e-mail handling with Echo2, 234–235
- Echo2 Java framework, 234–235, 235
- else clause of if statements (JavaScript), 51
- `encodeURIComponent` function (JavaScript), 89, 99
- error handling. *See also* security considerations
 - ! caution: on absence from AJAXLib, 175
 - ! caution: on absence from XHConn, 187

INDEX

- accessing different domains, 118–119
- advantages of small JavaScript files, 27
- Ajax using Firefox browser, 120–121
- avoiding server-side accesses, 105
- in case of missing document or PHP script, 111
- in cases of multiple responses to XMLHttpRequest object, 113
- common HTTP errors, 69
- displaying customized error messages, 269
- exception objects (JavaScript), 75
- importance of download status and error checking, 83, 84
- JavaScript using browser, 26–27
- reading input and validating via server, 102–105
- XML documents with Internet Explorer, 241, 242–243, 268–269
- XMLHttpRequest objects with script, 120

eval function (JavaScript), 94

events. *See also* mouse events

- browser, 28
- button clicks for execution of JavaScript code, 34
- common attributes in Ajax applications, 29

exception objects (JavaScript), 75

Extensible Hypertext Markup Language (XHTML). *See* XML

F

file handles, 328

Firefox

- debugging capability, 26, 27, 120–121
- determining presence and version, 54–55
- disconnecting mouse events from JavaScript, 136
- Greasemonkey, 120, 121
- methods and properties for XMLHttpRequest object, 63
- mouse event handling, 122, 124, 126, 128, 130
- mouse listener functions, 134
- predefined colors for text, 280
- selecting Google as home page, 7
- specifying italics, 274

font

- font-weight property in CSS, 276
- formatting size or bold, 24
- formatting with CSS style rules, 270–283

Fonz and Ajax, 19

for loops, 56–57, 58

foreground colors, setting in CSS, 280–281

<form> elements, 34–35

forward slashes (//)

- for commenting JavaScript, 25

functions

- callback, 154
- creating JavaScript, 30–31
- definition, 24, 30
- inner, 116
- passing arguments to, 38–39
- returning data from, 40–41
- unlimited lines of code, 31

G

games online, 18–19, 131

GET method

- compared to POST method, 162, 163, 166, 167
- passing data to server-side scripts, 88–89
- reading or passing type parameter, 86–87
- URL-encoding of data, 67

getAllResponseHeaders method, 106, 107

getData function, 66, 67, 70, 72

getElementById method, 33, 36

getElementsByTagName method (JavaScript), 82, 247, 260–261

getResponseHeader method, 108–109

global JavaScript variables, 48–49

- ! caution: on use of, 155, 159

Google as Ajax-enabled site, 7

Google Maps, 16–17

Google Search

- example of live search, 6–7

Google Suggest, 6, 7

- connecting to, 98–99

- creating JavaScript, 100

- displaying search results, 100–101

Greasemonkey (Firefox), 120, 121

greater than/less than operators (> <) (JavaScript), 52

H

HEAD request, 106–107, 110

header information

- analysis and extraction of specifics, 108–109
- retrieval, 106–107

HTML. *See also* dynamic HTML

- adding + and - buttons for calculator, 47

- block elements, 278

- coding JavaScript in script element, 24–25

- commenting, 25
- controls updates with Ajax tag library, 238–239
- event attributes, 28
- in PHP scripts, 77
- reading controls data, 87

HTML elements. *See also* SACK Ajax framework; shopping cart application

- form for enclosure of controls, 34–35
- absolute positioning, 288–289
- hiding, 293
- as JavaScript objects, 33
- mouse event handling, 123, 128, 129, 130, 131
- moving with CSS, 296–297
- overlapping with CSS, 298–299
- relative positioning, 290–291
- setting visibility, 292–293
- z-index property, 299

I

if statements (JavaScript), 50–51

images

- background image settings in CSS, 286–287
- displaying with Ajax and dynamic HTML, 142–143
- downloading with Ajax and dynamic HTML, 140–141
- switching without browser refresh, 236

inches as units of measurement, 135, 289

increment operators (JavaScript), 44, 45

indexing of JavaScript arrays, 58, 260

inner functions (JavaScript), 116

innerHTML property, 33

<input> elements, 34

instant messenger-type programs with browsers, 21

interactive menus. *See* menu systems

Interactive Website Framework. *See* IWF (Interactive Website Framework)

Internet Explorer

- checking before creating XMLHttpRequest object, 64–65
- collapsing or expanding XML element display, 241
- debugging capability, 26, 120, 121
- determining presence and version, 54–55
- disconnecting mouse events from JavaScript, 136
- error handling in XML documents, 241, 242–243, 268–269
- error parser, 268–269
- methods and properties for XMLHttpRequest object, 62, 63
- mouse event handling, 122, 124, 126, 128, 130
- mouse listener functions, 134

- predefined colors for text, 280
- selecting Google as home page, 7
- specifying italics, 274
- support of marquee element, 54
- tests for XML documents, 242–243

italics for text

- specifying in CSS, 274–275

IWF (Interactive Website Framework)

- avoiding return of cached data, 200
- connecting to server, 202–203
- downloading and installing, 200–201
- downloading data, 204–205
- multiple XMLHttpRequest support, 205
- parsing of XML, 205
- use of Extensible Hypertext Markup Language (XHTML), 203

J

Java-based application frameworks

- DWR (Direct Web Remoting), 230–233
- Echo2, 234–235

Java-based servers, 231

JavaScript. *See also* RICO JavaScript framework

- ! caution: on passing to browsers, 95
- accessing HTML text fields, 36–37
- addition operator (+), 40
- backgroundColor property, 285
- backgroundImage property, 287
- boldfacing text, 277
- button clicks for execution of functions, 34–35
- capitalization convention for names of variables, 43
- childNodes and siblingNodes properties, 245, 248–251
- color property, 281
- colorizing new text for noticeability, 282–283
- commenting, 25
- creating addition function for XML data extraction, 82–83
- creating scripts, 24–25
- creating within HTML <script> element, 24–25
- creating XMLHttpRequest object, 64–65
- displaying alert boxes, 28–29
- displaying scrolling text, 283
- downloading using Ajax, 94–95
- errors, viewing using browser, 26–27
- files in src attribute of HTML script element, 27
- functions, 30–31, 38–39, 40–41
- handling browser events, 28–29

INDEX

- nodeType and nodeValue properties, 244
- passing arguments to functions, 38–39
- positioning text and images, 32–33
- quotation marks (" ") for, 28
- reading text in text boxes, 37
- reserved words, 43
- SAJAX server-side Ajax Framework, handling in, 212–217
- sent by Google Suggest for live search, 96–97
- timing execution of, 30–31
- underlining and italics, 275
- use in Ajax, 3, 4, 5
- users turning off, 5
- for XML documents, 244–245

JavaScript arrays

- ! caution: on zero-basing, 59
- handling multiple data items, 58–59
- indexing of, 58, 260
- for returning multiple values, 41

JavaScript library, prototype, 211

JavaScript objects

- HTML elements as, 33
- named node maps, 262

JavaScript operators

- calculator example, 45–47
- definition, 44
- for manipulating variables, 44–47
- order of execution, 44
- shortcut versions, 44

JavaScript variables

- ! caution: on inadvertent declaration of, 49
- ! caution: on reinitialization of values, 48
- local and global, 48–49
- manipulating with operators, 44–47
- naming, 43
- storing data in, 42–43

K

keystrokes

- sending to Google Suggest, 98–99
- watching and handling with JavaScript, 97

L

left-aligning text in CSS, 278–279

LibAjax server-side Ajax framework

- ! caution: does not work with IIS servers, 224
- advantages and disadvantages, 224, 225

- downloading and installing, 224–225
- exporting more than one function, 227
- passing multiple parameters to functions, 229
- using GET and POST and other HTTP methods, 225
- writing browser-side code, 228–229
- writing server-side code, 226–227

libXMLRequest Ajax framework

- accessing XML from server, 178–179
- avoiding return of cached data, 177
- checking for XMLHttpRequest object creation, 181
- downloading and installing, 176–177
- downloading XML, 180–181
- pooling of XMLHttpRequest objects, 179
- synchronous *versus* asynchronous requests, 178

live searches

- connecting to Google Suggest, 96–99
- Google Search example, 6–7

LiveGrid custom RICO control, 206, 207

local JavaScript variables, 48–49

logical operators (JavaScript), 45, 53

login process with instant feedback, 14–15

loops (JavaScript)

- with for loops, 56–57, 58

M

maps, Google Maps, 16–17

<marquee> element, 54

menu systems

- ! caution: erase contents when closing, 149
- creating Ajax-driven, 144–145
- displaying with CSS, 146–147
- drop-down menu display, 100, 131
- hiding, 148–149
- with HTML <select> control, 145
- reading selections, responding to mouse clicks, 150–151
- sending selection to server, 151

methods

- definition, 24
- for Internet Explorer, 62

millimeters as units of measurement, 289, 291

MIME (Multipurpose Internet Mail Extensions), 320

mouse events

- click and double-click events, 128–129
- converting to standard format, 122–123, 125
- down and up events, 124–125
- enter and leave events, 130–131

fetching text in response to mouseovers, 92–93
HTML elements for, 123, 128, 129, 130, 131
move events, 126–127

movie reel image display example, 92–93

Mozilla

disconnecting mouse events from JavaScript, 136
methods and properties for XMLHttpRequest object, 63
mouse event handling, 122, 124, 126, 128, 130
mouse listener functions, 134
predefined colors for text, 280
specifying italics, 274

N

named node maps (JavaScript), 262

Netscape Navigator and XMLHttpRequest object, 63

node maps (JavaScript), 262

nodes, XML as, 244

nodeType and nodeValue properties (JavaScript), 244

<noscript> element, 25

NOT logical operator (JavaScript), 53

O

objects, 24

onblur event attribute, 103

onchange event attribute, 29, 35

onclick event attribute, 29, 34

ondblclick event attribute, 29

ondragdrop event attribute, 29

onkeydown event attribute, 29

onkeypress event attribute, 29, 35

onkeyup event attribute, 29, 103

online chat with Ajax, 20–21

online games, 18–19, 131

online shopping. *See* shopping cart application; shopping online

onload event attribute, 29

onmouse... event attributes, 29, 35, 122–123

onreadystatechange property, 68, 71

onresize event attribute, 29

onsubmit event attribute, 29

onunload event attribute, 29

open method

for opening XMLHttpRequest object, 66–67

OR logical operator (JavaScript), 53

outerHTML property, 33

P

page loads. *See* browser events

page refresh

avoiding for logins, 14–15

avoiding in live searches, 6–7

avoiding with drag and drop, 12–13

avoiding with dynamic HTML and CSS, 10–11

time penalty, 2

Pandorabots, 18

parentheses (())

enclosing arguments (data) for JavaScript, 24, 38

indicating JavaScript functions, 30

password protection for login page, 15

performance considerations

! caution: on responding to every user keystroke, 97

fetching data from Google, 6, 7

page refresh time penalty, 2

PHP

concatenating items into single text string, 305

connecting to Google Suggest, 98

creating arrays, 306–307

creating HTML to send to browser, 302–303

creating PHP-enabled pages, 300–301

do while loops, 325

echo statement, 302–303, 304, 305

foreach loops, 326–327

GET and POST methods, 315

handling variables, 304–305

HTML in, 77

if statement, test conditions, and else clauses, 312–313

in LibAjax server-side Ajax framework, 224

for loops, 322–323

operators, 308–309

reading data from check boxes, 316–317

reading data from radio buttons, 318–319

reading data from text fields, 314–315

reading data sent to server, 86–87

reading files on server, 328–329

in SAJAX server-side Ajax Framework, 212–217

sending XML to browser, 320–321

as server-side programming, 76–77, 153

text string handling, 310–311

while loops, 324–325

in Xajax server-side Ajax framework, 218–223

XML in, 79

pixels as units of measurement, 135, 289, 291

INDEX

Plastic Shore, 20–21

plus signs (+) in URL-encoded data, 88

pop-ups

- Ajax tag library, 237
- creating with CSS, 294–295

POST method

- ! caution: set header properly, 163
- compared to GET method, 162, 163, 166, 167
- data hidden in HTTP headers, 67
- passing data to server-side scripts, 90–91

properties

- definition, 24
- for Internet Explorer, 62

prototype JavaScript library, 211

Q

question mark (?) in URL-encoded data, 88

quotation marks

- double (" ")
 - for JavaScript text, 28
 - for XML document values, 240
- single (' ')
 - inside quoted text, 38

R

radio button creation and reading, 318–319

readyState property, 68

refreshing screen. *See* page refresh

relative positioning, 290–291

reserved words in JavaScript, 43

responseText property, 94

resultXML global variable, 171, 175

return keyword, 40

return value of functions, 40, 41

RICO JavaScript framework

- accessing server, 208–209
- demonstration, 10–11, 207
- downloading and installing, 206–207
- downloading XML data, 210–211
- passing data to RICO Ajax engine, 209

right-aligning text in CSS, 278–279

S

SACK Ajax framework

- accessing server, 190–191
- downloading and installing, 188–189

downloading data, 192–193

element and execute properties, 190, 191

notifying user of creation failure, 189

SAJAX server-side Ajax Framework

- ! caution: does not work with IIS servers, 212
- ! caution: on multiple concurrent XMLHttpRequests, 213
- advantages over JavaScript, 217
- downloading and installing, 212–213
- support for GET and POST methods, 213
- writing browser-side code, 216–217
- writing server-side code, 214–215

Sarissa Ajax framework

- accessing server, 196–197
- advantages and disadvantages, 194, 195
- downloading and installing, 194–195
- downloading XML data, 198–199
- synchronous and asynchronous downloads, 196–197

screen flicker, 2

screen measurement, X and Y axis, 289, 291

<script> elements (HTML), 24, 27, 40

scrolling image display, 286–287

scrolling text display, 283

search, live, through Google, 6–7

security considerations

- browser warnings upon URL navigation attempts, 98
- browsers turning JavaScript off, 25
- GET method compared to POST method, 163, 166, 167
- JavaScript, 5
- passing JavaScript to browsers, 95
- password protection, 15
- POST *versus* GET method, 67

semicolon (;) ending lines of JavaScript, 24

sendRPCDone function (JavaScript), 100

server-side Ajax frameworks, 153

server-side scripts

- with Ajax, 76–77
- for calling another domain, 118–119
- fetching text or images in response to mouseovers, 92–93
- passing data with GET method, 88–89
- passing data with POST method, 90–91
- reading data sent to server, 86–87
- validating returned data, 104–105
- with XML documents, 79

servers

- accessing with libXMLRequest Ajax framework, 178–179
- Java-based, 231

`setRequestHeader` method, 90

SHA-1 encryption algorithm, 15

shopping cart application

Ajax-driven, creating, 132–133

dragging HTML element, 134–135

dropping HTML element, 136–137

updating cart on server, 138–139

shopping online

drag and drop into carts, 12–13

instant message capability, 21

`siblingNodes` and `childNodes` properties (JavaScript), 245

slashes (//) for commenting JavaScript, 25

`` elements, 32, 40

string operators (JavaScript), 45

`<style>` elements, 101

synchronous *versus* asynchronous requests, 178, 180

T

table format displays with RICO, 206

test score example of looping (JavaScript), 56–59

text fields

Ajax framework to read from server, 154–157

displaying text in, 36–37

reading with PHP, 314–315

text formatting. *See also* CSS (Cascading Style Sheets)

colorizing new text for noticeability, 282–283

text nodes, 84

Tom Riddle's diary, 18–19

tool tips, 130, 236

true/false tests

with `if` statements (JavaScript), 50–51, 52

with logical operators (JavaScript), 53

with `while` and `do while` loops, 60–61

`try/catch` statement (JavaScript), 74

U

underlining of text in CSS, 274–275

underscore (`_`) for separating words in variable names, 43

units of measurement, 135, 289

URI (Universal Resource Indicator), 89

URL-encoding of data

! caution: visibility of transmitted data, 67

with `encodeURIComponent` function, 89, 99

GET method compared to POST method, 162, 163, 166, 167

manual, with code `%20`, 89

for use by server-side scripts, 88–89

URLs

assigning to `src` attribute, 27

relative *versus* absolute, for downloads, 72–73

typical put-and-take communications, 62

user experience

annoying screen flicker, 2

delays while shopping online, 12

V

validating data, 102–105

`var` keyword, 42, 43

W

Web applications, Web pages

advantages of IWF for, 200, 201

advantages of on-the-fly page updating, 10–11

online revolution, 2

positioning text and images, 32–33

problems with display in browsers, 2

typical put-and-take with URLs, 62

`while` loops (JavaScript), 60–61

“whitespace,” stripping out from XML data, 172, 173, 174

X

Xajax server-side Ajax framework

downloading and installing, 218–219

passing back multiple results, 219, 221

writing browser-side code, 222–223

writing server-side code, 220–221

XHConn Ajax framework

! caution: avoiding domain warnings from browser, 184

! caution: on lack of built-in error handling, 187

accessing server, 184–185

checking creation, 183

downloading and installing, 182–183

downloading data from server, 186–187

GET method *versus* POST, 184, 185

passing `XMLHttpRequest` object, 182, 186, 187

XHTML (Extensible Hypertext Markup Language). *See* XML

XML. *See also* RICO JavaScript framework

addition function for data extraction, 82–83

collapsing or expanding element display in Internet Explorer, 241

INDEX

- creating, 240–241, 264–265
- displaying as text, 247
- downloading
 - with `AJAXLib`, 172–173, 174–175
 - with `libXmlRequest`, 176–177, 178–179, 180–181
 - reading from server with Ajax frameworks, 158–159, 160–161, 166–167, 168–169
 - from server, 22–23, 80–81
 - for use in Ajax, 78–79
- elements
 - and document elements, 78, 79, 240, 242
 - extracting data from, 84–85, 169
 - finding by name, 260–261
 - retrieval, 246–247
- extracting
 - attributes, 262–263
 - data from elements, 82–83, 84–85, 169
 - document elements, 246–247
 - text data, 252–253
- IWF, handling in, 200–205
- JavaScript, handling in, 244–245
- navigating
 - differences between Internet Explorer, Mozilla, and Firefox, 254–255
 - with `firstChild` and `lastChild` properties, 248–249
 - with `nextSibling` and `previousSibling` properties, 250–251
- node handling, 244–245, 248–253
- PHP, handling in, 79, 320–321
- Sarissa, handling in, 194–197
- valid documents
 - creating, 264–265
 - reporting validation errors, 268–269
 - validating, 266–267
 - validity concept, 242
- well-formedness concept, 242–243
- whitespace
 - differences between Internet Explorer, Mozilla, and Firefox, 174
 - stripping out, 172, 173, 174, 256–259
- [XML declaration](#), 78
- [XMLHttpRequest object](#)
 - ! caution: set header properly, 163
 - checking for creation with `if` statement, 65, 66
 - checking for creation with `libXmlRequest` framework, 181
 - checking status of download, 68, 69
 - common HTTP errors, 69
 - connecting JavaScript function to `onreadystatechange` property, 68
 - creating array of, 114–117
 - creating with JavaScript, 64–65
 - debugger script for, 120
 - deleting, 117
 - download preparation, 68–69
 - downloading data, 70–71
 - downloading XML from server, 80–81
 - `getAllResponseHeaders` method, 106, 107
 - `getResponseHeader` method, 108–109
 - HEAD request, 106–107, 110
 - importance in Ajax applications, 62
 - methods for Apple Safari, 63
 - methods for Internet Explorer, 62
 - methods for Mozilla, Firefox, and Netscape, 63
 - open method, 70
 - opening, 66–67
 - passing data with `POST` method, 90–91
 - passing downloaded data to function, 71
 - pooling for reuse, 176, 179
 - properties for Apple Safari, 63
 - properties for Internet Explorer, 63
 - properties for Mozilla, Firefox, and Netscape, 63
 - `responseText` and `responseXml` properties, 246, 247
 - `responseText` property, 70, 94
 - `responseXML` property, 80, 81
 - SAJAX support for multiple, 213
 - `send` method, 70
 - `send` method for multiple arguments, 91
 - `setRequestHeader` method, 90
 - URLs, relative or absolute, 72–73
 - using two simultaneously, 112–113
- [XMLHttpRequest objects](#)
 - creating newer versions in Internet Explorer, 74–75



“Your browser does not support Ajax” message, 65

