

INDEX

Numbers and Symbols

- + (addition) operator, 13, 15, 18, 145, 385
- += (addition assignment) operator, 34, 191
- & (ampersand) character, 212, 300
- && (and) operator, 17, 20, 93
- [] (array), 58–59, 72
- = (assignment) operator, 24, 61, 334
 - as expression, 158
 - for default value, 45
 - in Egg, 201
- * (asterisk) character, 13, 145
- @ (at sign), 255
- \ (backslash) character
 - as path separator, 349
 - in regular expressions, 142–143, 154, 393
 - in strings, 14, 213
- ` (backtick) character, 14–15
- { } (block), 29
- ^ (caret) character, 143, 148, 158, 343
- :
- :
- ,
- (decrement) operator, 34
- / (division) operator, 13
- \$ (dollar sign), 26, 148, 152, 158
- " (double quotation mark) character, 14, 161, 196, 213
- == (equal to) operator, 17, 19, 64, 78
- ** (exponentiation) operator, 31
- > (greater than) operator, 16
- >= (greater than or equal to) operator, 17
- # (hash sign), 206, 322
- (hyphen) character, 13, 143, 228
- ++ (increment) operator, 34
- < (less than) operator, 16
- <= (less than or equal to) operator, 17
- * (multiplication) operator, 13, 18
- *= (multiplication assignment) operator, 33
- ! (not) operator, 17, 31
- != (not equal to) operator, 17
- ?? (nullish coalescing) operator, 20, 49, 314
- { } (object), 61, 64, 72
- || (or) operator, 17, 19, 93, 388
- % (percent) character, 300
- . (period) character, 27, 59, 72, 75
 - in regular expressions, 143, 153, 161
- | (pipe) character, 149, 393
- + (plus) character, 145, 161
- === (precisely equal to) operator, 19, 78, 112, 390
- !== (precisely not equal to) operator, 19
- ? (question mark) character, 18, 145, 153, 300
- % (remainder) operator, 14, 33, 285, 388, 399
- ' (single quotation mark) character, 14, 161, 214
- / (slash) character, 13, 35, 141–142, 153, 301, 349
- * (star) character, in regular expressions, 145
- (subtraction) operator, 13, 16, 18
- = (subtraction assignment) operator, 33
- ?: (ternary) operator, 18, 201
- _ (underscore) character, 26, 35, 148
- 200 (HTTP status code), 298, 346, 348
- 204 (HTTP status code), 351
- 2d (canvas context), 277
- 304 (HTTP status code), 358, 364, 370
- 400 (HTTP status code), 405
- 403 (HTTP status code), 349
- 404 (HTTP status code), 298, 349, 362–363
- 405 (HTTP status code), 301, 348

406 (HTTP status code), 402
500 (HTTP status code), 348

A

a (HTML tag), 212, 225, 227, 306, 330
absolute positioning, 230, 233, 241,
244, 250

absolute values, 74, 404

abstract data types, 95, 98

abstraction, 5, 39, 84, 220, 302, 335
in Egg, 195
with higher-order functions, 82
of the network, 210

abstract syntax tree, 196–199, 218–219

acceleration, 269

Accept header, 315, 402

access control, 139, 359

Access-Control-Allow-Origin header, 302

action, 320–323

activeElement property, 305

actor, 257, 262, 266–267

addEntry function, 64

addEventListener method, 236, 269, 347

addition, 13, 111

add method, 112

address, 75, 297

address bar, 211, 297, 299

ages example, 102–103

alert function, 213–214

algorithmic optimization, 380–381

alpha component, 332

alphanumeric characters, 143

alt attribute, 223

ALT key, 240

altKey property, 240

ambiguity, 206

analysis, 127, 131

ancestor elements, 263

Android, 240

angle, 231–232, 281–282, 401

angle brackets, 212

animation, 244, 249, 254, 293

bouncing ball, 295

platform game, 265, 269–270,
284–285, 291, 400

spinning cat, 230–232, 233

appendChild method, 222, 398

Apple, 215

applications, 1, 319, 356

arc method, 281–282, 401

arguments, 27, 45, 49, 72, 196

argv property, 341

arithmetic, 13, 18, 202

Array constructor, 322

Array.from function, 341

array-like objects, 104, 220, 222, 243,
306, 311, 344

Array prototype, 98, 102

arrays, 58

counting, 91

creation, 58, 89, 322, 389, 392

in Egg, 205

filtering, 86

flattening, 92

indexing, 58–59, 66, 70, 389, 399

iteration, 67, 82, 84

length, 59

as matrices, 256

methods, 69, 77, 84, 86, 87, 89,
92, 93

notation, 76

random element, 120

as regular expression match, 146

representation, 75

of rest arguments, 72

searching, 66, 69

as tables, 65

arrays in Egg (exercise), 205, 397

arrow functions, 43, 97, 190

arrow keys, 249

artificial intelligence, 115

artificial life, 254, 315

assert function, 138

assertions, 138

assignment, 24, 33, 158, 206, 398

assumptions, 136, 138

async function, 184–185, 190–192,
369, 395

asynchronous programming, 176–178,
184, 189–190, 271, 396

in Node.js, 339, 344, 346, 350, 353
reading files, 311

attributes, 212, 220, 224, 306, 322, 403

autofocus attribute, 305

automatic semicolon insertion, 24

automation, 123, 130

- automaton, 115
- avatar, 254
- average function, 88
- await keyword, 184–186, 191, 192, 395–396
- axis, 268, 277, 285–286, 405
- B**
- background, 254, 261, 266
- background (CSS), 250, 254, 262
- backtracking, 150, 153
- ball, 295, 401
- balloon (exercise), 249, 399
- banking example, 134
- baseControls constant, 333
- baseTools constant, 333
- bean counting (exercise), 53, 389
- beforeunload event, 246
- behavior, 160
- benchmark, 226, 373, 379
- best practices, 3
- bezierCurveTo method, 280
- big ball of mud, 163
- binary data, 3, 11, 344
- binary numbers, 11–12, 65, 131, 150, 310
- binary operators, 13, 16, 23
- binding, 4, 31, 61, 384
 - assignment, 24–25, 42
 - compilation of, 398
 - definition, 24, 39–40, 42, 204, 206
 - destructuring, 74
 - in Egg, 201, 202
 - global, 40, 128, 164, 272, 340–341
 - local, 40
 - model of, 25, 61, 63
 - naming, 26, 35, 50, 73, 129
 - from parameter, 40, 47
 - scope of, 40
 - as state, 31, 32, 63, 312
 - undefined, 136–137
 - visibility, 41
- bitfield, 242
- bitmap graphics, 283, 295
- bits, 3, 11, 12, 16, 65
- block comments, 36, 153
- block elements, 225, 227
- blocking, 176, 231, 248, 345

- blocks, 29, 32, 40–41, 43, 61, 134–135, 196
- blur event, 245, 246
- blur method, 305
- body (HTML tag), 212, 213, 218
- body (HTTP), 299–301, 346, 351, 352
- body property, 218, 219, 221, 301
- Boolean, 16, 30, 63, 142, 202
 - conversion to, 19, 28, 31
- Boolean function, 28
- border (CSS), 225, 227
- border-radius (CSS), 241
- bouncing, 255, 258, 265, 267, 295
- boundary, 148, 158, 160, 290, 393
- bounds of ranges, 86
- boxes, 139, 217, 254, 295, 401
- box shadow (CSS), 263
- br (HTML tag), 325, 403
- braces
 - block, 5, 29, 388
 - body, 83
 - class, 99
 - function body, 40, 43
 - object, 60–61, 64, 75
 - in regular expressions, 145
- branching, 150–151
- branching recursion, 48, 287
- break keyword, 33–34
- breakpoint, 132
- browsers, 170, 177, 209, 211–215, 254, 297, 299, 307, 330, 334, 374
 - environment, 26, 297
 - security, 302, 356
 - storage, 312, 314
 - window, 236
- browser wars, 215
- bubbling, 237, 238, 245–246
- Buffer class, 344, 346–347
- bugs, 81, 127, 131, 154, 160, 166, 215
- building Promise.all (exercise), 192, 397
- bundlers, 170
- button (HTML tag), 214, 236, 240, 250, 307, 313, 315, 323
- button property, 237, 242, 324
- buttons, 235, 299, 306, 316
- buttons property, 242, 324

C

- C (programming language), 373
- callback function, 176–178, 180, 235, 270, 323, 344, 346, 364, 369
- call method, 96, 102
- call protocol, 383
- call stack, 44, 46, 49, 60, 133–134, 136, 189, 383
- camel case, 35, 228
- cancelAnimationFrame function, 247
- canvas, 254, 275, 277–280, 283–288, 293–294, 377, 402
 - context, 277
 - path, 278
 - size, 276, 278
- canvas (HTML tag), 276, 320, 323, 330–331, 335, 404
- CanvasDisplay class, 288, 289, 291
- capitalization, 35, 100, 146, 228, 233, 347
- capture group, 146–148, 152, 360
- Carla the crow, 181–182, 186, 191
- carriage return, 157
- Cascading Style Sheets (CSS), 227–229, 260–264, 275, 278, 322, 365
- case conversion, 60
- case keyword, 34
- case sensitivity, 146, 394
- casual computing, 2
- catch keyword, 133–134, 136–137, 139, 189, 393
- catch method, 180
- cat’s hat (exercise), 233
- centering
 - image on screen, 231
 - player in viewport, 263–264
- certificate, 303
- change event, 305, 308, 326, 403, 406
- character category, 144
- characterCount function, 87–88
- character encoding, 344
- characters, 14–15, 90, 308
- characterScript function, 92–93, 391
- charCodeAt method, 90
- checkboxes, 304, 309, 316, 403
- checked attribute, 304, 309
- chessboard (exercise), 37, 388
- child nodes, 219–224
 - direct, 229
- childNodes property, 220–221, 223, 399
- children property, 221
- Chinese characters, 90, 92
- choice, 149
- Chrome, 215, 383
- circle, 231, 281
- circle (SVG tag), 276
- circles (exercise), 336, 404
- circular dependency, 173, 395
- class attribute, 222, 225, 228, 260, 262
- class declaration, 99–101
- classes, 96, 98–99, 111, 117, 255, 321
- class hierarchy, 110
- className property, 225
- clearing the canvas, 275, 284, 289–290, 402
- clearInterval function, 247
- clearRect method, 284, 402
- clearTimeout function, 247–248
- click event, 236–237, 241, 243, 322, 402
- client, 210, 302, 356, 365–366
- clientHeight property, 225
- clientWidth property, 225
- clientX property, 241, 243, 324–325
- clientY property, 241, 243, 324–325
- clipping, 290
- closePath method, 279
- closing tag, 212, 214
- closure, 46, 206, 397
- closure in Egg (exercise), 206, 397
- code, 6, 153, 254
 - reading and writing, 6, 115
 - structure of, 23, 31, 39, 163, 170
- code golf (exercise), 160
- codePointAt method, 90
- code unit, 90
- coin, 254–255, 268, 292
- Coin class, 259, 268
- collaboration, 209
- collection, 5, 58, 60–62, 77, 108
- collision detection, 265, 267–269, 400–401
- color, 276, 278, 290, 320, 332
- color (CSS), 227
- color code, 322

- color component, 322
- color field, 320, 322, 327
- color picker, 320, 327, 329
- color property, 321
- ColorSelect class, 327
- COMMAND key, 240, 335
- command line, 166, 339–341, 353
- comment field reset (exercise), 371, 406
- COMMENT_NODE code, 219
- comments, 35, 76, 153, 156, 206, 219, 356, 358, 363, 367
- comments in Egg (exercise), 206, 397
- CommonJS modules, 167, 168, 173, 341
- communication, 210, 302
- compareRobots function, 123
- comparisons, 16, 19, 202, 389
 - deep, 64, 78, 390
 - of NaN, 17
 - of numbers, 16, 27
 - of objects, 64
 - of strings, 16
 - of undefined values, 19
- compatibility, 5, 128, 209, 215, 335, 343
- compilation, 170, 204, 373–374, 380, 384, 397
- complexity, 3, 81, 110, 151, 229, 335
- components, 320, 321, 325, 333
- composability, 5, 88, 171
- computed properties, 59, 314
- computers, 1–2
- concatenation, 15, 70, 398
- concat method, 70, 92, 392, 398
- conditional execution, 18, 28, 34, 37, 201
- conditional operator, 18, 20, 201
- conditional request, 358
- configuration, 156
- connected graph, 123
- connection, 210, 297, 303, 356–357
- consistency, 35, 209, 220
- console.log function, 5, 8, 16, 27, 44, 46, 52, 131, 340, 380
- constants, 26, 73, 269
- const keyword, 26, 41, 63, 73, 74
- constructors, 35, 99–100, 110, 124, 128, 134, 147, 154, 391, 393
- Content-Length header, 299
- content negotiation (exercise), 315, 402
- Content-Type header, 299, 346, 348–349, 353
- context, 277
- context menu, 239
- continuation, 177
- continue keyword, 33
- control flow, 28, 84
 - asynchronous, 176, 185
 - conditional, 28
 - exceptions, 133–134
 - functions, 44
 - loop, 30–32
- CONTROL key, 240, 335
- control point, 280–281
- controls, 325, 327, 330, 333
- Conway’s Game of Life, 315–316
- coordinates, 111, 231, 241, 261, 264, 266, 277–278, 281, 285–286
- copy-paste programming, 51, 166
- copyright, 166
- correlation, 64–65, 67–68
- cosine, 73, 231
- countBy function, 91, 93
- counter variable, 31–32, 232, 388, 389, 397, 399
- CPU, 176
- crash, 136–137, 362, 371
- createElement method, 223, 322, 398
- createReadStream function, 347, 350
- createServer function, 345–348, 359–361
- createTextNode method, 223, 402
- createWriteStream function, 346, 351
- cross-domain requests, 302
- cryptography, 303
- CSS (Cascading Style Sheets), 227–229, 260–264, 275, 278, 322, 365
- ctrlKey property, 240, 335
- curl program, 352
- curly braces. *See* braces
- cursor, 308
- curve, 279–281
- cwd function, 349
- cycle, 173, 219

D

- Dark Blue (game), 253
- data, 2, 11, 57
- data attribute, 224, 250
- data event, 347
- data flow, 320, 335
- data format, 75, 219
- data loss, 371
- dataset, 66, 85
- data structures, 57, 171–172, 217, 316
 - collection, 58
 - immutable, 118
 - list, 77, 108
 - map, 102
 - stack, 60
 - tree, 196, 219, 293
- data URL, 330–331
- date, 143, 145, 147
- Date class, 147–148, 164, 168, 192, 395
- date-names package, 168
- Date.now function, 148, 192, 332, 386
- dblclick event, 241
- debouncing, 248
- debugger statement, 132
- debugging, 6, 127–129, 131, 134, 137–138, 160
- decentralization, 209
- decimal numbers, 11, 131, 150
- declaration (HTML), 227
- decodeURIComponent function, 300, 348, 360
- deep comparison, 64, 78, 390
 - exercise, 78, 390
- default behavior, 227, 239
- default keyword, 34
- default value, 20, 45, 278, 314, 334
- defineProperty function, 391
- degree, 281, 286
- DELETE method, 298–299, 301, 347, 350, 362
- delete operator, 61
- De Morgan’s laws, 391
- deoptimization, 384–385
- dependency, 164–167, 173, 214, 343
- deserialization, 76
- design, 172
- destructuring, 148
 - assignment, 405
 - binding, 74, 168, 334, 395
- developer tools, 27, 132, 136, 381
- dialect (JavaScript), 170
- dialog, 26–27
- diamond, 294, 400
- digit, 11–12, 131, 142–144, 145–147, 322
- Dijkstra, Edsger, 171
- dijkstrajs package, 171, 394
- Dijkstra’s algorithm, 171
- dimensions, 111, 225, 253, 255, 265, 277, 388
- direction (writing), 93
- directory, 341–344, 347–350, 353, 406
- directory creation (exercise), 353, 405
- disabled attribute, 306
- discretization, 254, 265, 270
- dispatch, 34, 320–322, 325, 333, 359, 403
- display, 260, 270–271, 288, 293–294
- display (CSS), 227, 250
- distance, 404
- distribution, 374
- division, 13–14, 378, 407
 - by zero, 14
- doctype, 212–213
- documentation, 339
- documentElement property, 218
- document format, 302, 315
- documents, 211, 217, 246, 275
- do loop, 31, 120
- DOM (Document Object Model), 218
 - attributes, 224
 - components, 320
 - construction, 220–223, 322
 - events, 236, 240
 - fields, 303, 307
 - graphics, 254, 260–263, 275–276, 293
 - interface, 219
 - modification, 222
 - querying, 221, 229
 - tree, 218
- domain, 211, 302, 312
- domain-specific language, 82, 130, 141, 205, 229
- DOMDisplay class, 260–261, 288
- dominant direction (exercise), 93, 391
- dom property, 321
- doneAt property, 332

- done property, 332
- double click, 241
- download, 7, 166, 330, 342, 352, 355, 371
- download attribute, 330
- draggable bar example, 242
- dragging, 242, 320, 328, 336
- draw function, 327, 336
- drawGraph function, 377
- drawImage method, 283–285, 288, 291, 402
- drawing, 217, 225–226, 231, 276–279, 287, 291–292, 319–320, 403
- drawing program example, 241, 319
- drawPicture function, 323, 330, 335, 403
- drop-down menus, 305, 310
- duplication, 166
- dynamic typing, 384

E

- ECMAScript, 5–6, 128
- editor (code), 32
- efficiency, 47, 77, 89, 204, 226, 254, 262, 277, 323, 335
- efficient drawing (exercise), 335, 403
- Egg language, 195–206, 219
- elegance, 48, 197
- ELEMENT_NODE code, 219, 398
- elements, 212, 219, 221, 223
- elements property, 306
- ellipse, 230, 232
- else keyword, 29
- elt function, 223, 322, 335, 367
- email, 303
- emoji, 15, 90, 158, 249
- empty set, 153
- encapsulation, 96, 103, 110, 236, 260
- encodeURIComponent function, 300, 357, 367
- encoding, 210
- encryption, 303
- end event, 347
- end method, 346, 348
- enemies example, 156
- engine, 373, 380, 382
- ENOENT (status code), 350
- ENTER key, 307
- entity, 212

- environment, 26, 200
- equality, 17
- error, 90, 127–129, 131–133, 136–137, 180
 - handling, 127, 132–133, 136, 344, 348, 350, 367, 370
 - message, 199, 315
 - recovery, 132
 - response, 298, 348, 351
 - tolerance, 213
- error event, 312, 351
- Error type, 134, 136–137, 350
- escaping
 - in HTML, 213–214
 - in regular expressions, 142–143, 154
 - in strings, 14, 196
 - in URLs, 300, 348, 357, 360
- ESC key, 272
- ES modules, 164–165, 214, 341–342
- Etag header, 358, 364, 370
- eval operator, 169
- evaluate function, 199–202
- evaluation, 199, 204
- even numbers, 30, 53
- event handling, 235–250, 254, 269–271, 272, 283, 293, 307–308, 322, 346, 400, 403
- event loop, 189
- event object, 237, 241, 243
- event propagation, 237–238, 245, 246
- event type, 237
- every method, 93
- everything (exercise), 93, 391
- evolution, 141, 335, 343
- exception handling, 133–137, 139, 180–181, 184, 189, 402
- exception safety, 136
- exec method, 146–147, 155–156
- execution order, 28, 43–44
- exercises, 2, 6, 7, 36, 131
- exit method, 340
- expected behavior, 239
- experimentation, 3, 7, 160
- exponent, 13, 161, 393–394
- exponentiation, 31, 33
- export keyword, 342
- exports object, 167, 169, 395

expression, 23–24, 27, 30–32, 42,
195–196, 200
expressivity, 205
extraction, 147

F

factorial function, 8
Fahrenheit, 106
false, 16–17
farm example, 50, 52, 149
fetch function, 300, 315, 346, 366,
369, 406
field, 240, 299, 303, 306, 309, 312, 316,
320, 371
file, 311, 341, 350, 406
 access, 169–170, 331, 344
 image, 319, 330
 resource, 298–299, 347, 349
 stream, 346
file extension, 349
file field, 304, 311
file format, 156
FileReader class, 311–312, 331
file reading, 311
file server, 365
 example, 347–353, 405–406
file size, 170
fileSizes function, 190–191
files property, 311
filesystem, 311, 344, 347, 349, 406
File type, 311
fill function, 329
filling, 277–279, 283, 294
fill method, 279, 322
fillRect method, 277, 284
fillStyle property, 278, 283, 322
fillText method, 283, 401
filter method, 86, 88, 92, 118, 391–392
finally keyword, 135, 139, 393
findInStorage function, 184
find method, 91
findRoute function, 122
finish event, 351–352
Firefox, 215, 383
firstChild property, 220
fixed positioning, 244
fixing scope (exercise), 206, 398
FizzBuzz (exercise), 37, 387

flattening (exercise), 92
flexibility, 6
flipHorizontally function, 286,
291, 400
flipping, 285–286
floating-point number, 13
flood fill, 325, 328
flow diagram, 150
focus, 240, 245, 305–306, 308–309,
335, 371
focus event, 245–246
focus method, 305
fold. *See* reduce method
font, 283
font-family (CSS), 228
font-size (CSS), 249
font-weight (CSS), 228
for attribute, 309
force, 375–376, 377–378, 380
force-directed graph layout, 375
forceDirected_noRepeat function, 380
forceDirected_noVector function, 382
forceDirected_simple function, 378
forceDirected_skip function, 380
forceSize function, 378
forEach method, 84, 104
for loop, 32–33, 67, 83, 93, 137, 380,
389–390
form, 299–300, 306–307, 353
form (HTML tag), 303–304, 306,
368, 406
formatDate module, 168
form property, 306
for/of loop, 67, 90, 106–107, 109, 392
fractal example, 287
fractional numbers, 13, 161, 254
frame, 284–285, 291, 402
framework, 52, 320
frequency tables, 64–65
function, 26, 39–54, 128, 203
 application, 26–27, 40, 44–45, 47,
72, 86, 136, 195–196, 200
 body, 40, 43
 callback. *See* callback function
 declaration, 42
 definition, 39–40, 42, 50
 higher-order, 42, 83–84, 86–88,
152, 270

- inlining, 383
- model of, 47
- naming, 50–51
- as property, 60
- purity, 52
- scope, 42, 167, 206
- as value, 40–42, 46, 83–84, 86, 237, 270, 400

Function constructor, 169, 202, 204, 315, 402

function keyword, 40, 42

Function prototype, 98, 102

G

game, 253–255, 269, 271, 272, 288

- with canvas, 293
- screenshot, 264, 293

GAME_LEVELS dataset, 271

game of life (exercise), 315, 403

garbage collection, 12, 382

garble example, 341

generation, 315–316, 403

generators, 185, 386, 407

getAttribute method, 224–225

getBoundingClientRect method, 226, 324

getContext method, 277

getDate method, 148

getElementById method, 222, 398

getElementsByClassName method, 222

getElementsByTagName method, 221, 223, 233, 398

getFullYear method, 148

getHours method, 148

getImageData method, 331–332

getItem method, 312, 314

GET method, 298–299, 301, 307, 347, 349, 357, 362

get method, 103

getMinutes method, 148

getMonth method, 148

getPrototypeOf function, 98, 100, 206, 398

getSeconds method, 148

getters, 105, 111, 258

getTime method, 147

getYear method, 148

GitHub, 298

global object, 128

global scope, 40, 164, 167, 202, 247, 341, 398

goalOrientedRobot function, 123

Google, 215

grammar, 23, 127, 157

Graph class, 376

graphics, 254, 260, 262, 275–277, 283, 293–294

graph layout, 375–376, 380

graphs, 116, 122, 171, 294, 374, 376

gravity, 269

greed, 153

grep, 353

grid, 254, 261, 266, 315, 403

groupBy function, 93

Group class, 112, 124, 185, 391–392

grouping, 13, 29, 146, 152, 393

groups (exercise), 112, 391–392

H

h1 (HTML tag), 212, 225

handleAction function, 366

hardcoding, 221, 295

hard disk, 171, 175

hard drive, 12, 311, 313, 339, 371

has method, 103, 112

hasOwn function, 104, 206, 362, 398

head (HTML tag), 212–213, 218

headers, 299, 302, 346, 357

headers property, 301, 315

head property, 218

height property, 336, 404

help text example, 245

hexadecimal numbers, 150, 300, 322, 332

hidden elements, 227, 250

higher-order function. *See* function: higher-order

history, 335

historyUpdateState function, 332

Hooke's law, 377

hot code, 374

href attribute, 212, 221, 224

HTML (HyperText Markup Language), 211, 217, 298, 312, 353

- notation, 212
- structure, 217, 219

html (HTML tag), 213, 218
HTTP (HyperText Transfer Protocol),
209–211, 297–300, 302–303,
345, 351, 353, 357
 client, 346, 352, 356
 server, 345, 347, 365
HTTPS, 211, 303
human language, 23

I

id attribute, 222, 228, 309
idempotence, 351, 405
identifier, 196
identity, 63
if keyword, 28, 158
 chaining, 29, 34, 388
If-None-Match header, 358, 364, 369
images, 223, 246, 275, 299
imagination, 253
IME (input method editor), 240
img (HTML tag), 212, 223, 227, 246,
275, 283, 284, 331
immediately invoked function
 expression, 167
immutable values, 63, 118, 258, 321–322,
328, 332, 403
import keyword, 165, 341, 359
includes method, 66–67, 391
indentation, 32
index, 59
 index.html, 365
indexOf method, 69–70, 112, 142,
154, 391
index property, 146
infinite loop, 33, 44, 137, 389
infinity, 14
infrastructure, 166
inheritance, 99, 109–111, 137, 350
INI file, 156
ini package, 166–167, 170, 342
initialization, 246
inline caching, 384–385
inline element, 225, 227
inlining, 383
inner function, 41, 398
innerHTML property, 245
inner loop, 151
innerWidth property, 245

in operator, 62, 103
input, 132, 235, 254, 305, 339, 362
input (HTML tag), 245, 303, 307, 309,
311, 327, 331
input event, 308
insertBefore method, 222
installation, 166
instance, 98, 100
instanceof operator, 110, 137
instruction, 3, 373
integers, 13
integration, 141, 219
interface, 100
 canvas, 275, 277
 design, 52, 141, 147, 155,
219–220, 278
 HTTP, 302, 357
 module, 164, 167, 169–170,
300, 342
 object, 96, 103–105, 112, 124, 257,
288, 307, 321
internationalization, 144
internet, 156, 210–211, 214
Internet Explorer, 215
interpolation, 15
interpretation, 6, 199–200, 204, 373
interview question, 37
inversion, 143
IP address, 211, 297
isDirectory method, 350, 405
isEven (exercise), 53, 388
isolation, 96, 163, 167, 214
iterator, 185
iterator interface, 106–107, 112

J

Jacques, the weresquirrel, 58
Java, 5
JavaScript, 5
 availability, 2
 flexibility, 6
 future, 26
 history, 5, 209
 in HTML, 213
 syntax, 23
 uses, 6
 versions, 6
 weaknesses, 6

- JavaScript console, 7, 16, 27, 132, 136, 315, 340
- JavaScript Object Notation. *See* JSON
- join method, 92, 102, 341
- JOURNAL dataset, 66–67
- journalEvents function, 67
- JSON (JavaScript Object Notation), 76, 170, 301, 314, 357, 362, 370, 406
- json method, 301
- JSON.parse function, 76, 406
- JSON.stringify function, 76
- JSX, 368
- jumping, 254, 269

K

- keyboard, 26, 235, 239, 254, 269, 272, 305–306, 308, 335
- keyboard bindings (exercise), 335, 403
- keyboard focus. *See* focus
- key code, 269
- keydown event, 239, 248, 270, 335, 399, 403
- key property, 240, 399, 403
- keyup event, 239, 270
- keywords, 24, 26, 225
- Khasekhemwy, 308
- kill process, 346

L

- label, 283, 295
- label (HTML tag), 309, 326
- labeling, 309
- landscape example, 41
- lastChild property, 220
- lastIndexOf method, 69
- lastIndex property, 155, 156
- Last-Modified header, 299
- latency, 170
- lava, 254–255, 263, 265–268, 292
- Lava class, 258, 267
- layering, 210
- layout, 225–227
- leaf nodes, 219
- leak, 214, 272
- learning, 2, 6
- left (CSS), 230–233
- LEGO, 164
- LengthList class, 109

- length property
 - for array, 59, 322
 - for string, 50, 53, 59, 71, 389
- let keyword, 24–25, 41, 63, 73–74, 128
- level, 254–255, 260–262, 271
- Level class, 255
- lexical scoping, 41–42
- library, 220, 320, 342
- license, 167
- line breaks, 14, 157
- line comments, 35, 153
- line drawing, 336, 404
- lines (code), 24, 31, 157, 203, 275, 277–282, 295, 401
- lineTo method, 279
- line width, 278, 285
- linewidth property, 278
- link (HTML tag), 264
- linked lists, 77, 108, 109, 390
- links, 212, 220–221, 239–240, 330
- list (exercise), 77, 390
- listening (TCP), 210, 345
- listen method, 345–346
- ListIterator class, 108
- literal expressions, 23, 141, 198, 200
- live data structure, 217, 223, 230, 399
- lives (exercise), 272
- live view, 356–357, 370, 406
- LoadButton class, 330
- load event, 246, 283, 291, 312, 402
- local binding, 46, 206, 389
- localhost, 345
- local scope, 40, 204
- localStorage object, 312–313, 366
- locked box (exercise), 139, 393
- logging, 131
- logical operators, 17
- long polling, 356–358, 362, 364, 369
- look-ahead tests, 149
- loop, 4, 30, 32, 36–37, 47, 67, 88–89, 156, 388–389, 401
 - body, 31, 83
 - termination, 33
- lycanthropy, 58, 64

M

- machine code, 3, 204, 373
- mailRoute array, 121

- malicious scripts, 214
- man-in-the-middle attack, 303
- map, 260, 306
- map (data structure), 102
- Map class, 103, 105
- map method, 86, 88, 92, 102, 118, 256, 326
- matching, 142, 148, 155, 160
 - algorithm, 149–151
- match method, 146, 156
- Math.abs function, 74, 404
- Math.acos function, 73
- Math.asin function, 73
- Math.atan function, 73
- Math.ceil function, 74, 266, 290
- Math.cos function, 73, 231–232, 401
- mathematics, 47, 84
- Math.floor function, 74, 120, 266, 290
- Math.max function, 27–28, 59, 72–73, 290
- Math.min function, 27, 53, 73, 290
- Math object, 53, 59, 73
- Math.PI constant, 73, 281
- Math.random function, 73, 120, 259, 316, 384
- Math.round function, 74
- Math.sin function, 73, 231–232, 259, 268
- Math.sqrt function, 66, 73, 391
- Math.tan function, 73
- max example, 72
- max-height (CSS), 263
- maximum, 27, 73, 87, 88
- max-width (CSS), 263
- Meadowfield, 115
- measuring, 381
- measuring a robot (exercise), 123, 392
- media type, 302, 315, 349
- memory, 3, 11, 382
 - call stack, 44
 - organization, 12, 24, 58, 63, 75
 - persistence, 371
 - speed, 175, 204
 - structure sharing, 78
- message event, 247
- meta key, 240
- metaKey property, 240, 335
- method attribute, 299
- method call, 96
- method property, 301
- methods, 60, 96, 100, 128, 346, 383
 - array, 69
 - HTTP, 298, 302, 346, 352, 357, 359
 - private, 100
- methods object, 348
- micro-optimization, 380–381
- Microsoft, 215
- MIME type, 315, 349
- mime-types package, 349
- mini application, 312
- minifiers, 170
- minimalism, 253
- minimum, 27, 53, 73
- minimum (exercise), 53, 388
- mirroring, 285–286, 295, 402
- mixer example, 95
- MKCOL method, 353, 405
- mkdir function, 353, 405
- modification date, 350
- modifier key, 240
- modularity, 96, 320
- modular robot (exercise), 173, 394
- module loader, 341
- module object, 169
- modules, 164, 167, 173, 260, 341–342, 359
 - design, 170
 - resolution, 165
- module system, 167
- monster (exercise), 272, 400
- Mosaic, 215
- motion, 254
- mouse, 26
- mouse button, 237–238, 241
- mousedown event, 238, 241, 243, 323–324, 402
- mousemove event, 242, 248, 250, 324, 336, 399
- mouse trail (exercise), 249, 399
- mouseup event, 241–243
- moveTo method, 279, 281
- Mozilla, 215
- multiple attribute, 310–311
- multiple-choice, 304–305, 309–310
- multiplication, 13, 257, 267
- multiplier function, 46
- mutability, 61, 63, 118

N

- name attribute, 306, 309
- namespace, 73
- naming, 4–5, 26
- NaN (not a number), 14, 17, 19, 127
- negation, 16–17
- neighbor, 315, 403
- nesting
 - arrays, 65
 - expressions, 23, 197
 - functions, 41
 - loops, 37, 388
 - objects, 218, 221
 - in regular expressions, 151
 - scope, 41
- Netscape, 5, 215
- network, 176, 209–210, 356
 - abstraction, 302
 - protocol, 210
 - security, 303
 - speed, 170, 175, 339
- newline character, 14, 37, 143, 153, 157, 256, 402
- new operator, 99
- next method, 107, 185, 392
- nextSibling property, 220
- node, 218–219
- node:fs package, 344, 345
- node:fs/promises package, 345
- node:http package, 345, 346
- Node.js, 6, 7, 27, 167, 177, 339–342, 344–352, 356–357, 359, 371
- NodeList type, 220, 229
- node_modules* directory, 341–342
- nodeName property, 233
- node:path package, 349
- node program, 340
- node:stream/consumers package, 362
- nodeType property, 219, 398–399
- nodeValue property, 221
- nonbreaking spaces, 144
- normalizing, 378
- not a number (NaN), 14
- note-taking example, 313
- notifications, 356
- NPM, 166–167, 171, 173, 341–342, 344, 349, 359–360, 371, 394

- npm program, 342–343, 349
- null, 18–20, 49, 59, 75, 78, 132
- number, 12, 63, 142, 161, 393
 - conversion to, 19, 28
 - notation, 12, 13
 - precision, 13
 - representation, 12
 - special values, 14
- Number function, 28, 35, 144
- Number.isNaN function, 29
- number puzzle example, 48

O

- object, 57, 61–62, 110, 384
 - creation, 76, 99, 314
 - identity, 63
 - as map, 259
 - as module, 167
 - mutability, 63
 - property, 27, 59, 73, 75, 97
 - representation, 75
- Object.create function, 98, 103, 203
- Object.keys function, 62, 78, 390, 398
- object-oriented programming, 95, 98, 104, 109, 117, 171
- Object prototype, 97, 98
- Object.prototype, 103
- object shape, 384
- obstacles, 265, 268
- offsetHeight property, 225–226
- offsetWidth property, 225
- onclick attribute, 214, 236
- onclick property, 322
- on method, 347
- OpenGL, 277
- opening tags, 212
- operators, 13, 15, 16, 19, 196, 202, 385
- optimization, 47, 52, 226, 248, 254, 262, 293–295, 345, 373–374, 380–386, 407
- option (HTML tag), 305, 310, 406
- optional arguments, 45, 77
- optional chaining, 75
- optional in pattern, 145
- options property, 310
- ordering, 210
- ordinal package, 168
- organization, 163

- outline, 278
- output, 16, 26–27, 131–132, 202, 339, 402
- overflow (CSS), 263
- overflow of numbers, 12
- overlap, 266
- overriding, 101, 104, 109, 395
- overwriting, 351, 353, 362

P

- p (HTML tag), 212, 225
- package, 166, 168, 343
 - package.json*, 342, 343
- package managers, 166
- padding (CSS), 261
- page reload, 246, 307, 312
- pageXOffset property, 226
- pageX property, 241, 243
- pageYOffset property, 226, 245
- pageY property, 241, 243
- panning, 325
- paragraph, 212
- parallelism, 176, 299
- parameters, 27, 40–41, 43, 45, 72, 96,
129, 169
- parentheses, 13
 - arguments, 27, 40, 43, 83, 196
 - with expressions, 23
 - in regular expressions, 146,
148–149, 158, 393
 - with statements, 29, 30, 32
- parentNode property, 220
- parent nodes, 237
- parseApply function, 198
- parseExpression function, 197
- parse function, 199
- parseINI function, 158, 166
- parser generator, 205
- parsing, 76, 127, 158, 195–198, 200, 202,
213, 217, 348, 364
- password field, 304
- passwords, 303
- path
 - canvas, 278–281, 400
 - closing, 279
 - filesystem, 341, 347
 - URL, 298, 301, 348, 357, 359
- pathfinding, 121, 171, 329
- patterns, 141–143, 154

- pausing (exercise), 272, 400
- percentages, 92, 245
- performance, 151, 170, 176, 192, 204,
226, 254, 293, 345, 373,
375, 380–381
- persistence, 312, 356, 371, 406
- persistent data structure, 117, 118, 124,
130, 321, 328, 332, 400
- persistent group (exercise), 124
- persistent map (exercise), 392
- PGroup class, 124, 392
- phase, 259, 268
- phi coefficient, 64–66
- phi function, 66, 74
- physics, 265, 269, 375, 399
- pi, 13, 73, 231, 259, 281
- pick function, 329
- PI constant, 73, 231
- PictureCanvas class, 323, 335
- Picture class, 321, 331
- pictureFromImage function, 331
- picture property, 321
- pictures, 275–276, 284, 293, 320, 332
- pie chart example, 282–283, 295, 401
- pipe method, 348, 351
- pixel art, 284
- PixelEditor class, 325, 333, 335
- pixels, 225, 232, 261, 275–277, 283–284,
293, 319, 321, 323–324,
327–332, 336, 403
- platform game, 253, 272
- player, 253–255, 263, 265, 268, 271, 284,
291–292
- Player class, 258, 268
- pointer, 220
- pointer events, 238, 323
- pointerPosition function, 324
- polling, 235
- pollTalks function, 369
- polymorphism, 104
- pop method, 60, 69
- ports, 210, 297, 345–346
- position, 226
- position (CSS), 230, 233, 244, 254,
262, 263
- postMessage method, 247
- POST method, 299–300, 307, 358
- power example, 47

precedence, 13, 17, 228–229
 predicate function, 86, 89, 92–93
 Prefer header, 358, 364, 369
 premature optimization, 48
 preventDefault method, 239, 244–246,
 269, 307, 324, 403
 previousSibling property, 220
 prime numbers (exercise), 386, 407
 primitiveMultiply (exercise), 139, 393
 privacy, 214
 private property, 139
 process object, 340, 349
 processor, 175, 383
 profiling, 48, 381
 programming, 1
 difficulty, 2
 history, 3
 joy, 1, 2
 size of, 82, 160
 style, 3, 24, 31–32, 35
 programming language, 1, 3, 195, 219,
 340, 373
 power, 5
 programs, 2, 24, 28
 progress bar, 244
 Promise.all function, 188, 190, 192,
 396–397
 Promise class, 178–180, 184, 189–190,
 192, 300–301, 312, 345,
 348, 370, 395–397
 Promise.reject function, 180
 Promise.resolve function, 178
 promises, 192, 397
 promptDirection function, 136–137
 promptNumber function, 132
 propagation, 237–238, 245–246
 proper lines (exercise), 336, 404
 property, 314, 384
 access, 27, 59, 96, 127, 334
 assignment, 61
 definition, 60, 64, 105
 deletion, 61
 inheritance, 97, 98, 101
 model, 61
 naming, 103, 106–107
 private, 100
 public, 100
 testing for, 62
 protocol, 210–211, 297, 298
 prototype, 97–99, 102–103, 109, 203,
 206, 397
 diagram, 102
 prototype property, 99–100
 pseudorandom numbers, 73–74
 public, 101
 public space (exercise), 353, 406
 publishing packages, 343
 pure function, 51–52, 77, 86, 171,
 316, 403
 pushing data, 356
 push method, 60, 67, 69, 391
 PUT method, 298–299, 347, 351, 357,
 362, 406
 Pythagoras, 391
 Pythagorean theorem, 404
 Python, 373

Q

quadratic curve, 280
 quadraticCurveTo method, 279, 401
 querySelectorAll method, 229, 309
 querySelector method, 230, 398
 query string, 299–300, 358, 364
 queue, 189
 quiet times (exercise), 191, 395
 quoting
 in JSON, 76
 of object properties, 61
 quoting style (exercise), 161, 393

R

rabbit example, 96–99
 radian, 231, 281, 286
 radio buttons, 304, 309
 radius, 336, 404
 raising (exception), 133
 random-item package, 394
 random numbers, 73–74, 259
 randomPick function, 120
 randomRobot function, 120
 range, 85, 143, 145
 range function, 5, 77, 389
 Range header, 301
 ray tracer, 293
 readability, 4–5, 35, 47, 51, 133, 163,
 200, 264, 295

- readable stream, 346–348, 362
- readAsDataURL method, 331
- readAsText method, 312
- readdir function, 344, 350, 405
- readdirSync function, 405
- read-eval-print loop, 340
- readFile function, 168, 344, 406
- readFileSync function, 345, 405
- reading code, 6, 115
- readTextFile function, 177
- real promises (exercise), 192, 396
- real-time events, 235
- reasoning, 17
- record, 60
- rect (SVG tag), 276
- rectangle function, 328, 404
- rectangles, 254, 265, 277, 278, 294, 328
- recursion, 44, 47–48, 53, 78, 184, 197, 198, 200, 221, 233, 287, 388, 390, 393, 398
- reduce method, 87–88, 92, 326, 391
- redundancy, 380
- ReferenceError type, 206
- RegExp class, 141–142, 154, 405
- regexp golf (exercise), 160
- regular expressions, 141–143, 151, 153–156, 160, 198, 353, 359, 360, 397, 405
 - alternatives, 149
 - backtracking, 150
 - boundary, 148
 - creation, 141, 154
 - escaping, 142, 154, 393
 - flags, 146, 152, 154, 394
 - global, 152, 155, 156
 - grouping, 146, 152
 - internationalization, 144
 - matching, 149, 150, 155
 - methods, 142, 147, 154
 - repetition, 145
- rejecting (a promise), 180, 190, 192, 396
- relative paths, 214, 341, 347, 406
- relative positioning, 230–231
- relative URL, 301
- remote access, 347
- remote procedure call, 302
- removeChild method, 222
- removeEventListener method, 236, 400
- removeItem method, 312
- rename function, 344
- rendering, 277
- renderTalkForm function, 368
- renderTalk function, 367
- renderUserField function, 367
- repeating key, 239
- repeat method, 71, 244
- repetition, 50, 145, 151, 154, 247
- replaceChild method, 222, 399
- replace method, 151, 152, 161, 393
- replaceSelection function, 308
- reportError function, 367
- repulsion, 375, 378
- request, 210, 298–299, 307, 346, 352, 356
- requestAnimationFrame function, 231, 246–247, 270, 295, 399
- request function, 346
- require function, 167–168, 173, 341–342
- reserved words, 26
- resolve function, 349
- resolving a promise, 178, 180, 190
- resolving module names, 165, 341
- resource, 210–211, 298–299, 302, 347, 362
- response, 298, 299, 302, 346, 348, 351
- Response class, 301
- responsiveness, 235, 339, 374
- restore method, 287–288
- rest parameters, 72
- result property, 312
- return keyword, 40, 44, 184, 388, 391
- return value, 27, 40, 132, 390
- reuse, 52, 110, 163, 166, 341
- reverse method, 77
- reversing (exercise), 77, 389
- rgb (CSS), 262
- right-aligning, 233
- rmdir function, 350, 353
- roadGraph object, 116
- roads array, 115
- roads module (exercise), 173, 395
- robot, 115, 117, 119, 121, 123, 173
- robot efficiency (exercise), 124, 392
- robustness, 357

- root, 219
- rotate method, 285–286, 288
- rotation, 294, 400
- rounding, 74, 132, 266, 290, 405
- roundTo example, 40, 45
- router, 356, 359
- Router class, 359–360
- routeRobot function, 121
- row (HTML), 233
- Ruby, 373
- rule (CSS), 228–229
- runAnimation function, 270, 272
- run function, 202
- runGame function, 271, 272
- runLayout function, 379
- runLevel function, 270, 272
- running code, 7
- runRobot function, 119, 392
- runtime error, 130–132, 138, 398
- Rust, 373

S

- Safari, 215
- sandbox, 7, 57, 214, 217, 302
- SaveButton class, 330
- save method, 287–288
- scalar replacement of aggregates, 383
- scale constant, 323
- scale method, 285, 286
- scaling, 261, 284–285, 291, 402
- scheduling, 189, 396
- scientific notation, 13, 161
- scope, 40–42, 46, 169, 200, 202, 206, 397–398
- script (HTML tag), 213–214, 246
- SCRIPTS dataset, 85, 87, 89, 91, 93
- scroll event, 244, 248
- scrolling, 239, 244, 263, 269, 289
- search method, 154
- search problem, 122, 149–150, 221, 353
- search tool (exercise), 353, 405
- sections, 157
- Secure HTTP, 211, 303
- security, 169, 214, 302–303, 311–312, 349, 359
- select (HTML tag), 305, 310, 313, 320, 325–326, 406
- selected attributes, 310

- selection, 308
- selectionEnd property, 308
- selectionStart property, 308
- selectors, 229
- self-closing tags, 212
- semantic versioning, 343
- sep binding, 349
- sequence, 145
- serialization, 76
- server, 210–211, 297–298, 301–302, 339, 345–347, 356, 359
- serve-static package, 360
- session, 314
- sessionStorage object, 314
- set, 142–143, 219
- Set (data structure), 112, 124
- setAttribute method, 224–225, 322
- Set class, 112, 124, 392
- setInterval function, 247, 284
- setItem method, 312
- set method, 103
- setters, 105
- setTimeout function, 177, 189, 247–248, 364, 370, 379
- shape, 275, 279, 281, 283, 294
- shapes (exercise), 294, 400
- shared property, 98, 101–102
- SHIFT key, 240, 403
- shiftKey property, 240
- shift method, 69
- short-circuit evaluation, 20, 49, 201, 391
- side effects, 24, 27, 33, 40, 52, 77, 171, 220, 222, 278, 300, 320
- sign, 12–13, 161, 393
- simulation, 117, 119, 253, 258, 315, 375, 399
- sine, 73, 231, 259, 268
- SkillShareApp class, 370
- skill-sharing project, 355–357, 359, 365
- skipSpace function, 198, 206
- slice method, 70, 85, 223, 389, 397
- slope, 405
- smooth animation, 231, 254
- SMTP protocol, 210
- socket, 357
- some method, 89, 93
- sorting data, 219
- source property, 155

- special form, 195, 200
- specialForms object, 200
- special return values, 132–133
- specificity, 229
- speed, 2, 295, 401
- spiral, 295, 401
- split method, 71, 116, 256, 395
- spread, 72, 109, 314, 322
- spread (triple dot) operator, 261
- spring, 375, 377
- sprite, 284, 291
- square, 28
- square brackets, 58–59, 74, 107, 143, 310, 314, 389
- square example, 39–40, 42–43
- square root, 66, 73, 391
- src attribute, 212, 214
- stack. *See* call stack
- stack overflow, 44, 47, 53, 388
- stack trace, 134
- staged compilation, 374
- standard environment, 26
- standard output, 340
- standards, 5, 26, 35, 86, 134, 144, 209, 215, 341
- star, 295, 401
- startPixelEditor function, 333
- Star Trek*, 280
- startState constant, 333
- startsWith method, 348
- state, 32, 117, 383
 - of application, 262, 320, 323, 332, 371
 - in binding, 24, 31, 33
 - of canvas, 278, 287
 - iterator, 186
 - in objects, 117, 256, 289
 - persistence, 328
 - transitions, 190, 320, 322
- statements, 23–24, 28, 30, 32, 40, 61
- stat function, 344, 349–350, 405
- static files, 357, 360
- static methods, 106, 112, 256
- static property, 106, 393
- static typing, 384
- Stats type, 350
- statSync function, 405
- status codes, 298, 340
- status property, 301, 366
- stopPropagation method, 237
- stream, 210, 346–348, 351
- strict mode, 128
- String function, 28, 104
- strings, 14, 58, 60, 63, 90
 - indexing, 53, 70, 71, 90, 146
 - length, 37, 90
 - methods, 70, 146
 - notation, 14
 - properties, 70
 - representation, 15
 - searching, 70
- stroke method, 279, 280
- strokeRect method, 277, 401
- strokeStyle property, 278
- strokeText method, 283
- stroking, 277, 278, 283, 294
- strong (HTML tag), 225, 227
- structure, 166, 212, 217, 320
- structure sharing, 78
- style, 227
- style (HTML tag), 228
- style attribute, 227, 228, 260
- style sheet. *See* CSS
- subclass, 110
- submit event, 304–307, 368, 406
- substitution, 52
- subtraction, 13, 111
- sum function, 5, 77
- summing (exercise), 77, 389
- summing example, 4, 81, 87, 203
- superclass, 110
- survey, 282
- SVG, 275–277, 293–294
- swapping bindings, 404
- swipe, 328
- switch keyword, 34
- Symbol function, 106
- Symbol.iterator symbol, 107
- symbols, 106
- synchronization, 370, 406
- synchronous programming, 176, 184, 345, 353
- syncState method, 321, 323, 326–327, 335, 406
- syntax
 - in Egg, 195–196
 - error, 26, 127, 128

- expression, 23
- function, 40, 42
- identifier, 26
- number, 12, 161
- object, 60
- operator, 13
- statement, 24, 28, 30, 32, 34, 133
- string, 14

SyntaxError type, 198

syntax tree, 196–199, 218–219

T

tabbed interface (exercise), 250, 399

tab characters, 14, 32

tabindex attribute, 240, 306, 335

TAB key, 306

table (HTML tag), 232, 254, 261, 403

table example, 398

tableFor function, 66

tables, 65, 66, 261, 262

tags, 211–212, 217, 228

talk, 355–356, 362–363

talkResponse method, 364

talksAbout function, 221

talkURL function, 367

Tamil language, 85

tampering, 303

tangent, 73

target property, 238

task management example, 69

TCP (Transmission Control Protocol), 210–211, 297, 357

td (HTML tag), 233, 261

temperature example, 105

template, 168, 371, 406

template literals, 15

templating language, 368

terminal, 340

testing, 123, 130

test method, 142

test runners, 130

test suites, 130

text, 14, 211–212, 217, 219, 283, 293–295, 308, 310, 344, 402

text-align (CSS), 233

textAlign property, 283, 401

textarea (HTML tag), 248, 304, 307–308, 313, 315, 406

textBaseline property, 283, 401

textContent property, 399, 402

text fields, 245, 304–308

textField function, 178, 181, 192, 395–396

text method, 301

TEXT_NODE code, 219, 399

text nodes, 219, 221, 223, 399

textScripts function, 91, 391

text wrapping, 293

th (HTML tag), 233

then method, 178–180, 396, 397

this binding, 60, 96–97, 99, 128

threads, 176–177, 189, 246–247

throw keyword, 133–134, 137, 139, 393

tiles, 291

time, 143, 145, 147, 231, 248, 265, 268, 270, 291, 332

timeline, 176, 189, 213, 231, 235, 246

timeout, 247, 357–358, 364

times method, 257

timestamp, 147, 192

time zone, 147

timing, 379

title (HTML tag), 212–213, 365

toDataURL method, 330

toLowerCase method, 60, 233

tool property, 321

tools, 141, 160, 170, 320, 325–329, 333, 336, 343

ToolSelect class, 326

top (CSS), 230–233

top-level scope. *See* global scope

toString method, 97, 102–104, 332, 347

touch, 243, 320

touchend event, 243

touches method, 265

touches property, 243, 325

touchmove event, 243, 324, 336

touchstart event, 243, 323, 324

toUpperCase method, 60, 130, 233, 347

tr (HTML tag), 233, 261

trackKeys function, 269, 272

transform (CSS), 275

transformation, 285–287, 295, 400

translate method, 285, 286

Transmission Control Protocol (TCP), 210, 211, 297, 357

transparency, 277, 284, 332

- transpilation, 204
- trapezoid, 294, 400
- traversal, 150
- treeGraph function, 377
- trees, 196, 219
- trial and error, 131, 281
- triangle (exercise), 36, 387
- trigonometry, 73, 231
- trim method, 71, 256
- true, 16
- trust, 214
- try keyword, 134–135, 393, 402
- type, 12, 15, 95, 110
- type attribute, 303, 307
- type checking, 129, 170
- type coercion, 18–19, 28
- type observation, 374, 384
- typeof operator, 15–16, 78, 390
- type property, 196, 237
- TypeScript, 129
- type variable, 129
- typing, 248
- typos, 127

U

- unary operator, 16, 23
- uncaught exception, 136, 181
- undefined, 18–20, 25, 40, 45, 59, 61, 75, 127–128, 132
- underline, 227
- UndoButton class, 333
- undo history, 332, 333
- Unicode, 15–16, 85, 90, 143–144
- uniformity, 196
- uniqueness, 228
- unit (CSS), 232, 244
- Unix, 350, 352, 353
- Unix time, 147
- unlink function, 344, 350
- unshift method, 69
- unwinding the stack, 133
- updated method, 362, 365, 406
- updateState function, 322
- upgrading packages, 166
- uppercasing server example, 347
- URL (Uniform Resource Locator), 211, 214, 276, 299, 301, 303, 346, 357, 367

- URL class, 348
- URL encoding, 300
- url package, 364
- urlPath function, 348
- usability, 239
- user experience, 235, 306, 356, 367
- user interface, 136, 320
- use strict, 128–129
- UTF-8, 344
- UTF-16, 15, 90

V

- validation, 132, 138, 195, 264, 307, 362
- value attribute, 304, 307, 310
- values, 12
- variable. *See* binding
- var keyword, 25, 40–41, 74
- Vec class, 111, 256–257, 267, 376, 401
- vector, 376
- vector (exercise), 111, 391
- vector graphics, 283
- Vec type, 378
- verbosity, 44, 176
- versions, 166, 212, 298, 343, 380
- viewport, 263–264, 289, 293
- VillageState class, 117
- virtual keyboard, 240
- virtual worlds, 115, 117, 119, 253
- viruses, 214
- vocabulary, 39, 82
- void operator, 26
- volatile data storage, 12

W

- waitForChanges method, 364
- waiting, 177
- walking, 291
- wave, 259, 268
- web, 5, 75, 209, 211, 214, 215, 297
- web application, 5, 312, 319
- web browser. *See* browsers
- WebDAV, 353
- webgl (canvas context), 277
- web page, 170
- websites, 214–215, 299, 339, 353, 355
- WebSockets, 357
- web workers, 246
- weekDay module, 164

- weekend project, 353
- weresquirrel example, 58, 60, 62, 64, 67, 69
- while loop, 4–5, 30, 32, 50
- whitespace, 206
 - in HTML, 221, 326, 399
 - indentation, 31–32
 - matching, 143, 144
 - syntax, 35, 196, 198, 397
 - trimming, 71, 256
 - in URLs, 357
- width property, 336, 404
- window, 237, 242, 246
- window object, 236
- with statement, 129
- word boundary, 148
- word character, 143–144, 148
- workbench (exercise), 315, 402
- work list, 122, 329
- World Wide Web (WWW), 5, 75, 209, 211, 214, 215, 297

- writable stream, 346, 348
- writeFile function, 344, 346, 406
- writeHead method, 346
- write method, 346
- writing code, 6, 115
- writing system, 85

X

- XML, 219, 276
- XML namespace, 276
- xmlns attribute, 276

Y

- yield keyword, 185
- your own loop (example), 93

Z

- zero-based counting, 53, 59, 147
- zeroPad function, 51
- zigzag, 401
- zooming, 293