

INDEX

Symbols and Numbers

&& operator, 17, 20, 95
* operator, 13, 18–19, 148
*= operator, 34
{ } (block), 29
{ } (object), 63, 66
- operator, 13, 16, 18–19
-- operator, 34
-= operator, 34
= operator, 24–25, 63, 348
 for default value, 48
 in Egg, 209–210
 as expression, 160, 162
== operator, 17, 19, 66, 80, 193
=== operator, 19, 80, 113, 410
! operator, 17, 31
!= operator, 17
!== operator, 19
/ operator, 13, 396
/= operator, 34
> operator, 16
>= operator, 17
< operator, 16
<= operator, 17
% operator, 14, 33, 297, 407, 408,
 418, 420
+ operator, 13, 15, 19, 148, 404
++ operator, 34
+= operator, 34, 199
?: operator, 18, 20, 209
[] (array), 60
[] (subscript), 60, 61
|| operator, 17, 20, 51, 95, 328, 408
2d (canvas context), 289
200 (HTTP status code), 312, 360, 364
204 (HTTP status code), 366, 367
304 (HTTP status code), 374, 380, 386
400 (HTTP status code), 425
403 (HTTP status code), 364
404 (HTTP status code), 312, 365,
 377–379
405 (HTTP status code), 316, 364

406 (HTTP status code), 422
500 (HTTP status code), 364

A

a (HTML tag), 222, 235, 237, 320, 344
Abelson, Hal, 202
absolute positioning, 240, 244, 253,
 257, 262
absolute value, 76, 424
abstract data type, 96
abstract syntax tree, *see* syntax tree
abstraction, 5, 41, 84, 86, 203, 316, 349
 in Egg, 230
 of the network, 221
acceleration, 281–282
Accept header, 329, 422
access control, 98, 141–142, 375
Access-Control-Allow-Origin header, 316
actions, 334, 336, 337
activeElement property, 319
actor, 269, 274, 280
add method, 113
addEntry function, 66
addEventListener method, 248, 282, 362
addition, 13, 113
address, 77, 311
address bar, 221, 311, 313
adoption, of tools and techniques, 145
ages example, 104
alert function, 223–224
algorithmic optimization, 397, 406
alpha, 346
alphanumeric character, 147
alt attribute, 233
ALT key, 252
altKey property, 252
ambiguity, 215
American English, 148
ampersand character, 223, 314
analysis, 129, 133
ancestor element, 275
Android, 252

angle, 242, 293, 294, 421
 angle brackets, 222
 animation, 256, 262, 271, 306, 419
 bouncing ball, 308, 421
 performance, 266
 platform game 277, 283, 296, 297,
 303, 305
 spinning cat, 240, 241, 244
 anyStorage function, 198–199, 200
 appendChild method, 232, 417
 Apple, 225
 application (of functions), *see* function
 application
 applications, 1, 333, 372
 arc, 293, 294
 arc method, 293, 421
 argument, 27, 47, 51, 74, 155, 204
 arguments object, 409
 argv property, 355
 arithmetic, 13, 19, 210
 array, 61, 64, 79, 95
 counting, 93
 creation, 91, 60, 336, 409, 413
 in Egg, 214
 filtering, 88
 indexing, 60, 68, 72, 409, 418
 iteration, 69, 85, 87
 length of, 61
 as matrix, 107–108, 268
 methods, 71, 79, 87–89, 92, 94, 95
 notation, 77
 random element, 122
 as regular expression match, 149
 representation, 77
 of rest arguments, 74
 searching, 68, 71
 as table, 67
 Array constructor, 336
 Array prototype, 100, 103
 Array.from function, 195, 231, 356
 array-like object, 230–232, 255, 320,
 325, 358
 arrays in Egg (exercise), 214, 416
 arrow function, 45, 99, 199
 arrow keys, 262
 artificial intelligence, 117, 213
 assert function, 140
 assertion, 140
 assignment, 25, 34, 160, 162, 215, 417
 assumption, 139, 140
 asterisk, 13, 148
 async function, 195–196, 199, 200, 386
 asynchronous programming, 181, 200
 animation, 284
 in Node.js, 354, 359, 361, 365, 368
 reading files, 325
 attribute, 222, 230, 235, 320, 337, 422
 autofocus attribute, 320
 automatic semicolon insertion, 24
 automation, 125, 132
 automaton, 117
 avatar, 266
 average function, 91
 await keyword, 195–196, 197, 199
 axis, 281, 289, 298, 424

B

Babbage, Charles, 58
 background, 266, 273, 278
 background (CSS), 262, 266, 274
 backslash character
 as path separator, 365
 in regular expressions, 146, 147,
 157, 158, 414
 in strings, 14, 223
 backtick, 14–15
 backtracking, 153–154, 156–157
 ball, 308, 421
 balloon (exercise), 262, 418
 banking example, 137
 Banks, Iain, 264
 baseControls constant, 348
 baseTools constant, 348
 bean counting (exercise), 56, 409
 beforeunload event, 258
 behavior, 164, 213
 benchmark, 236, 391, 397
 Berners-Lee, Tim, 218
 best practices, 3
 bezierCurveTo method, 292
 big ball of mud, 167, 168
 binary data, 3, 11, 358
 binary numbers, 11, 12, 68, 133,
 153, 324
 binary operator, 13, 16, 23

- binding
 - assignment, 25, 44
 - compilation of, 215, 403
 - definition, 24, 41, 215, 417
 - destructuring, 76
 - in Egg, 210, 212
 - exported, 173
 - global, 43, 130, 285, 354, 355
 - local, 43
 - model of, 25, 63, 65
 - naming, 26, 35, 52, 75, 131
 - from parameter, 42, 49
 - scope of, 43
 - as state, 31, 33, 65, 160, 326
 - undefined, 138
 - visibility, 43
- bitfield, 255
- bitmap graphics, 295, 308
- bits, 3, 11, 12, 16, 68
- block, 29, 32, 42, 43, 46, 63, 136, 137, 204
- block comment, 36, 156
- block element, 235, 237–238
- blocking, 182, 241, 260, 359
- blur event, 257–258
- blur method, 319
- body (HTML tag), 222–223, 228
- body (HTTP), 313–315, 360, 366, 367, 378
- body property, 228, 229, 231, 316
- bold, 237
- Book of Programming, The*, 10, 352
- Boolean, 16, 28, 31, 65, 146, 209, 210
 - conversion to, 19, 28, 31
- Boolean function, 28
- border (CSS), 235, 237
- border-radius (CSS), 253
- bouncing, 267, 270, 278, 280, 308
- boundary, 151, 153, 157, 161, 164, 302, 413
- bounds, of ranges, 88
- box, 142, 227, 266, 308, 421
- box shadow (CSS), 275
- br (HTML tag), 340, 422
- braces
 - block, 4, 29, 42
 - class, 102
 - function body, 46, 63, 408
 - object, 66, 76, 85
 - in regular expression, 148
- branching, 152, 153
- branching recursion, 50, 300
- break keyword, 33, 35
- breakpoint, 134
- British English, 148
- broadcastConnections function, 193
- browser, 2, 5, 174, 183, 219–225, 266, 267, 308, 313, 316, 321, 344, 348, 372
 - environment, 7, 26, 27, 311
 - security, 317, 392
 - storage, 326–328
 - window, 248
- browser wars, 225
- bubbling, *see* event propagation
- Buffer class, 358, 359, 361, 362
- bug, 83, 129, 133, 157, 159, 164, 168, 225
- building Promise.all (exercise), 200, 416
- bundlers, 174
- button, 247, 313, 320, 330
- button (HTML tag), 224, 248, 252, 262, 321, 327, 330, 337
- button property, 249, 255, 338
- buttons property, 255, 338

C

- C (programming language), 391
- cache, 172, 184
- call method, 99, 104
- call protocol, 400
- call stack, 46–47, 51, 62, 135, 136, 138, 197, 400
- callback functions, 184–190, 247, 283, 337, 358–359, 361, 380, 385
- calling (of functions), *see* function application
- camel case, 35, 238
- cancelAnimationFrame function, 260
- canvas, 266, 287, 289–292, 295–301, 305–307, 395, 421
 - context, 289
 - path, 290
 - size, 288, 290
- canvas (HTML tag), 288, 334, 337, 344, 345, 349, 423
- CanvasDisplay class, 301, 302, 304

- capitalization, 35, 101, 149, 238, 243, 362
- capture group, 149–151, 155, 376
- career, 265
- caret character, 147, 151, 161, 358
- carriage return, 161
- cascading, 238
- Cascading Style Sheets, *see* CSS
- case conversion, 62
- case keyword, 35
- case sensitivity, 149, 414
- casual computing, 2
- cat’s hat (exercise), 244
- catch keyword, 135–136, 138–139, 141, 197, 413
- catch method, 188
- CD, 11
- cell, 330
- Celsius, 110
- centering
 - image on screen, 241
 - player in viewport, 276
- certificate, 317
- change event, 319, 322, 340, 422, 425
- character, 14–15, 92–93, 322
- character category, 163
- character encoding, 358–359
- characterCount function, 90
- characterScript function, 94, 95, 411
- charCodeAt method, 92
- checkbox, 318, 323, 330, 422
- checked attribute, 318, 323
- chessboard (exercise), 38, 408
- chicks function, 198–199
- child node, 229, 230, 232
- childNodes property, 230, 231, 233, 419
- children property, 231
- Chinese characters, 92, 94
- choice, 152
- Chrome, 225, 398, 400, 402, 404
- circle, 241, 293
- circle (SVG tag), 288
- circles (exercise), 350, 423
- circular dependency, 178, 415
- circus, 71
- class, 101, 102, 113, 119, 267, 335
- class attribute, 232, 235, 239, 273, 275
- class declaration, 102
- class hierarchy, 112
- className property, 235
- cleaning up, after exceptions, 136
- clearing, the canvas, 288, 296, 301, 302, 421
- clearInterval function, 260
- clearRect method, 296, 421
- clearTimeout function, 259, 260
- click event, 248, 250, 253, 255, 337, 422
- client, 220, 316, 361, 372, 381, 382
- clientHeight property, 235
- clientWidth property, 235
- clientX property, 253, 255, 339
- clientY property, 253, 255, 339
- clipboard, 225
- clipping, 302
- closePath method, 291
- closing tag, 222, 224
- closure, 49, 214, 417–419
- closure in Egg (exercise), 214, 417
- code, 6, 156, 265
 - structure of, 23, 32, 41, 167, 175
- code golf, 164
- code unit, 92
- codePointAt method, 93
- coin, 266, 267, 280, 305
- Coin class, 271, 280
- collaboration, 219
- collection, 5, 60, 62, 64, 79
- collision detection, 277, 278, 280–282, 420, 421
- colon character, 18, 34, 63, 237
- color, 288, 290, 302, 334, 346
- color (CSS), 237, 238
- color codes, 336
- color field, 334, 336, 341
- color picker, 334, 341, 344
- color property, 335
- ColorSelect class, 341
- comma character, 204
- COMMAND key, 252, 349
- command line, 169, 353–355, 368
- comment, 35, 77, 156, 160, 214, 229, 372, 374, 379, 384, 417
- comment field reset (exercise), 388, 426
- COMMENT_NODE code, 229
- comments in Egg (exercise), 214, 417

- CommonJS modules, 171–173, 355–356
 - exercise, 177–178, 415
- communication, 220, 316
- community, 354
- compareRobots function, 125
- comparison, 16, 19, 31, 35, 80
 - deep, 210, 409
 - of NaN, 17
 - of numbers, 16, 27
 - of objects, 66
 - of strings, 16
 - of undefined values, 19
- compatibility, 5, 219, 225, 350, 357
- compilation, 174, 212, 213, 391, 392, 398, 400, 403, 417
- complexity, 3, 83, 112, 154, 239, 272, 349
- component, 334, 335, 340, 346
- composability, 5, 91, 175
- computed property, 61, 328
- computers, 1, 2
- concat method, 72, 95, 413, 418, 426
- concatenation, 15, 72, 418
- conditional execution, 18, 28, 34, 37, 209
- conditional operator, 18, 20, 209
- conditional request, 374
- configuration, 160
- connected graph, 125
- connection, 220, 311, 317, 372, 373
- connections binding, 192
- consistency, 35, 219, 230
- console.log function, 5, 8, 16, 27, 46, 48, 55, 133, 354, 362–363, 398
- const keyword, 25–26, 43, 65, 75, 76
- constants, 26, 75, 282
- constructor, 35, 101, 102, 112, 126, 130, 136, 150, 157, 411, 413
- content negotiation (exercise), 329, 422
- Content-Length header, 313
- Content-Type header, 313, 360, 365, 369
- context, 289
- context menu, 251
- continuation, 184
- continue keyword, 33
- control, 339, 341, 344, 347
 - control flow
 - asynchronous, 182, 196
 - conditional, 28–30
 - exceptions, 135, 136
 - function, 46, 86
 - loop, 30–31, 32–33
 - control point, 292, 293
 - convention, 35
 - convergent evolution, 184
 - Conway’s Game of Life, 330
 - coordinates, 113, 241–242, 253, 273, 276, 278, 279, 289, 293, 298
 - copy-paste programming, 53, 168
 - copyright, 169
 - correlation, 66, 67, 69, 70
 - corvid, 183
 - cosine, 75, 241
 - countBy function, 93, 95
 - counter variable, 30, 32, 242, 408, 409, 416, 418
 - CPU, 182
 - crash, 138, 140, 378, 387
 - createElement method, 234, 336, 417
 - createReadStream function, 362, 365
 - createServer function, 360, 362, 375, 376
 - createTextNode method, 233, 422
 - createWriteStream function, 361, 367
 - cross-domain request, 316
 - crow, 183, 184, 188, 194
 - crow-tech module, 185
 - crying, 149
 - cryptography, 317
 - CSS, 237–239, 273–277, 287, 290, 336, 382
 - CTRL key, 252, 349
 - ctrlKey property, 252, 349
 - curl program, 367
 - curly braces, *see* braces
 - cursor, 322
 - curve, 292–293
 - cutting point, 272
 - cwd function, 364–365
 - cycle, 229

D

- Dark Blue (game), 265
- data, 2, 11, 59
- data attribute, 235, 262

- data event, 362
- data flow, 334, 349
- data format, 77, 230
- data loss, 387
- data set, 68, 87
- data structure, 59, 175, 177, 306, 330
 - collection, 60
 - immutable, 120
 - list, 79
 - map, 104
 - stack, 62
 - tree, 204, 227, 229
- data URL, 344–345
- date, 147, 148, 150
- Date class, 150, 151, 169, 171
- date-names package, 171
- Date.now function, 151, 346, 405
- dblclick event, 253
- De Morgan's laws, 411
- debouncing, 260
- debugger statement, 134
- debugging, 6, 129–131, 133, 136, 139, 140, 164
- decentralization, 219
- decimal number, 11, 133, 153
- declaration, 237
- decodeURIComponent function, 314, 364, 376
- deep comparison, 66, 80
- deep comparison (exercise), 80, 410
- default behavior, 237, 251
- default export, 173
- default keyword, 35
- default value, 20, 48, 290, 328, 348
- defineProperty function, 411
- defineRequestType function, 185, 190
- degree, 293, 298
- delete method, 113
- DELETE method, 312–313, 315, 363, 366, 378
- delete operator, 63
- deoptimization, 401, 403, 404
- dependence, 66
- dependency, 168, 170, 173, 178, 224, 357
- deserialization, 77
- design, 168
- destructuring, 151
- destructuring assignment, 424
- destructuring binding, 76, 171, 348, 415
- developer tools, 7, 27, 134, 138
- dialect, 174
- dialog box, 26
- diamond, 307, 420
- digit, 11, 12, 133, 147–149, 336
- Dijkstra, Edsger, 116, 176
- Dijkstra's algorithm, 176
- dijkstra package, 176, 414
- dimensions, 113, 235, 265, 267, 278, 289, 401, 408
- dinosaur, 213
- direct child node, 239
- direction (writing), 95
- directory, 355, 358–359, 363, 365–366, 368, 425
- directory creation (exercise), 368, 425
- disabled attribute, 320
- discretization, 266, 278, 283
- dispatch, 34, 334–336, 340, 347, 375, 423
- display, 272, 273, 283, 284, 301, 305, 306
- display (CSS), 237, 262
- distance, 396, 423
- distribution, 392
- division, 13, 14, 396
- division by zero, 14
- do loop, 31, 122
- doctype, 222–223
- document, 221, 227, 258, 287
- document format, 317, 329
- Document Object Model, *see* DOM
- documentation, 353
- documentElement property, 228, 229
- dollar sign, 26, 151, 155, 161
- DOM, 228, 229
 - attributes, 235
 - components, 334
 - construction, 230, 232, 234, 336
 - events, 248, 252
 - fields, 318, 322
 - graphics, 266, 267, 273–276, 287, 288, 305
 - interface, 229–230
 - modification, 232–233
 - querying, 231–232, 239
 - tree, 229

dom property, 335
domain, 221, 313, 316, 327
domain-specific language, 84, 132, 145, 214, 239
DOMDisplay class, 273, 301
dominant direction (exercise), 95, 411
done property, 346
doneAt property, 346
dot character, *see* period character
double-click, 253
double-quote character, 14, 165, 204, 223
download, 7, 169, 344, 356, 367, 371, 387
download attribute, 344
draggable bar example, 254
dragging, 254, 334, 342, 350
draw function, 342, 350
drawGraph function, 395
drawImage method, 295–296, 298, 301, 303, 421
drawing, 227, 235, 236, 241, 272, 273, 288, 289, 292, 299, 303, 305, 333, 334, 422
drawing program example, 253, 333
drawPicture function, 338, 344, 350, 423
drop-down menu, 319, 324
duplication, 168
dynamic typing, 403

E

ECMAScript, 5, 6, 173
ECMAScript 6, 6, 426
economic factors, 349
ecstatic package, 376
Edge, 398, 400, 402
editor, 32
efficiency, 50, 79, 91, 192, 212, 236, 266, 275, 289, 338, 349
efficient drawing (exercise), 349, 423
Egg language, 203, 215, 229
electronic life, 266
elegance, 50, 205
element, 222, 229, 231, 234
ELEMENT_NODE code, 229, 418
elements property, 320
ellipse, 240, 242
else keyword, 29
elt function, 234, 336, 349, 384

email, 317
emoji, 15, 92, 162, 262
empty set, 156
encapsulation, 97, 98, 105, 112, 248, 272
encodeURIComponent function, 314, 373–374, 383
encoding, 220
encryption, 317
end event, 362
end method, 360, 361, 364
enemies example, 160
engine, 391, 398, 400, 402
engineering, 225
ENOENT (status code), 365
ENTER key, 321
entity, 223
enum (reserved word), 26
environment, 26, 208
equality, 17
error, 92, 129, 130, 133–135, 138, 139, 187, 188, 194
error event, 326, 367
error handling, 129, 135, 138, 359, 364, 365, 383, 386
error message, 207, 330
error recovery, 134
error response, 312, 364, 366
error tolerance, 223
Error type, 136, 138, 139, 365
ES modules, 173, 224
ESC key, 285
escaping
 in HTML, 222, 224
 in regular expressions, 146, 147, 158
 in strings, 14, 204
 in URLs, 314, 364, 373–374, 376
Escher, M.C., 286
ETag header, 374, 380, 386
eval operator, 170
evaluate function, 207–208, 210
evaluation, 170, 207, 213
even numbers, 30, 56
event handling, 247–262, 266, 282, 284, 285, 295–296, 305, 321, 322, 337, 362, 419, 422
event loop, 197
event object, 249, 253, 255
event propagation, 249, 250, 257, 258

- event type, 249
- every method, 95
- everything (exercise), 95, 411
- everywhere function, 192
- evolution, 145, 349, 357
- exception handling, 135, 136,
 - 138–142, 187, 188, 196, 197,
 - 200, 416, 422
- exception safety, 138
- exec method, 149–150, 158, 160
- execution order, 28, 45, 46
- exercises, 2, 7, 37, 133
- exit method, 354
- expectation, 251
- experiment, 3, 7, 164
- exploit, 225
- exponent, 13, 165, 414
- exponentiation, 31, 33
- export keyword, 173
- exports object, 171–173, 356, 415
- expression, 23, 24, 27, 31, 33, 44, 203,
 - 204, 208
- expressivity, 214
- extension, 355
- extraction, 150

F

- factorial function, 8
- Fahrenheit, 110
- fallthrough, 35
- false, 16
- farm example, 52, 54, 152
- fetch function, 315, 329, 361, 383,
 - 385, 425
- field, 252, 313, 318, 320, 323, 326, 330,
 - 334, 388
- Fielding, Roy, 310
- file, 168, 325, 426
 - access, 172, 174, 345, 358, 359
 - image, 333, 344
 - resource, 312, 313, 363, 365, 366
 - stream, 361
- file extension, 365
- file field, 318, 325
- file format, 160
- file reading, 325
- file server, 381
- file server example, 363, 365–369, 425
- file size, 174

- file system, 325, 358, 359, 363, 364, 425
- File type, 325
- FileReader class, 325, 326, 345
- files property, 325
- fill function, 343
- fill method, 291, 336
- fillColor property, 336
- filling, 289–291, 295, 306
- fillRect method, 289, 296
- fillStyle property, 290, 295
- fillText method, 295, 420
- filter method, 88, 91, 94, 120, 191,
 - 411, 413
- finally keyword, 137, 142, 413
- findIndex method, 94
- findInStorage function, 194, 195
- findRoute function, 124, 193
- finish event, 367
- Firefox, 225, 398, 400, 402, 404
- firewall, 372
- firstChild property, 230
- fixed positioning, 257
- fixing scope (exercise), 215, 417
- FizzBuzz (exercise), 37, 407
- flattening (exercise), 95
- flexibility, 6
- flipHorizontally function, 304, 420
- flipHorizontally method, 298
- flipping, 297–298
- floating-point number, 12–13
- flood fill, 340, 343
- flooding, 192
- flow diagram, 152, 153
- focus, 252, 257, 319, 320, 322, 323,
 - 349, 388
- focus event, 257–258
- focus method, 319
- fold, *see* reduce method
- font, 295
- font-family (CSS), 238
- font-size (CSS), 262
- font-weight (CSS), 238
- for attribute, 323
- for loop, 32, 33, 69, 85, 95, 139, 397,
 - 409, 410
- for/of loop, 69, 93, 106, 107, 109,
 - 398, 412
- force, 393–398
- force-directed graph layout, 393

- forceDirected_noRepeat function, 397
- forceDirected_noVector function, 401
- forceDirected_simple function, 396
- forceDirected_skip function, 398
- forEach method, 87
- form, 313, 314, 320, 321, 369
- form (HTML tag), 318, 320, 385, 425
- form property, 320
- formatDate module, 171, 173
- fractal example, 300
- fractional numbers, 13, 165, 266
- fragmentation, 402
- frame, 296, 297, 303, 421
- framework, 54, 334
- frequency table, 67
- fs package, 358, 359
- Fuller, Margaret, 370
- function, 5, 26, 41, 45, 130, 203, 204, 211
 - application, 26, 27, 42, 46, 47, 50, 74, 88, 138, 203, 208
 - body, 42, 46
 - callback, *see* callback function
 - declaration, 45
 - definition, 41, 45, 52
 - higher-order, 45, 85, 86, 88, 89, 91, 155, 283
 - inlining, 400
 - model of, 49
 - naming, 52–54
 - as property, 62
 - purity, 55
 - scope, 44, 169, 214
 - as value, 42, 44, 48, 85, 86, 88, 249, 283, 419
- Function constructor, 170, 172, 210, 213, 330, 422
- function keyword, 42, 45
- Function prototype, 100, 103
- future, of JavaScript, 6, 26, 45, 308

G

- game, 265–267, 282, 284, 285, 301
 - with canvas, 305
 - screenshot, 277, 305
- game of life (exercise), 330, 422
- GAME_LEVELS data set, 284
- garbage collection, 12, 400–403
- garble example, 355

- gardening, 371
- gaudy home pages, 262
- generation, 330, 422
- generational garbage collection, 403
- generator, 196
- get method, 105
- GET method, 312, 313, 315, 321, 361, 363, 365, 373, 377–378
- getAttribute method, 235
- getBoundingClientRect method, 236, 339
- getContext method, 289
- getDate method, 151
- getElementById method, 232, 417
- getElementsByClassName method, 232
- getElementsByTagName method, 232, 233, 243, 418
- getFullYear method, 151
- getHours method, 151
- getImageData method, 345, 346
- getItem method, 326, 328
- getMinutes method, 151
- getMonth method, 151
- getPrototypeOf function, 100, 101, 215, 417
- getSeconds method, 151
- getter, 109, 113, 270
- getTime method, 150
- getYear method, 151
- GitHub, 312
- global object, 130
- global scope, 43, 170, 210, 259, 354, 355, 417
- goalOrientedRobot function, 125
- Google, 225
- gossip property, 192
- grammar, 23, 129, 160
- graph, 118, 124, 176, 193, 306, 392, 394, 405
- graph layout, 393, 394, 397, 398, 400
- graphics, 266, 272, 275, 287–289, 295, 305, 306
- GraphNode class, 394, 400
- grave accent, *see* backtick
- gravity, 282
- greater than, 16
- greed, 156–157
- grep, 368
- grid, 266, 273, 278–279, 330, 422
- Group class, 113, 114, 126, 197, 411, 412

groupBy function, 95
grouping, 13, 29, 149, 155, 413
groups (exercise), 113, 114, 411, 412

H

h1 (HTML tag), 222, 235
hack, 173
handleAction function, 382
hard disk, 175, 181, 184
hard drive, 12, 325, 327, 353, 387
hard-coding, 231, 307
has method, 105, 113
hash mark character, 214, 336
hasOwnProperty method, 105, 215, 417
head (HTML tag), 222, 223, 228
head property, 228
header, 313, 316, 360, 373
headers property, 315, 316, 329
height property, 350, 423
help text example, 257
hexadecimal numbers, 153, 314,
336, 346
Hières-sur-Amby, 183
hidden element, 238, 262
higher-order functions, *see* function,
higher-order
history, 5, 349
historyUpdateState function, 346
Hoare, C.A.R., 82
Hooke's law, 395
hooligan, 375
Host header, 313
hot code, 392
href attribute, 222, 232, 235
HTML, 221, 312, 326, 369
notation 221
structure 227
html (HTML tag), 223, 228
HTTP, 219–221, 311–317, 366, 367,
369, 373,
client, 361, 367, 372
server, 360, 363, 375, 381
http package, 360–361
HTTPS, 221, 317, 361
https package, 361
human language, 23
Hypertext Markup Language, *see*
HTML

Hypertext Transfer Protocol, *see* HTTP
hyphen character, 13, 147, 238

I

id attribute, 232, 239, 323
idempotence, 189, 366, 425
identifier, 204
identity, 65
if keyword, 28, 162
 chaining, 29, 34, 408
If-None-Match header, 374, 380, 385
image, 233, 258, 287, 313
imagination, 265
IME (input method editor), 252
img (HTML tag), 222, 233, 237, 258,
287, 295, 296, 345
immutable, 65, 120, 270, 335, 336, 342,
346, 423
implements (reserved word), 26
import keyword, 173
in operator, 64, 105
includes method, 68, 69, 399, 411
indentation, 32
index, 61
index property, 149
index.html file, 381
index.js file, 355
indexOf method, 71, 73, 94, 113, 146,
158, 411
infinite loop, 33, 47, 139, 409
infinity, 14
infrastructure, 169
inheritance, 100, 111–112, 139, 365
INI file, 160
ini package, 169, 173, 175, 356
initialization, 258
inline caching, 403, 404
inline element, 235, 237
inlining, 400
inner function, 44, 418
inner loop, 154
innerHeight property, 257
innerWidth property, 257
input, 134, 247, 266, 319, 353, 378
input (HTML tag), 257, 318, 322, 323,
325, 341, 345
input event, 322
insertBefore method, 232, 233
installation, 169

instance, 101
instanceof operator, 112, 139
instructions, 3, 391
integers, 13
integration, 145, 230
interface, 98
 canvas, 287, 289
 design, 54, 145, 150, 155, 158, 229,
 230, 272, 290
 HTTP, 316, 373
 module, 168–172, 175, 315, 356
 object, 105, 106, 109, 114, 126, 189,
 269, 301, 322, 335
interface (reserved word), 26
internationalization, 162
internet, 160, 220, 221, 224
Internet Explorer, 225
interpolation, 15
interpretation, 7, 170, 207–208,
 212, 391
interview question, 37
inversion, 147
invoking (of functions), *see* function
 application
IP address, 221, 311, 313
isDirectory method, 366, 425
isEven (exercise), 56, 408
isolation, 97, 167, 170, 224
iterable interface, 107, 412
iterator, 196
iterator interface, 106, 107, 114, 398

J

Jacques, the weresquirrel, 60
Java, 5
JavaScript, 5
 availability of, 2
 flexibility of, 6
 history of, 5, 219
 in HTML, 223
 syntax, 23
 uses of, 6
 versions of, 6
 weaknesses of, 6
JavaScript console, 7, 16, 27, 134, 138,
 330, 354
JavaScript Object Notation, *see* JSON
join method, 94, 104, 356
journal, 60, 62, 64, 66, 69

JOURNAL data set, 68
journalEvents function, 69
JSON, 77, 175, 184, 193, 315, 328, 373,
 386, 425
 json method, 315
 JSON.parse function, 77, 426
 JSON.stringify function, 77
JSX, 384
jump, 4
jump-and-run game, 265
jumping, 266, 282
just-in-time compilation, 392

K

Kernighan, Brian, 128
key code, 282
key property, 252, 418, 423
keyboard, 26, 247, 251, 266, 281, 282,
 285, 319, 320, 322, 349
keyboard bindings (exercise), 349, 423
keyboard focus, *see* focus
keydown event, 251, 260, 282, 349,
 418, 423
keyup event, 251, 282
keyword, 24, 26, 235
Khasekhemwy, 322
kill process, 360
Knuth, Donald, 40

L

label, 295, 307
label (HTML tag), 323, 341
labeling, 323
landscape example, 44
Laozi, 180
lastChild property, 230
lastIndex property, 158–160
lastIndexOf method, 71
Last-Modified header, 313
latency, 174
lava, 266, 267, 275, 278, 280, 305
Lava class, 270, 280
layering, 194, 220
layout, 235–237
laziness, 236
Le Guin, Ursula K., 2
leaf node, 229
leak, 225, 285

- learning, 2, 6, 371
- left (CSS), 240–242, 244
- LEGO, 168
- length property
 - for arrays, 61, 336
 - for strings, 53, 56, 61, 74, 409
- less than, 16
- let keyword, 24, 25, 43, 65, 75, 76, 130
- level, 266, 267, 273, 275, 284
- Level class, 267
- lexical scoping, 44
- library, 230, 334, 356, 357
- license, 169
- line, 24, 32, 161, 287, 289–294, 307, 420
- line break, 14, 161
- line comment, 35, 156
- line drawing, 350, 424
- line width, 290, 297
- lines of code, 211
- lineTo method, 290–291
- lineWidth property, 290
- link (HTML tag), 277
- linked list, 79, 410, 426
- links, 222, 230–231, 251, 252, 344
- linter, 173
- Liskov, Barbara, 96
- list (exercise), 79, 410
- listen method, 360
- listening (TCP), 220, 360
- literal expression, 23, 146, 206, 208
- live data structure, 227, 233, 240, 419
- live view, 372, 373, 387, 426
- lives (exercise), 285
- load event, 258, 295–296, 303, 326, 421
- LoadButton class, 344
- local binding, 48, 215, 409
- local scope, 43, 212
- localhost, 360
- localStorage object, 326–327, 383
- locked box (exercise), 141, 413
- logging, 133
- logical operators, 17
- long polling, 372–374, 378, 380, 385
- loop, 4, 30, 32, 37, 38, 50, 69, 85, 90, 91, 160, 189, 408, 409, 420
 - termination of, 33
- loop body, 31, 85
- lycanthropy, 60, 66

M

- machine code, 3, 213, 391
- macro-optimization, 406
- magic, 99, 203
- mailRoute array, 123
- maintenance, 169
- malicious script, 224
- man-in-the-middle, 317
- map, 272, 321
- map (data structure), 104
- Map class, 105, 109, 195
- map method, 88, 91, 94, 99, 104, 120, 191, 268, 340
- Marcus Aurelius, 246
- match method, 149, 159
- matching, 146, 151, 152, 158, 164
 - algorithm, 152–154
- Math object, 56, 61, 75
- Math.abs function, 76, 424
- Math.acos function, 75
- Math.asin function, 75
- Math.atan function, 75
- Math.ceil function, 76, 278, 302–303
- Math.cos function, 75, 241, 242, 421
- mathematics, 50, 86
- Math.floor, 76, 122, 278, 302–303
- Math.max function, 27, 61, 74, 75, 302
- Math.min function, 27, 56, 75, 302
- Math.PI constant, 75, 293
- Math.random function, 75, 122, 271, 330, 404
- Math.round function, 76
- Math.sin function, 75, 241, 242, 271, 281
- Math.sqrt function, 68, 75, 411
- Math.tan function, 75
- Matrix class, 107–108, 335
- matrix example, 107–108, 111
- MatrixIterator class, 108
- max-height (CSS), 275–276
- maximum, 27, 75, 90
- max-width (CSS), 275–276
- McConnell, Steve, 390
- Meadowfield, 117
- measurement, 397
- measuring, 399
- measuring a robot (exercise), 125, 412
- media type, 317, 329, 365
- meetups, JavaScript, 371

- memory, 3, 11
 - call stack, 24
 - organization, 12, 47, 60, 65, 77
 - persistence, 387
 - speed, 181, 213, 400, 402, 406
 - structure sharing, 79
- mesh, 221
- message event, 259
- meta key, 252
- metaKey property, 252, 349
- method, 62, 100, 101
 - array, 71
 - HTTP, 312, 317, 360, 367, 373, 375
 - interface, 98
- method attribute, 313
- method call, 98
- method property, 315
- methods object, 363
- micro-optimization, 397, 399, 406
- Microsoft, 225
- Middle East, graph of, 393
- mime package, 365
- MIME type, 329, 365
- mini application, 326
- minifiers, 175
- minimalism, 265
- minimum, 27, 56, 75
- minimum (exercise), 56, 408
- minus, 13, 165
- Miró, Joan, 332
- mirror, 298, 308, 421
- mirroring, 297–298
- MKCOL method, 368–369, 425
- mkdir function, 368–369, 425
- modification date, 366
- modifier key, 252
- modular robot (exercise), 177, 414
- modularity, 97, 334
- module, 168, 169, 177, 272, 355, 356, 375
 - design, 175
- module loader, 355
- module object, 172
- module system, 169
- modulo (remainder) operator, 14, 33, 297, 407, 408, 418, 420
- Mongolian vowel separator, 162
- monster (exercise), 285, 419
- Mosaic, 225
- motion, 266
- mouse, 26
 - button, 249, 250, 253
 - cursor, 253
- mouse trail (exercise), 262, 418
- mousedown event, 250, 253, 255, 337, 338, 422
- mousemove event, 254, 260–262, 338, 350, 418
- mouseup event, 253–255
- moveTo method, 290, 293
- Mozilla, 225
- multiple attribute, 324, 325
- multiple-choice, 318–319, 323, 324
- multiplication, 13, 269–270, 280
- multiplier function, 49
- music, 265
- mutability, 63, 65, 120

N

- name attribute, 320, 324
- namespace, 75
- naming, 4, 5, 25–26
- NaN (not a number), 14, 17, 18, 130
- negation, 16, 17
- neighbor, 330, 422
- neighbors property, 190
- nerd, 158
- nesting
 - of arrays, 67
 - of expressions, 23, 205
 - of functions, 44
 - of loops, 38, 408
 - of objects, 228, 231
 - in regular expressions, 154
 - of scope, 44
- Netscape, 5, 225
- network, 182, 219, 372
 - abstraction, 194, 316
 - protocol, 220
 - reliability, 188
 - security, 317
 - speed, 174, 181, 353
- network function, 194
- new operator, 101
- newline character, 14, 38, 147, 156, 161, 268, 422
- next method, 107, 197, 412
- nextSibling property, 230

Nietzsche, Friedrich, 226
node, 228, 229
node program, 354
node-fetch package, 361
Node.js, 6, 7, 27, 171, 183, 353–369,
372, 373, 375, 387, 398
NodeList type, 230, 239
node_modules directory, 355, 356
nodeName property, 243
nodeType property, 229, 418, 419
nodeValue property, 231
nonbreaking space, 162
normalizing, 396
not a number (NaN), 14, 17, 18, 129
notation, 173
note-taking example, 327
notification, 372
NPM, 169, 171, 173, 174, 176, 177,
355–358, 365, 375, 376,
387, 415
npm program, 356, 357, 365
null, 18, 19, 51, 61, 77, 80, 134
number, 12, 65, 146, 165, 414
 conversion to, 19, 28
 notation, 12–13
 precision of, 13
 representation, 12
 special values, 14
Number function, 28, 35
number puzzle example, 50–52
Number.isNaN function, 29

O

object, 59, 63–65, 97, 112
 creation, 77, 101, 328, 401, 403
 identity, 65
 as map, 272
 as module, 169
 mutability, 65
 property, 27, 61, 75, 76, 99
 representation, 77
Object prototype, 99, 100
object shape, 404
Object.assign function, 328, 336
Object.create function, 100, 104, 211
Object.keys function, 64, 80, 195,
410, 417
object-oriented programming, 97, 101,
105, 106, 111, 119, 175

Object.prototype, 104
observation, 399
obstacle, 277, 278
offsetHeight property, 235, 236
offsetWidth property, 235
on method, 362
onclick attribute, 224, 248
onclick property, 337
OpenGL, 289
opening tag, 222
operator, 13, 16, 19, 204, 210, 404
 application, 13
optimization, 50, 55, 236, 260, 266,
275, 306, 308, 359, 392,
398–400, 403, 406, 426
option (HTML tag), 319, 324, 425
optional, in pattern, 148
optional arguments, 48, 78
options property, 324
ordering, 220
ordinal package, 171–172
organic growth, 167
organization, 167
outline, 289
output, 16, 26, 27, 133, 134, 211,
353, 422
overflow (CSS), 275–276
overflow, with numbers, 12
overlap, 278
overlay, 238
overriding, 103, 105, 111, 415
overwriting, 367, 369, 378

P

p (HTML tag), 222, 235
package, 168, 171, 355, 357
package (reserved word), 26
package manager, 169
package.json file, 357
padding (CSS), 274
page reload, 258, 321, 326
pageX property, 253, 255
pageXOffset property, 236
pageY property, 253, 255
pageYOffset property, 236, 257
Palef, Thomas, 265
panning, 339
paragraph, 222
parallelism, 182, 313

- parameter, 27, 42, 43, 46–48, 74, 76, 99, 131, 172
- parent node, 249
- parentheses
 - arguments, 23, 41, 46, 85, 204
 - expression, 13
 - in regular expressions, 149, 151, 152, 162, 413
 - statement, 27, 29, 31, 33
- parentNode property, 230
- parse function, 207
- parseApply function, 206
- parseExpression function, 205
- parseINI function, 161, 168
- parsing, 77, 129, 161, 203–206, 208, 211, 223, 227, 364, 380
- password, 317
- password field, 318
- path
 - canvas, 290–293, 420
 - closing, 291
 - file system, 355, 363
 - URL, 312, 315, 363, 364, 373, 375
- path package, 365
- pathfinding, 123, 176, 193, 343
- pathfinding (exercise), 405, 406, 426
- patience, 350
- pattern, 145–147, 157
- pausing (exercise), 285, 419
- pea soup, recipe analogy, 84
- peanuts, in weresquirrel example, 70–71
- percent sign, 314
- percentage, 94, 257
- performance, 154, 174, 212, 236, 266, 306, 359, 391, 393, 397, 399
- period character, 27, 61, 74, 147, 156, 165, 336
- persistence, 326, 372, 387, 425
- persistent data structure, 119, 120, 126, 132, 335, 342, 346, 419
- persistent group (exercise), 126
- persistent map (exercise), 413
- PGroup class, 126, 413
- phase, 271, 281
- phi coefficient, 66–68
- phi function, 68, 76
- phone, 252
- physics, 277, 281, 393, 418
- physics engine, 278
- pi, 13, 75, 241, 271, 293
- PI constant, 75, 242
- pick function, 343
- picture, 287, 288, 296, 306, 334, 346
- Picture class, 335, 345
- picture property, 335
- PictureCanvas class, 337, 349
- pictureFromImage function, 345
- pie chart example, 294, 295, 307, 420
- ping request, 190
- pink, 336
- pipe analogy, 220
- pipe character, 152, 414
- pipe method, 364, 367
- pipeline, 175
- pixel, 235, 242, 253, 267, 273, 288, 289, 295, 296, 302, 306, 308, 333, 335, 339, 342, 343, 345, 350, 423
- pixel art, 296
- PixelEditor class, 340, 347, 349
- pizza, in weresquirrel example, 67
- platform game, 265, 285
- Plauger, P.J., 128
- player, 265–267, 275, 278, 281, 284, 296, 303, 305
- Player class, 270, 281
- plus character, 13, 148, 165
- pointer, 230
- pointer events, 253–256, 337
- pointerPosition function, 338
- polling, 247
- pollTalks function, 385
- polymorphism, 105–106
- pop method, 62, 71
- Popper, Karl, 234
- port, 220, 311, 360
- pose, 296
- position, of elements on screen, 236
- position (CSS), 240, 244, 257, 266, 275
- POST method, 313, 314, 321, 374
- postMessage method, 259
- power example, 42, 48, 50
- precedence, 13, 17, 239
- predicate function, 88, 92, 95
- Prefer header, 374, 380, 385
- premature optimization, 50

- preventDefault method, 251, 256–258, 282, 321, 339, 423
- previousSibling property, 230
- primitiveMultiply (exercise), 141, 413
- privacy, 225
- private (reserved word), 26
- private properties, 98, 141–142
- process object, 354–355, 364–365
- processor, 181, 400
- profiling, 50, 399
- program, 2, 23, 28
- program size, 83, 84, 164, 272
- programming, 1
 - difficulty of, 2
 - history of, 3
 - joy of, 1, 2
- Programming Is Terrible*, 166
- programming language, 1–2
 - creating, 203, 213
 - DOM, 229
 - history of, 3
 - machine language and, 391
 - Node.js and, 354
 - power of, 5
- programming style, 3, 24, 32, 35, 272
- progress bar, 256
- project chapter, 117, 203, 265, 333, 371
- promise, 200, 416
- Promise class, 186, 187, 189, 195, 197, 198, 200, 315, 326, 359, 361, 363, 386, 416
- Promise.all function, 190, 199, 200, 416
- Promise.reject function, 187
- Promise.resolve function, 186, 190
- promises package, 359
- promptDirection function, 139
- promptInteger function, 134
- propagation, of events, 249, 250, 257, 258
- proper lines (exercise), 350, 424
- property
 - access, 27, 61, 129, 348, 403
 - assignment, 63
 - definition, 63, 66, 109
 - deletion, 63, 98
 - inheritance, 99, 101, 103
 - model of, 63
 - naming, 105–107
 - testing for, 64
- protected (reserved word), 26
- protocol, 220, 221, 311–312
- prototype, 99–104, 111, 211, 215, 417, 426
 - diagram, 103
- prototype property, 101
- pseudorandom numbers, 75
- public (reserved word), 26
- public properties, 98
- public space (exercise), 369, 425
- publishing (packages), 358
- punch card, 3
- pure function, 55, 79, 88, 175, 330, 422
- push method, 62, 69, 71, 411, 426
- pushing data, 372
- PUT method, 312–313, 363, 367, 373, 378, 425
- Pythagorean theorem, 411, 423
- Python, 391

Q

- quadratic curve, 292
- quadraticCurveTo method, 292, 420
- query string, 314, 374, 380
- querySelector method, 240, 417
- querySelectorAll method, 239, 324
- question mark, 18, 148, 157, 314
- queue, 198
- quotation mark, 14, 165
- quoting
 - in JSON, 77
 - of object properties, 63
- quoting style (exercise), 165, 413

R

- rabbit example, 98, 100–102
- radians, 242, 293, 298
- radio buttons, 318, 323
- radius, 350, 423
- radix, 11
- raising (exceptions), 135
- random numbers, 75, 271
- random-item package, 414
- randomPick function, 122
- randomRobot function, 122
- range, 88, 147, 148

- range function, 5, 78, 409
- Range header, 316
- ray tracer, 306
- readability, 4, 5, 35, 50, 54, 135, 167, 208, 276, 307
- readable stream, 361, 362, 364, 378
- readAsDataURL method, 345
- readAsText method, 326
- readdir function, 359, 366, 425
- readdirSync function, 425
- read-eval-print loop, 354
- readFile function, 172, 358, 425
- readFileSync function, 359, 424
- reading code, 6, 117
- readStorage function, 184
- readStream function, 378, 379
- real-time events, 247
- reasoning, 17
- recipe analogy, 84
- record, 62
- rect (SVG tag), 288
- rectangle, 266, 278, 289, 307, 342
- rectangle function, 342, 423
- recursion, 47, 50, 56, 80, 189, 195, 205, 206, 208, 231, 243, 300, 394, 408, 410, 413, 416, 418
- reduce method, 89, 91, 94, 95, 340, 411
- redundancy, 397
- ReferenceError type, 215
- RegExp class, 146, 157, 424
- regex golf (exercise), 164
- regular expressions, 145–165, 206, 368, 375, 376, 417, 424
 - alternatives, 152
 - backtracking, 153
 - boundary, 151
 - creation, 146, 157
 - escaping, 146, 158, 414
 - flags, 149, 155, 157, 414
 - global, 155, 158, 159
 - grouping, 149, 155
 - internationalization, 162
 - matching, 152, 158
 - methods, 146, 150, 158
 - repetition, 148
- rejecting (a promise), 187, 189, 198
- relative path, 172, 224, 355, 363, 425
- relative positioning, 240, 241
- relative URL, 315
- remainder (modulo) operator, 14, 33, 297, 407, 408, 418, 420
- remote access, 363
- remote procedure call, 316
- removeChild method, 232
- removeEventListener method, 248, 419
- removeItem method, 326
- rename function, 359
- rendering, 289
- renderTalk function, 384
- renderTalkForm function, 385
- renderUserField function, 383
- repeat method, 73, 257
- repeating key, 251
- repetition, 52, 148, 154, 157, 260
- replace method, 155, 165, 413
- replaceChild method, 233, 418
- replaceSelection function, 322
- reportError function, 383
- repulsion, 393, 395
- request, 185, 189, 220, 312, 313, 321, 360, 361, 367, 372
- request function, 189, 361, 362
- request type, 185
- requestAnimationFrame function, 241, 258, 260, 283, 308, 418
- requestType function, 190
- require function, 171, 172, 178, 355, 356, 365, 375
- reserved words, 26
- resolution, 172, 355
- resolve function, 364
- resolving (a promise), 186, 187, 189, 198
- resource, 220, 221, 312, 313, 317, 363, 377
- response, 185, 189, 312, 313, 316, 360, 364, 366
- Response class, 315
- responsiveness, 247, 353,
- rest parameters, 74
- restore method, 299, 300
- result property, 326
- retry, 189
- return keyword, 42, 47, 101, 196, 408, 411
- return value, 27, 42, 134, 185, 410
- reuse, 54, 112, 167–169, 356
- reverse method, 79

- reversing (exercise), 79, 409
- rgb (CSS), 274
- right-aligning, 243
- rmdir function, 366, 368
- roadGraph object, 118
- roads array, 117
- roads module (exercise), 177, 415
- robot, 117, 119, 121, 123, 125, 177
- robot efficiency (exercise), 125, 412
- robustness, 373
- root, 229
- rotate method, 298, 300
- rotation, 307, 420
- rounding, 76, 134, 278, 279, 302, 424
- router, 372, 375
- Router class, 375, 376
- routeRequest function, 194
- routeRobot function, 123
- routing, 192
- rows, in tables, 243
- Ruby, 391
- rules (CSS), 238, 239
- run function, 211
- runAnimation function, 283, 285
- runGame function, 284, 285
- runLayout function, 396
- runLevel function, 283, 285
- running code, 7
- runRobot function, 121, 412
- run-time error, 132–134, 140, 417
- Rust (programming language), 391

S

- Safari, 225
- sandbox, 7, 59, 224, 227, 316
- save method, 299, 300
- SaveButton class, 344
- scalar replacement of aggregates, 400, 402
- scale constant, 337–339
- scale method, 297, 299
- scaling, 273, 296, 297, 303, 421
- scalpel (exercise), 200, 416
- scheduling, 197, 354
- scientific notation, 13, 165
- scope, 43, 44, 48, 168, 170–173, 208, 210, 214, 215, 417
- script (HTML tag), 223, 224, 258
- SCRIPTS data set, 87, 89, 92, 93, 95

- scroll event, 256, 260
- scrolling, 251, 256–257, 275–276, 282, 301
- search method, 158
- search problem, 124, 152, 154, 232, 368, 405
- search tool (exercise), 368, 424
- section, 161
- Secure HTTP, 221, 317, 361
- security, 224, 225, 316, 317, 325, 327, 364, 375
- select (HTML tag), 319, 324, 327, 334, 340, 425
- selected attribute, 324
- selection, 322
- selectionEnd property, 322
- selectionStart property, 322
- selector, 239
- self-closing tag, 222
- semantic versioning, 357
- semicolon, 23, 24, 33, 237
- send method, 185, 188
- sendGossip function, 191
- sep binding, 364–365
- sequence, 148
- serialization, 77
- server, 220, 221, 311–313, 315, 316, 353, 360, 362, 363, 372, 375
- session, 328
- sessionStorage object, 328
- set, 146, 147, 229
- Set (data structure), 113, 126
- Set class, 113, 126, 413
- set method, 105
- setAttribute method, 235, 337
- setInterval function, 260, 296
- setItem method, 326
- setter, 110
- setTimeout function, 184, 197, 259, 260, 380, 386
- shape, 287, 290, 291, 293, 295, 307
- shapes (exercise), 307, 420
- shared property, 100, 103
- SHIFT key, 252, 423
- shift method, 71
- shiftKey property, 252
- short-circuit evaluation, 20, 51, 209, 411
- SICP, 202

- side effect, 24, 27, 34, 42, 54, 65, 79, 88, 159, 175, 199, 230, 232, 233, 236, 290, 299, 314, 334, 335
- sign, 12, 165, 414
- sign bit, 12
- signal, 11
- simplicity, 213
- simulation, 119, 121, 265, 270, 330, 393, 418
- sine, 75, 241, 271, 281
- single-quote character, 14, 165, 224
- singleton, 126
- skill, 333
- SkillShareApp class, 386
- skill-sharing project, 371–373, 375, 381
- skipSpace function, 206, 214
- slash character, 13, 35–36, 146, 156, 315, 364, 425
- slice method, 72, 73, 88, 233, 409, 416
- slope, 424
- sloppy programming, 261
- smooth animation, 241
- SMTP, 220
- social factors, 349
- socket, 372–373
- some method, 92, 95, 191, 376, 426
- sorting, 229
- source property, 158
- special form, 203, 208
- special return value, 134, 135
- specialForms object, 208
- specificity, 239
- speed, 1, 2, 308, 421
- SpiderMonkey, 400
- spiral, 307, 420
- split method, 118, 268
- spread, 74, 336
- spread operator, 274
- spring, 393, 395
- sprite, 296, 303–304
- spy, 256
- square, 28
- square brackets, 60, 61, 74, 76, 107, 147, 324, 328, 409
- square example, 41–42, 45, 46
- square root, 68, 75, 411
- src attribute, 222, 224
- stack, *see* call stack
- stack overflow, 47, 50, 56, 408
- stack trace, 136
- staged compilation, 392
- standard, 5, 26, 35, 88, 136, 162, 349, 354, 355
- standard environment, 26
- standard output, 354, 362–363
- standards, 219, 225
- star, 307, 420
- Star Trek* insignia, 292
- startPixelEditor function, 347
- startState constant, 347
- startsWith method, 364
- stat function, 359, 365, 366, 425
- state
 - of application, 275, 334, 342, 346, 347, 388
 - in binding, 24, 31, 32, 34, 400
 - of canvas, 289, 299
 - in iterator, 197
 - in objects, 119, 268, 301
 - transitions, 198, 336, 337
- statement, 23, 24, 28, 31, 32, 42, 63
- static (reserved word), 26
- static file, 373, 376
- static method, 110, 113, 268, 413
- static typing, 403
- Stats type, 366
- statSync function, 425
- status code, 312, 354–355
- status property, 315, 383
- stdout property, 362–363
- stopPropagation method, 250
- storage function, 187
- stream, 220, 361–363, 364, 367, 378
- strict mode, 130
- string, 14, 60, 62, 65, 92
 - indexing, 56, 72, 74, 92, 149
 - length, 37, 92
 - methods, 73, 149
 - notation, 14
 - properties, 72
 - representation, 15
 - searching, 73
- String function, 28, 105
- stroke method, 290–292
- strokeRect method, 289, 421
- strokeStyle property, 290
- strokeText method, 295

- stroking, 289, 290, 295, 306
- strong (HTML tag), 235, 237
- structure, 168, 222, 227, 334
- Structure and Interpretation of Computer Programs*, 202
- structure sharing, 79
- style, 237
- style (HTML tag), 238, 239
- style attribute, 237–239, 273
- style sheet, *see* CSS
- subclass, 111
- submit, 318, 320, 321
- submit event, 321, 384, 425
- substitution, 54
- subtraction, 13, 113
- sum function, 5, 78
- summing (exercise), 78, 409
- summing example, 4, 83, 89, 211
- superclass, 111
- survey, 294
- Sussman, Gerald, 202
- SVG, 287–289, 305, 306
- swapping bindings, 424
- swipe, 342
- switch keyword, 34
- symbiotic relationship, 183
- symbol, 106
- Symbol function, 106
- Symbol.iterator symbol, 107
- SymmetricMatrix class, 111
- synchronization, 387, 426
- synchronous programming, 182, 195, 359, 368
- syncState method, 335, 338, 340, 341, 349, 426
- syntax
 - of Egg, 203, 204
 - error, 26, 129, 130
 - expression, 23
 - function, 42, 45
 - identifier, 26
 - number, 12, 165
 - object, 63
 - operator, 13
 - statement, 24, 26, 28–34, 135
 - string, 14
- syntax tree, 204–205, 207, 228–229
- SyntaxError type, 206

T

- tab character, 14, 32
- TAB key, 320
- tabbed interface (exercise), 262, 419
- tabindex attribute, 252, 320, 349
- table (HTML tag), 243, 266, 274, 422
- table example, 417
- tableFor function, 68
- tables, 67, 68, 274
- tags, 221–222, 227, 239, *see also names of specific tags*
- talk, 371, 372, 377–379
- talkResponse method, 380
- talksAbout function, 231
- talkURL function, 383
- Tamil, 87
- tampering, 317
- tangent, 75
- target property, 250
- task management example, 71
- TCP, 220, 221, 311, 373
- td (HTML tag), 243, 274
- Tef, 166
- temperature example, 110
- template, 171, 388, 426
- template literals, 15
- tentacle (analogy), 25, 63, 65
- terminal, 354
- termite, 183
- ternary operator, 18, 20, 209
- test method, 146
- test runners, 132
- test suites, 132
- testing, 125, 132
- text, 14, 221, 222, 227, 229, 295, 305–307, 322, 324, 358, 422
- text field, 257, 318, 319, 322
- text method, 315
- text node, 229, 231, 233, 419
- text wrapping, 305
- text-align (CSS), 243
- textAlign property, 295, 420
- textarea (HTML tag), 260, 318, 322, 327, 330, 425
- textBaseline property, 295, 420
- textContent property, 418, 422
- TEXT_NODE code, 229, 419
- textScripts function, 94, 411
- th (HTML tag), 243

- then method, 186–188, 191, 416
- theory, 133
- this binding, 62, 98–99, 101, 130
- thread, 182, 183, 198, 259
- throw keyword, 135, 136, 139, 141, 413
- tile, 303
- time, 147, 148, 150, 184, 241, 261, 277, 278, 280, 283, 303, 346
- time zone, 150
- timeline, 182, 197, 223, 241, 247, 258
- timeout, 188, 259, 373, 374, 380
- Timeout class, 189
- times method, 269
- timing, 396
- title, 382
- title (HTML tag), 222, 223
- toDataURL method, 344
- toLowerCase method, 62, 243
- tool, 145, 164, 175, 334, 339, 340, 342–344, 347, 350, 357
- tool property, 335
- ToolSelect class, 340
- top (CSS), 240–242, 244
- top-level scope, *see* global scope
- toString method, 99, 100, 103–105, 346, 362
- touch, 255, 334
- touchend event, 255
- touches method, 278
- touches property, 255, 339
- touchmove event, 255, 339, 350
- touchstart event, 255, 337, 339
- toUpperCase method, 62, 132, 243, 362
- tr (HTML tag), 243, 274
- trackKeys function, 282, 285
- transform (CSS), 287
- transformation, 297–299, 308, 420
- translate method, 298, 299
- Transmission Control Protocol, 220, 221, 311, 373
- transparency, 289, 296, 346
- transpilation, 213
- trapezoid, 307, 420
- traversal, 152
- tree, 100, 204, 229
- treeGraph function, 394
- trial and error, 133, 282, 293
- triangle (exercise), 37, 407
- trigonometry, 75, 241

- trim method, 73, 268
- true, 16
- trust, 224
- try keyword, 136, 137, 190, 413, 422
- type, 12, 16, 112
- type attribute, 318, 321
- type checking, 131, 174
- type coercion, 18, 19, 28
- type observation, 392, 401, 403
- type property, 204, 249
- type variable, 131
- typeof operator, 16, 80, 410
- TypeScript, 131–132
- typing, 260
- typo, 129

U

- Ullman, Ellen, xx
- unary operator, 16, 23
- uncaught exception, 138, 188
- undefined, 18, 19, 25, 42, 47, 61, 63, 77, 129, 130, 134
- underline, 237
- underscore character, 26, 35, 98, 151, 157
- undo history, 346, 347
- UndoButton class, 347
- Unicode, 15, 17, 87, 92, 147, 162, 163
- unicycling, 371
- Uniform Resource Locator, *see* URL
- uniformity, 204
- uniqueness, 239
- unit (CSS), 242, 257
- Unix, 366–368
- Unix time, 150
- unlink function, 359, 366
- unshift method, 71
- unwinding the stack, 135
- upcasing server example, 362
- updated method, 378, 381, 425
- updateState function, 336
- upgrading, 169
- upload, 325
- URL, 221, 224, 288, 313, 315, 317, 360, 373, 383
- URL encoding, 314
- url package, 364, 380
- urlToPath function, 364
- usability, 251

- use strict, 130
- user experience, 247, 320, 372, 383
- user interface, 138, 334
- users' group, 371
- UTF-8, 358, 359
- UTF-16, 15, 92

V

- V8, 398
- validation, 134, 140, 203, 277, 321, 378, 379
- value, 12, 186
- value attribute, 318, 322, 324
- var keyword, 25, 43, 76
- variables, *see also* binding
- Vec class, 113, 268, 269, 280, 394, 396, 421
- vector, 394, 400
- vector (exercise), 113, 411
- vector graphics, 295
- verbosity, 46, 182
- version, 169, 222, 312, 357, 398
- viewport, 275–277, 301, 302, 305
- VillageState class, 119
- virtual keyboard, 252
- virtual world, 117, 119, 121
- virus, 224
- vocabulary, 41, 84
- void operator, 26
- volatile data storage, 12

W

- waitForChanges method, 380
- waiting, 184
- walking, 303
- warning, 357
- wave, 271, 280, 281
- web, *see* World Wide Web
- web application, 5, 326, 333
- web browser, *see* browser
- web page, 174
- web worker, 259
- WebDAV, 369
- webgl (canvas context), 289
- website, 224, 225, 313, 353, 369, 371
- WebSockets, 373
- weekDay module, 169–170
- weekend project, 369

- weresquirrel example, 60, 62, 64, 66, 69, 71
- while loop, 4, 30, 32, 53, 160
- whitespace
 - in HTML, 231, 340, 419
 - indentation, 32
 - matching, 147, 162
 - syntax, 204, 206, 214, 417
 - trimming, 73, 268
 - in URLs, 373–374
- Why's (Poignant) Guide to Ruby*, 22
- width property, 350, 423
- window, 250, 255, 258
- window object, 248
- with statement, 131
- word boundary, 151
- word character, 147, 151, 162
- work list, 124, 343
- workbench (exercise), 330, 422
- world, of a game, 265
- World Wide Web, 5, 77, 219, 221, 224, 225, 311
- writable stream, 360–363, 364
- write method, 360, 361
- writeFile function, 359, 361, 425
- writeHead method, 360
- writing code, 6, 117
- writing system, 87
- WWW, *see* World Wide Web

X

- XML, 230, 288
- XML namespace, 288
- xmlns attribute, 288

Y

- yield (reserved word), 26
- yield keyword, 197
- your own loop (example), 95
- Yuan-Ma, 10, 352

Z

- Zawinski, Jamie, 144
- zero-based counting, 56, 61, 150
- zeroPad function, 54
- zigzag shape, 420
- zooming, 305