

# Index

---

- Abstraction, 311
  - cut-and-paste, 162–163, 178
  - for DRY, 30
- Abstraction layers, 164–165
- accept method
  - Friendship, 417–418
  - FriendshipController, 432
- accept\_one\_side method, 418
- Accepted friendships, 412
- Access protection for pages
  - smart way, 152–154
  - stupid way, 150–151
  - testing, 155
- ACID properties, 64
- Action caching, 510
- Action Mailer package, 389
  - configuration, 390–391
  - linking and delivering reminders, 392–395
  - password reminders, 391–392
  - testing reminders, 396–399
- ActionController class, 77
- Active Record
  - avatars, 366, 378
  - console tool, 52–53
  - element access, 185, 291
  - fixtures, 306–308
  - friendships, 415, 418
  - searches, 309–311
  - sessions, 132–133
  - spec, 282
  - user model, 52
  - user specs, 265, 268
- ActiveRecordStore method, 132
- acts\_as\_ferret plugin, 330, 332
- add method, 380
- add\_column method, 62
- add\_to\_base method, 379–380
- AddAuthorizationToken class, 193
- ADDDATE function, 347
- AddUserTimestamps class, 62
- Adjectives for modifiers, 443
- Administration basics in deployment,
  - 511–514
- after\_filter function, 152
- age method, 317–318
- Agile framework, 62
- Ajax comments, 479
  - beginning, 485–486
  - creating, 490–492
  - debugging and testing, 501–504
  - destroying, 492–494
  - new, 486–490

- Ajax comments (*continued*)
  - visual effects
    - cancel button, 499
    - creating, 495–499
    - degrading gracefully, 499–501
- Alphabetical index, 313–316
- Alternate layouts, 29
- and operator, 191–192
- Another RJS Testing System (ARTS), 501
- Apache server, 508
- APIs (Application Programming Interface)
  - Rails, 23–24
  - REST style, 444–445
- Application controller
  - for authorization, 195–197
  - for user specs, 266
- application error page, 514
- Application Programming Interface (APIs)
  - Rails, 23–24
  - REST style, 444–445
- ApplicationController class, 23–24
  - logins, 195–197, 214
  - searches, 336–337
  - user specs, 266–267
- ApplicationHelper module, 167, 197, 211
- Arguments, optional, 93
- around\_filter function, 152
- Arrays
  - associative, 34
  - clearing, 338
  - concatenating, 335
  - friendships, 426
  - Ruby, 115
  - subtraction operator, 291
- ArsDigita, 6
- ARTS (Another RJS Testing System), 501
- assert method, 100
- assert\_email\_field method, 240, 242
- assert\_equal method, 102, 124
- assert\_form\_tag method, 239
- assert\_in\_delta method, 208
- assert\_input\_field method, 239–240
- assert\_length method, 244–245, 299
- assert\_match method, 397
- assert\_password\_field method, 240, 242
- assert\_redirected\_to profile\_for method, 409
- assert\_screen\_name\_field method, 241
- assert\_status method, 419
- assert\_submit\_button method, 239
- assert\_tag method, 105, 109, 148
- assigns function, 102, 108
- Associations, comments, 481–482
- Associative arrays, 34
- Asterisks (\*) in searches, 342
- At sign symbols (@) in ERb, 31–32
- attr\_accessor construct
  - passwords, 231
  - remember me, 185
- Authorization
  - cookies, 193–195
  - login. *See* Login systems
- authorization\_token cookie, 201
- authorize method
  - session variables, 149
  - user specs, 274–275
- authorized? method, 481
- avatar\_tag function, 370
- Avatars, 365
  - deleting, 382–385
  - friendships, 430
  - ImageMagick and convert for, 374–377
  - manipulating, 373–374
  - models, 365–368
  - partials, 371–373
  - saving, 377–378
  - testing, 385–387
  - upload page, 368–371

- upload preparation, 365–366
- validations, 378–382
- barely\_invalid\_string method, 244
- before\_filter method
  - AvatarController, 368
  - CommentController, 483
  - cookies, 196–197
  - email address updates, 229
  - EmailController, 401
  - FaqController, 292
  - FriendshipController, 423, 432
  - logins, 216–217
  - PostsController, 457
  - SpecController, 272–273
  - UserController, 152–154, 284
- begin . . . rescue blocks
  - for browsing, 359–361
  - for search exceptions, 339–341
- belongs\_to method
  - blogs, 447, 454–455
  - comments, 481–482, 484
  - FAQs, 290
  - friendships, 414, 416, 426–427
  - specs, 265, 332
- Bignum class, 189
- blank? method, 287
- Blind\_down effect, 495
- Blind\_up effect, 497
- Blocks, 31, 48, 71–72
- Blog class, 447, 455
- blog\_id attribute, 454
- @blog variable, 468
- blogs. *See* RESTful blog
- Body content in pages, 32
- @body variable, 392, 404
- Boolean contexts, 82
- Boolean functions, 169
- breakup method, 418, 433
- browse page, 343–345
- Browsing for users, 343
  - browse page, 343–345
  - find\_by\_asl, 345–350
  - geographical location, 350
    - GeoData, 352–355
    - local databases, 351–352
    - names, 355–358
    - validation, 358–363
- button\_to function, 81
- button\_to\_function function, 499
- Caching, 510
- Calculations for distances on spheres, 353–355
- Cancel button, 499
- cancel method, 432
- cancel URL modifier, 443
- Capistrano version control system, 511
- capitalize! method, 357
- capitalize\_each! method, 356–357
- Cascading Style Sheets (CSS), 39–41
- Chains of filters, 153
- check\_authorization method, 196–197, 214
- Clark, Mike, 8
- Class functions, 172
- Classes
  - adding methods to, 188–189
  - in modules, 92, 165
  - superclasses, 291
- clear method, 398
- clear\_password! method
  - password updates, 231–232
  - refactoring, 173–176
- Clearing passwords, 173–176
- Clients, HTTP, 77
- CocoaMySQL, 44
- Code duplication. *See* Refactoring
- collection option, 282
- Colons (:) for symbols, 35
- Comment class, 481

- Comments, 26
  - Ajax. *See* Ajax comments
  - controller and preemptive partial, 482–484
  - documentation, 93, 95
  - models and associations, 481–482
  - resources, 480–481
  - routing, 484–485
- comments\_path method, 489
- CommentsController class, 483, 503–504
- Common ids, user specs, 264
- Communities, 303
  - avatars. *See* Avatars
  - browsing. *See* Browsing for users
  - building, 303–304
  - final page, 363–364
  - index action, 308–309
    - alphabetical, 313–316
    - find for, 309–311
    - results, 316–320
    - working with, 311–313
  - pagination, 321–323
  - results summaries, 323–325
  - sample user setup, 304
    - collecting data, 305
    - loading data, 305–307
  - searching. *See* Searching
- CommunityController class, 312
- concat function, 334–335
- Concatenating arrays, 335
- Conditional statements, 82
- confirm function, 383
- confirm option, 382
- connect method
  - blog resources, 446
  - map, 440–441
- console tool
  - for help, 118–120
  - user model validations, 52–53
- Constants, 55–56
- content\_tag function, 269
- content\_type attribute, 381
- Controllers
  - comments, 482–484
  - communities, 303–304
  - controlling our, 255–256
  - FAQ, 292–293
  - generating, 20–22
  - personal profiles, 258–260
  - Posts, 450–454, 456–459
  - site, 22–24, 99–100
  - spec, 266–268, 440–442
  - user registration, 65–66
- convert utility, 374–377
- cookie\_expires function, 206
- cookie\_value function, 206
- Cookies, 181, 192–193
  - authorization, 193–195
  - finding, 195–197
  - logout!, 197–199
  - remember me, 186–192
  - secure, 199–201
  - summary, 201–203
  - testing, 203–212
- cookies variable, 187
- correct\_error\_message function, 245
- correct\_password? function, 236
- correspond method, 401–403, 407
- count method, 474
- Create, Read, Update, Delete (CRUD)
  - operations, 439–441, 448
- create action and method
  - blog posts, 471
  - comments, 490–491
  - Friendship, 417
  - FriendshipController, 423
  - friendships, 415–416
  - PostsController, 451–452, 457–459
  - visual effects, 498, 501

- create\_reminder method, 397
- create\_table method, 413
- CreateBlogs class, 446
- CreateComments class, 480
- CreateFriendships class, 413–414
- CreateGeoData class, 351–352
- CreatePage class, 289
- CreatePosts class, 449–450
- CreateSpecs class, 261
- cross-site scripting (XSS) attacks, 282
- CRUD (Create, Read, Update, Delete) operations, 439–441, 448
- CSS (Cascading Style Sheets), 39–41
- CURDATE function, 347
- Cut-and-paste abstraction, 162–163, 178
- cycle function, 317
  
- Daftuar, Sumit, 7
- darcs version control system, 511
- database.yml file, 45–46, 99–100, 507
- Databases
  - choosing, 44
  - configuration testing, 98–99
  - creating, 45
  - data integrity, 63–64
  - geographic data, 351–352
  - normalization, 414
  - sessions, 132–133
  - user models, 43–46
  - wiping, 51
- date field, 397
- date\_select function, 271
- Dead Ferret, 333–334
- debug function for forms, 75–77
- Debugging
  - Ajax comments, 501–504
  - logins, 135–138
- decline method, 432
  
- def keyword, 23
- Defaults
  - index actions, 24
  - REST tests, 474–476
- defined? function, 250
- delete method
  - avatars, 383–385
  - cookies, 195
  - FriendshipController, 432–433
  - Rake, 307
- delete\_all method, 307
- DELETE requests
  - blogs, 467
  - REST, 440–441
- Deleting
  - avatars, 382–385
  - friendships, 418
- deliver\_friend\_request method, 423–424
- deliver\_message method, 402, 404
- deliver\_reminder method, 391, 395
- Dependent comments, 482
- Deployment, 505
  - administration basics, 511–514
  - minimal production servers, 508–509
  - production mode, 506–508
  - scaling, 509–511
  - software and hardware options, 506
- Design, 1
- destroy method
  - comments, 492–494
  - friendships, 418
  - PostsController, 452, 458
  - visual effects, 498–499
- Development environments, 15
- development.log file, 79, 512
- Development servers, 18–19
- Dictionaries, 34
- display action, 257
- Distances on spheres calculations, 353–355

- do statements, 48, 71
- Documentation comments, 93, 95
- Don't Repeat Yourself (DRY) philosophy, 30
- Double-blind email system, 399
  - correspond function, 401–403
  - email link, 399–400
  - messages, 403–406
  - testing, 406–409
- DRY (Don't Repeat Yourself) philosophy, 30
- Duck typing, 459
- Duplicate blog posts, 471–473
- duplicate? method, 471–473, 481
- Duplicated code. *See* Refactoring
- each method, 314
- edit method
  - FaqController, 292
  - passwords, 236
  - PostsController, 451, 457
  - user specs, 267
- edit page testing, 242–243
- edit\_profile action, 151
- edit URL modifier, 441–442, 467
- elsif keyword, 190
- Email, 389
  - Action Mailer package, 389
    - configuration, 390–391
    - linking and delivering reminders, 392–395
    - password reminders, 391–392
    - testing reminders, 396–399
  - addresses
    - testing, 122, 127–128
    - updating, 226–229, 242–243
    - validating, 57–60, 63
  - double-blind email system, 399
    - correspond function, 401–403
    - email link, 399–400
    - messages, 403–406
    - testing, 406–409
  - email attribute, 46
  - EmailController class, 394, 398, 401–403
  - EmailControllerTest class, 398, 409
  - @emails variable, 398
  - Embedded Ruby (ERb)
    - overview, 27–28
    - templates, 30–31
  - Empty strings, 276, 286–287
  - encode method, 397
  - Encryption of passwords, 61
  - Environment variables, 157–160
  - ERb (Embedded Ruby)
    - overview, 27–28
    - templates, 30–31
  - Error messages
    - avatars, 379–382
    - user registration forms, 82–88
  - error\_messages\_for function, 82, 358, 382
  - errors attribute, 114–115
  - errors object, 237
  - Escape, HTML, 282
  - Exceptions
    - browsing, 359–361
    - raising, 79–80
    - searching, 339–341
  - Exclamation points (!) with functions, 168–169, 194
  - exist? method, 367
  - exists? method
    - Avatar, 366–367
    - Friendship, 416–417
  - Expiration of cookies, 188
  - FAQ (Frequently Asked Questions)
    - adding to hub, 294–298
    - controller, 292–293
    - editing, 293–294
    - model, 288–292
    - testing, 298–299
  - Faq class, 290–291

- FaqController class, 292
- FaqTest class, 299
- Ferret, 327
  - dead, 333–334
  - installing, 330–332
  - searching with, 330–336
- Fielding, Roy, 438
- fieldset tag, 67
- Filters, chains of, 153
- find method
  - blog posts, 473
  - community index actions, 309–311
  - user specs, 267
  - users, 226
- find\_by method, 142
- find\_by\_asl method, 361
  - distance search, 354–355
  - implementing, 345–350
- find\_by\_authorization\_token method, 196
- find\_by\_contents method, 332–336
- find\_by\_email method, 394–395
- find\_by\_first\_name\_and\_last\_name method, 311
- find\_by\_screen\_name method, 53–54
- find\_by\_screen\_name\_and\_password method, 108, 142, 311
- find\_by\_sql method, 311, 349
- find\_by\_user\_id\_and\_friend\_id method, 415–416, 418
- find\_by\_zip\_code method, 352, 354
- first method, 206
- Fixnum class, 188
- Fixtures
  - community data, 305–308
  - loading, 112
- fixtures function, 112
- flash container, 85–88
- Float class, 189
- for element in array do structure, 115
- Foreign keys with friendships, 414
- forget! method, 218–219
- form\_for function
  - error messages, 82
  - password updating, 230
  - user registration, 67–72, 77, 107
- form\_tag function, 328
- Formats, REST style, 444–445
- Forms
  - appearance, 66–70
  - debug for, 75–77
  - error messages, 82–88
  - refactoring, 176–177
  - refinements, 72–75
  - structure, 70–72
- Forwarding, URL. *See* Friendly URL
  - forwarding
- Fragment caching, 510
- Free productivity, 1–3
- Frequently Asked Questions. *See* FAQ (Frequently Asked Questions)
- friend\_id attribute, 426–427
- friend\_request method, 424
- @friend variable
  - friendships, 423, 431
  - UserMailerTest, 407
- Friendly URL forwarding, 156
  - login, 158–161
  - refactoring, 177–179
  - register, 161–162
  - request variable, 156–158
  - testing, 162–163
- friendly\_url\_forwarding\_aux method, 179
- Friendship class, 414, 416–418
- friendship\_status method, 421
- FriendshipController class, 423, 432–434
- FriendshipControllerTest class, 434–435
- FriendshipHelper module, 421

- Friendships, 411
  - management
    - actions, 431–433
    - has\_many, 426–428
    - hub friendships, 428–431
  - models, 411–412
    - abstract, 412–413
    - completing, 417–418
    - overview, 413–415
    - pending friendships, 415–416
    - requests, 416–417
    - testing, 419–420
  - requests, 412, 420
    - controlling, 423–425
    - links, 420–422
    - testing, 433–435
- FriendshipTest class, 419–420
- @from variable, 397
- full\_messages method, 58, 82
- full\_name method, 277
- Full-strength validations, 59–61
- Functional programming, 123
- Functional tests, 97
- Functions
  - class, 172
  - on-the-fly, 108
- generate command, 47, 260–261
- generate script, 20–22
- geo\_data.csv file, 351–352
- Geographical location in browsing, 350
  - GeoData, 352–355
  - local databases, 351–352
  - names, 355–358
  - validation, 358–363
- GET requests
  - blogs, 462, 465
  - REST, 440–441
  - user registration, 77–78
- Global variables, 214
- Graphical representation of people.
  - See* Avatars
- Greenspun, Philip, 6
- Hairy code, 111
- Hansson, David Heinemeier, 52, 439
- Hartl, Michael, 7–9, 511
- has\_many method
  - blogs, 455
  - comments, 481–482
  - friendships, 426–428
- has\_one method, 265, 291, 332, 426, 428, 447
- Hash keys, symbols as, 77
- Hashes
  - cookies, 187–188, 199–200
  - links, 33
  - overview, 34
- Haversine formula, 354
- HeidiSQL, 44
- Help and helper files
  - ApplicationHelper module, 167, 197, 211
  - console tool, 118–120
  - friendships, 421
  - registration, 92–94
  - user information tests, 237–242
- Help page, 28, 30, 32
- helper method, 285
- hide\_edit\_links? function
  - avatars, 372
  - friendships, 428
- @hide\_edit\_links? variable, 300–301
- href option, 500
- HTML escape, 282
- HTML skeletons, 32
- HTTP (hypertext transfer protocol), 77–78, 438–441
- httpd.conf file, 509
- Hubs
  - FAQ for, 294–298
  - friendships, 428–431

- personal profiles, 277–280, 285–288
  - user registration, 89–90
- humanize function, 281
- Hypertext transfer protocol (HTTP), 77–78, 438–441
- if statements, 82
- image\_tag function, 370
- ImageMagick package, 374–377
- Importing projects into RadRails, 16–17
- Incremental development, 104
- Index action and method, 151–152
  - blogs, 460, 469
  - communities, 308–309
    - alphabetical, 313–316
    - find for, 309–311
    - results, 316–320
    - working with, 311–313
  - controllers, 21
  - default, 24
  - FAQ, 294–296
  - personal profiles, 259–260
  - PostsController, 451, 457
- index page
  - avatars, 368
  - blogs, 461, 466
  - content, 26
  - redirection to, 146, 151
  - testing, 155
  - users, 88, 108
- index URL modifier, 442
- Inflectors, 49, 325
- Inheritance, 23
- initialize method
  - Avatar, 366–367
  - Faq, 290–291
  - Message, 401
- Injection attacks, 310
- Inline RJS, 488
- input tags, 72, 105–106, 247, 269
- inserts\_fixtures function, 307
- inspect method, 78–80
- Inspecting variables, 78–80
- Installing
  - Ferret, 330–332
  - Ruby, 14
  - RubyGems, 14–15
- Instance variables, 32
- Instances of classes, 53
- Integration tests, 97, 209–215
- Integrity, user model, 63–64
- Interests, personal. *See* FAQ (Frequently Asked Questions)
- Interpolation, variables, 89
- irb program
  - for console, 52
  - for hashes, 34
- join method, 367
- Join operations, 355
- Key/value pairs
  - cookies, 187–188
  - hashes, 34
- Latitude in sphere distance calculations, 353–355
- Layouts, 28–30
  - alternate, 29
  - ERb, actions, and instance variables, 30–32
  - hashes, 34
  - link\_to, 35–36
  - navigation, 32–34, 37
  - splitting pages, 32
  - style, 36–37
  - symbols, 35
- Leading underscores in filenames, 246
- Left joins, 355

- Length
  - passwords, 120–121
  - screen names, 117–118
- lib directory, 287
- Libraries
  - Active record, 43
  - Ajax, 495
  - Ferret, 330
  - JavaScript, 479, 486
  - ORB, 52
- link\_to function, 33
  - comments, 487
  - communities, 317
  - overview, 35–36
  - style, 36–37
  - symbols and strings in, 154
- link\_to\_remote function, 487–489
- link\_to\_unless\_current function, 37, 91, 148, 320
- Links
  - friendship requests, 420–422
  - navigation, 32–34
  - in registration, 90–94
- Linux
  - Ruby installation on, 14
  - RubyGems installation on, 15
- load\_post function, 483
- load task, 307
- Local databases of geographic data, 351–352
- local error page, 514
- Local requests, 513
- Local variables, 72, 214
- location method, 277
- logged\_in? method
  - cookies, 197
  - refactoring, 164–168
- @logged\_in\_user variable, 400
- logger function, 79
- Logical operators, 191–192, 267–268
- login method, 160–161, 168–169, 178
- login! method
  - cookies, 197
  - refactoring, 168–172
- Login systems, 131
- logging in, 134
  - debugging, 135–138
  - registration, 134–135
  - testing, 142–146, 203–209
  - tracking login status, 134
  - view and action, 138–142
- logging out
  - process, 146–147
  - testing, 147–148, 209
- navigation, 148–150
- page protection, 150
  - smart way, 152–154
  - stupid way, 151
  - testing, 155
- refactoring, 164
  - clear\_password!, 173–176
  - form handling, 176–177
  - friendly forwarding, 177–179
  - logged\_in?, 164–168
  - login!, 168–172
  - logout!, 172–173
  - testing, 179
- remembering in. *See* Remember me feature
- session state, 131–133
- updating, 250–252
- URL forwarding, 156
  - login, 158–161
  - register, 161–162
  - request variable, 156–158
  - testing, 162–163
- logout method, 146
- logout! method
  - cookies, 197–199
  - refactoring, 172–173

- Long output suppression, 119
- Longitude in sphere distance calculations, 353–355
- Magic code, 111
- Magic columns, 61–63
- Magic numbers, 55
- Maintaining session state, 131–133
- Make files, 49–50
- make\_profile\_vars method, 469
- maxlength attribute
  - login tests, 143
  - testing, 107
  - user registration, 72–73
- memcached system, 132, 510
- merge! function, 205, 240
- Message class, 401
- Message digests, 199–201
- message method, 404–405
- Messages, email. *See* Email
- Metaprogramming, 54
- method\_missing function, 56
- Methods
  - adding to classes, 188–189
  - defined, 72
  - parentheses with, 53
- Migrations
  - magic columns, 61–62
  - user model, 46–47
    - raking, 49–51
    - skeleton files, 47–48
- MIME (Multipurpose Internet Mail Extensions), 369
- Minimal production server, 508–509
- Minus signs (–) for newlines, 87
- Mixins, 165
- mod\_proxy\_balance method, 506
- mod\_rewrite method, 505
- Model-view-controller (MVC)
  - architecture, 20
- Models
  - avatars, 365–368
  - comments, 481–482
  - friendships. *See* Friendships
  - user. *See* User models
- Modifiers, URL, 441–443
- Modules, classes in, 92, 165
- Mongrel server, 506, 508–510
- Multipart encoding, 369
- Multipurpose Internet Mail Extensions (MIME), 369
- Mutation operations, 169, 194
- MVC (model-view-controller) architecture, 20
- MySQL, 44–45
- mysqladmin command, 45
- name method, 317
- Named routes, 282–285
- Names
  - conventions, 37–38
  - database, 44
  - screen. *See* Screen names
  - tables, 48–49
- nav\_link function, 93, 95, 320
- Navigation
  - layouts, 32–34, 37
  - testing, 148–150
- Negation operator (!) with functions, 168–169
- New comments, 486–490
- new method
  - comments, 488, 502
  - PostsController, 451, 457
  - visual effects, 495, 497, 500–501
- new\_post\_path method, 460
- new URL modifier, 441–442
- newlines, 87
- Nginx webserver, 506
- nil, meaning of, 153
- nil? method, 81
- Normalization, database, 414

- not keyword, 168, 191–192
- Numeric class, 189
- Object-relational mapping (ORM), 52
- On-the-fly functions, 108
- Operator precedence, 191–192
- Optional arguments, 93
- Optional characters, 36
- or\_else function, 285–287
- or operator precedence, 191–192
- or-equals operator (||), 267–268
- Order of operator execution, 191–192
- ORM (object-relational mapping), 52
- OS environment variable, 376
- OS X
  - Ruby installation on, 14
  - RubyGems installation on, 15
- Outside requests, 513
- Page caching, 510
- Page protection, 150
  - smart way, 152–154
  - stupid way, 151
  - testing, 155
- Pages, splitting, 32
- paginate function, 322, 325, 336–338
- paginated? function, 460
- Pagination
  - communities, 321–323
  - in searching, 336–339
- pagination\_links function, 322–323
- Paginator class, 336–337
- Pairs of users in friendships, 411–412
- param\_posted? function, 176, 266
- params variable, 75–77
- Parentheses (), optional, 53
- Partials, 245
  - advanced, 247–250
  - avatar, 371–373
  - comment, 482–484
  - login and register views, 250–252
  - simple, 246–247
  - spec, 281–282
- password attribute, 46, 231
- password\_confirmation attribute, 231
- password\_errors function, 236–237
- password\_field function, 248
- Password reminders
  - Action Mailer package, 391–392
  - linking and delivering, 392–395
  - testing, 396–399
- Passwords
  - clearing, 173–176
  - databases, 45–46
  - encryption, 61
  - length, 120–121
  - login, 142
  - partials for, 247–252
  - testing, 120–121, 126–127
  - updating, 229–237, 242–243
  - user registration, 72
  - validating, 63
- Paths for comments, 485
- Pending friendships, 412, 415–416
- Percent symbols (%) in searches, 309
- Personal profiles, 255
  - controller for, 258–260
- FAQ
  - adding to hub, 294–298
  - controller, 292–293
  - editing, 293–294
  - model, 288–292
  - testing, 298–299
- hub main content, 285–288
- hub view for, 277–280
- named routes and URLs, 282–285
- public-facing, 299–302
- spec box for, 280–282
- URLs, 256–258

- user specs, 260
  - editing, 266–272
  - models, 260–265
  - protecting, 272–273
  - testing, 273–276
- PHPMyAdmin program, 44
- Platforms, 13–14
- pluralize function, 325
- Post class, 454, 473, 482
- post function, 107
- POST requests
  - blogs, 462, 465, 467
  - REST, 440–441
  - user registration, 77–78, 80–82
- post\_url method, 454
- PostgreSQL, 44
- Posts, blog, 447–450
  - creating, 461–463
  - duplicate, 471–473
  - editing, 467–468
  - publishing, 468–471
  - routing, 455–456
  - showing, 463–467
- posts\_path method, 459, 462
- posts URL modifier, 443
- PostsController class, 450–454, 456–459
- PostsControllerTest class, 473–477
- Pound server, 506
- Precedence of operators, 191–192
- Preemptive partials, 482–484
- Pretty URLs, 25
- Prochazka, Aurelius, 5–8
- Production environment, 506
- Production mode, 505–508
- Production servers, minimal, 508–509
- Productivity, free, 1–3
- profile\_for function, 284, 312, 402, 409, 483
- profile\_url function, 284, 317
- ProfileController class, 370–371, 422
- ProfileHelper class, 423
- Profiles. *See* Personal profiles
- protect method, 152–153
  - comments, 483
  - login forwarding, 158–159
  - in refactoring, 165–166
  - user specs, 272–273
- protect\_blog method, 458–459
- Protecting
  - pages, 150
    - smart way, 152–154
    - stupid way, 151–152
  - testing, 155
  - user specs, 272–273
- public error page, 514
- Public-facing profiles, 299–302
- Publishing blog posts, 468–471
- PUT requests
  - blogs, 467
  - REST, 440–441
- Question marks (?)
  - boolean functions, 169
  - variables, 300
- radio\_button function, 271
- radio\_button\_tag function, 345
- RadRails IDE, 15–17
- Rails
  - API, 23–24
  - scaffolds, 447–448
  - URLs, 24–25
  - views, 26–28
- rails program, 15–17
- rails\_space\_production database, 507
- raise function, 79–80
- Rake tasks, 306–307
- rake utility, 99

- Rakefiles
  - loading data, 306–307
  - migrations, 49–51
- Ranges, 55
- read\_fixture method, 397
- Rebuilding indexes, 334
- @recipients variable, 397
- RecordInvalid exception, 194
- redirect\_to method, 161
- redirect\_to\_forwarding\_url method, 177
- redirect\_url method, 161
- Redirection, return statement after, 151–152
- Refactoring, 131
  - login system, 164
    - clear\_password!, 173–176
    - form handling, 176–177
    - friendly forwarding, 177–179
    - logged\_in?, 164–168
    - login!, 168–172
    - logout!, 172–173
    - testing, 179
  - remember me feature, 215–216
    - completed, 222–223
    - forget!, 218–219
    - remember!, 216–218
    - remember me checkbox, 219–220
    - ternary operator, 220–221
- Reflection, 54
- register action and method
  - complete, 88
  - in refactoring, 171, 175–178
  - User controller, 66, 78, 91
- Registering users. *See* User registration
- Regular expressions
  - testing, 122–128
  - in validations, 59–60
- Related functions, 38
- Relationships in friendships, 411–412
- reload! function, 56
- Remember me feature, 181–182
  - cookies, 192–193
    - authorization, 193–195
    - finding, 195–197
    - logout!, 197–199
    - secure, 199–201
    - summary, 201–203
  - refactoring, 215–216
    - completed, 222–223
    - forget!, 218–219
    - remember!, 216–218
    - remember me checkbox, 219–220
    - ternary operator, 220–221
  - remember\_me attribute, 184–186
  - remember me box, 182–184
  - remember me cookie, 186–192
  - testing
    - integration, 209–215
    - login, 203–209
    - logout, 209
  - remember! method, 216–218
- RememberMeTest class, 210–212
- remind method, 394–395
- reminder method, 391–392
- Reminders
  - Action Mailer package, 391–392
  - linking and delivering, 392–395
  - testing, 396–399
- remote\_form\_for method, 489
- replace function, 356
- Representational State Transfer (REST) style.
  - See* REST (Representational State Transfer) style
- request method, 416, 420, 423
- request\_uri function, 157
- request variable, 156–158
- Requests
  - blogs, 467

- friendship, 412, 416–417, 420
  - controlling, 423–425
  - links, 420–422
  - testing, 433–435
- HTTP, 77–82
- REST, 440–441
- rescue blocks
  - for browsing, 359–361
  - for search exceptions, 339–341
- Resources
  - comments, 480–481
  - REST, 438
  - RESTful blog, 445–447
- resources method, 440–442, 446
- respond\_to method, 441, 452–453
- Responders, HTTP requests, 441
- REST (Representational State Transfer) style, 437–438
  - CRUD with, 439–441, 448
  - formats, 444–445
  - semicolons in, 442–444
  - URL modifiers, 441–442
- RESTful blog, 437–438
  - comments. *See* Ajax comments
  - managing, 459–461
  - model connections, 454–455
  - posts
    - creating, 461–463
    - duplicate, 471–473
    - editing, 467–468
    - publishing, 468–471
    - routing, 455–456
    - showing, 463–467
  - Posts controller, 456–459
  - scaffolds, 445
    - posts, 447–450
    - Posts controller, 450–454
    - resources, 445–447
  - testing, 473–477
- Results summaries for searches, 323–325
- return keyword
  - need for, 93
  - after redirection, 151–152
- reverse! function, 169
- rhtml files, 22, 25
- RJS (Ruby JavaScript), 479, 489, 495–496
- RMagick interface, 374
- Rolled back transactions, 417
- Roundy, David, 511
- Routes
  - changing, 25–26
  - personal profiles, 282–285
- Routing
  - blog posts, 455–456
  - comments, 484–485
- Ruby
  - arrays, 115
  - installing, 14
- Ruby JavaScript (RJS), 479, 489, 495–496
- RubyGems, 14–15
- Salted hashes, 200
- Sandbox, 512
- sanitize method
  - blog posts, 465
  - FAQs, 297–298
  - spec, 282
- save method, 377–378
- save! method, 194
- Saving avatars, 377–378
- scaffold command, 448
- scaffold\_resource, 449–450, 473
- Scaffolds, 3
  - Rails, 447–448
  - RESTful blog, 445
    - posts, 447–450
    - Posts controller, 450–454
    - resources, 445–447
- Scaling, 509–511

- schema\_info table, 51
- screen\_name attribute, 46
- Screen names
  - hashing, 200
  - length, 117–118
  - regular expressions for, 60
  - testing, 117–118, 124–126, 128
  - UserMailer, 392
  - validating, 60, 63
- Searching, 327
  - exceptions in, 339–341
  - with Ferret, 330–332
  - with find, 309–311
  - with find\_by\_contents, 332–336
  - pagination in, 336–339
  - testing, 341–343
  - views in, 328–330
- Secure hashing algorithms, 200–201
- Security. *See* Cookies; Login systems
- Self-handling forms, 77
- self keyword
  - eliminating, 217
  - friendships, 416
  - local variables, 174
  - need for, 170, 172
- Semicolons (;)
  - edit modifier, 467
  - in URLs, 442–444
- @sent\_on variable, 397
- server\_settings, 390
- Servers
  - development, 18–19
  - HTTP, 77
  - minimal production, 508–509
- Session state, maintaining, 131–133
- session variable
  - debugging with, 135–138
  - login status, 131–132, 134
- set\_table\_name function, 49
- setup\_friends method, 423, 431, 433
- SHA1 secure hashing algorithm,
  - 200–201
- Shared-nothing architectures, 509
- Short-circuit evaluation, 191, 268
- show method
  - blogs, 469
  - email, 400
  - PostsController, 451, 457
  - profiles, 257
- Simple Mail Transfer Protocol (SMTP), 390
- Site controller
  - generating, 20–22
  - layouts, 29–31, 67
  - overview, 22–24
  - for routes, 25–26
  - tests, 99–100
    - functional, 104
    - nontrivial, 100–103
    - overkill, 103
- Site documentation, 95
- SiteController class, 22–23, 31, 99, 101
- SiteControllerTest class, 100–101
- Size, avatars, 375
- size attribute
  - avatars, 381
  - login tests, 143
  - testing, 107
  - user registration, 72
- Skeletons, HTML, 32
- SMTP (Simple Mail Transfer Protocol), 390
- Spec box, 280–282
- Spec class, 262, 265
- @spec variable, 274
- SpecControllerTest class, 273–274
- Specs, user. *See* User specs
- Specs controller, 440–442
- Sphere distance calculations, 353–355
- split function, 312–313

- Splitting
  - pages, 32
  - strings, 312–313
- sprintf function, 118
- SQL injection attacks, 310
- Strings
  - empty, 276, 286–287
  - splitting, 312–313
  - vs. symbols, 154–155
- Strong mutation functions, 194
- Style
  - development, 38–41
  - layouts, 36–37
- stylesheet\_link\_tag function, 39–41
- @subject variable, 397
- Subtraction operator for arrays, 291
- Subversion version control system,
  - 331, 511
- successful\_conversion? function, 377–379
- super command, 291
- Superclasses, 291
- Suppressing long output, 119
- Symbols
  - as hash keys, 77
  - overview, 35
  - vs. strings, 154–155
- system function, 375–376
  
- Table names, 48–49
- tag function, 269
- tag URL modifier, 443
- tail command, 513
- TDD (Test-Driven Development), 98
- Templates, ERb, 30–31
- Ternary operator, 220–221
- test\_about method, 101
- test\_accept method, 419
- test\_blank method, 276
- test\_breakup method, 419
- test\_catch\_blog\_id\_mismatch method, 477
- test\_correspond method, 408–409
- test\_create method, 434–435
- test\_create\_comment method, 503
- test\_create\_post method, 475–476
- test\_delete\_comment method, 503–504
- test\_destroy\_post method, 476
- Test-Driven Development (TDD), 98
- test\_edit\_page method, 242–243
- test\_edit\_success method, 274
- test\_email\_maximum\_length method, 127
- test\_email\_with\_invalid\_examples method,
  - 123–124, 127–128
- test\_email\_with\_valid\_examples method,
  - 122, 127
- Test environment, 506
- test\_get\_edit method, 476
- test\_get\_index method, 475
- test\_get\_new method, 475
- test\_help method, 101
- test\_index method, 103
- test\_index\_authorized method, 155
- test\_index\_unauthorized function, 155
- test\_login\_failure\_with\_nonexistent\_screen\_name method, 145
- test\_login\_failure\_with\_wrong\_password method, 145
- test\_login\_friendly\_url\_forwarding method,
  - 162–163, 178
- test\_login\_page method, 143–144, 203
- test\_login\_success method, 144, 168, 204, 206
- test\_login\_success\_with\_remember\_me method, 207
- test\_logout method, 148, 168, 209
- test\_max\_lengths method, 299
- test\_message method, 408
- test\_navigation\_not\_logged\_in method,
  - 148–149
- test\_new\_comment method, 503
- test\_password\_length\_boundaries method, 245

- test\_password\_maximum\_length method, 120–121, 127
- test\_password\_minimum\_length method, 120, 126
- test\_password\_reminder method, 399
- test\_register\_friendly\_url\_forwarding method, 163, 178
- test\_registration\_failure method, 109–110, 175–176
- test\_registration\_page method, 104–106, 241–242
- test\_registration\_success method, 107, 135, 167–168
- test\_remember\_me method, 211–213
- test\_remind method, 408
- test\_reminder method, 396–397
- test\_request method, 419
- test\_screen\_name\_length\_boundaries method, 245
- test\_screen\_name\_maximum\_length method, 118, 126
- test\_screen\_name\_minimum\_length method, 117, 125–126, 243
- test\_screen\_name\_with\_invalid\_examples method, 124–125, 128
- test\_screen\_name\_with\_valid\_examples method, 124, 128
- test\_should\_create\_post method, 474
- test\_show\_post method, 476
- test\_truth method
  - RememberMeTest, 211
  - SiteControllerTest, 100
  - UserTest, 111
- test\_unauthorized\_delete\_comment method, 504
- test\_unauthorized\_redirected method, 477
- test\_uniqueness\_of\_screen\_name\_and\_email method, 116, 125
- test\_update\_post method, 476
- test\_user\_invalidity method, 113–114, 125
- test\_user\_validity method, 113, 125
- Tests, 97
  - avatars, 385–387
  - blogs, 473–477
  - comments, 501–504
  - database configuration, 98–99
  - double-blind email system, 406–409
  - FAQ, 298–299
  - friendship models, 419–420
  - friendship requests, 433–435
  - login
    - invalid, 145–146
    - valid, 142–145
  - logout, 147–148
  - navigation, 148–150
  - page protection, 155
  - password reminders, 396–399
  - philosophy, 98
  - reflections on, 214–215
  - remember me feature
    - integration, 209–215
    - login, 203–209
    - logout, 209
  - search, 341–343
  - site controller, 99–100
    - nontrivial, 100–103
    - overkill, 103
  - testing, 212–214
  - user information updates, 237
    - advanced, 243–245
    - edit page, 242–243
    - helpers, 237–242
  - user models, 111–113
    - console, 118–120
    - password length, 120–121
    - regular expressions, 122–128
    - running, 128–129
    - screen name length, 117–118

- uniqueness, 115–117
- validation, 113–115
- user registration, 103
  - additional, 111
  - basic, 104–107
  - functional, 104
  - running, 110
  - successful, 107–108
  - unsuccessful, 108–110
- user specs, 273–276
- text\_area function, 293–294
- text\_field\_for function, 268–270
- text\_field\_tag function, 329, 345
- TextMate IDE, 15
- Thomas, Dave, 8, 200
- thumbnail\_tag method
  - avatars, 370
  - comments, 483
- thumbnail\_url method, 366, 430
- time\_ago\_in\_words function, 465
- Time classes, 189
- Timestamps, 62, 207–208
- @title variable, 394–395
- Titles in pages, 32
- tmp directory, 385
- to field, 397
- to\_i function, 337
- to\_s function, 206
- to\_xml method, 453
- Tracking login status, 134
- Transactions, friendships, 416–417
- try\_to\_login method, 144, 149,  
204–205, 275
- try\_to\_update method, 235
- Underscores (.) in filenames, 246
- Uniform Resource Identifiers (URIs), 157
- uniq! function, 334–335
- Uniqueness, testing, 115–117
- Unit tests, 97
- update method, 452, 458
- update\_attributes method, 227–228
- Updating user information. *See* User  
information updates
- upload method, 369, 373–374
- Upload page, 368–371
- uploaded\_file function, 386
- URIs (Uniform Resource Identifiers), 157
- URL forwarding, 156
  - login, 158–161
  - refactoring, 177–179
  - register, 161–162
  - request variable, 156–158
  - testing, 162–163
- url method, 366
- URLs
  - blogs, 462
  - forwarding. *See* URL forwarding
  - modifiers, 441–442
  - personal profiles, 256–258, 282–285
  - pretty, 25
  - REST, 438–439
  - semicolons in, 442–444
  - UserMailerTest, 408
- Use default location option, 17
- User class, 426–428, 447, 482
- User controller, 65–66
- user\_id field, 261, 265
- User information updates, 225–226
  - email addresses, 226–229
  - partials, 245
    - advanced, 247–250
    - login and register views, 250–252
    - simple, 246–247
  - passwords, 229–237
  - testing, 237
    - advanced, 243–245
    - edit page, 242–243
    - helpers for, 237–242

- User models, 43
  - data integrity, 63–64
  - database for, 43–46
  - migrations, 46–47
  - raking, 49–51
  - skeleton files, 47–48
- tests, 111–113
  - console, 118–120
  - password length, 120–121
  - regular expressions, 122–128
  - running, 128–129
  - screen name length, 117–118
  - uniqueness, 115–117
  - validation, 113–115
- validations, 51–52
  - console, 52–53
  - example, 56–57
  - full-strength, 59–61
  - improving, 57–58
  - magic columns, 61–63
  - simple, 54–56
- User profiles. *See* Personal profiles
- User registration, 65, 134–135
  - example user, 95
  - hub stub, 89–90
  - linking in, 90–94
  - page for, 77–82
  - register function, 88
- tests, 103
  - additional, 111
  - basic, 104–107
  - functional, 104
  - running, 110
  - successful, 107–108
  - unsuccessful, 108–110
- views
  - appearance, 66–70
  - debug for, 75–77
  - error messages, 82–88
  - partials, 250–252
  - refinements, 72–75
  - structure, 70–72
- User specs, 260
  - editing, 266–272
  - models, 260–265
  - protecting, 272–273
  - testing, 273–276
- user variable, 84, 142
  - @user variable, 84
    - email, 404
    - friendships, 423, 431
    - UserMailer, 392
    - UserMailerTest, 407
    - working with, 142
- UserMailer class, 391–392, 404–405, 424
- UserMailerTest class, 396–397, 407–408
- users function, 114
- users.yml file, 112–113, 341–342, 406–407
- valid? method
  - messages, 401–402
  - passwords, 236
- valid\_file? method, 381
- valid\_float? method, 361
- valid\_input? method, 360–362
- validate method, 57–58
- validates\_format\_of method, 59, 122, 264
- validates\_inclusion\_of method, 263–264
- validates\_length\_of method, 55, 59, 121, 263
- validates\_uniqueness\_of method, 55
- Validations
  - avatars, 378–382
  - basic, 113–115
  - browse, 358–363
  - user model, 51–52
    - console, 52–53
    - example, 56–57

- full-strength, 59–61
  - improving, 57–58
  - magic columns, 61–63
  - simple, 54–56
- Variables
- inspecting, 78–80
  - instance, 32
  - interpolation, 89
  - local and global, 72, 214
- VC caching, 510
- Version control systems, 331, 510–511
- Vertical bars (|)
- links, 33
  - logical operator, 267–268
- Views
- login, 138–142, 250–252
  - personal profiles, 277–280
  - Rails, 26–28
  - search, 328–330
  - user registration
    - appearance, 66–70
    - debug for, 75–77
    - error messages, 82–88
    - refinements, 72–75
    - structure, 70–72
- visual\_effect method, 495
- Visual effects, Ajax
- cancel button, 499
  - creating, 495–499
  - degrading gracefully, 499–501
- Weak mutation functions, 194
- Web services with REST, 438
- Weblogs. *See* RESTful blog
- Wildcard symbols in searches, 309, 342
- Windows
- Ruby installation on, 14
  - RubyGems installation on, 14–15
- Wiping databases, 51
- xhr method, 502–504
- XHR object, 486, 489
- XHTML, 39
- XMLHttpRequest object, 486, 489
- XSS (cross-site scripting) attacks, 282
- YAML, 46
- Yellow fade technique, 497
- .yml configuration files, 46
- ZeroDivisionError exception, 79