

Index

Symbols/ Numerics

-> (arrow operator), 153
[] (brackets), 152–153
: (colon), 146, 624
/* and */ (comments), 453
{ } (curly braces), 140
- (leading dash), 146
/ (path separator), 649
+ (plus) icon
 Model editor's use of, 680, 682
 Property List editor's use of, 246, 595, 596
/tmp directory, 477, 649
@class keyword, 506
@dynamic statements, 690
@end compiler directive, 143, 147, 149
@implementation directive, 141, 148–149
@interface directive, 141, 143, 144, 148
@private directive, 141
@property declarations, 155–156, 157, 336–337
@protected directive, 141
@public directive, 141
@selector directive, 507, 508
@synthesize statements
 adding, 478
 generating accessor methods, 156–157, 337, 487, 488
 synthesizing hotels property, 707
 using in implementation block, 155
pragma mark statement, 339
#import statements, 506

A

ABMutableMultiValueRef properties, 779
ABNewPersonViewController, 774
ABPeoplePickerNavigationController class, 774–775
ABPersonViewController class, 774
ABRecordGetRecordType function, 776
ABRecordRef, 776
ABUnknownPersonViewController, 774
accelerometer
 accessing, 814–815
 basing response on timestamp, 818
 detecting motion with, 15–16, 38
 illustrated, 812
 respondToShake method, 818–821
 simulating, 70
 uses for, 812–813
accessor methods
 about, 153–154
 creating, 336–337
 generating with @synthesize statements, 156–157, 337, 487, 488
 getting data with, 158–162
 using properties with, 162–163
accessory
 cell architecture for, 460
 defining cell's type of, 463
accuracy, 553, 808–809
action sheets, 87

ad hoc provisioning profile, 96, 105
AddHotelController class
 addHotelController:didFinishWithSave: method, 765–766
 adopting ABPeoplePickerNavigationControllerDelegate protocol, 775–776
 creating view controllers, 755–757
 delegation code for, 768–769
 didFinishWithSave: method, 765–766
 entering hotel data for, 741–743
 outlets for, 750, 751
 viewDidLoad method, 763–764
AddHotelController.h file, 747–749, 763–765
AddHotelController.m file
 dismissing modal view controller, 757–763, 765–766, 783
 implementing saveToContacts method, 783–785
Address Book
 ABPeoplePickerNavigationController interface, 774
 adding contacts to Hotels list, 773, 782–785
 adding hotels to, 741, 782–785
Address Book objects, 776
canceling lookup in, 782
database interaction with, 776

- Address Book (*continued*)
 - hotel names in, 778–779
 - saving addresses in, 782, 783–785
- AddressBook framework, 782
- AddressBookUI framework, 782
- addresses
 - coding street, 780–781
 - creating method to find, 559–562
 - forward geocoding with, 559
 - retrieving coordinates for, 570–574
 - reverse geocoding to find, 555
 - saving to contact, 783–786
- adopting protocols, 260–261, 630–632
- advertising, 124, 131–132
- airplane mode, 545, 788, 789
- alert views, 87
- alerts
 - displaying, 492–493
 - Internet failure, 545, 651
 - terminating applications for, 280–281
 - using offline, 490–491
- alloc method
 - function of, 150, 195
 - memory management rules for, 225–226
- amEditing state, 759, 760–761
- animation
 - adding to Photo view, 366–375
 - background color and timing intervals for, 315–316
 - connecting timer to, 316–317
 - creating timers for images, 315
 - delay and display values for, 820–821
 - hiding view for, 821
 - listing for falling words, 312–315
 - shaking device to begin, 818–821
 - speed slider for, 335, 340–344, 345–347
 - testing, 318, 332–334
 - text field connections to code, 347–349
 - UIView class methods used for, 317
- Annotation class
 - code listing for, 672
 - generating .h and .m files for, 689–690
 - loading objects in Map.m of, 711
 - managed object model for, 672–676
- Annotation entity
 - adding, 680
 - adding relationship to, 682–686
 - inverse relationship for, 684–686
 - latitude, longitude attributes, 681–682
- Annotation.h file, 689–691
- Annotation.m file, 689–691
- annotations
 - about, 540
 - adding dynamically, 661–662
 - creating, 547–550
 - displaying, 550–551
 - doFind: modifications for, 664–665
 - getting with MapController.m, 662–663
 - illustrated, 552
 - marking new locations with, 568–569
 - placemarks, 557
 - refreshing, 662–663
 - removing, 568, 666–667
 - viewDidLoad
 - modifications in MapController, 663–664
- API key, 573
- App Purchase feature, 125–128
- App Store. *See also* developing applications; distributing applications
 - about, 1
 - App Purchase feature versus, 127
 - application standards for, 114–115
 - banking information for, 112
 - competition on, 117
 - customer reviews in, 129
 - distributing applications via, 46–47
 - listing applications on, 19
 - managing iTunes Connect apps, 106–108
 - marketing products on, 118–120
 - provisioning applications for, 104–106
 - sending metadata and artwork to, 109–112
 - submitting information to, 108
 - uploading application to, 112–114
- Apple. *See also* App Store Human Interface Guidelines, 475, 514
 - iPhone application styles, 20

- registering with, 43–50
- requires signature
 - authentication, 95
- standards for
 - applications, 39–41, 114–115
- using trademarks from, 115
- Apple Developer
 - Connection Reference Library, 307
- Apple Developer Forums
 - Beta, 53
- Apple device UDID, 97
- Apple ID
 - creating, 44, 46
 - using in development
 - provisioning, 97, 101–102
- application delegate. *See* delegates
- application level, 189
- Application Loader, 113
- `applicationDidBecomeActive`: method, 281
- `applicationDidFinishLaunching`: method
 - adding last view code to, 479–480
 - responding to events, 281
 - triggering application delegate with, 352
 - using stored data, 488–489
- applications. *See* also developing applications;
 - distributing applications
 - accelerometer events in, 815
 - accessing media from, 370–375
 - ads in, 124, 131–132
 - analytics for, 124
 - Apple categories of, 20
 - building and running in SDK, 65–67
 - Cancel and Save buttons for, 745, 763–765
 - can't access current location, 809–811
 - categorizing, 20, 119
 - characteristics of great, 12–14
 - configuring property lists, 484–487
 - defining scope of, 427–432
 - designing, 25–27, 36–38
 - determining audience for, 121
 - device-guided design of, 14, 17–18, 38–39
 - digital signatures for, 95–97
 - direct manipulation and feedback in, 33
 - dismissing keyboard in, 761–762
 - distributing via App Store, 46–47
 - enabling access to stored data, 486
 - exploiting iPhone features in, 14–16
 - finding memory leaks, 212–215, 405–407, 408–413
 - focusing content of, 27–28
 - gestures for, 16, 32
 - getting location and heading events for, 808
 - icon added to, 376–377
 - impact of battery life on, 18
 - implementing user preferences, 482–487
 - including Web page links, 128
 - increasing functionality of, 535
 - interface consistency
 - across, 31–32
 - Internet access with, 15, 37
 - iTunes affiliate links on, 128
 - knowing user's location, 15, 37
 - knowing where you are in task, 28
 - loading asynchronously, 632–634
 - localizing, 36, 436, 452–455
 - low-memory warnings in, 282–283
 - making function obvious in, 32–33
 - marketing, 118–120
 - memory management
 - rules for, 225, 284
 - migrating to Core Data schemas, 672–673
 - modal view controllers in, 756–763, 765–766
 - modeling on real world tasks, 29–30
 - opening in New Project window, 59–60
 - possibilities for iPhone, 1–2
 - preferences used in, 490–493
 - pricing, 120–121
 - providing quality information, 36
 - provisioning for App Store, 104–106
 - reading preference settings in, 487–489
 - responding to
 - interruptions, 280–281
 - restoring state of, 479–481
 - running in Instruments, 402–405

- applications (*continued*)
 - saving state for, 475–479
 - screen shots of, 111–112
 - selecting knowledge base for, 30
 - selling, 112
 - Settings bundle added to, 327–328
 - simulating, 70
 - speed and performance of, 35
 - state information for
 - interface, 758–760
 - submitting App Store information, 108
 - submitting icons for, 111
 - supplying default data for, 769–771
 - templates for, 59, 274–277
 - terminating, 279–280
 - tracking orientation and motion, 15–16, 38
 - turning off autocorrection for, 567, 772
 - uninstalling from iPhone Simulator, 69
 - updating for attention, 131
 - uploading to App Store, 112–114
 - URL loading system for, 635–636
 - windows in, 82–83
- applicationWillResignActive: method, 281
- applicationWillTerminate: method calling, 475
- receiving message from, 283
- saving current state and unsaved data to disk, 280
- saving section and row information in, 477, 479
- architecture
 - cell's accessory, 460
 - extending with
 - composition techniques, 528–530
 - program, 4
 - RoadTrip's, 433–436
- arguments
 - array elements as, 231
 - assigning to instance variable, 159
 - designing for generic `WebViewController`, 616–617
 - specifying in methods, 146, 624
 - syntax for multiple, 146
- arrays. *See also* mutable arrays
 - about, 229, 230–231, 606–607
 - autorelease pool as, 223
 - changing from dictionary to, 592–593
 - counting elements in, 607
 - dictionaries versus, 236, 608
 - fast enumeration in, 232–233, 607–608
 - fixed, 230, 233
 - holding dictionary nodes in, 329
 - multidimensional, 234–235
 - mutable, 230
 - as property list object, 240
 - releasing and de-allocating, 231, 233
 - saving and restoring, 237
 - using C, 234
- `arrayWithContentsOfURL`: method, 637
- arrow operator (`->`), 153
- artwork
 - licensing to use App Store, 107
 - submitting to App Store, 111–112
- asynchronous loading
 - alerts for Internet failures during, 651
 - Car Information modifications for, 658–659
 - Car Servicing modifications for, 655
 - for data and files, 632–634
 - implementing `Sight` for, 647–651
 - making URL connections for, 639–640
 - modifying Weather app for, 659–660
- attributes
 - adding latitude and longitude, 681–682
 - configuring Table view, 445
 - Core Data entity, 676–682
 - displaying window, 276–277
- Attributes Inspector
 - formatting labels with, 743, 744
 - opening, 299
- audio, 16
- authentication
 - handling challenges to, 645–646
 - signature, 95
 - synchronous loading and, 637
 - URL, 636–637
- autocorrection, 567, 772
- autorelease pools, 222–225, 272
- axes of accelerometer, 812, 815–817

B

Back button
 function of, 469
 hiding, 515–518
 illustrated, 471

background color, 315–316

banking information
 required for sales, 112
 uploading, 112–113

base class, 175

battery life, 18, 39

behavior
 adding with model
 objects, 90
 customizing with `UIKit`
 framework, 93
 design patterns adding,
 90–92
 subclassing to add, 90, 93

blog publicity, 129–130

bounce scroll, 445

brackets (`[]`), 152–153

branding, 131

breakpoints
 activating from Debugger
 strip, 389
 setting, 386, 388, 394–396

bugs, 382–385

Build and Analyze feature,
 212–215, 396–399

build configuration, 65

Build Results window
 compiling code while
 open, 385
 viewing errors in, 382, 384

bundles
 adding to project,
 327–328, 483–484
 opening, 520–521
 Xcode, 244, 458–459, 605

buttons
 Back, 469, 471, 515–518
 Cancel, 567, 745, 763–764

configuring attributes for,
 296, 297, 745

connecting to action
 method, 749–754

deleting from template
 code, 716

Find, 561–562

Get from Contacts, 748,
 773

hiding, 515–518

Info, 296–299

Light Info, 361, 363, 364

Locate, 552–554

Save, 745, 763–764, 765

sending action with Done,
 299–302

simulating events of, 288

Sleep/Wake, 281

tapping Cancel, 567

C

C language, 137, 234

C++ language, 137

calendar alerts, 280–281

Call states, 68

callouts, 551

camera, 16

campground information,
 430

Cancel button, 567, 745,
 763–764

cancel messages, 640

cancel: method, 764

cancelConnection
 messages, 641

canceling
 ABPeoplePicker
 Navigation
 Controller, 774

Address Book lookup, 782

searches, 566, 567

URL requests and
 connections, 640, 641

Car Model Information
 feature
 data source for, 496
 illustrated, 497
 implementing
 initWithTrip:
 method, 521
 using single view
 controller with, 498

Car Servicing Information
 feature, 250
 data source for, 496
 downloading Web data
 for, 632–634
 illustrated, 497
 implementing, 523–527
 using single view
 controller with, 498

CarInformation object,
 507, 518–523

CarInformation.h file,
 522, 658

CarInformation.m file,
 522, 658–659

CarServicing object, 507,
 518–519

CarServicing.h file,
 525–526, 655

CarServicing.m file,
 524–526, 655, 656–658

case sensitivity, 3, 142

categories
 about, 264
 Apple’s application, 20
 class extensions and,
 603–604
 defining informal
 protocols with, 267
 importance of application,
 119
 uses for, 267, 603

cell objects
 about, 459
 adding subviews to
 content view, 460

- cell objects (*continued*)
 - architecture of, 460
 - creating for table views, 461–466
 - custom subclasses for, 461
- cells
 - cell identifiers for, 462–463
 - defining accessory type of, 460, 463
 - displaying `Sight` object, 612–613
 - setting label properties for, 464–465
- centering map on location, 542–544
- circular dependencies, 506
- class diagram, 175–176
- class extensions, 603–604
- class interface. *See* interface files
- class methods, 151, 223–225
- classes. *See also* subclasses; superclasses
 - accessing properties within, 157
 - adding methods as categories, 264–266
 - adopting formal protocols, 260–261, 630–632
 - allocating objects for new, 149–152
 - categories extending, 267
 - class clusters, 152, 198
 - class diagram, 175–176
 - Core Data Stack, 693, 700
 - declaring class interface, 139, 141, 143–147
 - delegation in, 261–263
 - deriving from
 - `NSManagedObject`, 670
 - designated initializer for, 152, 204
 - extending, 603–604
 - Foundation container, 229–230
 - Foundation utility, 228–229
 - generating `Managed Object`, 686–688
 - header files for, 307
 - implementing, 139, 141, 148–149
 - informal protocols for, 258
 - inheritance in, 164–165
 - instantiating, 145
 - introducing, 137–139
 - naming custom, 90
 - naming entities versus, 676
 - `NSObject`, 144, 181, 220
 - Objective-C versus C++, 137
 - objects as instances of, 138
 - parts of, 139
 - regenerating Core Data, 692
 - sending messages to, 151
 - super- and subclasses, 175–176
- Classes folder, 447, 499
- cleaning up Undo manager, 730, 731
- Clartune, 29–30
- client-server computing, 15
- `CLLocationManager` class, 808
- Cocoa Foundation, 778
- Cocoa framework
 - arrays in, 606–607
 - design patterns used in, 187
 - designated initializers in, 204
 - developing in, 136–137
 - property lists for, 589
- code folding, 306
- Code Sense feature, 305–306
- code signing, 95
- code stubs, 461
- coding. *See* programming
- collections
 - arrays, 607
 - arrays as ordered, 230
 - enumeration in arrays, 232–233
- colon (:), 146, 624
- comments, 453, 492–493
- compiler directives, 143, 148–149
- compilers
 - compiling early RoadTrip, 492–493
 - creating accessors, 336–337
 - implementing class with `#import`, 506
 - introducing classes with `@class`, 506
 - warnings during builds, 385
- composition technique, 189
- `connection:DidFailWithError:` message, 642–643
- `connection:DidFinishLoading:` message, 643
- `connection:didReceiveAuthenticationChallenge:` message, 645–646
- `connection:DidReceiveData:` message, 641–642
- `connection:didReceiveResponse:` message, 641

- connectionError:
 - method, 648, 651
 - connections
 - error messages for, 648
 - limitations of, 39
 - making quickly, 35
 - warning users about
 - lost, 41
 - constants. *See also*
 - Constants.h file
 - accelerometer values as, 814–815
 - extensibility with, 450
 - importing, 448–449
 - using in declarations, 447
 - using kCell
 - Identifier, 614
 - Constants.h file
 - adding views as key in, 352
 - defining dictionary keys
 - in, 457
 - editing, 322–324
 - modifying number of
 - sections in, 446–451
 - contacts
 - accessing, 16
 - adding, 741, 773, 782–785
 - browsing and selecting, 774, 775, 777–781
 - data storage for, 776
 - container classes
 - array objects in, 230–235
 - dictionaries in, 236–238
 - Foundation, 229–230
 - container views, 86
 - containers
 - container classes, 229–238
 - as property list objects, 240, 591
 - reading and writing from, 245
 - content
 - compelling, 27–30
 - recording and playing, 38
 - views displaying, 84
 - Content views
 - about, 83, 495–496
 - communicating size of, 355
 - moving from table view
 - to, 468
 - view hierarchy and, 84, 85
 - context
 - incorporating device, 14, 17–18, 38–39
 - managed object, 670, 692–693, 700–701, 725, 726
 - control flow, 311
 - control structures
 - example of switch, 174–175
 - replacing with
 - polymorphism, 184–186
 - controller objects. *See also*
 - view controllers
 - about, 81, 188
 - control flow in, 311
 - controls, 85, 86, 92–93
 - creating and initializing, 475
 - illustrated, 82
 - view controllers, 87–89
 - cookies, 637
 - copy method, 225–226
 - copying existing
 - functionality, 40
 - copyrights, 109
 - Core Data. *See also*
 - managed object
 - models
 - accessing managed
 - ObjectContext
 - property, 700–701
 - adding attributes, 676–682
 - advantages of, 670
 - Core Data Framework
 - and, 697
 - creating Hotel class for, 671–672
 - entities in, 674, 682–685
 - fetch requests, 705–706
 - generating classes for
 - data model, 686–691
 - implementing application
 - templates with, 694–697
 - instantiating Core Data
 - Stack, 706–713
 - inverse relationships for, 684–686
 - limitations of, 671
 - managed object context
 - and, 670, 692–693
 - managed object model
 - for, 672–673
 - no dealloc in
 - implementation files, 690
 - object persistence with, 669–670, 692–693
 - regenerating classes, 692
 - template for, 715–716
 - Undo feature for, 726, 728
 - using NSManagedObjectContext
 - class, 692
- Core Data Framework, 697
- Core Data sample project
 - accessing managed
 - ObjectContext
 - property, 700–701
 - adding Core Data
 - Framework to, 697
 - creating experimental, 694–697
 - generating .h and .m
 - class files, 689–690

Core Data sample project
(continued)
 instance variables, properties, and method for, 698–700
 modifying class files for, 690–691
 persistence store coordinator, 693, 701–703
 setting up Core Data Stack, 693–694

Core Data Stack
 accessing managed `ObjectContext` property, 700–701
 accessing managed `ObjectModel` property, 703–704
 classes found in, 693, 700
 creating project's, 694–697
 instantiating, 706–713
 setting up, 693–694

Core Data template, 715–716

Core Foundation, 778

Core Location framework, 806–807

credentials, 636–637, 645–646

cross dissolve modal view, 757

curly braces (`{ }`), 140

currency symbols, 237

current location
 denying access to, 809–811
 displaying, 539–540
 method finding, 546
 zooming to, 552–554

`currentLocation` property, 808

customers
 acquiring, 121
 encouraging reviews by, 129

knowing your, 122, 124, 125
 marketing apps to, 118–120
 pricing apps for, 120–121
 reaching with Web page links, 128
 tracking downloads by, 122–124

D

data. *See also* stored data mode
 accessor methods for retrieving, 158–162
 available on Web server, 579
 caching iPhone, 434
 confirming server received data, 641–642
 entering default, 769–771
 getting from objects, 158–162
 hard coding for `Sight` objects, 588–589
 loading asynchronously, 632–634
 minibook on, 6
 parsing location, 574
 placemark storage of, 556–557
 preference, 320–321
 property list, 250–251, 256, 590–591, 600–601
 records in databases, 776
 registering from accelerometer, 812–813
 retrieving from dictionary, 252
 saved in variables, 251
 saving downloaded, 648–650
 storing essential car, 430, 431

using stored, 482
 view controllers providing, 436
 writing URL data to disk, 636

data entry
 canceling, 763–764
 composing e-mail text, 798–801
 keyboard input for, 772
 saving data, 763–764, 765
 selecting new Road Card entry, 792–793

data entry view
 controls for hotel's, 743–747
 view controller for hotel, 741–743

Data Model template, 673–676

data models, 673–674

data source
 for Car Model Information feature, 496
 for `UITableView` objects, 441, 582

databases
 interacting with Address Book, 776
 records for, 776

`datatip`, 386, 387, 388

`dealloc` method
 de-allocating objects, 208–211
 modifying for managed object context, 725
 omitted in Core data implementation files, 690
 releasing `NSObject` superclass objects with, 220
 tracing objects using, 221

Debugger Console, 66, 392–393

- Debugger (Xcode Text editor)
 - datatip in, 386, 387, 388
 - Debugger strip, 389–390
 - debugging code while running, 389–390, 394
 - looking at stack in, 388
 - opening and using, 390–391
 - setting breakpoints, 394–396
 - using Xcode Text editor’s, 385–390
- debugging. *See also* Debugger
 - importance of, 377, 381
 - kinds of bugs, 382–385
 - Mini Debugger for, 392
 - opening and using Debugger window, 390–391
 - overview of miniguide on, 5
- declared properties
 - accessing, 157
 - function of, 153–154
 - releasing assigned objects, 157–158
 - using, 162–163
- DeepThoughts. *See also* Utility Application template
 - about, 22–23
 - about low-memory warnings, 282–283
 - accessing media in, 370–375
 - adding second view, 352–356
 - changing Main view’s background color, 315–316
 - delegate and controller objects in, 311
 - “falling” words animation in, 305, 312–315
 - finding memory leaks, 212–215, 405–407, 408–413
 - Flipside view, 278–279, 337–344
 - icon for, 376–377
 - initialization in, 277–279
 - initializing Main view, 289–290
 - Instruments app used with, 402–407
 - main function and lifecycle of, 272–274
 - photo selection and display, 352
 - property list for falling words, 328–332
 - scrolling in, 352, 355, 364–366
 - Settings bundle added to, 327–328
 - Sleep/Wake button interruptions in, 281
 - task performed by, 319
 - template functionality in, 274–276, 305
 - terminating, 279–280
 - testing animation for, 318
 - testing user preferences for, 332–334
 - user preferences for, 320–326
 - view initialization in, 311
 - writing Main view code, 305–306
 - zombie testing, 414–423
- DeepThoughts
 - AppDelegate.h file, 352–353
 - AppDelegate.m file, 353–355
- default data, 769–771
- delaying animation, 820–821
- delegate methods, 258, 623–624
- delegate objects
 - about, 258, 623–624
 - control flow for, 311
 - defining system view controller’s protocol with, 773–774
 - returning based on selection, 652–653
 - specifying, 276
 - UITableView, 441, 582
- delegates. *See also* delegate objects
 - AddHotelController, 765–766, 768–769
 - choosing from pop-up menu, 502–503
 - DeepThoughts, 277–279
 - derivation of, 279
 - fetching result controller, 724–725
 - initializing view controllers in, 352–355
 - launching, 273
 - location manager and accelerometer, 813
 - selecting, 750, 752–753
 - terminating applications, 280
- delegation. *See also* delegate objects; delegates
 - about, 165, 257–258, 623–624
 - ABPeoplePicker Navigation Controller, 775
 - AddHotelController, 768–769
 - adding to class, 261–263
 - declaring protocols, 259
 - encapsulation and inheritance with, 351
 - using delegates, 273
 - WebViewController, 625–630
 - when dismissing modal view controllers, 757

- Delegation pattern, 90, 91–92
- `deleteHotel:` method, 736–737
- deleting
 - `CarServicing.m` methods, 657–658
 - hotels, 735–737
 - user preferences, 486–487
- design
 - compelling application, 25–27
 - device-guided, 14, 17–18, 38–39
 - drawing from user experience, 34–36
 - orientation and motion in interface, 15–16, 38
- design patterns. *See also specific pattern*
- Delegation, 90, 91–92, 258
 - implementing object encapsulation with MVC, 187–189
 - MVC, 79–82
 - Target-Action, 91, 92–93
 - using, 78–79
- designated initializers, 152, 204
- destination in Xcode Model Editor, 682, 683
- Detail view (Xcode), 60, 61
- developers
 - Development Provisioning Profiles for, 97–98
 - exploring the iPhone Dev Center, 51–53
 - iPhone Developer Program for, 19, 46–51
 - licensing Apple artwork and images, 107
 - provisioning devices for development, 98–100
 - putting ads in apps, 124–125
 - registered iPhone, 44–46
 - developing applications. *See also developers; Development Certificate*
 - building user interface, 747
 - Cocoa framework for, 136–137
 - cycles in, 20–21, 52
 - Development Provisioning Profile for, 97–98, 100–103
 - free and paid application versions, 120
 - including App Purchase feature, 125–128
 - including Web page links, 128
 - reasons for, 19
 - using SDK, 57–58
 - Development Certificate
 - creating, 100–103
 - function of, 97
 - tied to Development Provisioning Profile, 97–98, 102
 - Development Provisioning Assistant, 101–103
 - Development Provisioning Profile, 97–98, 100–103
- device
 - designing apps for, 14, 17–18, 38–39
 - knowing user’s location, 15, 37
 - obtaining Development Provisioning Profiles for, 97–103
 - tracking location changes for, 809
- dictionaries
 - arrays versus, 236, 608
 - changing from array to, 592–593
 - checking to *see* if in plist, 251
 - creating, 237–238, 251–252, 596–599
 - dictionary of, 246–249
 - entering key-value pairs, 595–599
 - entries, keys, and values in, 236–237, 609
 - hiding, 598
 - holding nodes in arrays, 329
 - managing dictionary of, 250
 - mutable, 237, 245
 - as property list object, 240
 - reading files into mutable, 245
 - retrieving data from, 252
 - RoadTrip, 455–457
 - saving and restoring, 237
 - setting up key-value pairs, 328
 - static, 237, 610
 - updating, 253
 - uses for, 236
 - digital signature, 96–97
 - directories
 - paths for saved data, 649
 - setting path to `Documents`, 703
 - disclosure triangles, 246, 596
 - dismissing
 - `ABPeoplePicker Navigation Controller`, 775
 - keyboard, 761–762
 - modal view controller, 757–763, 765–766, 783
 - display views, 84, 86
 - distance
 - calculating location and, 806–812
 - returning between specified coordinates, 820

distance filter
 setting, 808, 809
 updating location by
 amount in, 811–812

distributing applications
 creating Distribution
 Provisioning Profile, 96,
 104–106
 joining iPhone Developer
 Program, 19, 46–51
 process for, 96–97,
 104–106

distribution certificate, 96

Distribution Provisioning
 Profile, 96, 104–106

Documentation window
 (Xcode Text editor),
 307–308

doFind: method, 567–568

downloadData: message,
 648

downloading
 application blocked
 during, 637
 SDK, 54–56
 tracking, 122–124
 Web data for apps,
 632–634

dynamic binding, 187

E

e-mail
 composing message for,
 796–803
 testing on device, 789
 unavailable in airplane
 mode, 788
 using iPhone, 787–789

Editor view (Xcode), 60,
 62–63

embedded.mobile
 provision, 106

encapsulation
 delegation and
 inheritance with, 351

encapsulating objects,
 135, 140

implementing with
 inheritance, 175

MVC pattern and object,
 187–189

polymorphism versus, 172

understanding, 169,
 170–171

enhance-ability
 inheritance and, 176–177
 role of polymorphism in,
 184–186
 ways to achieve, 169

entities
 adding, 674–676
 attributes for, 676–679
 creating relationships
 between, 682–685
 fetching, 705, 718–720
 updated when class files
 generated, 690

entries in dictionaries, 236

errors
 alerts about Internet
 failures, 545, 651
 app blocked until data
 downloaded, 637
 connection failure,
 642–643, 648
 location manager failed,
 810–811
 persistent store, 702–703
 reverse geocoding
 placemark, 557
 URL connection, 642–643

event loop
 handling interruptions
 with, 281
 setting up , 273–274
 in Touch Up Inside
 process, 303

events
 Info button, 296–299
 processing Touch Up
 Inside, 302–303

timing accelerometer, 814
 viewing possible Info
 button, 296–297
 views for handling touch,
 84

extensibility
 achieving, 169
 allowing with inheritance,
 176–177
 composition techniques
 for, 528–530
 modifying view
 controllers for, 582–584
 role of polymorphism in,
 184–186

UIWebView view
 controller providing,
 498

F

“falling” words animation
 adjusting speed of, 335,
 340–344
 changing color and timing
 for, 315–316
 code listings for
 animation, 312–315
 creating property list for,
 328–332
 default image for, 368–369
 effects of, 305
 testing, 318

fast enumeration
 about, 6
 using in arrays, 232–233,
 607–608

fetches results controller
 creating, 717–720
 delegates for, 724–725
 using in Table view, 706,
 724

fetchesResults
 Controller method,
 670, 717–720

- fetching
 - entities, 705, 718–720
 - estimating batch size, 709
 - objects from persistent store, 693
 - saved objects, 670
 - using fetch requests, 705–706
- files. *See also specific types of files*
 - adding to Xcode project, 368–369
 - creating new project, 322–324
 - .h and .m suffixes for, 141, 142
 - loading asynchronously, 632–634
 - naming Objective-C, 142
 - nib, 75, 272
 - reading into mutable dictionary, 245
 - reading/writing dictionary to, 609
 - setting paths to, 649
 - specifying path in code to, 244
 - .xib, 75
- File's Owner proxy object
 - about, 75, 276
 - assigning class name to, 360
 - connecting methods to Light Info button, 363
 - initializing for `MainView.xib`, 289, 290
- Find button, 561–562
- find: method, 561–562
- FindLocation
 - Controller class
 - creating, 562
 - implementing, 562–563, 566–567
 - importing MapKit and MapController, 563–564
 - interface for, 565–566
- first responder
 - First Responder proxy object, 75, 276
 - setting with undos and redos, 732
- fixed arrays, 230, 233
- fixing memory leaks, 215–221
- Flipside view (DeepThoughts)
 - about, 63
 - connecting interface objects for, 244–249, 344–349
 - displaying in Interface Builder, 73
 - Done button of, 299–302
 - entering substitute text in, 305
 - implementing, 278–279
 - initializing, 293–295
 - inspecting Utility Application template's, 292–295
 - interface object methods for, 337–344
 - showing in Simulator, 66, 67
- FlipsideView subclass, 295
- FlipsideView
 - Controller object, 299–302
- FlipsideView
 - Controller.h file, 335
- FlipsideView
 - Controller.m file, 337–340
- Focus Ribbon column, 306
- folders
 - adding and managing, 500
 - Classes, 447, 499
 - specifying path to, 244
- for in feature, 232, 607–608
- formal protocols
 - adopting, 260–261
 - declaring, 259, 625
 - defined, 258, 624
 - method implementation in, 263
- forums
 - Apple Developer Forums Beta, 53
 - publicity on, 129–130
- Foundation framework.
 - See also* arrays; dictionaries
 - about, 227
 - container classes, 229–230
 - utility classes of, 228–229
- frameworks
 - about, 64, 77–78
 - adding AddressBook and AddressBookUI, 782
 - Core Data, 697
 - Core Location, 806–807
 - delegation in, 165
 - design patterns and, 79
 - developing apps in Cocoa, 136–137
 - extending with delegation and categories, 257–258
 - Foundation, 227–230
 - functionality added with, 78
 - implementing behavior with Delegation pattern, 90, 91–92
 - iPhone's, 51
 - MapKit, 532–533
 - MessageUI, 789
 - object-oriented nature of, 79–80
 - setting up event loop with UIKit, 273–274
 - simulator limitations for, 70

Store Kit, 126, 127
 using iPhone OS, 271
 free applications, 120
 function prototypes, 145, 146
 function (use)
 form following, 31
 making obvious in app, 32–33
 functions (programming).
See also methods
 basing modules on, 168
 used as methods in
 Objective-C, 138, 145

G

GDB (GNU Source-Level Debugger), 392
 geocodeLocation:
 method, 573
 Geocoder class
 coding interface file for, 569–570
 creating and
 implementing files for, 569–575
 error messages for, 572, 574
 implementation code for, 570–572
 methods used by, 572–574
 geocoding, 555, 559–562
 gestures. *See also* shaking
 iPhone
 defined, 16
 Shake, 68, 733, 734, 762
 simulating, 68–69, 70
 supported in Map view, 538
 swipe, 368–369
 using standard iPhone, 32
 Get from Contacts button, 748, 773

getCarServicingData:
 method, 526–527
 getData: method,
 650–651
 getter methods, 153, 154, 155, 336
 GNU Source-Level
 Debugger (GDB), 392
 goBack: method, 514–515
 Google
 free geocoding from, 559, 574–575
 Google Maps, 539
 requesting API key from, 574–575
 GPS (Global Positioning System), 542
 grouped table view, 442–443, 445
 groups, project, 356
 Groups & Files view
 (Xcode), 63

H

.h file suffix, 141, 142
 Hardware menu (iPhone Simulator), 68
 header files
 defining `id` and `nil`
 object types in, 143
 designating, 141
 locating for symbol, 307
 switching window between
 implementation and,
 306
 Help menu (Xcode Text editor), 309
 hiding
 Back button, 515–518
 dictionary entries, 597
 Objective-C methods, 527, 602–603
 horizontal modal view, 757

horizontal scrolling, 355
 horizontalAccuracy,
 553
 Hotel class
 creating interface for,
 671–672
 generating .h and .m files
 for, 688–689, 690
 Hotel entity
 inverse relationships for,
 684–686
 naming, 676
 HotelController class
 adding, 713–714
 adding template code to,
 715–716
 interface file for, 714–715
 uncommenting code in,
 712–713
 HotelController.h file
 adding state information
 to, 758–759
 adding undo code to
 object, 728–730
 setting up, 714–715
 HotelController.m file
 adding fetchedResultsController
 method,
 717–720
 changing Table view
 methods in, 723–724
 deleting hotels, 736–737
 implementing, 715
 including section titles,
 724
 initializing `amEditing`
 state, 759, 760–761
 maintaining managed
 object context in, 725
 viewDidLoad method
 changes for, 716–717
 Hotel.h file, 688
 Hotel.m file, 689, 690

- hotels
 - adding contacts to Hotel list, 773
 - adding new, 670, 671–672, 720–724
 - annotation objects used by, 711
 - Cancel and Save buttons for, 745, 763–765
 - data entry interface, 741–749
 - deleting, 735–737
 - deleting added, 735–737
 - dismissing keyboard after entering, 761–762
 - displayed by city, 719
 - displaying on maps, 726, 727
 - entering default data for, 769–771
 - estimating batch size for fetching, 709
 - fetch requests for, 705–706
 - illustrated, 742
 - list of, 725–726
 - modifying `Trip.h` file with properties for, 710–711
 - putting in Address Book, 741, 782–785
 - shaking to undo entries, 733–735, 762
 - showing on maps, 726, 727
 - undoing, 726–737
 - viewing information about, 430
 - Human Interface Guidelines, 475, 514
-
- I
- IBAction keyword, 747–748, 749, 783–784
 - IBOutlet keyword, 335, 747
 - icons
 - adding application, 376–377
 - Apple standards for, 114–115
 - DeepThoughts, 376–377
 - `MainWindow.xib` file, 276
 - plus, 246, 595, 596, 680, 682
 - submitting application's, 111
 - used in book, 6–7
 - id object type, 142–143
 - image picker, 370
 - images. *See also* animation; photos
 - adding `RoadTrip Sight` objects, 599, 601
 - Apple standards for, 115
 - changing to thumbnails, 614–616
 - creating timers for animated, 315
 - licensing to use Apple images, 107
 - immersive applications, 20
 - implementation files
 - `@dynamic` statements for Core Data, 690
 - `@end` compiler directive in, 149
 - `dealloc` method omitted in Core Data, 690
 - implementing classes, 148–149
 - methods designating, 141
 - switching window between header and, 306
 - indexes
 - accessing array elements with, 231
 - saving path to, 477–478
 - Info button
 - processing Touch Up Inside events for, 298–299, 302–303
 - using Utility Application template's, 295–296
 - informal protocols, 258, 263, 267
 - inheritance
 - advantages of, 176–177
 - classes and, 164–165
 - delegation and encapsulation with, 351
 - implementing, 178–184
 - overriding, 176, 177
 - simplifying code with, 172–176
 - using with polymorphism, 184, 186–187
 - ways to use, 177–178
 - `init` method
 - about, 195
 - invoking superclass's, 197–199
 - returning pointer to object created with, 509
 - returning `self` with, 200
 - initializers
 - code examples implementing, 200–206
 - designated, 152, 204
 - initializing objects
 - about, 195–196
 - code implementing initializers, 200–206
 - initializing instance variables, 199
 - in Objective-C, 151–152
 - objects returned by superclass, 198–199
 - returning `self`, 200
 - starting Flipside view with `MainViewController` object, 293–295
 - steps in, 196

- subclass initialization
 - steps, 197
 - types of methods, 193
- initWithContents
 - OfFile: method
 - classes implementing, 480
 - using writeToFile::
 - with, 591, 609
- initWithHotel:trip:
 - method, 754–755
- initWithName: method, 604–605, 667, 707
- initWithTrip:::
 - initializer
 - fast enumeration using, 608
 - implementing Car
 - Servicing
 - Information with, 523–524
 - using with Hotel
 - Controller, 715
 - working with controllers, 507, 508–509
- insertNewObject
 - deleting code from, 766–768
 - modifying for AddHotel
 - Controller, 755–757
- insetting menu text, 464–466
- instance methods, 151
- instance variables. *See also* outlets
 - accessing, 162–163, 632
 - adding to interface, 478
 - assigning arguments to, 159
 - declaring, 143, 144–145, 488
 - defined, 138, 145
 - entities versus, 676
 - inheritance's effect on, 176–177
 - initializing, 199
 - naming, 90
 - pointing properties to, 261
 - scoping, 140–141
 - using Debugger to check, 396
- instantiating
 - classes, 145
 - multiple views, 352–356
 - objects, 137, 138
- Instruments application
 - about, 51, 401–402
 - choosing executable to
 - trace in, 418–423
 - enabling zombies, 416–423
 - finding memory leaks, 405–407
 - ObjectAlloc instrument, 404–405
 - starting compiled app in, 402–405
 - testing for de-allocated
 - objects, 407, 410, 414–416
 - using Leaks instrument, 407–413
- Interface Builder
 - about, 51
 - connecting button to
 - action method, 749–754
 - connecting interface and
 - action in, 749–754
 - connecting interface
 - objects in, 344–349
 - copying objects between
 - nib files, 361
 - inspecting Main view in, 288–292
 - opening Attributes
 - Inspector, 299
 - reviewing Flipside view in, 292–295
 - starting, 288
 - using, 73–76
- viewing MainWindow.
 - xib file in, 275–277
- interface files
 - adding @end directive, 143, 147
 - adding instance variables
 - in, 144–145
 - characteristics of Core
 - Data, 690
 - declaring, 143–147
 - method declarations in, 145–147
 - structure of Objective-C, 139, 141
 - using @interface in, 141, 143, 144, 148
- Internet
 - accessing with
 - applications, 15, 37
 - alerting users to
 - unavailability, 545, 651
 - data caching when
 - unavailable, 434
 - retrieving content from, 496
 - reverse geocoding errors, 557
- interruptions, 280–281
- inverse relationships, 684–686
- iPhone. *See also* iPhone Simulator
 - accessing media on, 370–375
 - application design for
 - computing, 39
 - avoiding autorelease
 - coding for, 224
 - designing apps to use
 - strengths, 36–38
 - developing apps for, 19, 20–21, 52
 - e-mailing from, 787–789
 - effect of SMS message on
 - app, 280

- iPhone (*continued*)
 - functional features of, 31
 - guidelines for preferences, 320
 - handling app memory for, 282–284
 - limitations of, 17–18
 - provisioning for
 - development, 98–100
 - recording and playing content with, 38
 - resource effect on performance, 401–402
 - shaking to undo entries, 733–735, 762
 - taking screen shots with, 111–112
 - tapping entry to see next view, 499–504
 - iPhone Dev Center
 - downloading SDK from, 54–56
 - exploring, 51–53
 - illustrated, 45, 50
 - provisioning process from, 99–100
 - resources of, 53
 - iPhone Developer Program, 19, 46–51
 - iPhone Developer Program Portal
 - development and distribution
 - provisioning profiles from, 101
 - illustrated, 50
 - provisioning via, 99–100
 - iPhone Distribution Certificate, 104–105
 - iPhone OS
 - accelerometers in, 15–16
 - accessing contacts, 16
 - frameworks from, 271
 - functionality added with, 78
 - gestures for applications, 16, 32
 - including user’s location in apps, 15
 - Internet access by applications, 15, 37
 - playing audio and video, 16
 - requirements for programming, 3–4
 - running out of memory, 18
 - tracking orientation and motion, 15–16, 38
 - using application features of, 14–16
 - using iPhone pictures and camera, 16
 - virtual memory handling, 207, 282–284
 - iPhone Simulator
 - about, 51, 52
 - alert testing on, 545
 - button event simulation with, 288
 - current location display on, 540
 - debugging while running, 389–390, 394
 - developing applications on, 3–4
 - gestures in, 68–69, 70
 - Hardware menu options for, 68
 - Instruments to test apps in, 402, 405–407
 - limitations of, 70
 - resetting and retesting with, 334
 - testing user preferences in, 332–334
 - uninstalling apps and resetting, 69
 - viewing applications in, 66–67
 - iPhone Software Development Kit. *See* SDK
 - isSourceType
 - Available: class method, 370
 - iTunes Connect
 - about, 106
 - illustrated, 108
 - managing applications in App Store, 106–108
 - requesting promotional codes, 130
 - tracking customer downloads, 122–124, 125
 - uploading application to App Store, 112–114
-
- ## K
-
- kABGroupType records, 776
 - kABMultiDictionary
 - PropertyType, 779
 - kABMultiString
 - PropertyType, 779
 - kABPersonFirstName
 - Property constant, 778
 - kABPersonLastName
 - Property constant, 778
 - kABPersonOrganization
 - Property constant, 778
 - kABPersonType records, 776
 - kAccelerometer
 - Frequency constant, 814
 - kCellIdentifier
 - constants, 614

key-value pairs, 328, 595–599, 609

keyboard

- defining input for, 772
- dismissing, 761–762
- limitations of, 18
- using inappropriate type of, 41

Keychain, 96, 97

keys

- adding to `Constants.h` file, 457
- creating user preference keys for `DeepThoughts`, 321–322
- dictionary, 236, 456–457, 609
- looking up values for, 237–238
- setting up key-value pairs, 328, 595–599
- updating values of, 253

L

labels. *See also* text fields

- adding with `# pragma mark`, 339
- formatting text font color and size, 464
- multi-value, 780
- property settings for, 464–465
- saving attributes, 745
- static text in, 743, 744

landscape mode

- seeing `Simulator` in, 68
- viewing maps in, 539

`lastView` instance

- variable
- adding to file, 478
- saving, 477
- writing data to file from, 478

late binding, 187

latitude, 541–542

latitude attribute, 681–682

leading dash (-), 146

Leaks instrument, 407–413

Library, 743

licensing artwork and images, 107

lifecycle

- iPhone application, 272–274
- object, 207
- responding to interruptions in, 280–281

Light Info button, 361, 363, 364

links

- adding Web page, 128, 512–515
- disabling, 515–518
- including iTunes affiliate, 128

`loadHotels` method, 708–710

loading

- asynchronous file, 632–634
- `DeepThoughts` nib file, 272
- elements of requests for, 636
- navigation controller view, 468–469
- Sight objects, 604–610
- user interface quickly, 279

`loadSights` method, 602–603, 604

`localizable.strings` file, 454

localizing applications, 36, 436, 452–455

Locate button, 552–554

location

- accuracy of, 555, 809
- asking to use current, 809, 810
- centering map display on, 542–544
- denying use of current, 809–811
- displaying current map, 539–540
- failure to find right away, 811
- finding right away, 809
- marking with annotations, 568–569
- pinning on map, 428–429
- simulation limitations for, 70
- tracking changes in, 546
- updating distance to/from, 811–812
- using Core Location framework for, 806–807
- zooming to current, 552–554

`locationManager`

- denying application access to current location, 809–811
- `DidFailWithError:` message, 810–811
- `didUpdateToLocation:fromLocation` argument, 811–812
- setting up, 808–809

logic errors, 384–385

longitude, 541–542

longitude attribute, 681–682

low-memory warnings, 282–283

M

.m file suffix, 141, 142

Mac OS

Debugger Console in, 66,
392–393
required for programming,
3

requirements for SDK, 54
version used for this
book, 232

virtual memory handling
in, 207

Mail. *See also* e-mail

illustrated, 788
using, 787–789

main function, 272–274

Main view (DeepThoughts)

about, 63
animating, 312–317
background color and
animation timing in,
315–316

copying view to Photo
view, 361–364

Info button of, 295–299
inspecting Utility

Application template's,
288–292

testing animation in, 318
writing code for, 305–306

Main view (Road Trip)

about, 439
developing menu lists for,
455–459
making selections in,
474–475

returning to, 469
tapping entry to get,
499–504

MainView class, 291–292

MainViewController
object

initializing Flipside view
with, 293–295

initializing two instances
of, 352–355

reading preferences into
file for, 321, 324–326

releasing instances of,
355–356

reusing code for, 366–368
used in Utility Application
template, 291

MainViewController.h
file, 340

MainViewController.m
file

adjusting animation speed
in, 340, 341–344
setting preferences in, 321

MainView.xib file

copying and reusing code
from, 361–364
creating from File's Owner
object, 289, 290

MainWindow.xib file

icons for, 276
modifying for scrolling,
364–366
reviewing functionality of,
275–277

Managed Object Class

Generation dialog,
686–688

managed object context
code accessing property
for, 700–701

function of, 670
maintaining, 725

sending fetch request to,
706

Undo manager for, 726
using objects in, 692–693

managed object models

about, 670, 672–673
adding attributes to,
676–679

creating, 673–674

entities added for, 674–676

persistent store
incompatible with,
702–703

relationships between
entities in, 682–685

managedObjectContext
property, 700–701

managedObjectModel
property, 703–704

MapAnnotation class,
568–569

MapController.h file
declaring doFind:

method in, 567–568

importing, 534

modifying map
annotations in, 665
setting up, 531, 532

MapController.m file

doFind: method code
for, 567–568

getting annotations from
Map, 662–663

modifying map
annotations in, 665–666
setting up, 531, 532

MapController.xib file,
535–537

Map.h file, 532, 660–667

MapKit framework

about, 1
reverse geocoding feature
in, 555–556
selecting and using,
532–533, 538–539

Map.m file

adding annotations
dynamically, 661–662
creating, 532

loading annotation
objects in, 711

maps. *See also* reverse
geocoding

about MKMapView class,
535, 536, 538–539

- adding toolbar to, 559–561
- annotating, 547–552
- canceling searches, 566, 567
- centering on region, 540–545
- creating Find button for, 559–562
- displaying hotels on, 726, 727
- files needed for, 531–532
- focusing content on task in, 28
- Google, 539, 574–575
- illustrated, 440, 536
- initialization code needed, 533–535
- reverse geocoding for, 555–559
- setting up regions for, 568
- showing current location, 539–540
- specifying latitude and longitude, 541–542
- titles for, 544
- using MapKit framework, 532–533
- viewing in landscape mode, 539
- marketing
 - adding links for, 138
 - apps on App Store, 118–120
 - buying ads and publicity, 131–132
 - competition on App Store, 117
 - determining audience for app, 121
 - encouraging App Store reviews, 129
 - requesting promotional codes, 130
 - social networking and, 129–130
 - updating applications for attention, 131
 - viewing sales reports, 122–124
- Marketing Master, 120
- MarketingProfs, 120
- memory. *See also* memory leaks; memory management
 - allocating object, 149–150, 195
 - array, 221
 - designing applications for, 39
 - raising and releasing, 206–207
 - reusing cells from, 462–463
 - running out of, 18
 - simulating, 68, 70
 - virtual memory handling, 207, 282–284
- memory leaks
 - about, 207, 402
 - finding, 212–215, 405–407, 408–413
 - fixing, 215–221
- memory management
 - about, 195, 207
 - allocating, releasing, and deallocating objects, 208–211
 - application performance and, 284
 - autorelease pools, 222–225
 - basic rules of, 225–226, 284
 - deallocating objects in `HotelController.m`, 725
 - iPhone OS' virtual memory and, 207, 282–284
- objects in arrays, 221, 231, 233
- releasing multiple instances of `MainViewController`, 355–356
- templates and, 287
- `menuOffset` methods, 465–466, 474
- message expressions, 152–153
- messages
 - object communications via, 168
 - sending from objects to, 151, 152–153
 - using array elements as arguments in, 231
- `MessageUI` framework, 789
- metadata, 106, 109–111
- methods. *See also specific method*
 - accessor, 153–154, 158–162
 - array, 231
 - autorelease pools and class, 223–225
 - class, 151, 223–225
 - connecting timer to animation, 316–317
 - connecting to objects, 335
 - connecting views to, 361–364
 - creating with categories, 264–266
 - declaring, 143, 145–146
 - defined, 140
 - defining in protocols, 258, 624
 - delegated, 258, 623–624
 - designating implementation files, 141
 - effect of inheritance on, 176–177
 - hiding Objective-C, 527

methods (*continued*)
 implementing with
 compiler directive, 148–149
 instance versus class, 151
 listing in protocols, 259
 multiple arguments in, 146
 naming, 90, 147, 624
 NSDictionary, 236–237, 609
 types of initialization, 193
 using categories, 603–604
 migrating app to Core Data, 672–673
 Mini Debugger, 392
 MKAnnotation protocol, 547–548, 549
 MKMapView class
 annotating maps, 547–552
 features of, 538–539
 setting up, 535, 536
 MKPlacemark, 556–557
 MKReverseGeocoder class, 556–558
 modal view controllers
 dismissing, 757–763, 765–766, 783
 using, 756–757
 modal views, 88
 model objects
 about, 80, 188
 adding application
 behavior via, 90–91
 creating RoadTrip row, 455–459
 illustrated, 82
 RoadTrip, 433, 435–436
 specifying, 89
 modular programming, 167–168
 motion detection, 15–16, 38
 multi-threading, 510
 multi-value labels, 780
 multi-value properties, 776, 779

multidimensional arrays, 234–235
 multiple inheritance, 176
 multitouch interface
 limitations of, 17–18, 39
 sense of control with, 33
 mutable arrays
 defined, 221
 example and uses of, 229–231
 placeholders in, 230, 607
 releasing, 231, 233
 restoring, 237
 mutable dictionaries, 237, 245
 MVC (Model-View-Controller) pattern. *See also* controller objects
 about, 79–81
 adding application
 behavior with, 90
 advantages of, 528
 example of, 81–82
 implementing, 189–192
 object encapsulation with, 187–189
 table views using, 455–456
 using view controllers, 87–89

N

Name attribute, 677–679
 naming
 classes, 90
 delegate methods, 624
 Development Provisioning Profile, 102
 entities, 676
 entity attributes, 677, 678
 methods, 90, 147
 Objective-C files, 142
 and saving projects, 60, 444

navigation. *See also* Navigation-Based Application template
 disabling links for, 514
 implementing for
 hierarchical content, 300
 navigation controllers for, 469–471
 tapping entry to see next view, 499–504
 using navigation bars, 87, 468–470
 zooming to current location and back to region, 552–554
 navigation bars
 displaying table views with, 468
 illustrated, 469
 in-application navigating with, 87
 managing with navigation controllers, 470
 Navigation-Based Application template
 modifying, 446–455
 navigation controller included in, 468–469
 using, 442, 443
 using Core Data with, 694–697, 713
 navigation controllers
 found in Navigation-Based Application template, 468–469
 navigating hierarchy, 469–471
 new method, 225–226
 New Project window, 59–60
 nib files
 about, 75, 359
 connecting outlets and action methods in, 749–754

- connection between
 - object and outlets in, 335
 - copying objects between, 361
 - creating Photo view, 359–364
 - creating view controller without using, 582–584
 - loading DeepThoughts, 272
 - nil objects
 - about, 142–143
 - returned by superclass initialization, 199
 - sending messages without generating runtime error, 396
 - nil values, 237
 - NPR Addict, 37
 - NSArray arrays, 233, 606–608
 - NSData arrays, 591
 - NSDictionary class
 - fast enumeration in, 238
 - methods of, 236–237, 609
 - NSEntityDescription object, 708
 - NSFetchedResultsController, 706, 717
 - NSLocalizedString macros, 453–455
 - NSLog statements, 221
 - NSManagedObject class
 - contents and uses of, 692
 - deriving Core Data objects from, 670
 - NSMutableArray arrays
 - about, 229, 606–608
 - creating RoadTrip, 456
 - NSArray versus, 233
 - placeholders in, 230, 607
 - using as wrapper, 591
 - NSMutableDictionary arrays, 456–457
 - NSNull object, 607
 - NSObject class
 - about, 144, 434
 - application delegate derived from, 279
 - de-allocating objects in superclass, 220
 - default for new classes, 181
 - NSManagedObject versus, 692
 - NSString object, 315–316
 - NSTimer class, 315
 - NSURLConnection class
 - about, 636, 637
 - delegate methods for, 638
 - initiating connections with, 639–640
 - synchronous downloading with, 637
 - NSURLDownload class, 636
 - NSURLRequest class
 - creating request for car servicing information, 525, 632–633
 - function of, 512
 - problems with construction of, 633–634
 - using with NSURLConnection, 639–640
 - NSURLRequest object
 - making asynchronous server connections with, 636
 - receiving from model object, 614, 616, 617, 619, 621, 622
 - sending message to delegate via, 629
 - NSUserDefaults class
 - identifying preferences for, 321–322
 - moving preferences into, 489
 - setting up preferences with, 320–321
 - NSZombieEnabled environment variable, 416, 417
-
- ## O
-
- object-oriented programming. *See also* encapsulation; inheritance; polymorphism
 - composition in, 189
 - creating accessor methods, 336–337
 - encapsulation in, 135, 140, 169, 170–171
 - frameworks in, 79–80
 - goals of extensibility and enhance-ability, 169
 - how inheritance works in, 172–178
 - implementing inheritance, 178–184
 - knowledge of, 4
 - modules in, 168
 - Objective-C’s use of, 135–137
 - polymorphism in, 169–170, 171–172
 - procedural versus, 137, 168
 - subclassing in, 90
 - ObjectAlloc instrument, 404–405
 - objectAtIndex message, 231
 - Objective-C
 - about, 135–137
 - accessor methods in, 153–154

- Objective-C (*continued*)
 - allocating objects, 149–152
 - case sensitivity of, 3, 142
 - categories and extensions in, 603–604
 - class implementation and interface, 139, 141
 - declaring class interfaces, 143–147
 - fast enumeration, 6, 232–233, 607–608
 - getting started in, 53
 - hiding methods in, 527, 602–603
 - `id` and `nil` object types in, 142–143
 - initialization in, 151–152, 193
 - introducing objects and classes in, 137–139
 - methods in, 145–147
 - naming conventions in, 142
 - protocols in, 258–261
 - raising and terminating objects, 206–211, 283
 - reference counting, 207–211, 283
 - scoping instance variables, 140–141
 - sending messages to objects, 151, 152–153
 - specifying instance variables, 144–145
 - using composition technique, 189
 - versions of, 136
 - working with declared properties, 153–163, 336
- Objective-C Declared Properties feature, 336
- objects. *See also* memory management
 - adding to arrays, 221, 230–231, 233
 - adding undo code to new hotel, 728–730
 - allocating, 149–152
 - annotation, 547–550
 - behavior in object-oriented program, 168–170
 - choosing right, 80
 - composite, 190
 - connecting methods to objects, 335
 - controller, 81
 - copying between nib files, 361
 - data and function members within, 140
 - delegate, 258
 - derivation of application delegate, 279
 - dynamic nature of, 150–151
 - encapsulation of, 135, 140, 169, 170–171
 - File’s Owner proxy, 75, 276
 - First Responder proxy, 75, 276
 - formal protocols for delegate, 91
 - getting data from, 158–162
 - `id` and `nil`, 142–143
 - initializing, 151–152
 - instantiating, 137, 138
 - introducing, 137–139
 - inverse relationships between, 684–686
 - lifecycle of, 207
 - managed object context for, 670, 692–693
 - managing memory for, 195, 207
 - memory allocation for, 149–150
 - model, 80, 82
 - MVC pattern and object encapsulation, 187–189
 - persistence of, 669–670, 692–693
 - polymorphism and, 171–172
 - raising and terminating, 206–211, 283
 - relationships between, 722
 - releasing and de-allocating, 208–211, 283
 - releasing assigned, 157–158
 - representing strings as, 315–316
 - retain count for, 208, 210, 211
 - returned by superclass initialization, 198–199
 - saving relationships between, 669–670
 - sending messages to, 151, 152–153
 - serializable, 239, 590
 - storing data in placemark, 556–557
 - storing in dictionaries, 236–238, 608–610
 - tracing de-allocation of, 221
 - view, 75, 81
 - working with in stacks, 470
 - wrapper, 591
- operating systems. *See* iPhone OS; Mac OS
- operations, 140

outlets
 adding for text fields, 747
 adding referencing, 750, 752
 connecting methods to objects with, 335
 declaring, 335, 336, 501–504
 defined, 501, 747
 setting up `FindLocationController`'s, 564–565
 viewing `AddHotelController`, 750, 751
 overriding inheritance, 176, 177

p

paid versions of applications, 120
 paradigm, 168
`parseLocationData`: method, 574
 path
 saving `indexPath`, 477–478
 setting to application
 `Documents`
 `Directory`, 703
 specifying to property lists, 244
 path separator (`/`), 649
 people picker for `AddressBook`, 774, 775–776, 777–781
`peoplePicker`
 `NavigationController`:
 `shouldContinueAfterSelectingPerson`:
 delegate method, 777–781

performance
 effect of iPhone's resources on, 401–402
 loading user interface quickly, 279
 memory and application, 284
 persistence
 defined, 228
 providing object, 669–670, 692–693
 persistence object stores, 693
 persistence stack, 692
 persistence store
 coordinator, 693, 701–703
 persistent store
 about, 670
 errors using, 702–703
 phone number selections, 779–789
 Photo view
 animating, 366–375
 creating nib file for, 359–364
 implementing, 356–366
 photos
 accessing from iPhone, 370–375
 applications accessing user's, 16
 selecting and displaying in app, 352
`PhotoViewController`
 object
 adding default image, 368–369
 reusing code for, 366–368
`PhotoViewController.h`
 object, 366
`PhotoViewController.m`
 object, 366–368

`PhotoView.h` file, 357
`PhotoView.m` file, 357
`PhotoView.xib` file, 359–364
 pinning map locations, 428–429
 placeholders in mutable arrays, 230, 607
 placemarks, 556–557
 plain table view
 about, 442–443
 changing to grouped table view, 445
 polymorphism
 adding new view using, 351
 concept of, 164, 169–170, 171–172
 encapsulation versus, 172
 replacing control structure with, 184–186
 using inheritance with, 184, 186–187
 popping objects, 470
 preferences
 adding in `Flipside` view, 334–344
 defined, 319
 guidelines for iPhone, 320
 implementing user, 482–487
 `NSUserDefaults`, 320–322, 489
 reading into
 `DeepThoughts`, 324–326
 `RoadTrip`, 486, 490–493
 setting up user, 320–321
 `Settings` app, 327–334
 testing `DeepThoughts` user, 332–334
 using offline data, 432

- PreferenceSpecifiers
 - array, 485
- prematurely de-allocated
 - blocks, 402
- press releases, 129–130
- pricing applications, 120–121
- primitives, 239, 240, 591
- procedural programming, 137, 168
- process counter (PC), 389, 394
- productivity applications, 20
- programming. *See also*
 - memory management; Objective-C; SDK
 - adding application behavior, 90–91
 - adding labels with `# pragma mark` statement, 339
 - changing values of constants, 614
 - code examples for, 3
 - code folding, 306
 - composition techniques for `Trip` model, 528–530
 - control flow in, 311
 - delegation in, 165
 - design patterns for, 78–79
 - development cycles for, 20–21, 52
 - enclosing code blocks in curly braces, 140
 - examples of rejected features, 40–41
 - finding memory leaks, 212–215, 405–407, 408–413
 - fixing memory leaks, 215–221
 - frameworks in, 64, 77–78
 - inheritance in, 164–165, 177–184
 - initializing objects, 151–152
 - iPhone Dev Center resources for, 53
 - iterative nature of, 436–437, 758
 - memory management code, 195
 - modular, 167–168
 - procedural, 137, 168
 - properties, 154–157, 162–163
 - requirements for, 3–4
 - specifying model objects, 89
 - specifying path in code to files, 244
 - transparent code, 167
 - types of bugs, 382–385
 - uncommenting code, 527, 585–587, 712–713
 - using device-guided design, 14
 - view controllers, 87–89
 - writing reusable code, 177–178, 186
 - Xcode preferences for, 71–73
- progress indicators, 644
- projects. *See also* Core Data
 - sample project
 - adding files to Xcode, 368
 - constants file for, 322–324
 - creating group for, 356
 - finding items in, 309–310
 - interface for Xcode, 60–65
 - naming Xcode, 60
 - property lists added to, 240–242, 592–599
 - resetting Simulator for retesting, 334
 - saved searches in, 310
 - saving, 60, 344, 356, 444
 - Settings bundle added to, 327–328, 483–484
 - starting, 58–60
 - promotional codes for applications, 130
 - properties
 - adding, 154–157, 162–163
 - function of declared, 153–154
 - multi-value, 776, 779
 - readonly, 699
 - releasing, 157–158
 - single-value, 776, 778
 - specifying read-only, 155
 - Property List editor, 592, 596
 - property list objects
 - about, 236
 - changing array to dictionary, 592–593
 - containers as, 240, 591
 - primitives, 239, 240, 591
 - property lists. *See also* arrays; dictionaries
 - adding to project, 240–242, 592–599
 - configuring for application, 484–487
 - creating dictionary of dictionaries, 246–249
 - dictionary, 251–252
 - falling words, 328–332
 - initializing `Sight` objects with, 601–603
 - modifying, 250–256
 - new entry for, 251–252
 - objects for, 239–240, 591
 - reading files into mutable dictionary, 245
 - retrieving data from existing dictionaries, 252
 - RoadTrip, 435–436, 581, 589–601
 - saving objects in, 669

- Settings application, 482
 - specifying path to, 244
 - storing data in, 600–601
 - symbol strings, 245
 - uses for, 238–239
 - using, 242–244, 589–591
 - protocols
 - adopting, 260–261, 630–632
 - data source and delegate, 441, 582
 - declaring, 259, 624–625
 - defined, 91
 - defining methods in, 258, 624
 - for delegate objects, 91
 - implementing, 276, 290
 - informal and formal, 258
 - URL loading system
 - support for, 637
 - using inheritance to create, 177
 - provisioning
 - about, 56
 - applications for App Store, 104–106
 - Development Provisioning Profile, 97–98, 100–103
 - Distribution Provisioning Profile, 96, 104–106
 - profiles required for Xcode, 97–98
 - publicity
 - on blogs and forums, 129–130
 - buying, 132
 - pushing objects, 470
- R**
-
- read-only property, 155, 699
 - real-time access
 - allowing during data loading, 632–634
 - providing for Weather, 430, 431, 496, 497
 - recording and playing
 - content, 38
 - records
 - database, 776
 - dismissing view controller for, 782–783
 - RedLaser, 35
 - redo feature, 732
 - reference coordinate system, 542
 - reference counting, 207–211, 283
 - refreshAnnotations
 - method, 662–663
 - region
 - configuring map, 568
 - how they work, 541–545
 - illustrated, 545
 - properties for, 538
 - returning from current
 - location to, 552–554
 - setting span of, 541–542, 553–554
 - updating and setting, 541
 - registered iPhone developers. *See* developers
 - relational databases, 671, 672–673
 - relationships
 - adding for entities, 682–686
 - creating objects and their, 722
 - inverse, 684–686
 - release messages, 208–211, 225–226
 - releasing
 - animated objects, 317
 - array objects, 231, 233
 - assigned objects, 157–158
 - de-allocating and, 208–211
 - fetching objects, 720
 - imagePicker
 - Controller, 375
 - MKReverseGeocoder, 556–558
 - relevance in user interface, 34
 - reloading Table views, 725
 - removing annotations, 568, 666–667
 - rendering, simulating, 70
 - resetting iPhone Simulator, 69, 334
 - restoring
 - application state, 479–481
 - arrays and dictionaries, 237
 - resuming applications
 - correctly, 481
 - retain count
 - basic rules for, 225–226
 - finding current, 221
 - keeping for objects, 208, 210, 211
 - retain messages
 - array objects receive, 231, 233
 - owning objects you send, 284
 - returnCarInformation
 - method, 522
 - returnCarServing
 - Information method, 524–525, 632–633
 - returnWeather: method, 511, 621
 - reusable code
 - designing view controller as, 498, 501–504
 - inheritance and, 177–178
 - MainViewController
 - code modifications, 366–368
 - managing data in property lists, 589
 - modifying MainView.xib, 361–364
 - reusing cells from memory, 462–463
 - writing, 186

- reverse geocoding. *See also* geocoding
 - adding delegate for, 534
 - defined, 555
 - Internet errors for, 557
 - using in RoadTrip, 556–559
- Road Cards
 - adding controller for, 789
 - composing message for, 796–803
 - creating interface for, 790–796
 - sending from sights, 794–796
- RoadCardController.m file, 789
- RoadTrip. *See also* hotels; Table view; URL loading system
 - about, 1, 5–6, 23
 - accessing accelerometer in, 814–815
 - adding contacts to Hotel list, 773, 782–785
 - alerts in, 492–493, 545, 651
 - annotating maps in, 547–552
 - calculating location and distance in, 806–812
 - centering map on region, 540–545
 - changing images to thumbnails, 614–616
 - compiling and running, 616
 - composing e-mail messages, 796–803
 - composition techniques for Trip model, 528–530
 - controller and nib files for, 499–501
 - creating cells for table views, 461–466
 - declaring nib file outlets, 501–504
 - defining scope of, 427–432
 - deleting hotels, 735–737
 - dismissing modal view controllers, 757–763, 765–766, 783
 - displaying hotels by city, 719
 - editing sections in Constants.h file, 446–451
 - entering dictionary key-value pairs, 595–599
 - illustrated, 12, 428
 - images for Sight objects in, 599, 601
 - implementing Car Information object, 507, 518–523
 - implementing user selections in, 472–475
 - landscape mode maps in, 539
 - localizing, 436, 452–455
 - map functionality in, 531–537
 - model objects for, 433, 435–436
 - modifying to use Core Data, 694–700
 - new hotels added to, 670, 671–672, 720–724
 - preferences in, 490–493
 - program architecture for, 433–436
 - reading preference settings in, 487–489
 - responding to user selections, 467–469, 582
 - reverse geocoding in, 556–559
 - Road Cards from, 787
 - saving state, 436, 475–479
 - section titles, 452, 667, 724
 - selecting Sights in main view, 585–587
 - sending card from sights, 794–796
 - setting label properties, 464–465
 - setting up Root.plist preferences, 485–486
 - shaking device to undo entries, 733–735
 - showing user location on map, 539–540
 - sight lists in, 579–581
 - steps to generic WebViewController, 616–617
 - stored data in, 436, 482–483
 - table views in, 439–440, 581–582
 - tracking changes in location, 546
 - user interface design of, 29
 - using URL loading system, 635–636
 - view controllers for, 436
 - views, 433, 435
 - window layout for, 83
 - zoom to current location and back, 552–554
- RoadTrip_Prefix.pch file, 699–700
- RoadTrip.xcdatamodel file, 674
- Root.plist file, 328–332, 482
- RootViewController class
 - about, 444–445
 - activating, 470
 - allocating and initializing data for view controller with, 507

- Core Location framework
 - with, 813
 - making UI
 - Accelerometer
 - Delegate, 813
 - modifying for RoadTrip, 446–455
 - uncommenting code in, 527, 585–587
 - using delegation with, 626–629
 - viewDidLoad method of, 497
 - working with generic
 - WebViewController, 620
 - RootViewController.m file
 - allocating and initializing
 - MapController, 533–535
 - uncommenting selection in, 585–587
 - updating for Core
 - Location framework, 807
 - rows
 - configuring for table view, 441
 - creating model for, 455–459
 - defined, 440
 - deselecting to change views, 474
 - implementing action for selected, 472–475, 580, 581–582
 - modifying number of section, 446–451
 - supplying text for, 582
 - runtime
 - class objects created at, 149
 - late binding at, 187
 - responding to
 - interruptions, 280–281
 - runtime errors, 383–384
 - runtime system
 - about Objective C's, 136, 137
 - sending object releases to, 209
-
- S
- Safari bookmarks, 32
 - sales reports for
 - applications, 122–124
 - Save button, 745, 763–764, 765
 - save: method, 765
 - saveData: method, 648–650
 - saveToContacts: method, 783–785
 - saving
 - Address Book entries, 782, 783–785
 - application state, 279–280, 475–479
 - car servicing data, 525–526
 - data with application state, 279–280
 - downloaded data, 648–650
 - label attributes, 745
 - lastView instance variable, 477
 - projects, 60, 344, 356, 444
 - variable's data, 251
 - savingCarServicingInformation method, 525–526
 - screen
 - designing, 39
 - resolution of, 17
 - size of, 17, 561
 - screenshots, 111–112
 - Scroll view, adding, 355
 - scrolling
 - adding horizontal, 352, 355
 - bounce scroll, 445
 - modifying MainWindow for, 364–366
 - Scroll view, 355, 364–366
 - scrollView method, 364–366
 - SDK (iPhone Software Development Kit). *See also* frameworks; Interface Builder; iPhone Simulator
 - building and running applications, 65–67
 - development cycle with, 57–58
 - downloading, 54–56
 - Interface Builder in, 73–76
 - introduction of, 1
 - joining iPhone Developer Program, 19, 46–51
 - MapKit framework, 532–533, 538–539
 - registering to download, 43, 44–46
 - starting Xcode projects in, 58–60
 - tools in, 51–52
 - using iPhone Simulator, 68–70
 - Search Bar
 - adding, 563–564
 - canceling searches in, 566, 567
 - turning off autocorrection for, 566–567
 - sections
 - adding, 446–451
 - computing offset for menu, 465–466
 - saving, 477

- sections (*continued*)
 - table view, 440, 441
 - titles for, 441, 452, 724
- SEL selectors, 507
- Select and Paste menu, 762–763
- selections. *See* user selections
- selectors
 - implementing view controller with, 498
 - SEL, 507
- self method
 - interchanging with `super` method, 181, 182
 - returning, 200
 - sending messages to superclass with, 197
- serializable objects, 239, 590
- ServerManager class
 - adding delegate methods to, 639–644
 - allocating and initializing instance of, 648
 - code initializing, 644–645
 - implementing, 638–639
- setter methods, 153, 154, 155, 336
- Settings application
 - adding preferences in, 327–334, 482
 - preferences to configure in, 320
- Settings bundle, 327–328, 483–484
- shaking iPhone
 - axes of accelerometer, 812, 815–817
 - beginning animation after, 818–821
- simulating, 68
 - to undo entries, 733–735, 762
- Sight class
 - adopting protocol for, 630–632
 - providing title information for, 667
- Sight objects
 - about, 581
 - adding images for RoadTrip, 599, 601
 - coding, 587–588
 - displaying, 587–589, 611–614
 - hard coding data for, 588–589
 - initializing self with dictionary, 610–611
 - loading, 604–610
 - property lists initializing, 601–603
- Sight.h file, 630–631
- SightListController class
 - adding methods to, 611–612
 - creating without using nib file, 582–584
 - displaying Sight objects in, 587–589
 - purpose of, 581
 - responding to sight selections, 654
 - uncommenting out selection of in RootViewController.m, 585–587
- SightListController.h file, 584
- SightListController.m file, 583–584
- Sight.m file
 - implementing webView ControllerNSURL Request:process Method, 647–648
 - modifying, 631
- Sight.plist
 - creating, 592–599
 - illustrated, 590
 - locating and loading, 605–606
- sights
 - entering dictionary key-value pairs for, 595–599
 - sending Road Card from, 794–796
- simulating swipe gesture, 368–369
- Simulator. *See* iPhone Simulator
- single-value properties, 776, 778
- Sleep/Wake button, 281
- SMS messages, 280
- Software Development Kit. *See* SDK
- span
 - computing for Locate button, 553–554
 - horizontalAccuracy computations for, 553
 - map's, 538
 - setting region's, 541–542
- special characters in Xcode, 237
- speed
 - animation, 335, 340–344, 345–347
 - application, 35
- SQLite, 669, 671
- stacks. *See also* Core Data Stack
 - dealing with aborted, 414
 - inverting, 412, 413

- persistence, 692
 - pushing controller onto, 475, 499
 - viewing and debugging, 388
 - working with objects in, 470
 - starting
 - compiled app in Instruments, 402–405
 - Flipside view with Main ViewController object, 293–295
 - Interface Builder, 288
 - launching Xcode, 58
 - projects, 58–60
 - specifying view displayed when, 481
 - state
 - adding information about, 758–759
 - checking application, 759–760
 - defined, 279
 - events in saving, 476–479
 - restoring application, 479–481
 - saving at termination, 279–280
 - setting `amEditing`, 760–761
 - simulating Call, 68
 - Static Analyzer (Xcode), 212–215, 396–399
 - static dictionaries, 237, 610
 - static text, 743
 - Store Kit framework, 126, 127
 - stored data mode
 - adding instructions in `applicationDidFinishLaunching:` method, 488–489
 - enabling access to stored data, 486
 - PreferenceSpecifiers
 - array selecting, 485
 - retrieving last data version, 482
 - working in, 432
 - street. *See* addresses
 - strings
 - reformatting for third-party geocoding, 573
 - representing as objects, 315–316
 - syntax for localization, 453–454
 - subclasses
 - adding methods to existing classes without, 603–604
 - adding new methods as categories, 264–266
 - creating, 179–181
 - custom cell, 461
 - customizing behavior with, 93
 - defined, 90, 164, 175
 - hierarchy of, 175–176
 - steps prior to initializing, 197
 - subviews
 - about, 84, 85
 - controls as, 85
 - setting table view label properties in, 464–465
 - super method, 181, 182
 - superclasses
 - creating, 178–179
 - declaring in class interface, 143
 - defined, 164
 - initializing, 151–152
 - invoking `init` method for, 197–199
 - understanding, 175
 - superviews, 84, 85
 - swipe gesture, 368–369
 - switch statements
 - adding title sections with, 452
 - avoiding complexity of, 174–175, 186
 - constants with, 450
 - overuse of, 136
 - symbol strings, 245
 - symbols, 307
 - synchronous downloading, 637
 - syntax
 - errors in, 382–383
 - localization strings, 453–454
 - method declarations, 145–147
 - system view controllers
 - AddressBook UI controllers, 773–774
 - using iPhone's, 773–775
-
- ## T
-
- tab bars, 87
 - Table view
 - about RoadTrip, 435, 439–440, 581–582
 - cell accessory types, 460, 463
 - cells in, 440, 459
 - computing menu section offset, 465–466
 - configuring attributes for, 445
 - data source and delegate protocols for, 441, 582
 - fetching results controller in, 706, 724
 - handling user selections on, 472–475
 - label properties in, 464–465

- Table view (*continued*)
 - modifying to add hotels, 723–724
 - moving to content view from, 468
 - navigating to with navigation bars, 468
 - plain and grouped, 442–443
 - reloading, 725
 - section titles in, 452
 - sections of, 440, 441
- tableView:cellForRowAtIndexPath:method, 723–724
- tableView:didSelectRowAtIndexPath:method
 - adding to SelectListController, 654
 - delegation changes in, 627–628
 - implementing, 472–475
 - modifying CoreData template's, 736
 - user preference implementation with, 490–491, 582
- tableView:titleForHeaderInSection:method, 724
- tapping
 - Cancel button, 567
 - entry to see next view, 499–504
 - reverting to default data on, 770
- Target-Action pattern
 - adding application behavior with, 91, 92–93
 - controls' use of, 86
 - view controllers with, 88
- target location, 809
- tasks
 - consistency with user's world, 28–29
 - knowing where you are in, 28
 - modeling applications on real world, 29–30
- templates. *See also specific templates*
 - adding second view to, 352–355
 - advantages of using, 274–277
 - Core Data with, 694–697, 713
 - Data Model, 673–676
 - memory management and, 287
 - repurposing code from, 715–716
 - selecting application, 59
 - using delegation, encapsulation, and inheritance in, 351–352
- terminating applications
 - beginning process of, 274
 - safe shut down when, 279–280
 - saving data and state when, 475–479
 - when answering phone or replying to SMS message, 280
- testing
 - alerts on Simulator, 545
 - application on real device, 402
 - DeepThoughts animation, 318
 - device in Airplane Mode, 545, 789
 - user preference settings, 332–334
- text. *See also* text fields
 - adding to cells, 461–466
 - connecting to interface, 347–349
 - connecting to text fields, 747, 748–754
 - entering e-mail, 798–801
 - formatting font color and size of label, 464
 - insetting menu, 464–466
 - modifying cell, 460
 - protocols for keyboard input of, 772
 - providing DeepThoughts substitute, 305
 - static, 743
 - supply for table view rows, 582
- Text Editor navigation bar (Xcode), 60, 61–62
- text fields
 - adding blank, 745
 - connecting name outlet item to, 750, 751, 753
 - displaying Select and Paste menu when pressed, 762–763
 - outlets connecting to text, 747, 748–754
 - resuming display of, 761–762
 - turning off autocorrection for, 772
- text views, 87
- third-party geocoding
 - reformatting strings for, 573
 - using, 559–562
- three line icon, 246, 595, 596
- thumbnails, 614–616
- time stamp, 720, 818
- timers for animation, 315–317, 820–821

- titles
 - implementing, 441, 452, 667
 - including section heads for hotel, 724
 - localizing section, 436, 452–455
 - section, 441, 452–455
 - setting map, 544
 - Sight, 667
 - Tony's Tips for iPhone Users, 32, 37, 40, 41
 - toolbar
 - adding to Map view, 559–561
 - Xcode, 60, 61
 - touch-based interface. *See* multitouch interface
 - touch events
 - handling with views, 84
 - processing, 302–303
 - Touch Up Inside events
 - Info button reaction to, 298–299, 302–303
 - processing, 302–303
 - tracing, de-allocation of objects, 221
 - tracking, orientation and motion, 15–16, 38
 - trademarks, 115
 - Transaction class, 261–263
 - transparency in code design, 167
 - triangles, disclosure, 246, 596
 - trip instance variable, 522
 - Trip object
 - composition techniques for, 528–530
 - initializing, 505–507
 - loadSights method for, 602–603
 - passing to delegate for selections, 628
 - returning delegate object based on selection, 652–653
 - returnWeather: method for, 511
 - sending loadWebView: message via, 512
 - updating methods for, 654
 - using view controllers with, 497, 504–505
 - weatherRealtime method, 511–512, 621–622
 - Trip.h file
 - adding properties and methods for hotels, 710–711
 - removing annotation methods from, 666–667
 - Trip.m file
 - adding items for Core Data Stack use, 698–700
 - modifying annotation methods in, 668
 - pasting Core Data Stack code into, 695–696
 - tripSelector: method
 - replacing with delegate assignment, 628
 - used in generic controller, 620
-
- ## U
-
- UIAccelerometer class, 812
 - UIAccelerometerDelegate, 813
 - UIApplicationDelegate object, 307, 308
 - UIApplicationMain function
 - calling with main function, 272
 - lifecycle after calling, 272–274
 - sending application delegate launch messages, 273
 - work done by, 274–276
 - UIImagePickerController class, 370
 - UIImagePickerController
 - Delegate protocol, 276
 - UIKit framework
 - customizing behavior with, 93
 - handling interruptions with, 281
 - illustrated, 274
 - MVC design pattern in, 82
 - responses to low-memory warnings, 282–283
 - setting up event loop with, 273–274
 - UIModalTransitionStyleFlipHorizontal, 757
 - UIScrollView class
 - adding scrolling with, 355
 - modifying windows for scrolling with, 364–366
 - UISearchBar class, 563–564
 - UITableView objects, 441
 - UITableView view, 435, 440
 - UITableViewCell class, 460

- UITableViewController
 - adding sections to, 446–451
 - using in RoadTrip, 444–445
- UITableViewDataSource
 - protocol, 441, 445
- UITableViewDelegate
 - protocol, 441
- UITextInputTraits
 - protocol, 772
- UIView class
 - about, 291
 - FlipsideView subclass of, 295
 - setting up new Photo view subclass of, 356–358
 - using for animation block, 317
- UIViewController class
 - about, 291
 - setting up new Photo view subclass of, 358–359
- UIViewController
 - SubClass template, 499, 500, 741–742
- UIWebView class, 501
- UIWebView Delegate
 - protocol, 515
- UIWebView view controller
 - extensibility with, 498
 - implementing goBack: method from, 514–515
- UIWindow class, 359
- uncommenting code, 527, 585–587, 712–713
- Undo feature. *See also* Undo manager
 - Core Data support for, 726, 728
 - deleted user preferences, 486
 - shaking to use, 733, 734, 762
- Undo manager
 - cleaning up, 730, 731
 - display reverts to default with, 770–771
 - setting up, 729–730
- undoManagerDidRedo:
 - method, 732
- undoManagerDidUndo:
 - method, 732
- uninstalling apps from Simulator, 69
- updating
 - applications for attention, 131
 - dictionaries, 253
- uploading app to App Store, 112–114
- URL loading system
 - about, 635–636
 - authentication and credentials for, 636–637, 645–646
 - canceling and releasing connections, 641
 - canceling requests, 640
 - Car Information
 - modifications for, 658–659
 - Car Servicing
 - modifications for, 655
 - connection errors, 642–643
 - cookies and, 637
 - handling authentication challenges, 645–646
 - implementing Sight for asynchronous downloads, 647–651
 - loading Weather asynchronously, 659–660
 - protocol support in, 637
 - writing data to disk, 636
- URLs. *See also* URL loading system
 - canceling and releasing connections to, 641
 - canceling requests to, 640
 - connection errors to, 642–643
 - making asynchronous connection to, 639–640
 - retrieving geocoded information using, 573
- user experience
 - creating, 25–27
 - designing apps with depth, 26–27
 - engaging users, 33
 - interface design for, 34–36
 - making tasks consistent with, 28–29
 - modeling actions on real world, 29–30
 - standard gestures and, 32
- user interface. *See also* multitouch interface
 - adding instance variable to, 478
 - Address Book's AB
 - PeoplePicker
 - Navigation
 - Controller, 774
 - Apple rejection policies for, 39–41
 - asking to use current location, 809, 810
 - building application's, 747
 - Cancel and Save buttons for hotels, 745, 763–765
 - canceling Address Book lookup, 782
 - compelling design of, 25–27
 - consistency in, 31–32
 - creating to add hotels, 741–749

- designing experience
 - with, 34–36
 - developing, 52
 - dismissing keyboard after entry, 761–762
 - dismissing modal view
 - controllers, 757–763, 765–766, 783
 - engaging users with, 33
 - form following function, 31
 - Human Interface
 - Guidelines for, 475, 514
 - implementing user preferences for, 482–487
 - incorporating device
 - context in, 14, 17–18, 38–39
 - Library components for, 743
 - loading quickly, 279
 - Mail, 787–789
 - making function obvious, 32–33
 - making tasks intuitive, 28–29
 - modal view controllers
 - for, 756–757
 - modeling on real world tasks, 29–30
 - providing quality information, 36
 - relevance in, 34
 - Road Cards, 790–796
 - speed and performance of, 35
 - state information for, 758–760
 - tracking orientation and motion, 15–16, 38
 - transition view styles for, 757
 - using knowledge base for, 30
 - view objects and
 - consistency of, 86
 - Xcode project window, 60–65
 - user interface objects, 162
 - user selections. *See also*
 - buttons; keyboard
 - adding contacts to Hotel list, 773
 - choosing functions from rows, 580, 581–582
 - choosing street addresses, 781
 - creating
 - WebViewController in response to, 582
 - deselecting rows to change view, 474
 - implementing in
 - RoadTrip, 472–475
 - making new Road Card entry, 792–793
 - making phone number selections, 779–789
 - preparing for Sights main view, 585–587
 - responding to, 467–469, 582
 - returning delegate objects based on, 652–653
 - saving new addresses, 782, 783–785
 - selecting Address Book contacts, 774, 775, 777–781
 - using generic controller, 620
 - userLocation value, 546
 - Utility Application template
 - Flipside view of, 292–295
 - Info and Done buttons for, 295–302
 - initializing Main view, 289–290
 - Main view of, 288–292
 - memory warning methods in, 282–283
 - reviewing, 271
 - running, 287–288
 - views in, 287
 - utility applications. *See also* Utility Application template
 - Apple’s perspective on, 20
 - characteristics of, 319
 - choosing template for, 287
 - overview of guide, 5
 - value added by, 26
 - utility classes of
 - Foundation framework, 228–229
-
- ## U
- values
 - changing constants, 614
 - combining with key, 246–249
 - looking up, 236, 237–238
 - setting up for key-value pairs, 328, 595–599
 - storage limitations for
 - nil, 237
 - updating key, 253
 - using, 236–237, 609
 - variables. *See also* instance variables
 - NSZombieEnabled environment, 416, 417
 - saving data in, 251
 - versions of Objective-C, 136
 - vertical modal view, 757
 - vibrating applications, 40
 - video, 16

- view controllers
 - about navigation
 - controllers, 468–469
 - about
 - UIViewController
 - class, 291
 - coding to retrieve display
 - content, 497
 - creating without using nib
 - file, 582–584
 - delegation with, 623, 625–629
 - designing reusable, 498, 501–504
 - dismissing modal, 757–763, 765–766, 783
 - illustrated, 89
 - initializing in application
 - delegate, 352–355
 - modal, 756–757
 - providing data with, 436
 - RoadTrip, 436
 - setting up for AddHotel
 - Controller, 747–749
 - steps to old school
 - generic, 616–623
 - system, 773–774
 - Target-Action pattern
 - with, 88
 - Trip model object with, 497, 504–505
 - using, 87–89
 - view objects
 - about, 75, 80, 81, 188
 - illustrated, 82
 - viewDidLoad method
 - AddHotelController's, 763–764
 - adding to RoadTrip Web
 - view, 510–511
 - Back Button navigation
 - with, 512–515

- changing background
 - color with, 315
- creating row model array
 - and dictionary with, 457–458
- hard coding Sight object
 - data, 588–589
- HotelController's, 716–717
- MapController.m's, 663–664
- updating for delegation, 628–629
- when to send, 311
- views
 - adding Search Bar to, 563–564
 - adding second, 352–356
 - animating, 312–317
 - connecting to methods, 361–364
 - container and display, 86
 - Content, 83, 84, 85
 - creating nib file for new, 359–364
 - data entry, 741–743
 - defined, 83
 - hiding animation, 821
 - hierarchy for, 84–85
 - implementing Photo, 356–366
 - interacting with users
 - with, 84
 - kinds of UIViewController, 85–87
 - modal, 88
 - RoadTrip, 433, 435
 - Scroll, 355, 364–366
 - single view controller for
 - multiple, 498, 501–504
 - specifying display for
 - launch, 480–481

- table, 440
- timing display for
 - animation, 820–821
- view initialization, 311
- viewWillDisappear:
 - method, 733, 758

W

- Weather feature
 - implementing URL loading
 - for, 659–660
 - offline message for, 497, 498
 - real-time access for, 430, 431, 496, 497
 - specifying content for, 496–498
- Weather model object
 - implementing, 507
 - using generic controller
 - with, 619, 620–622
- weatherRealtime
 - method, 511–512
- Web servers. *See also* URLs
 - downloading data from, 632–634
 - providing application data
 - on, 579
 - verifying connection with, 641
- Web sites
 - adding links to, 128, 512–515
 - disabling links to, 515–518
 - loading data from, 622–623
 - Neal's, 3
 - retrieving content from, 496, 619–622
 - weather, 435

Web views
 about RoadTrip, 435
 defined, 87
 interface guidelines for, 514
 managing Web site links on, 512–515

WebViewController class
 adding delegation to, 625–630
 adopting protocol for, 630–632

CarInformation object
 initialization for, 507, 518–523

CarServicing object
 initialization for, 518–519

configuring Back button, 512–515
 creating, 582
 declaring initializer for, 507, 509
 generic, 616–617, 619–622
 initializing with message needed to send, 507–508
 interface for, 508
 instantiating and initializing, 626–629
 protocol declarations in, 624–625

WebViewController.h file, 625–626

WebViewController.m file, 499, 500, 510–511

webViewControllerNS
 URLRequest:processMethod, 647–648

WebViewController.xib file, 501, 502–503

windows
 about application, 82–83
 displaying attributes of, 276–277
 modal views for dialogs, 88
 navigation controller view for, 468–469
 programming function of, 87
 switching between file, 306

wrapper objects, 591

writeToFile:: method, 591, 609, 649, 650

writing URL data to disk, 636

X

Xcode. *See also* nib files
 about, 4, 51
 adding files to project, 368–369
 adding new classes to projects, 181
 build configuration in, 65
 bundles in, 458–459, 605
 documentation for, 306–310
 finding memory leaks, 212–215, 396–399, 405–407, 408–413
 fixing memory leaks, 215–221

Library interface
 components in, 743
 naming projects, 60

project window features in, 60–65
 provisioning profiles required for, 97–98
 saving projects, 60, 344, 356
 setting preferences for, 71–73
 special characters in, 237
 starting new projects, 58–60
 welcome screen, 58

Xcode Model Editor, 674–676, 682–686

Xcode Text editor
 debugging in, 385–391, 394–396

Documentation window for, 307–308
 features of, 305–306
 finding items in project, 309–310
 Help menu for, 309
 Quick Help in, 306–307

.xib files. *See* nib files

Z

zombies
 enabling, 416–423
 finding de-allocated objects, 414–416
 tracking down, 407–413

zoom
 to current map location, 552–554
 zoom levels for region, 542

