

# Index

## ■ Special Characters and Numerics

- +isMultithreaded method, 19
- 2s.png file, 263
- 32-bit RGBA format, 84
- 3D model exporting, 65–69
- 3D objects, as GUI items, 243–246
- 3ds Max, 59, 61, 65–66, 68
- 4-channel (RGBA) image, 148
- 8 bits per channel, 148
- 16-bit integer type, 62

## ■ A

- acceleration, 218
- accelerometer data, 214
- action dictionary entry, 317
- action-loc-key, 321
- actionWithDuration parameter, 271
- active record fetchers for Core Data tool, 129
- Active Record Fetching for Core Data library, 131
- Active Record Fetching helper library, 130
- Active Target XCode toolkit, 255, 281
- Add Existing Framework option, 48, 128
- Add New Layer option, 241
- addChild function, 270
- addDependency: method, 26
- addOp: message, 35
- addOp: method, 28
- addOperation: method, 25
- Advanced RISC Machine (ARM), 5, 142–144, 157
- afterRender method, 187
- AI (Artificial Intelligence), 158
- Alert notification option, 314
- Align With View menu option, 211
- ambient light setting, 213
- ambient light settings, 213, 216
- Android phones, 311
- animateCard function, 279
- animated warrior model, 219
- animation import settings, 233–237
- animations, 232
- Apache MINA framework, 318
- APChar class, 175–176
- APFontMap class, 173
- APFontmap.m file, 185
- APGlyph class, 174, 179
- API (Application Programming Interface), 6, 93–94, 193, 252, 317
- API key, 10–13, 15
- APNS (Apple Push Notification Service), 318–322
- APNSConnection class, 319
- APNSFeedbackDecoder class, 319
- APNSNotificationEncoder class, 319
- Apple Development Devices Announcement, 324
- Apple documentation, 17
- Apple Image Picker, 147
- Apple iPad Programming Guide, 281
- Apple OpenGL guide, 72
- Apple Push Notification servers, 316, 323, 325
- Apple Push Notification Service (APNS), 318–322
- Apple World Wide Developers Conference, 4
- application delegate class, 8
- application delegate header file, 8
- Application Programming Interface (API), 6, 93–94, 193, 252, 317
- applicationDidFinishLaunching method, 8, 314
- applications
  - Cocoa, data-driven architecture
    - choosing approach, 104
    - connecting data to UI, 102–104
    - data model design, 101–102
  - data driven, 102
  - English-only, 94
  - fCookie, 167–168
  - Interestingness, adding JSON parsing framework to, 10
  - Interestingness, changing to use NSOperationQueues, 37–38

- Interestingness, changing to use
    - NSOperationQueues and blocks, 50–52
  - Interestingness, converting to use official version of blocks, 53–54
  - iPhone, unit testing, 109–134
  - texture atlas, opening, 173
  - Applications directory, 168
  - applyDamage method, 239
  - APRenderController class, 178, 181–182, 184, 187
  - APRenderController instance, 180
  - APRenderController.m file, 183–184
  - Apress web site, 142, 167
  - aps dictionary, 319, 321–322
  - APTText class, 177–180, 183
  - arandom(time(NULL)) method, 266
  - arc4random function, 183
  - arc4random() method, 266
  - ARM (Advanced RISC Machine), 5, 142–144, 157
  - armV6 architecture, 144
  - Artificial Intelligence (AI), 158
  - ascenders, 166
  - ascent variable, 174
  - ASCII (text-based format), 65
  - ASCII file format, 66
  - assert macros, 118
  - asset management interface, 194
  - asset pipeline, 198–199
  - assets, importing, 204
  - assets/converted folder, 86
  - assets/converted/geometry folder, 76
  - Asynchronous methods, 17, 19
  - attack animation, 237–239
  - attack command, 239
  - attack method, 238
  - attack() method, 239
  - attribution, 95
  - audio handling, 194
  - Audio Listener component, 241
  - Audio MIDI Setup app, 280
  - audio player coding
    - connecting play/pause button, 304–305
    - connecting skip controls, 305
    - overview, 302–303
    - providing player state in user interface (UI), 305
    - understanding audio sessions, 306–307
  - audio sessions, 306–307
  - Audio Toolbox framework, 306
  - Audiobooks app, 289–292
  - Audio-Centric apps for iPhones
    - designing
      - book information screen, 292–293
      - book selection screens, 292
      - Home Screen, 291–292
      - player screen, 293–294
      - for target users, 290
    - example project
      - coding audio player, 302–307
      - overview, 296–297
      - setting up user interface (UI), 298–301
    - implementation, 295
    - overview, 289
  - AudioExampleViewController.h file, 298, 303
  - AudioExampleViewController.xib file, 299
  - AudioSession class, 306
  - autorelease pool, 18–19, 21, 28, 33, 41, 52
  - AVAudioPlayer class, 295–296, 303–305
  - AVAudioPlayer variable, 303
  - AVFoundation classes, 295, 303
  - AVFoundation framework, 302–303
- ## B
- backButtonClicked method, 305
  - backingHeight dimension, 179
  - backingWidth dimension, 179
  - Badge notification option, 314
  - barrel button object, 243–245
  - Barrel mesh component, 200–201
  - barrelButton object, 245–246
  - barrelButton prefab object, 243
  - barrel-cylinder material component, 201
  - barrelTextureHigh image, 204
  - barrelWithCollider prefab object, 231
  - BBAffordable script, 237, 239
  - BBCharacterController script, 220–221, 228, 236, 238
  - BBSimpleFollower script, 229
  - BBSpawnBarrelsAction script, 245
  - BBTouchable class, 227–228, 238, 244
  - BBTouchableButton script, 244–245
  - BBTouchInputController script, 226–228, 246
  - beforeRender method, 184
  - beta program, extending, 324–325
  - BIN variable, 143
  - bits per channel, 148
  - bits per pixel (bpp), 85
  - Blender, 59, 61, 63, 65, 68
  - block literal expression, 50
  - Block statements, 48
  - blockOperationWithBlock: method, 52
  - blocks
    - adding PLBlocks framework, 48–50
    - changing Interestingness application to use NSOperationQueues and blocks, 50–52

- converting Interestingness application to use official version of blocks, 53–54
  - overview, 46–47
- Blocks Group folder, 50
- Blocks language extension, 4–6, 45, 48, 50
- blurring, 156–157
- Bonzai font, 168
- book information screens, 292–293
- book selection screens, 292
- bookmarking, 296
- box collider component, 219–220, 225–226
- Box Collider menu option, 226
- Box2d application template, 256
- Box2d physics library, 252, 256, 286
- bpp (bits per pixel), 85
- Build and Debug button, 255
- Build and Go button, XCode toolkit, 255
- Build and Run button, 152
- Build and Run option, 118, 125, 297
- Build Settings option, 247
- Build tab, 114–115, 147
- Build Target, 112, 115, 122
- building, for devices, 247
- Bumgarner, Bill, 110
- buttonPressed: method, 31, 35
- buzz.aif file, 319
- Byröd, Martin, 137

## C

- C Sharp Script menu item, 205
- calculate function, 278
- callback method, 314
- CallFunc method, 272
- Camera menu option, 241
- cameras, 69, 141, 241
- cancelAllOperations message, 25
- Capsule collider component, 225, 230, 234
- Capsule Collider menu option, 230
- card CCSprite object, 272
- Card class, 262
- Card.h file, 261–262, 264, 267
- cardHomePoint method, 274, 285
- cardImages NSMutableArray object, 274
- Card.m file, 261–264, 267, 285
- cards, 262–264, 284
- card-slide.wav file, 280
- Carraud, Emmanuel, 138
- cascade initialization, 154
- categories, Objective-C, 124–127
- CCDirector object, 252, 259–260, 270, 274
- CCLabel object, 253, 273
- CCLayer object, 252–253, 259, 272
- CCMenuItem object, 260, 283
- ccp() method, 270
- CCScene object, 252, 259, 261, 270
- CCSprite class, 271
- CCSprite object, 252–253, 271–272, 274
- CCTextureCache object, 279
- ccTouchesBegan method, 276
- cd command, 171
- center Y property, 219
- CGBitmapContextCreate() method, 148
- CGImage class, 147, 149, 154
- CGPoint object, 270
- CGPointMake() method, 270
- CGRect object, 277
- character block model, 230
- character controller component, 219, 222–223
- Character Controller menu option, 218
- Character game object, 218
- Character Model variable, 235–236
- Character object, 218, 220–222, 224, 228, 230, 236–237
- CharacterController component, 218–220, 222–224, 228, 234
- characterModel.animation.CrossFade("Idle") method, 236
- characterModel.animation.CrossFade("Walk") method, 236
- characters, main, 218–224
- charCount variable, 178
- chars array, 177
- Checkers solver app, 159
- Cheetah3D, 63, 65, 204, 232
- child view controller, 104
- Chipmunk application template, 256
- Chipmunk physics library, 252, 256, 270
- Classes folder, 7, 32, 40, 256, 258
- Clear Flags option, 241
- client implementations, of Push Notification service, 314–316
- client protocol, 317
- Cocoa APIs, 123, 194
- Cocoa application, data-driven architecture
  - choosing approach, 104
  - connecting data to UI
    - delegates, 102–103
    - notifications, 103
    - setter propagation, 104
  - data model design, 101–102
- Cocoa frameworks, 19, 110
- Cocoa Threads, 17
- Cocoa Touch applications, 18, 165
- Cocoa Touch class templates, 7
- Cocoa Unit Testing Framework space, 110
- Cocoa wrapper, 247
- Cocos2d

- adding sound, 280–281
- game logic
  - cards, 262–264
  - deck, 264–266
  - events, 275–280
  - overview, 261
  - SimpleHand.m file, 266–269
  - sprites, 270–272
- getting started with, 252–253
- installing
  - configuring sample code, 254–255
  - installing XCode project templates, 255–256
  - starting new project, 256
- making scenes, 258–261
- origins of, 251
- reasons for using, 252
- supporting iPad, 281–285
- Video Poker, 257–258
- cocos2d-iphone.xcodeproj XCode project, 254
- CocosDenshion Cookbook, 281
- CocosNode object, 258
- code outline, example of, 64–65
- code.tar.gz file, 171
- coding audio player
  - connecting play/pause button, 304–305
  - connecting skip controls, 305
  - overview, 302–303
  - providing player state in user interface (UI), 305
  - understanding audio sessions, 306–307
- colliders, 224–228
- collision detection, 219
- collision dynamics, 219
- color matching, 159
- color options, 203
- color shader, 186
- color space object, 148
- colorRenderBuffer, 179
- colorSpace reference, 148
- .COM bubble, 289
- comment tab, MusicPlayer.xcdatamodel file, 130
- communication flow, of Push Notification service, 314
- compile errors, 207
- compiler flags, 143
- compiler option, Build tab, 114
- component name right click menu, 220
- Computer Vision, on iPhones
  - defined, 140–141
  - face detector application, creating
    - adding image conversion functions, 147–149
    - creating simple GUI, 149
    - face blurring, 156–157
    - loading Haar cascades, 152–154
    - loading images from photo library, 151–152
    - performing face detection, 154–156
    - setting up OpenCV, 142–144
    - setting up XCode, 145
  - improving performance, 157–159
  - other projects, 159
  - overview, 137–139
  - reasons for, 141–142
- concurrency, improving responsiveness using non-responsive user interfaces
  - adding JSON parsing framework to Interestingness application, 10
  - building Interestingness user interface, 7–9
  - composing RESTful request for list of Interestingness images, 10–12
  - overview, 6
  - RESTful request and JSON parser, 12–13
  - UITableViewDataSource protocol methods, 14–15
  - with NSThread and NSObject objects, 19–24
  - with operation objects
    - NSOperation and NSInvocationOperation, 26–45
    - NSOperationQueue, 25–26
    - overview, 24
    - with operation objects and blocks, 45–54
    - overview, 3–5
    - using concurrent solutions, 17–18
- Connecting IBAction methods, 301–302
- connection:didFailWithError: method, 19
- connectionDidFinishLoading: method, 19
- connection:didReceiveData: method, 19
- connection:didReceiveResponse: method, 19
- constraints, external, 97
- consumption
  - direct-client, 99
  - server-intermediary, 100–101
- content creation tools, 61, 63, 68–69, 71
- content pipeline, creating, 61–64
- contentSize method, 277
- Continuous Integration environment, 134
- Continuous Integration system, 134
- Controller called attack() method, 238
- controls, connecting, 305
- conversion routines, 75
- converted data rendering on iPhone, 86–88
- converted shark model, 64, 87
- converter directory, 64–65, 86
- converter tool, running, 86

- converter/converter.xcodeproj file, 86
- converter/main.cpp file, 69, 73
- ConvertHierarchy() method, 69, 73
- CoolerUnity folder, 195
- Coordinate systems, 68, 172
- coordinates, mapping, 171–172
- Copy Files option, 121–122
- Copy items into destination group's folder
  - check box, 145–146
- copy to destination folder option, 10
- Core Audio framework, 295
- Core Data entities, 128
- Core Data Entity Modeling tool, 127, 129
- Core Data framework, 110, 127–128, 130–132, 292
- Core Data helper files, 131
- Core Data models, 127–132
- Core Data stack, 128, 131
- Core Data store, 117, 128, 292
- Core Data team, 110
- CoreData framework, 117
- CoreData.framework library, 128
- CoreGraphics.framework, 114
- coroutines, 209–210, 239
- cpv() method, 270
- cpVect object, 270
- CPVRTexture object, 85
- Create Empty menu option, 218, 227
- create header file checkbox, 7
- createObjects method, 183
- cross platform development, 192–193
- cross platform support, 192
- Cube menu option, 219
- cubelt variable, 46–47, 50
- cubeltTimesAFactor block variable, 50
- Culling mask drop down menu, 241
- Culling Mask option, 241
- curCard parameter, 275
- CURL utility, 143
- currentTime attribute, 305
- custom scripting, 205–207
- cv\* functions, 155
- cvaux/include header file, 144
- cvCreatelImage() method, 148
- cv.h file, 147
- cvHarrDetectObjects() method, 158
- cv/include header file, 144
- cvLoadImage method, 154
- cxcore/include header file, 144
- delegates, 102–103
  - notifications, 103
  - setter propagation, 104
- coverage and accuracy, 94
- source consumption, design patterns for, 99–101
- data exchange file format, vs. in-game file format, 63–64
- data model design, 101–102
- data sources, choosing
  - API design, 93–94
  - data coverage and accuracy, 94
  - economics, 95–96
  - overview, 92
  - trials, 96
- dataWithContentsOfURL method, 174
- Deal button, 257, 284
- dealCards function, 280
- dealloc method, 8, 28, 35
- Debug configuration, 111, 115
- deck, 264–266
- Deck class, 273
- Deck.h file, 261, 264
- Deck.m file, 261, 265
- default NSManagedObjectContext instance, 130
- default XCode Project, 111
- defaultFrameBuffer, 179, 181
- Delaunay Triangulation, 159
- delegates, 102–103
- dependencies property, NSOperation class, 26
- descent variable, 174
- detachNewThreadSelector:toTarget:withObject : method, 23
- Detect Faces button, 151
- detectFaces method, 151
- DEV variable, 143
- developer accounts, 324–325
- development environment certificate, 324
- device token, 314–315, 317, 319
- devices, building for, 247
- deviceToken object, 315
- deviceToken variable, 315
- die() method, 238
- Direct Dependencies section, 112
- direct management, 295
- direct-client consumption, 99
- directional attribute, 223
- disaggregateInterestingnessList: method, 19, 21–22, 40
- Dismiss button, 322
- dispatch queues, 46
- display, setting up, 181–182
- Distribution mode, 325
- doButtonAction() method, 245

## **D**

### data

- connecting to UI

drag tool, 203  
 Dragon's Den television program, 139  
 drawOnFaceAt:InImage method, 154, 156  
 duration attribute, 128  
 duration method, 132–133  
 DXT format, 80  
 DXTC format, 80

**E**

EAGLView view, 180  
 eBuddy application
 

- client / server protocol, 317
- eBuddy iPhone application, 312–313
- eBuddy Messenger application, 312
- eBuddy Pro application, 323–324
- fitting parts together, 322–323
- overview, 311
- server to APNS, 318–322

 eBuddy Pro application, 323–324  
 economics, 95–96  
 edge angle property, 152  
 edge intensity property, 152  
 edge polarity property, 152  
 Edit Project Settings menu option, 147  
 Emotion detector project, 159  
 empty container object, 203  
 empty game object, 199  
 empty methods, 206  
 empty objects, 199–200  
 English-only applications, 94  
 Entity class, 127  
 Entity data, 130  
 Entity instance, 127  
 Entity Modeling tool, 127  
 Epsilon values, 76  
 error variable, 315  
 ES1Renderer.mm file, 87  
 ESRenderer class, 179  
 events
 

- combining many actions together, 278–279
- detecting sprite taps, 276–277
- overview, 275
- switching sprite image, 279–280

 example.mp3 file, 302  
 Existing Files option, Add menu, 86, 302  
 exporter plug-ins, 65  
 exporting 3D models, 65–69  
 Extensible Markup Language (XML), 6, 154, 158  
 external constraints, 97

**F**

face blurring, 156–157  
 face detector application, creating
 

- adding image conversion functions, 147–149
- creating simple GUI, 149
- face blurring, 156–157
- loading Haar cascades, 152–154
- loading images from photo library, 151–152
- performing face detection, 154–156
- setting up OpenCV, 142–144
- setting up XCode, 145

 FaceDetect project, 145  
 FaceDetectViewController.h file, 147, 149, 151  
 FaceDetectViewController.m file, 148, 151, 154  
 FaceDetectViewController.xib file, 149  
 factory method, 258  
 Faux Character model, 220, 228, 234  
 Faux Character object, 219, 229  
 faux gui layer objects, 241, 244  
 faux-joysticks, 215  
 FBX exporter dialog box, 65–66  
 FBX files, reading, 66–69  
 FBX models, converting into triangle data, 73–76  
 FBX reader library, 63  
 FBX scene node, 77  
 FBX SDK, 65–68, 74  
 FbxResourceConverter::convert() method, 74, 76  
 FbxResourceReader class, 66–67  
 FbxResourceUtils class, 75  
 FbxResourceUtils::processTriangles() method, 74, 76  
 fCookie application, 167–168  
 FeedbackNotification class, 319  
 FeedbackReceiver class, 319  
 fetchImage: method, 23  
 FetchImageOperation class, 39–44, 52  
 FetchImageOperation.h file, 43, 50  
 fetchInterestingnessList method, 19, 21–22, 39–40  
 file formats, data exchange vs. in-game, 63–64  
 File's Owner element, 300  
 File's Owner icon, 30  
 Files' Owner icon, 30  
 File's Owner icon, 31  
 find\_if method, 76  
 FindObjectOfType() method, 221  
 flashlight app, 297  
 flexible content pipeline, creating, 61–64  
 Flickr API key, 11, 15  
 flickr.interestingness.getList API, 6, 11, 19, 37  
 FlightTrack application

- choosing data sources
  - API design, 93–94
  - data coverage and accuracy, 94
  - economics, 95–96
  - overview, 92
  - trials, 96
- current status, 105–106
- data-driven Cocoa application architecture
  - choosing approach, 104
  - connecting data to UI, 102–104
  - data model design, 101–102
- design patterns for data source
  - consumption, 99–101
- overview, 91
- release of, 105
- source-driven user interface design
  - challenges, 97–98
  - overview, 96
  - techniques, 98–99
- floating-point (FP), 158
- Floor plane object, 216
- font techniques, 163
- font0.plist file, 171, 173
- font0.png file, 171, 173
- Fontery classes
  - APChar, 175–176
  - APGlyph, 174
  - APText, 177–179
  - overview, 173
- Fontery folder, 173
- fontName string, 185
- Fonts group, 180
- fonts in OpenGL. *See* OpenGL fonts
- format parameter, 11
- FortuneCookie project, 173
- fortunes
  - creating, 182–184
  - displaying, 184–187
- Fortunes array, 182–183
- forwardButtonClicked method, 305
- Foundation.framework, 114
- FP (floating-point), 158
- framebuffers, 179, 181
- framework, JSON parsing, 10
- Frameworks folder, 48, 128
- FTFlightListController, 104
- FTFlightManager class, 104
- functions, adding image conversion, 147–149
- adding touchable objects, 230
- animation import settings, 233–237
- animations, 232
- building for devices, 247
- inputs and colliders, 224–228
- interacting with objects, 237–239
- main characters, 218–224
- multiple cameras, 241
- prefabs, 231
- Unity interface
  - adding lights, 213–214
  - asset pipeline, 198–199
  - coroutines, 209–210
  - custom scripting, 205–207
  - game view, 197, 210–211
  - hierarchy view, 198
  - importing assets, 204
  - inspector view, 198
  - meshes, renderers, and materials, 200–203
  - playing, 208–209
  - project view, 198
  - reasons for using, 193–195
  - scene view, 196–197
  - Transform component, 199
  - using iPhones as game pads, 214
  - user interface (UI), 240
- game pads, using iPhones as, 214
- game view, Unity interface, 197, 210–211
- gameObject class, 206–207, 218, 227, 245
- gameObject instance variable, 206, 229
- gameObject.Find() method, 245
- gameObject.transform component, 207
- games, starting on iPhone, 59–60
- gaming consoles, 141
- GCC (GNU Compiler Collection), 147
- GCC 4.2 compiler, 53
- GCD (Grand Central Dispatch), 4, 17, 45
- Generation Gap pattern, 129
- genfont utility, 173
- Genres features, 291
- Geometry class, 72–73, 76
- Geometry::Vertex class, 72
- Get Info option
  - MediaPlayerTests target Control-click menu, 112
  - MediaPlayer.xcdatamodel file, 130
  - target control-click menu, 302
- GET method, 11
- GetComponent() method, 222, 227
- getImageForURL: method, implementing, 42–45
- GetMouseButton() method, 226
- GetNodeAttribute()->GetAttributeType() method, 71

## G

- game development
  - 3D objects as GUI items, 243–246
  - adding terrain, 216–218

GHTestCase class, 111, 117, 124  
 GHUnit file, 111  
 GHUnit framework, 110–111, 114, 117–119  
 GHUnit libraries, 115  
 GHUnit unit test runner application, 119  
 GHUnitTestCase class, 117  
 glDisableClientState method, 186  
 glDrawArrays method, 187  
 glEnableClientState method, 186  
 GlobalVariables.h file, 261–262  
 glTexCoordPointer method, 186  
 glTranslatef method, 184  
 glyphName variable, 174  
 GNU Compiler Collection (GCC), 147  
 Google Code repository, 10  
 Google Mac Toolkit, 111  
 Google Street View, 156  
 GPU (Graphics Processing Unit), 5, 72, 79–80, 85  
 Grand Central Dispatch (GCD), 4, 17, 45  
 graphical user interface (GUI), 80, 97, 149, 240  
 graphics pipeline, 181, 186  
 Graphics Processing Unit (GPU), 5, 72, 79–80, 85  
 grep utility, 144  
 Grey Barrel object, 205  
 Group photo fun project, 159  
 Groups & Files pane, XCode, 86, 112  
 GUI (graphical user interface), 80, 97, 149, 240  
 GUI items, 3D objects as, 243–246  
 GUI.Box object, 240  
 GUI.Button object, 240  
 GUICamera attributes, 241–242  
 GUICamera object, 241, 244, 246  
 GUICamera position property, 243  
 GUIText class, 240  
 GUIText object, 240  
 GUITexture class, 240  
 Gutenberg, Johannes, 164–167, 171, 175–176, 180

## H

Haar cascades, loading, 152–154  
 Haar object detection, 152–153, 159  
 haarcascade\_frontalface\_alt\_tree.xml file, 153  
 Handford, Gabriel, 110  
 handLabel object, 284  
 handleTouchAtPoint( ) method, 226, 246  
 handleTouchAtPointForAllCameras( ) method, 246  
 handling textures
 

- converting images into PVRTC format, 84–85

image compression vs. texture
 

- compression, 79–80
- Imagination's PVRTC format, 80–81
- reading PNG images, 82–84

 header files, 144, 154  
 height property, 219, 230  
 HelloOperationQueues
 

- building, 28
- building user interface for, 30–31
- implementing, 33–36

 HelloOperationQueues project, Xcode, 28  
 HelloOperationQueuesViewController class, 28  
 HelloOperationQueuesViewController.m file, 33  
 HelloOperationsViewController.xib file, 30  
 HelloWorldScene.h file, 256, 258, 260  
 HelloWorldScene.m file, 256, 258, 260  
 hierarchy view, Unity interface, 198  
 hit.transform.gameObject method, 227  
 Home Screen, 291–292  
 HotelPal application, Mobiata, 100–101  
 HowToPlay node, 258  
 HowToPlayScene.h file, 258–259  
 HowToPlayScene.m file, 258–259  
 HTTP (HyperText Transfer Protocol), 6, 10–11  
 HyperText Preprocessor (PHP), 11

## I

IB (Interface Builder), 30–31, 127, 142, 149, 151, 299  
 IBAction methods, 299, 301  
 IBOutlet declarations, 299  
 "Idle" (characterModel.animation.CrossFade) method, 236  
 idle animation, 223, 237–238  
 Ignore Raycasts layer, 229, 241  
 IM (Instant Messenger), 312, 323  
 Image class, 83  
 image compression, vs. texture compression, 79–80  
 image conversion functions, adding, 147–149  
 Image Picker, Apple, 147  
 image tagging, 141  
 Image View, 149–150  
 imageData element, 148  
 imageData object, 52  
 imageDictionary class, 44  
 imagePickerController:didFinishPickingImage:editingInfo method, 154  
 Image::R8G8B8 property, 83  
 Image::R8G8B8A8 property, 83  
 images
 

- converting into PVRTC format, 84–85
- loading from photo library, 151–152



- Images folder, 204, 216
- imageTitles mutable array, 13
- imageURL variable, 13, 40
- imageURLs array, 14
- ImageView boundaries, 149
- imageView outlet, 150
- imageView property, 14, 40, 43
- imageview variable, 149
- Imagination PowerVR SDK, 80, 84
- Imagination Technologies, 4–5
- Imagination's PVRTC format, 80–81
- importing 3D art assets into iPhone games
  - advantages of writing tools, 60–61
  - creating flexible content pipeline
    - data exchange vs. in-game file formats, 63–64
    - tools problem, 61–62
  - example code outline, 64–65
  - exporting 3D models, 65–69
  - handling textures
    - converting images into PVRTC format, 84–85
    - image compression vs. texture compression, 79–80
    - Imagination's PVRTC format, 80–81
    - reading PNG images, 82–84
  - rendering converted data on iPhone, 86–88
  - starting iPhone games, 59–60
  - traversing scene contents
    - converting FBX models into triangle data, 73–76
    - converting triangle data into in-game formats, 76–78
    - OpenGL triangle data, 71–73
    - overview, 69
    - types of nodes, 70–71
- importing assets, 204
- In Memory store, 131
- Indexed Triangle list, 71
- Info option, Interestingness target CTRL-Click menu, 15
- Info.plist file, 324
- in-game file format, vs. data exchange file format, 63–64
- in-game formats, converting triangle data into, 76–78
- init method, 28, 260, 276
- initWithAPI method, 181
- initWithContentsOfURL: method, 41
- initWithHand method, 267
- initWithImageURL:target:targetMethod
  - instantiation method, 40
- initWithMsg:dependency: method, 33
- initWithStyle: method, 9, 12, 38, 43
- InMemory store data type, 128
- InputManager object, 227–228
- inputs, 224–228
- inspector view, Unity interface, 198
- installing Cocos2d
  - configuring sample code, 254–255
  - installing XCode project templates, 255–256
  - starting new project, 256
- Instant Messenger (IM), 312, 323
- int variable, 50
- interacting with objects, 237–239
- Interestingness-Version5-PLBlockExtensions folder, 52
- Interestingness algorithm, Flickr, 6
- Interestingness application
  - adding JSON parsing framework to, 10
  - changing to use NSOperationQueues, 37–38
  - changing to use NSOperationQueues and blocks, 50–52
  - converting to use official version of blocks, 53–54
- Interestingness, composing RESTful request for list of images, 10–12
- Interestingness images
  - composing RESTful request for list of, 10–12
  - implementing NSInvocationOperation to fetch list of, 39–40
- Interestingness project, 9–10, 48, 50, 52
- Interestingness user interface, building, 7–9
- InterestingnessAppDelegate class, 7
- InterestingnessAppDelegate file, 8–9
- InterestingnessTableViewController class, 8–9, 20–21, 38, 40, 42–43, 48, 50
- InterestingnessTableViewController.h file, 11
- Interface Builder (IB), 30–31, 127, 142, 149, 151, 299
- internal iPod Library, 119
- iPad Programming Guide, 281
- iPad simulator, 119
- iPad, supporting, 281–285
- iPhone Developer Program Portal, 324
- iPhone SDK, 80, 179–180, 289, 295, 306
- iPhone simulator, 111, 118–119, 123
- IPHONE\_OS\_VERSION\_MAX\_ALLOWED value, 282
- iPhoneInput class, 226
- lplImage class, 148, 152, 154, 158
- lplImage format, 148, 152
- lplImage image, 148
- lplImageFromUIImage:image method, 158
- iPod library, 117
- iris scanners, 141
- isBefore method, 126

isCancelled method, 28, 33  
 isCancelled property, 26  
 isConcurrent method, 27–28  
 isConcurrent property, 26  
 isEqual method, 126–127  
 isExecuting method, 28  
 isExecuting property, 26  
 isFinished KVO property, 34  
 isFinished method, 28  
 isFinished property, 26  
 iSolve team, 138–139  
 isometric rendering, 216  
 isometric view, 215–216  
 isReady property, 26

## J

J2ME (mobile Java), 311  
 Jackson, Benjamin, 252  
 Javascript language, 194–195  
 JavaScript Object Notation (JSON), 6, 11, 20, 316–317, 319  
 Jobs, Steve, 3  
 JobUnit class, 32–33  
 JobUnit.h file, 33  
 JPEG format, 79  
 JSON (JavaScript Object Notation), 6, 11, 20, 316–317, 319  
 JSON parsing framework, 10–13  
 JSON.h file, 10, 12  
 jsonResultString class, 13

## K

Kanis, Dimitri, 60–61  
 kAudioSessionCategory\_MediaPlayback category, 306  
 Key Value Observing (KVO), 25–26, 34  
 keychains, XCode, 325  
 key-value coding (KVC), 25, 134  
 KfBxAxisSystem method, 68  
 KfBxGeometryConverter class, 74–75  
 KfBxImporter class, 66–67  
 KfBxMesh class, 71, 73  
 KfBxNode class, 70–71  
 KfBxNodeAttribute::eMESH node type, 71  
 KfBxScene class, 66–67, 69  
 KfBxScene::GetRootNode() method, 69  
 KfBxSdkManager class, 66–67  
 KfBxSystemUnit method, 68  
 kFortunes array, 182–183  
 KVC (key-value coding), 25, 134  
 KVO (Key Value Observing), 25–26, 34

## L

L flag, 143  
 Lamarche, Jeff, 7  
 Lamming, Oliver, 138  
 Languages features, Home Screen, 291  
 Layer attribute, 228  
 Layer drop-down menu, 228  
 Layer menu, GUICamera object, 241  
 layer property, 239  
 lerp variable, 230  
 Lew, Daniel, 100  
 lib directory, 64  
 libPVRTexLib library, 85  
 libPVRTexLib library, 84  
 Library component palette, 30  
 Library link, Home Screen, 291  
 library, loading images from, 151–152  
 Library panel, 299  
 Librivox collection, 290  
 Librivox project, 289  
 light objects, 211, 214  
 lights, adding, 213–214  
 Linear Interpolation process, 229  
 lineHeight variable, 178  
 lineWidth variable, 178  
 Link Binary with Libraries section, 121  
 Linked Libraries list, 303  
 Linked Libraries section, 114  
 linker errors, 147  
 linker flags, 115, 147  
 linker option, 114  
 lipo program, 144  
 listener component, 241  
 loading Haar cascades, 152–154  
 loadInterestingnessList class, 21  
 loadInterestingnessList method, 12, 19  
 Localizable.strings files, 321  
 localization setting, 321  
 loc-args key, 321  
 lossy compression format, 80

## M

Mac OS X Runtime folder, 48  
 Mac Toolkit, Google, 111  
 Mach threads, 19  
 Machine Learning library, 159  
 MacPorts package, 170  
 MacPorts tool, 169–170  
 Magical Panda Software, 124, 131  
 MagicSolver.com Ltd, 139  
 Main Camera object, 199, 211, 230, 241, 244, 246

- main characters, 218–224
- main Image View class, 150
- main Interface Builder window, 300
- main method, 28, 32–33
- main nib file, 7
- Main Scene class, 261
- main window nib file, 7, 30
- MainMenuScene scene, 260
- MainMenuScene.h file, 260
- MainMenuScene.m file, 260–261, 283
- mapkey variable, 174
- mapLibrary variable, 174
- mapping, 171–172
- Mark, David, 7
- material components menu, 201
- Material menu item, 204
- materials, 200–203
- Materials folder, 204
- maxConcurrentOperationCount property, 25, 36
- MediaPlayer.framework library, 120, 123
- Mesh collider components, 225–226
- Mesh Filter menu item, 200
- Mesh Renderer component, 200–201, 203–204
- Mesh Renderer menu item, 200
- Mesh Scale Factor setting, 234
- meshes, 200–203
- message.aif file, 319
- Messenger application, eBuddy, 312
- mFactor variable, 50
- MIDI (Musical Instrument Digital Interface), 280
- min\_neighbors parameter, 158
- MINA (Multipurpose Infrastructure for Network Applications), 318
- mipmaps, 85
- ml/include header file, 144
- mobile Java (J2ME), 311
- mobile network connections, 79
- Mobile photo editing tools project, 159
- mock objects, 120–123
- mode attribute, 149
- Models folder, 203–204, 230
- Model-View-Controller (MVC), 110
- mogenerator plug-in, XCode, 130
- mogenerator tool, 129–131
- Mono scripting project, 194
- MonoBehaviour object, 206
- Moore's law, 140
- morphological operators, 159
- mouse button mapping, 226
- Move() method, 222–224
- move speed property, 223
- moveDirection vector, 223
- movement vector, 222–223
- movementDestination variable, 222–224
- movementDirection.y property, 222
- moveSpeed property, 223
- moveTowards() command, 239
- mp\_isAfter: method, 126
- mp\_isBefore method, 126
- mp\_isBefore:(NSDate \*)otherDate method, 125
- mp\_isEqual: method, 126
- MP3 files, 296, 302
- MPMediaItem class, 120, 123
- MPMediaItem instance, 123
- MPMediaItem object, 123
- MPMediaItemPropertyGenre parameter, 123
- MPMediaItemPropertyTitle parameter, 123
- MPMediaItems class, 119
- MPMediaPlayer framework, 117
- MPMediaQuery class, 123
- multi-architecture files, 144
- multiple cameras, 241
- Multipurpose Infrastructure for Network Applications (MINA), 318
- music player application, 128
- Musical Instrument Digital Interface (MIDI), 280
- MusicPlayer build target, 112
- MusicPlayer Direct Dependent library, 113
- MusicPlayer project, 111–112
- MusicPlayerTests Build Target, 112
- MusicPlayerTests General tab, 112
- MusicPlayerTests target, 112, 115–116, 121
- MusicPlayer.xcdatamodel Data Model, 128, 130
- MusicPlayer.xcdatamodel file, 130
- MVC (Model-View-Controller), 110
- MyAppDataDidChangeNotification, 103
- MyDelegateProtocol protocol, 103

## N

- name attribute, 130
- Narrators features, 291
- National Television System Committee (NTSC), 140
- NDA (Non-Disclosure Agreement), 5, 251
- New Copy Build Phase option, 121
- New File menu item, XCode, 129
- New File window, 128
- NewBehaviorScript script, 206
- node attribute, 70–71
- node method, 258
- nojasoncallback=1 parameter, 11
- Non-Disclosure Agreement (NDA), 5, 251
- Non-uniform rational B-spline (NURBS), 71
- normal attribute, 71
- normalized values, 172
- Notification class, 319

- notifications, 103
  - NotificationSender class, 319
  - NSArray property, 104
  - NSBlockOperation class, 24, 46, 52–53
  - NSCurrentLocaleDidChangeNotification method, 321
  - NSData class, 14, 41, 148
  - NSDate class, 124–125
  - NSDate+Helpers.h file, 125
  - NSDate+Helpers.m file, 125
  - NSDate+HelpersTests class, 124
  - NSDictionary class, 13, 19, 22, 24, 41
  - NSFetchedResultsController class, 292
  - NSInvocationOperation. See NSOperation class and NSInvocationOperation
  - NSLog statements, 50
  - NSManagedObject class, 127, 130–131
  - NSManagedObjectContext class, 128, 130–131
  - NSManagedObjectModel class, 127, 131
  - NSMutableArray object, 272
  - NSNotificationCenter, Apple, 103
  - NSObject class, 19–24
  - NSOperation class and NSInvocationOperation
    - building HelloOperationQueues, 28
    - building user interface for HelloOperationQueues, 30–31
    - changing Interestingness app to use NSOperationQueues, 37–38
    - creating work units by subclassing, 32–33
    - implementing FetchImageOperation subclass of, 40–41
    - implementing getImageForURL: and storeImageForURL: methods, 42–45
    - implementing HelloOperationQueues, 33–36
    - implementing to fetch list of interestingness images, 39–40
    - overview, 26
    - subclassing, 27–28
  - NSOperationBlock sample project, 50
  - NSOperationQueue class
    - changing Interestingness application to use, 37–38, 50–52
    - overview, 25–26
  - NSOperationQueue+PLBlocks.h file, 50, 52–53
  - NSOperationQueue+PLBlocks.m file, 50, 52–53
  - NSOperationQueueDefaultMaxConcurrentOperationCount constant, 25
  - NSOrderedAscending constant, 126
  - NSPersistentStore class, 131
  - NSPersistentStoreCoordinator class, 131
  - NSString class, 11, 13, 43
  - NSThread class, 19–24
  - NSURL class, 40, 43
  - NSURLConnection class, 19
  - NTSC (National Television System Committee), 140
  - numberOfSectionsInTableView: method, 9, 14
  - NURBS (Non-uniform rational B-spline), 71
- ## O
- O flag, 143
  - Obj-C language, 32, 40, 60, 102, 171, 179, 194–195, 206, 228
  - objects
    - interacting with, 237–239
    - touchable, adding, 230
  - objects, operation, 24–45
    - NSOperation and NSInvocationOperation, 26–45
      - building HelloOperationQueues, 28
      - building user interface for HelloOperationQueues, 30–31
      - changing Interestingness app to use NSOperationQueues, 37–38
      - creating work units by subclassing, 32–33
      - implementing FetchImageOperation subclass of, 40–41
      - implementing getImageForURL: and storeImageForURL: methods, 42–45
      - implementing HelloOperationQueues, 33–36
      - implementing to fetch list of interestingness images, 39–40
      - subclassing, 27–28
      - NSOperationQueue, 25–26
    - observeValueForKeyPath:ofObject:change:context: method, 34
  - OCMock Framework, 121–122
  - OCMock object, 121
  - OCMock static library, 121
  - OCMock.framework folder, 121–122
  - OCMock.framework item, XCode, 122
  - OCUnit framework, 110–111, 117
  - OCUnit library, XCode, 110
  - off-road robot vehicles, 141
  - off-screen framebuffer, 187
  - offset vector variable, 229
  - On screen display, 296
  - OnControllerColliderHit() method, 224
  - onGUI() method, 240
  - onTimer method, 305
  - Open Computing Language (OpenCL), 142–144
  - opencv-1.1.0 folder, 145

- opencv-1.1.0/data/haarcascades directory, 153
  - opencvlib directory, 143–146
  - OpenGL fonts, 163–187
    - creating texture atlas, 168–171
    - opening applications, 173
    - texture mapping, 171–172
  - fCookie application, 167–168
  - Fontery classes, 173–179
    - APChar, 175–176
    - APGlyph, 174
    - APText, 177–179
  - fortunes
    - creating, 182–184
    - displaying, 184–187
  - history of, 164–165
  - Pragmatic Fontery system, 167
  - setting up display, 181–182
  - terminology, 165–166
- OpenGL triangle data, 71–73
  - opengl/Classes/EAGLView.mm file, 87
  - opengl/opengl.xcodeproj file, 86
  - open-source Checkers library, 138
  - open-source Connect-4 library, 138
  - operation objects, 24–54
    - blocks, 46–54
      - adding PLBlocks framework, 48–50
      - changing Interestingness application to use NSOperationQueues and blocks, 50–52
      - converting Interestingness application to use official version of blocks, 53–54
  - NSOperation and NSInvocationOperation, 26–45
    - building HelloOperationQueues, 28
    - building user interface for HelloOperationQueues, 30–31
    - changing Interestingness app to use NSOperationQueues, 37–38
    - creating work units by subclassing, 32–33
    - implementing FetchImageOperation subclass of, 40–41
    - implementing getImageForURL: and storeImageForURL: methods, 42–45
    - implementing HelloOperationQueues, 33–36
    - implementing to fetch list of interestingness images, 39–40
    - subclassing, 27–28
  - NSOperationQueue, 25–26
  - Operation Objects class hierarchy, 24
  - operation queues, GCD, 46
  - operationCountOutput outlet, 30
  - operationCountOutput outlet, 30
  - operationOutput outlet, 30
  - operations property, NSOperationQueue class, 25
  - optimized texture format, 64
  - Options checkbox, UITableViewController class, 7
  - Options menu item, 260
  - OptionsScene.h file, 261
  - OptionsScene.m file, 261
  - Other Linker Flags option, Build tab, 114
  - Other Linker Flags setting, settings window, 147
  - otherlibs/highgui header file, 144
  - overlay scene, 241
  - Overview dropdown menu, 255
  - Overview item, XCode toolkit, 254, 281
- ## P
- pagination, 292
  - paintball sentry guns, 141
  - pan/zoom/tumble tool, 197
  - Papyrus font, 178
  - parameters dictionary, 317
  - parent object, 228, 244
  - parent transforms, 220
  - path-finding algorithm, 224
  - pause playback control, 293, 296, 304
  - Per Diem calculator, 289
  - per\_page parameter, 11
  - performance, 157–159
  - performSelectorInBackground:withObject: method, 19, 39–40
  - performSelectorOnMainThread:withObject:waitUntilDone: method, 19, 21, 40–41
  - periodicRotate() method, 210
  - perspective transform, 159
  - Phone-based barcode scanners, 141
  - photo library, loading images from, 151–152
  - PHP (HyperText Preprocessor), 11
  - php5 language, 170
  - physics libraries, 191
  - Physics.Raycast() method, 227
  - pipelines, 61–64
  - placeholder model, 219
  - Plane menu option, GameObject menu, 216
  - Plausible Blocks SDK package, 48
  - Plausible Labs extensions, 50
  - Plausible Labs PLBlocks 1.1 Beta, 5
  - Play & Pause, UIButton class, 298
  - Play board games using the camera project, 159
  - play button, 304
  - play button, Unity editor, 208–209, 211, 228

- Play Game menu item, 260
- player screens, 293–294
- player setup, 296
- player state, providing in UI, 305
- Playlist entities, 128, 130–133
- PlaylistControllerTests class, 118
- PlaylistControllerTests.h file, 117
- PlaylistControllerTests.m file, 117–118
- PlaylistItem class, 120
- PlaylistItem header, 120
- PlaylistItem objects, 119
- PlaylistItem.h file, 120
- PlaylistTests test case, 131
- PlaylistTests test case file, 130
- play/pause button, 304–305
- playPauseButton class, 304–305
- playPauseButtonClicked method, 304, 306
- playPauseClicked method, 305
- PLBlockOperation class, 50, 52–53
- PLBlocks, 48–50, 53
- plist (property list) file, 171
- PLIST file, 169, 171, 173
- PLIST property, 180
- plus icon, Direct Dependencies section, 112
- PNG (Portable Network Graphics) format, 63, 82–84
- PNG\_COLOR\_TYPE\_RGB property, 83
- PNG\_COLOR\_TYPE\_RGBA property, 83
- PngResourceReader class, 82
- PngResourceReader::read() method, 82
- Point light, 213–214
- Point Light menu option, GameObject menu, 214
- PokerGameScene.h file, 261, 272–273
- PokerGameScene.m file, 261, 273–274, 280, 284–285
- poker-music.mp3 file, 280
- polygons, 71, 73–74
- port command, 169
- portability, 226
- Portable Network Graphics (PNG) format, 63, 82–84
- Portable Operating System Interface [for Unix] (POSIX), 17, 19
- portrait mode, iPhone, 172
- position attribute, 71
- position method, 277
- position property, 200–201, 216, 219, 224, 234
- position transformation, 199
- POSIX (Portable Operating System Interface [for Unix]), 17, 19
- Powered by X footer, 95
- PowerVR SDK, 80, 84
- PowerVR Texture Compression (PVRTC) format, 64, 80–81, 83–85
- pragmatic font techniques, 163
- Pragmatic Fontery system, 167–168
- Prefab menu option, Assets menu, 231
- prefab objects, 230–231, 239, 245
- prefabs, 231
- Prefabs folder, 231, 243
- preprocessor directive, 282
- preprocessor option, Build tab, 114
- preview window, 247
- price/performance ratio, 193
- primitive collider component, 226
- primitive collider shapes, 225
- primitive shapes, 194
- primitive solid objects, 225
- printf function family, 77
- private keys, 325
- production environment certificate, 324
- Project menu, 115
- Project Target, 117
- project view, 198, 203–205, 207, 216, 226, 233, 243
- <ProjectName>Tests convention, 112
- Properties tab, Interestingness target CTRL-Click menu, 15
- property list (plist) file, 171
- protocols, client / server, 317
- Provider certificate, 314
- push notifications, 311–325
  - Apple Push Notification service, 313–316
    - client implementations, 314–316
    - communication flow, 314
    - client implementations, 314–316
    - communication flow, 314
  - eBuddy application, 311–323
    - client / server protocol, 317
    - eBuddy iPhone application, 312–313
    - eBuddy Messenger application, 312
    - fitting parts together, 322–323
    - server to APNS, 318–322
  - eBuddy Pro application, 323–324
    - extending beta program, 324–325
- Push Server, 312–313, 317, 322
- Push timeout, 312
- push\_destination parameter, 317
- push\_timeout parameter, 318
- push\_type parameter, 318
- .pvr file, 85
- PVRTC (PowerVR Texture Compression) format, 64, 80–81, 83–85
- PVRTexTool, 80–81
- PVRTTextureUtilities::ProcessRawPVR() method, 85
- Python language, 61, 251, 260
- Pyweek challenge, 251

## Q

QA (Quality Assurance), 109  
 Qc.png file, 263  
 quads, 71, 74, 240  
 Quality Assurance (QA), 109  
 quaternion, 206  
 queuePriority property, NSOperation class, 26

## R

radius property, CharacterController component, 219  
 rand() method, 266  
 random(). method, 266  
 range setting, light object, 214  
 raycast method, 227, 229  
 raycasting code, 246  
 Reachability API, 7  
 readMediaItem method, 120  
 Really Simple Syndication (RSS), 6  
 real-time processing, 82  
 Real-time, real-life pong project, 159  
 rectangle object, 277  
 regions, 140  
 relative priority level, operation object, 27  
 Release configuration XCode, 111  
 remapped inputs, 226  
 remove (rm) command, 171  
 Remove Component option, gear-menu, 219  
 removeDependency: method, 26  
 removeOperation: method, 25  
 render method, 184–185  
 Render Settings menu option, Edit menu, 213  
 renderers, 200–203  
 renderObjects method, 184, 187  
 renewing subscriptions, 95  
 Rentzsch, Jonathon, 129  
 resizeFromLayer method, 181  
 ResizeMe Mac application, 285  
 Resource option, New File window, 128  
 Resources folder, 180, 280  
 Resources group, 149, 153, 299, 302  
 respondsToSelector method, 125  
 RESTful request, 10–13  
 RESTful Web Services, 6, 16, 20, 23  
 revenue share, 96  
 revision control system, 325  
 rewind playback control, 293  
 RGB format, 79, 83–84  
 RGBA (4-channel) image, 148  
 RGBA format, 83–85  
 right-handed coordinate system, 68  
 rm (remove) command, 171

rotate method, 209  
 rotate script, 218  
 rotate tool, 197  
 RotateBarrel class, 206  
 RotateBarrel script, 206, 208, 227  
 Rotating Barrel object, 199–200, 203, 205, 208–209, 232  
 rotating barrel script, 221  
 rotation parameter, model, 69  
 rotation part, transform component, 206  
 rotation property, camera object, 216  
 rotation script, 220  
 rotation transformation, 199  
 Round Rect Buttons, 30, 150, 299  
 RSS (Really Simple Syndication), 6  
 Ruby language, 289  
 runAction parameter, 271  
 run-time memory, 79

## S

S3 Texture Compression (S3TC) format, 80  
 S3TC (S3 Texture Compression) format, 80  
 save: method, 128  
 scale parameter, model, 69  
 scale property, Faux Character object, 219  
 scale transformation, 199  
 scale\_factor parameter, 158  
 scale\_factor parameter, 158  
 scanf function family, 77  
 scanned barcodes, 141  
 scene contents, 69–78  
   converting FBX models into triangle data, 73–76  
   converting triangle data into in-game formats, 76–78  
   OpenGL triangle data, 71–73  
   types of scene nodes, 70–71  
 scene nodes, 69–71  
 scene segmentation, 140  
 scene view, 196–197  
 scenes, 258–261  
 ScreenPointToRay() method, 227  
 screens  
   book information, 292–293  
   book selection, 292  
   Home Screen, 291–292  
   player, 293–294  
 scripting, custom, 205–207  
 scripts folder, 205, 220, 226  
 Search bar, browser screens, 292  
 section navigator, browser screens, 292  
 Seeker demo, 4  
 Seeker program, 4

- segmentation, 159
- sendAlert function, 272
- SendMessage() method, 244
- SendMessageOptions.DontRequireReceiver constant, 244
- SenTest framework, 110
- seq dictionary entry, 317
- Sequences, 272
- serial dispatch queues, GCD, 46
- serial queues, GCD, 46
- Serlet, Bertran, 4
- server protocol, 317
- server-intermediary consumption, 100–101
- Set Active Target Project menu item, 115
- setFlights: method, 104
- setQueuePriority: method, 27
- setSuspended:NO method, 25
- setSuspended:YES method, 25
- setter propagation, 104
- settings, animation import, 233–237
- Settings button, project view, 233
- settings window, 147, 234–235
- setUp method, 117
- setup time, 180
- setupPortraitMode method, 182
- setVerticallyFlipped() method, 85
- shader component, 201
- shadow positions, 141
- shape, objects, 141
- shark files, 86
- shark model, 61, 79
- shark.fbx file, 86
- shark.png file, 86
- silent switch, 306
- simple follower script, 230
- Simple Object Access Protocol (SOAP), 11
- SimpleAudioEngine.h file, 280
- SimpleHand class, 273
- SimpleHand object, 266–267, 272
- SimpleHand.h file, 261
- SimpleHand.m file, 261, 266–269
- Single precision floating-point numbers, 62
- Singleton design pattern, 221
- size property, collider component, 230
- Skinny-Controller-Fat-Model paradigm, 127
- skip back playback control, 296
- Skip Backward, UIButton class, 298
- skip controls, connecting, 305
- skip forward playback control, 296
- Skip Forward, UIButton class, 298
- skip playback control, 293
- sleep mode, 319–320
- smart caching, 292
- Snow Leopard+, 143
- Snowferno game, 194–195
- SOAP (Simple Object Access Protocol), 11
- Software Bisque, 4
- Song entities, 128, 130–133
- SongTests test case file, 130
- sortedHand object, 267
- Sound notification option, 315
- Source Code area, Apress web site, 167
- SourceForge, 143
- Spawn object, 271, 275, 278
- Spawned Barrel object, 245
- Sphere collider component, 225
- Sphere collider components, 225
- Spot light, 213–214
- Springboard, 312, 314, 320
- sprites, 270–272
  - detecting sprite taps, 276–277
  - load and display, 270
  - manipulating, 271–272
  - switching sprite image, 279–280
- spriteWithFile method, 274
- SQLite store, 127
- square magnitude, movement vector, 222
- square size requirements, 81
- src directory, 65
- src/resource/fbx/FbxResourceConverter.cpp file, 73
- src/resource/fbx/FbxResourceUtils.cpp file, 74–75
- src/resource/fbx/FbxResourceUtils.h file, 75
- src/resource/Geometry.cpp file, 72
- src/resource/geometry/GeometryResourceReader.cpp file, 78
- src/resource/geometry/GeometryResourceWriter.cpp file, 77
- src/resource/png/PngResourceReader.cpp file, 82
- src/resource/png/PngResourceReader.h file, 82
- src/resource/pvrtc/PvrtcResourceWriter.cpp file, 84
- SSLContext class, 319
- stages, animations, 232
- Start() method, 206, 210, 222, 229, 238
- StartCoroutine() method, 210
- static accessor method, 221
- static library file, GHUnit framework, 111
- status bar, 207
- std::find\_if method, 76
- Stick Wars application, 252
- StickWars application, 286
- stock rand() function, 183
- storeImageForURL: method, 40, 42–45, 52
- string variable, 178
- stringWithContentsOfURL:encoding:error method, 13



- stringWithFormat: method, 13
- subclassing, 27–28
  - creating work units by, 32–33
- subscription, 95
- subscriptions, auto-renewing, 95
- sudo command, 170
- Sudoku Magic app, 159
- Sudoku solver application, 138
- Supported Audio Formats in SDK Version 3.0, 296
- surface color, 140
- Surprise Me features, Home Screen, 291
- Surveillance software, 141
- swizzled option, 85
- synchronous methods, 19
- system queues, GCD, 46
- SystemConfiguration.framework API, 7

## T

- tableView:cellForRowAtIndexPath: method, 9, 14
- tableView:cellForRowAtIndexPath method, 42–43
- tableViewController class, 8
- tableViewController instance variable, 8
- tableView:didSelectRowAtIndexPath:, 9
- tableView:numberOfRowsInSection: method, 9, 14
- Tag attribute, 228
- Tag drop-down menu, 228
- tag property, 31
- tap interface, 215
- tappedSprite function, 277
- tap-to-move interface, 215
- tar command, 171
- tarball technique, 171
- target field drop-down menu, 230
- Target Info window, 112
- target item, Groups & Files pane, 112
- Target variable, main camera object, 230
- targetObject class, 41
- Targets group, Xcode toolkit, 254
- Taylor, Adam, 232–233
- tearDown method, 117
- template matching algorithm, 159
- Terminal application, 141–142, 144, 169, 255, 285
- terrain, 216–218
- Test Build target window, 114
- test folder, code directory, 115
- Test target, 121, 125
- testFirstUnitTest class, 118
- testing iPhone application units, 109, 134
- Tests folder, XCode, 115, 117
- text-based format (ASCII), 65
- texture atlas
  - creating, 168–171
  - opening applications, 173
  - texture mapping, 171–172
- texture compression, vs. image compression, 79–80
- texture mapping, 171–172
- Th.png file, 263
- tiling, texture placement setting, 217
- time elapsed label, UILabel class, 298
- time remaining label, UILabel class, 298
- Time.deltaTime variable, 223
- Tip Calculator app, 289
- title attribute, Song entity, 128
- title property, MPMediaItem class, 120
- title Property, PlaylistItem class, 120
- title property, Round Rect Button, 150
- Titles link, Home Screen, 291
- TLS (Transport Layer Security), 314
- toggleCard function, 280
- tools, advantages of writing, 60–61
- touch inputs, 214–215, 218, 226, 228, 246
- touch manager object, 221, 228
- Touch Up Inside event, 31
- Touch Up Inside method, 301
- touchable objects, 230, 244
- transaction-based model, 95
- transform component, 198–199, 203, 206–207
- transform properties, 198
- Transport Layer Security (TLS), 314
- triangle data, 73
  - converting FBX models into, 73–76
  - converting into in-game formats, 76–78
- Tribble, Bud, 4
- TripDeck, 99
- TrueType fonts, 163, 168–169, 171, 179
- .ttf extension, 168, 170
- Twiddled option, 85
- typedef feature, C language, 47
- typeface, 166
- typefaces, 166

## U

- UIApplication class, 314–315
- UIApplicationDelegate class, 314–316
- UIApplicationLaunchOptionsRemoteNotificationKey key, 316
- UIButton class, 298
- UIImage class, 41, 148–149, 152
- UIImagePickerControllerDelegate interface, 147

- UIImageView class, 150
  - UIKit classes, 17–18
  - UIKit.framework, 114
  - UILabel class, 30, 298, 300
  - UINavigationController objects, 104
  - UINavigationControllerDelegate interface, 147
  - UITableView class, 6, 102
  - UITableViewController class, 6–9
  - UITableViewDataSource protocol methods, 14–15
  - UITableViewDelegate Protocol class, 9
  - UIUserInterfaceIdiomPad variable, 282
  - UIViewController template, 7
  - Uniform Resource Locator (URL), 316
  - unit test harness, GHUnit framework, 114
  - Unit Test Project, 121
  - Unitron text editor, 193
  - Unity interface, 192–214
    - adding lights, 213–214
    - asset pipeline, 198–199
    - coroutines, 209–210
    - custom scripting, 205–207
    - game view, 197, 210–211
    - hierarchy view, 198
    - importing assets, 204
    - inspector view, 198
    - meshes, renderers, and materials, 200–203
    - playing, 208–209
    - project view, 198
    - reasons for using, 193–195
    - scene view, 196–197
    - Transform component, 199
    - using iPhones as game pads, 214
  - universal files, 144
  - unlock screen, 319
  - unusedChars array, 177
  - unusedChars method, 177
  - update loop, 224, 238–239
  - Update() method, 206–207, 210, 222, 224, 226, 229, 246
  - Upgrade Current Project for iPad option, XCode Project menu, 281
  - URL (Uniform Resource Locator), 316
  - user interfaces, 240
    - building for HelloOperationQueues, 30–31
    - building Interestingness, 7–9
    - connecting data to, 102–104
      - delegates, 102–103
      - notifications, 103
      - setter propagation, 104
    - non-responsive, 6–15
      - adding JSON parsing framework to Interestingness application, 10
      - building Interestingness user interface, 7–9
      - composing RESTful request for list of Interestingness images, 10–12
      - RESTful request and JSON parser, 12–13
      - UITableViewDataSource protocol methods, 14–15
      - providing player state in, 305
      - setting up, 298–301
      - source-driven design, 96–99
        - challenges, 97–98
        - techniques, 98–99
      - userInfo variable, 316
      - utility classes, AVFoundation, 295
      - Uunity editor, 209
- ## V
- valueForKey method, 123
  - valueForKey: method, 123
  - Vector3.right property, 207
  - version control systems, 134
  - vertexCount variable, 178
  - vertices, 198, 203
  - VertexPredicate class, 75–76
  - VertexPredicate::operator() method, 75–76
  - vertices array, 177
  - Video Poker, 257–258, 281
  - video windows feature, 143
  - Video\_PokerAppDelegate.m file, 261
  - view-based application, 111, 145, 296–297
  - view-based XCode project, 112
  - viewDidLoad: method, 33, 154
  - viewWillAppear: method, 12, 22, 39, 48
  - virtual viewfinder, 159
- ## W
- "Walk" (characterModel.animation.CrossFade) method, 236
  - walking animation, 223, 233, 237–238
  - warrior animation, 236
  - Wavefront OBJ file format, 62
  - Weather app, Apple, 94
  - webcam support feature, 143
  - Wenderlich, Ray, 286
  - wget command, 171
  - while arc4random() method, 266
  - Window-based Application template, XCode, 7
  - winSize function, 274
  - winSize property, 270

- Wolfram Alpha computational knowledge engine, 285
- work units, creating by subclassing, 32–33
- workMsg property, 32
- workOperation operation, 25
- workQueue class, 33
- World Wide Developers Conference (WWDC)
  - Apple, 4–6
- wrapper classes, 303
- Wright, Richard S., 4
- writing tools, advantages of, 60–61
- WWDC (World Wide Developers Conference)
  - Apple, 4–6

**X**

- XCode, 145, 255–256
- .xib file, 149
- XML (Extensible Markup Language), 6, 154, 158
- XY values, 186

**Z**

- z character, 174
- Z range, 182
- Z-Axis management, 270
- z-Delete utility, 82, 234