

# Index

## • A •

About pages  
  adding images, 45  
  creating for Facebook apps, 43–46,  
  279–280  
  defined, 16–17  
administration methods in Facebook  
  API, 53  
AJAX (Asynchronous JavaScript and XML).  
  *See also* FBJS  
  defined, 159  
  FBJS AJAX object members, 159  
  providing dynamic feedback to users,  
  160–165  
animation. *See also* FBJS; images  
  adjusting size/visibility of block-level  
  events, 156–159  
  adjusting speed/duration, 156  
  FBJS animation methods, 151–152  
  tweening  
    changing background color, 153–155  
    defined, 152  
    increasing font size, 152–153  
    moving objects, 155  
API Test Console, 167–170  
appinclude.php sample listing, 286–287  
applications. *See* Facebook applications;  
  Speed Dial application  
Appsaaholic application, 319  
attachment-preview tag in FBML, 102  
attachments. *See also* Facebook  
  applications  
  adding to applications, 303–304  
  adding to Walls, 14–15  
  coding in PHP, 304–305  
  defined, 303  
  integrating Facebook apps via, 14–15, 21  
  overview, 35  
  viewing, 305–306

Attack! application, 315–316  
attributes  
  derived attributes, 141  
  for fb:name tag, 104–105  
  for fb:pronoun tag, 105–106  
authentication methods in Facebook  
  API, 53

## • B •

browser user-agent strings, 309–310  
buttons  
  Add Music button in iLike app, 14  
  creating in navigation headers using  
  FBML, 184  
  Facebook, emulating in iframe canvas  
  pages, 224–225, 229–232

## • C •

callback metaphor, 22–23  
canvas pages. *See also* Facebook  
  applications; Speed Dial application  
  adding CSS styles to FBML pages  
  embedded styles, 216–217  
  external style sheets, 218  
  inline styles, 216  
  overview, 215–216  
  in Speed Dial sample app, 266–267  
conditional FBML tags for  
  defined, 98–99  
  fb:default, 112  
  fb:else, 108, 111  
  fb:if, 111  
  fb:if-can-see, 107  
  fb:if-can-see-photo, 108  
  fb:if-is-app-user, 108–109  
  fb:if-is-friends-with-  
  viewer, 109  
  fb:if-is-group-member|, 109

canvas pages (*continued*)

- fb:if-is-in-network, 109
- fb:if-is-user, 109
- fb:if-user-has-added-app, 108
- fb:switch, 111–112
- overview, 106
- creating data entry forms using FBML
  - defined, 190
  - fb:editor-\* control elements, 191–194
  - fb:editor form element, 190–192
- creating navigation headers using FBML
  - adding borders, 189
  - adding image banners, 190
  - fb:action links, 184
  - fb:create-button, 184
  - fb:dashboard header, 183–185
  - fb:header customization, 185, 188–190
  - fb:help link, 184
  - fb:tabs and fb:tab-item links, 185–188
  - fb:title of app home page, 187
  - getting rid of padding, 189–190
  - overview, 181–182
- creating using FBML versus iframes, 179–181
- defined, 12–13, 179
- emulating Facebook styles in iframe pages
  - buttons, 224–225, 229–232
  - clearfix utility class and, 225
  - dashboards, 223–226
  - Facebook color palette and, 222
  - including graphics, 225
  - navigation tabs, 226–228
  - overview, 220–221
  - subtitle sections, 228–229
  - text formatting, 221–222
  - two-column lists, 232–234
- using external .js files in, 149–150
- integrating Facebook apps into, 13, 18–19
- making accessible to non-Facebook users, 307
- CDs application example, 182, 183
- clearfix utility class, 225
- color palette in Facebook, 222

- conditional FBML tags. *See also* FBML
  - for canvas pages, 106–109
  - defined, 98–99
  - fb:default, 99, 112
  - fb:else, 98, 108, 111
  - fb:if, 111
  - fb:if-can-see, 96, 107
  - fb:if-can-see-photo, 96, 108
  - fb:if-is-app-user, 108–109
  - fb:if-is-friends-with-viewer, 109
  - fb:if-is-group-member, 109
  - fb:if-is-in-network, 109
  - fb:if-is-user, 98, 109
  - fb:if-multiple-actors, 243
  - fb:if-user-has-added-app, 108
  - fb:switch, 98, 111–112
  - fb:visible-to\*, 109–111
  - overview, 106
  - for profile boxes, 106, 109–111
  - warning, 111
- content, defined, 2
- cookie methods in Facebook API, 54, 310
- CSS styles. *See also* Facebook applications; styles; styling
  - adding to FBML canvas pages/profile boxes
    - embedded styles, 216–217
    - external style sheets, 218
    - fbml.setRefHandle in, 219–220
    - inline styles, 216
    - overview, 215–216
    - to profile boxes using fb:ref, 219–220
    - in Speed Dial sample app, 266–267
    - and updating user profiles en masse, 219–220
    - wide and narrow profile boxes, 218–219
  - emulating Facebook styles in iframe
    - canvas pages using
      - buttons, 224–225, 229–232
      - clearfix utility class and, 225
      - dashboards, 223–226
      - Facebook color palette and, 222
      - including graphics, 225
      - navigation tabs, 226–228
      - overview, 220–221
      - subtitle sections, 228–229
      - text formatting, 221–222
      - two-column lists, 232–234

## • D •

### dashboards

- creating in FBML canvas pages, 183–185
- emulating in iframe canvas pages, 223–226

### data entry forms. *See also* forms

- defined, 190
- `fb:editor-*` control elements, 191–194
- `fb:editor` form element, 190–192

### Data Store API site, 325

### database, creating the Speed Dial, 253–254

### database queries. *See* FQL

### database terminology, 122

### `data.getCookies` API method, 54, 310

### `data.setCookies` API method, 54, 310

### `db.phpsample` listing, 290–291

### dependencies, avoiding, 39

### derived attributes, 141

### developer links, 27, 198

### Developer site. *See also* Facebook

#### applications

#### registering applications on

- base options, 31–34, 250–251
- fixing errors, 36
- installation options, 33–35, 251–252
- integration point options, 35–36, 252
- New Application page, 31
- overview, 30, 250
- and viewing in My Applications page, 36–37, 253

#### registering feed story templates, 245–247

### developer tools. *See also* Facebook

#### Platform

#### API Test Console, 167–170

#### Feed Preview Console, 167, 173–175

#### Firebug FBJS debugger, 168, 176

#### overview, 167–168

### dialog box tags in FBML, 100

### discussion boards

#### creating, 113–114

#### manning, 323

### `display.php` sample listing, 287–290

### DOM (Document Object Model) properties.

#### *See also* FBJS

#### accessing, overview, 144–146

#### get methods for accessing, 145–146

#### set methods for setting, 145–148

#### setting `innerHTML` property, 147–148

## • E •

### editor form/control tags in FBML, 100, 190–194

### elements, defined, 2

### e-mail alerts, integrating Facebook apps via, 21

### embedded CSS styles, adding to FBML canvas pages/profile boxes, 216–217

### end tags, 2

### event data, retrieving with FQL queries, 129–138

### events, adjusting size/visibility of block-level, 156–159

### events methods. *See also* Facebook API defined, 51

#### `events.get`, 62–69

#### `events.getMembers`, 69–72

#### properties of, 62

#### warning, 72

### external CSS style sheets, adding to FBML canvas pages/profile boxes, 218

## • F •

### Facebook. *See also* Developer site;

#### News Feed

#### color palette, 222

#### Developer page, 30

#### groups, 72

#### home page, 10–11

#### My Applications page, 36–37, 246–247, 253

#### versus MySpace, 10, 16

#### News Feed, 10–11

#### overview, 9, 10

#### pages, 83

#### profile page components. *See also individual subentries*

##### canvas, 12–13

##### defined, 12

##### Left Nav sidebar, 12–13

##### Mini-Feed, 12, 13

##### profile boxes, 12–14

##### Walls, 14–15

- Facebook (*continued*)
  - social graph, 15–16
  - styles, 215, 220–221
  - Web site addresses. *See also* Web site addresses
    - API Test Console, 168
    - application ads info, 324
    - developer links, 27, 198, 263, 325
    - Facebook UI styles, 263
    - FBML Test Console, 170
    - Feed Preview Console, 173
    - home page, 10
    - mobile site, 302
    - My Applications page, 246
    - prohibited practices, 321
    - sample files, 5
- Facebook API (Application Programming Interface). *See also* Facebook Platform;
  - feed methods
    - administration methods, 53
    - API Test Console, 167–170
    - authentication methods, 53
    - calling the, 49–50
    - cookie storage methods, 54, 310
    - defined, 24, 49
    - evaluating call responses, 55–56
    - events methods
      - defined, 51
      - `events.get`, 62–69
      - `events.getMembers`, 69–72
      - properties of, 62
      - warning, 72
    - FBML and FQL methods, 54
    - feed methods, 52
    - versus FQL, 26, 122
    - friends methods
      - defined, 51
      - `friends.appUsers`, 60–61
      - `friends.areFriends`, 58–60
      - `friends.get`, 57–58
      - overview, 56
    - groups methods
      - defined, 51
      - `groups.get`, 73–80
      - `groups.getMembers`, 80–83
      - overview, 72–73
      - warning, 82
    - marketplace methods, 54, 310–312
    - notifications methods
      - `notifications.get`, 53
      - `notifications.send`, 53, 237, 247–248, 275
      - `notifications.sendEmail`, 53
    - overview, 22–23
    - pages methods
      - defined, 52
      - overview, 83
      - `pages.getInfo`, 84–86
      - `pages.isAdmin`, 87
      - `pages.isAppAdded`, 86
      - `pages.isFan`, 87
      - properties of, 83–84
    - photos methods, 52
    - profile methods, 53
    - REST operations, 56
    - tip, 24
    - users methods
      - defined, 51
      - overview, 87
      - `users.getInfo`, 88–92
      - `users.getLoggedInUser`, 87
      - `users.hasAppPermissions`, 93
      - `users.isAppAdded`, 93
  - Facebook applications. *See also* Facebook Platform; Speed Dial application
    - adding CSS styles to FBML canvas
      - pages/profile boxes
        - embedded styles, 216–217
        - external style sheets, 218
        - `fbml.setRefHandle in`, 219–220
        - inline styles, 216
        - overview, 215–216
        - to profile boxes using `fb:ref`, 219–220
        - and updating user profiles en masse, 219–220
        - wide and narrow profile boxes, 218–219
      - adding discussion boards, 113–114
      - application directory, 16, 17
      - architecture of, 22–24
      - attachments
        - adding to applications, 303–304
        - adding to Walls, 14–15
        - coding in PHP, 304–305
        - defined, 303

- integrating with Facebook, 14–15, 21
- overview, 35
- viewing, 305–306
- building
  - adding About pages, 43–46
  - adding images to icons, 41–42
  - avoiding dependencies, 39
  - overview, 29
  - registering with Facebook before, 30–37
  - setting default FBML for profile boxes, 42–43, 198–199
  - simple, using PHP, 37–41
  - tips, 32–34, 36–38, 41
  - warnings, 32, 38
- canvas pages. *See also* canvas pages
  - using iframes versus FBML, 179–181
  - making accessible to non-Facebook users, 307
  - redirecting to new URL within, 310
- CDs application example, 182, 183
- creating data entry forms in canvas pages
  - using FBML
    - defined, 190
    - `fb:editor-*` control elements, 191–194
    - `fb:editor` form element, 190–192
- creating mobile versions of, 302–303
- creating navigation headers in canvas pages using FBML
  - adding borders, 189
  - adding image banners, 190
  - `fb:action` links, 184
  - `fb:create-button`, 184
  - `fb:dashboard` header, 183–185
  - `fb:header` customization, 185, 188–190
  - `fb:help` link, 184
  - `fb:tabs` and `fb:tab-item` links, 185–188
  - `fb:title` of app home page, 187
  - getting rid of padding, 189–190
  - overview, 181–182
- emulating Facebook styles in iframe canvas pages
  - buttons, 224–225, 229–232
  - `clearfix` utility class and, 225
  - dashboards, 223–226
  - Facebook color palette and, 222
  - including graphics, 225
  - navigation tabs, 226–228
  - overview, 220–221
  - subtitle sections, 228–229
  - text formatting, 221–222
  - two-column lists, 232–234
- `fbRecipe` sample app
  - adding a dashboard, 183–185
  - adding a header, 188–190
  - adding tabs, 185–188
  - customizing profile box content, 202–213
  - using
    - `feed.publishTemplatizedAction` in, 244–245
    - updating profile box content, 199–201
- getting and setting cookies, 54, 310
- handling Web browser issues, 308–309
- integration points in Facebook
  - About pages, 16–17
  - application directory, 16–17
  - attachments, 14–15, 21
  - canvas, 13, 18–19
  - e-mail alerts, 21
  - giving users choices about, 17–18
  - Google Analytics, 309
  - inviting friends to add apps, 18
  - Left Nav, 13, 19–20
  - Marketplace, 310–312
  - News Feed, 21
  - profile action links, 20–21
  - profile boxes, 13, 19–20, 195
  - requests to users, 21
  - setting during registration, 35–36, 252
  - Super Wall example, 16–22
  - user privacy settings, 21–22
- My Flickr application, 13–14
- optimizing speed, 301–302
- Photos application, 181
- recommended apps
  - Appsaholic, 319
  - Attack!, 315–316
  - criteria used for, 313
  - Flixster, 318
  - iLike, 14, 196, 314–315
  - iRead, 316
  - Local Picks from Trip Advisor, 182, 314

Facebook applications (*continued*)

- Quizzes, 317
- Runlicious from Introplay, 319
- Scrabulous, 315
- Top Friends, 318
- Where I've Been, 196, 197, 318
- Workout Olympiad from Introplay, 196, 197, 201, 319
- registering with Facebook
  - base options, 31–34, 250–251
  - fixing errors, 36
  - installation options, 33–35, 251–252
  - integration point options, 35–36, 252
  - New Application page, 31
  - overview, 30, 250
  - and viewing in My Applications page, 36–37, 253
- storing user session keys, 306–307
- tips for making apps popular
  - attend to user reviews/feedback, 324
  - avoid social app faux pas, 321–322
  - brand effectively, 322
  - communicate wisely with users, 323
  - engage potential users with About page, 323
  - man discussion boards, 323
  - overview, 321
  - promote on Facebook, 324
  - react quickly to user growth, 325
  - stay on top of platform changes, 324
  - think and act social, 322
- Facebook Marketplace methods in
  - Facebook API, 54, 310–312
- Facebook Platform. *See also* Facebook API; FBJS; FBML; FQL
  - defined, 16
  - developer tools
    - API Test Console, 167–170
    - FBML Test Console, 167, 170–173
    - Feed Preview Console, 167, 173–175
    - Firebug FBJS debugger, 168, 176
    - overview, 167–168
  - Facebook API, 24, 49
  - Facebook JavaScript, 26, 143–144
  - Facebook Markup Language, 25, 95
  - Facebook Query Language, 25–26, 121
  - overview, 3–4, 9

Facebook widgets/UI elements. *See* FBML

- FBJS (Facebook JavaScript)
  - debugging with Firebug, 168, 176
  - defined, 26, 143–144
  - DOM properties
    - accessing, overview, 144–145
    - get methods for accessing, 145–146
    - set methods for setting, 145–148
    - setting `innerHTML` property, 147–148
  - styling elements using `setStyle()`, 148–149
  - tips and techniques, 150
  - using external .js files on canvas pages, 149–150
  - using the FBJS Animation library
    - adjusting size/visibility of block-level events, 156–159
    - adjusting speed/duration, 156
    - animation methods, 151–152
    - changing background color, 153–155
    - increasing font size, 152–153
    - moving objects, 155
    - open-source version of, 153
    - overview, 150–151
    - tweening animation, 152–155
  - working with AJAX in
    - defined, 159
    - FBJS AJAX object members, 159
    - providing dynamic feedback to users, 160–165
- FBML (Facebook Markup Language). *See also* Facebook Platform
  - attachment tag, 102
  - canvas pages using iframes versus, 179–181, 250
  - conditional tags
    - for canvas pages, 106–109
    - defined, 98–99
    - `fb:default`, 99, 112
    - `fb:else`, 98, 108, 111
    - `fb:if`, 111
    - `fb:if-can-see`, 96, 107
    - `fb:if-can-see-photo`, 96, 108
    - `fb:if-is-app-user`, 108–109
    - `fb:if-is-friends-with-viewer`, 109

- fb:if-is-group-member, 109
- fb:if-is-in-network, 109
- fb:if-is-user, 98, 109
- fb:if-multiple-actors, 243
- fb:if-user-has-added-app, 108
- fb:switch, 98, 111–112
- fb:visible-to\*, 109–111
- overview, 106
- for profile boxes, 106, 109–111
- warning, 111
- creating data entry forms in canvas pages
  - defined, 190
  - fb:editor-\* control elements, 191–194
  - fb:editor form element, 190–192
- creating navigation headers in canvas pages
  - adding borders, 189
  - adding image banners, 190
  - fb:action links, 184
  - fb:create-button, 184
  - fb:dashboard header, 183–185
  - fb:header customization, 185, 188–190
  - fb:help link, 184
  - fb:tabs and fb:tab-item links, 185–188
  - fb:title of app home page, 187
  - getting rid of padding, 189–190
  - overview, 181–182
- default FBML for app profile boxes, 42–43, 198–199
- defined, 25, 95
- dialog box tags, 100
- editor form/control tags, 100, 190–194
- Facebook field placeholders, 96
- Facebook widgets/UI elements
  - adding Walls to pages, 96, 114–115
  - benefits of, 112
  - building Walls, 96, 115–116
  - creating discussion boards, 113–114
  - defined, 96, 101–102, 112–113
  - fb:board, 113–114
  - fb:comments, 114–115
  - fb:wall, 96, 115–116
- fb:error tag, 96
- fb:friend-selector tag, 258
- fb:google-analytics tag, 309
- fb:mobile tag, 302–303
- fb:photo tag, 96, 108
- fb:random tag, 268–269
- fb:redirect tag, 310
- fb:ref tag, 219–220
- fb:swf tag, 112
- fb:user-agent tag, 308–309
- FBML methods in Facebook API, 54
- FBML Test Console, 167, 170–173
- general markup tags, 96
- HTML tags, 97
- media tags, 99, 112
- miscellaneous tags, 102–103
- overview, 22–23
- page navigation tags, 101
- presentation tags, 96
- profile tags, 99
- request form tags
  - for condensed forms, 120
  - defined, 100–101
  - fb:multi-friend-input, 258
  - fb:multi-friend-selector, 116–120
  - fb:req-choice, 116–119
  - fb:request-form, 116–120, 278–279
  - for invitation forms, 278–279
  - for standard-sized forms, 116–119
- status message tags, 99
- user and group tags
  - defined, 97–98
  - fb:name, 96, 103–105
  - fb:profile-pic, 106
  - fb:pronoun, 105–106
  - fb:user, 98
- fbml.setRefHandle tag, 219–220
- fbRecipe sample application. *See also* Facebook applications
  - adding a dashboard, 183–185
  - adding a header, 188–190
  - adding tabs, 185–188
  - customizing profile box content, 202–213
  - updating profile box content, 199–201
  - using feed.
    - publishTemplatizedAction in, 244–245

- feed methods. *See also* Facebook API; Mini-Feed; News Feed
  - `feed.publishActionOfUser`, 52, 237, 238–240
  - `feed.publishStoryToUser`, 52, 236–238
  - `feed.publishTemplatizedAction` defined, 52
  - `fb:if-multiple-actors` tag, 243
  - fixing grammar in aggregated stories, 243
  - parameters for, 241–242
  - publishing to non-app users, 240, 245
  - registering story templates, 245–247
  - restrictions on, 237
  - story aggregation, 240
  - using in `fbRecipe` application, 244–245
  - using tokens, 240–243
  - warning, 241
- Feed Preview Console, 167, 173–175
- fields. *See* tables
- Firebug debugger, 168, 176
- Firefox browser user-agent strings, 308
- Flixster application, 318
- footers, canvas page, 270
- `foreach` loop warnings, 72, 82
- forms. *See also* FBML
  - data entry forms
    - defined, 190
    - `fb:editor-*` control elements, 191–194
    - `fb:editor` form element, 190–192
  - request forms
    - condensed forms, 120
    - defined, 100–101
    - `fb:multi-friend-selector`, 116–120
    - `fb:req-choice`, 116–119
    - `fb:request-form`, 116–120, 278–279
    - invitation forms, 278–279
    - standard-sized forms, 116–119
- FQL (Facebook Query Language). *See also* Facebook Platform
  - defined, 25–26, 121
  - versus Facebook API, 26, 122
  - `fql.query` API method, 54, 123, 125
  - manipulating results using FQL functions
    - combining two fields example, 139–142
    - defined, 139
    - derived attributes and, 141
    - JSON results format and, 141
    - in `SELECT` clauses, 139–141
    - in `WHERE` clauses, 141–142
  - overview, 22–23
  - versus SQL, 123–124
  - writing basic queries
    - changing field order in results, 127–128
    - `FROM` clause in, 124
    - indexable table fields, 126–127
    - result formats, 125–126
    - retrieving subfield data from arrays, 128
    - searchable tables, 124–125
    - `SELECT` clause in, 124, 127
    - using `WHERE` operators in, 128–129
    - `WHERE` clause in, 124, 126
  - writing complex queries with `IN` operator
    - defined, 129, 133
    - events attached to a page, 129–138
    - first names of friends with current app installed, 138
    - pages a user is fan of, 138–139
- friends. *See also* Speed Dial application
  - with current app installed, 138
  - inviting to add applications, 18
  - sending notifications to conditionally, 275
  - Top Friends application, 318
- friends methods. *See also* Facebook API
  - defined, 51
  - `friends.appUsers`, 60–61
  - `friends.areFriends`, 58–60
  - `friends.get`, 57–58
  - overview, 56
- friends tags. *See also* FBML
  - `fb:friend-selector`, 258
  - `fb:if-is-friends-with-viewer`, 109
  - `fb:multi-friend-input`, 258
  - `fb:multi-friend-selector`, 116, 117–120
- Friendster, 10
- `FROM` clause in FQL queries, 124

## • G •

get methods in FBJS for accessing DOM properties, 144–146  
get\_cell\_contents() PHP function, 272–273  
get\_profile\_fbml() PHP function, 271, 273  
Google Analytics, 309  
graphics. *See* images; animation  
groups, defined, 72  
groups methods. *See also* Facebook API; user and group tags  
defined, 51  
groups.get, 73–80  
groups.getMembers, 80–83  
overview, 72–73  
warning, 82

## • H •

headers, canvas page, 257. *See also* navigation headers  
HTML tags in FBML, 97

## • I •

if tags. *See* conditional FBML tags  
iframe canvas pages. *See also* canvas pages; CSS  
emulating Facebook styles in  
buttons, 224–225, 229–232  
clearfix utility class and, 225  
dashboards, 223–226  
Facebook color palette and, 222  
including graphics, 225  
navigation tabs, 226–228  
overview, 220–221  
subtitle sections, 228–229  
text formatting, 221–222  
two-column lists, 232–234  
versus FBML canvas pages, 179–181, 250  
iframes, defined, 32, 179

iLike application. *See also* Facebook applications  
Add Music button, 14  
defined, 314–315  
overview, 196  
images. *See also* animation  
adding to About pages, 45  
adding to application icons, 41–42  
creating image banners in FBML, 190  
defining links to in News Feed, 236, 237  
including in iframe canvas pages, 225  
IN operator. *See also* FQL  
defined, 129, 133  
using in FQL queries  
events attached to a page, 129–138  
first names of friends with current app installed, 138  
pages a user is fan of, 138–139  
indexable table fields in FQL, 126–127  
index.php sample listing, 280–286  
inline CSS styles, adding to FBML canvas pages/profile boxes, 216  
innerHTML DOM property, setting in FBJS, 147–148  
installation options for Facebook apps, 33–35, 251–252  
integration points for Facebook apps. *See also* Facebook applications  
About pages, 16–17  
application directory, 16–17  
attachments, 14–15, 21  
canvas, 13, 18–19  
e-mail alerts, 21  
giving users choices about, 17–18  
Google Analytics, 309  
inviting friends to add apps, 18  
Left Nav, 13, 19–20  
Marketplace, 310–312  
News Feed, 21  
profile action links, 20–21  
profile boxes, 13, 19–20, 195  
requests to users, 21  
setting during registration, 35–36, 252  
user privacy settings, 21–22

Internet Explorer 6.0 user-agent strings, 308

Introplay's Runlicious application, 319

Introplay's Workout Olympiad application.  
*See also* Facebook applications  
 defined, 319  
 profile box, defined, 196–197  
 profile box, updating, 201

invitation forms, 278–279

invite.php sample listing, 296–297

iRead application, 316

## • J •

JavaScript, Facebook. *See* FBJS

JavaScript, using on canvas pages, 149–150

JavaScript and XML, Asynchronous.  
*See* AJAX

JSON format, FQL functions and, 141

JSONP, defined, 169

## • K •

keys, storing user session, 306–307

## • L •

Left Nav sidebar, 12–13, 19, 20

Local Picks application from Trip Advisor,  
 182, 314

lower camel case, 148

## • M •

marketplace methods in Facebook API, 54,  
 310–312

markup languages, 2

media tags in FBML, 99, 112

Microsoft Internet Explorer 6.0 user-agent  
 strings, 308

Mini-Feed. *See also* News Feed  
 defined, 12–13  
 feed.publishActionOfUser method,  
 52, 237–240

feed.  
 publishTemplatizedAction method  
 defined, 52  
 fb:if-multiple-actors tag, 243  
 using in fbRecipe app, 244–245  
 fixing grammar in aggregated  
 stories, 243  
 parameters for, 241–242  
 publishing to non-app users, 240, 245  
 registering story templates, 245–247  
 restrictions on, 237  
 story aggregation, 240  
 using tokens, 240–243  
 warning, 241

versus News Feed, 13, 238

publishing stories  
 at same time as News Feed, 238–240  
 to non-app users, 240, 245  
 restrictions on, 237  
 templatized, to at same time as News  
 Feed, 240–247  
 tips for, 247  
 testing with Feed Preview Console, 167,  
 173–175

mobile Facebook applications, 302–303

My Applications page  
 registering feed story templates, 246–247  
 viewing registered apps on, 36–37, 253

My Flickr application, 13–14

MySpace, 10, 15

## • N •

narrow profile boxes. *See also*  
 profile boxes  
 adding content with fb:narrow,  
 202–205, 207–212  
 customizing style properties for, 218–219  
 in Speed Dial application, 271–272,  
 274–275

navigation headers, creating using FBML.  
*See also* canvas pages; FBML  
 adding borders, 189  
 adding image banners, 190

- emulating tabs in iframe canvas pages, 226–228
- fb:action links, 184
- fb:create-button, 184
- fb:dashboard header, 183–185
- fb:header customization, 185, 188–190
- fb:help link, 184
- fb:tabs and fb:tab-item links, 185–188
- fb:title of app home page, 187
- getting rid of padding, 189–190
- overview, 181–182
- New Application page in Developer site, 31
- News Feed. *See also* Mini-Feed
  - defined, 10–11
  - defining image links, 236–237
  - feed.publishActionOfUser method, 52, 237–240
  - feed.publishStoryToUser method, 52, 236–238
  - feed.
    - publishTemplatizedAction method defined, 52
    - fb:if-multiple-actors tag, 243
    - fixing grammar in aggregated stories, 243
    - parameters for, 241–242
    - publishing based on user actions, 276
    - publishing to non-app users, 240, 245
    - registering story templates, 245–247
    - restrictions on, 237
    - story aggregation, 240
    - using in fbRecipe app, 244–245
    - using tokens, 240–243
    - warning, 241
  - integrating Facebook apps into, 21
  - versus Mini-Feed, 13, 238
  - overview, 235–236
  - publishing stories
    - conditionally, 277
    - to current users, 236–238
    - to non-app users, 240, 245
    - restrictions on, 237
    - to at same time as Mini-Feed, 238–240
    - templatized, based on user actions, 276
    - templatized, to at same time as Mini-Feed, 240–247
    - tips for, 247
    - testing with Feed Preview Console, 167, 173–175
  - notifications methods. *See also* Facebook API
    - notifications.get, 53
    - notifications.send
      - to current user, 247–248
      - defined, 53, 247
      - to friends conditionally, 275
      - to lists of users, 248
      - restrictions, 237
    - notifications.sendEmail, 53
  - notify\_friend() function in PHP, 275
- 0 ●
- operators. *See* FQL
- optimizing application speed, 301–302
- p ●
- page navigation tags in FBML, 101
- pages, 83. *See also* About pages; canvas pages; profile pages
- pages methods. *See also* Facebook API
  - defined, 52
  - overview, 83
  - pages.getInfo, 84–86
  - pages.isAdmin, 87
  - pages.isAppAdded, 86
  - pages.isFan, 87
  - properties of, 83–84
- photo tags in FBML, 96, 108
- Photos application, 181
- photos methods in Facebook API, 52
- Poison Control. *See* Speed Dial application
- popularity tips. *See also* Facebook applications
  - attend to user reviews/feedback, 324
  - avoid social app faux pas, 321–322
  - brand effectively, 322

- popularity tips (*continued*)
    - communicate wisely with users, 323
    - engage potential users with About page, 323
    - man discussion boards, 323
    - overview, 321
    - promote on Facebook, 324
    - react quickly to user growth, 325
    - stay on top of platform changes, 324
    - think and act social, 322
  - presentation tags in FBML, 96
  - profile boxes. *See also* profile pages
    - adding CSS styles to
      - embedded styles, 216–217
      - external style sheets, 218
      - `fbml.setRefHandle` in, 219–220
      - inline styles, 216
      - overview, 215–216
      - using `fb:ref`, 219–220
      - wide and narrow box styles, 218–219
    - comparing approaches to, 196–197
    - defined, 12–14, 196
    - displaying personalized content, 198
    - FBML conditional tags for, 106, 109–111
    - FBML tags for adding content
      - defined, 99
      - `fb:action`, 201, 205
      - `fb:narrow`, 202–205, 207–212
      - `fb:subtitle`, 201–203, 205
      - `fb:user-item`, 201–204, 206
      - `fb:user-table`, 201–204, 206–207
      - `fb:wide`, 201–202, 204, 211
    - integrating Facebook apps into, 13, 19–20, 195
    - My Flickr profile box, 13–14
    - overview, 195
    - rendering canvas page content to, 271–273
    - setting content in Speed Dial app, 271–275
    - setting default FBML for, 42–43, 198–199
    - showing/hiding content with
      - `fb:visible-to-*` tags, 109–111
    - updating with `profile.setFBML`, 198–201
    - updating with `render_profile_box()`, 273–274
    - user profiles in
      - adding profile action links to, 212–213
      - updating en masse using `fb:ref`, 219–220
      - what is displayed in, 198
  - profile methods in Facebook API, 53
  - profile pages. *See also* canvas pages; Facebook applications; profile boxes
    - canvas, 12–13
    - defined, 12
    - Left Nav sidebar, 12–13, 19–20
    - Mini-Feed, 12–13
    - overview, 195
    - profile action links
      - adding to user profiles, 212–213
      - defined, 197–198
      - `fb:profile-action` tag, 212–213
      - integrating Facebook apps via, 20–21
      - overview, 195
      - `profile.setFBML` tag, 199–201, 212–213
      - updating, 199–201
    - profile boxes, 12–14
    - Walls, 14–15
  - profile tags in FBML, 99
  - `profile.phpsample` listing, 292–294
  - publishing stories. *See* News Feed; Mini-Feed
  - `publish.php` sample listing, 294–296
- 
- **Q** •
  - querying Facebook data. *See* FQL
  - Quizzes application, 317
- 
- **R** •
  - random quote displays, 268–269
  - Recipe application. *See* `fbRecipe` sample app
  - records/rows in tables, 122
  - redirecting Facebook apps to new URLs, 310

- registering
    - Facebook apps. *See also* Facebook applications
      - base options, 31–34, 250–251
      - fixing errors, 36
      - installation options, 33–35, 251–252
      - integration point options, 35–36, 252
      - New Application page, 31
      - overview, 30
      - and viewing in My Applications page, 36–37, 253
    - feed story templates, 245–247
  - `render_profile_box()` PHP function, 273
  - request form tags. *See also* FBML; forms
    - for condensed forms, 120
    - defined, 100–101
    - `fb:multi-friend-selector`, 116–120
    - `fb:req-choice`, 116–119
    - `fb:request-form`, 116–120, 278–279
    - for invitation forms, 278–279
    - for standard-sized forms, 116–119
  - REST (REpresentational State Transfer) operations, 56
  - rows/records in tables, 122
  - Runlicious application from Introplay, 319
- S ●
- Safari browser user-agent strings, 308
  - Scrabulous application, 315
  - SELECT clauses. *See also* FQL
    - in FQL queries, 124, 127
    - using FQL functions in, 139–141
  - session keys, storing user, 306–307
  - set methods. *See also* FBJS
    - for accessing DOM properties, 145–148
    - listed, 145–146
    - `setInnerFBML()`, 145, 147
    - `setInnerText()`, 148
    - `setInnerXHTML()`, 145, 147
    - `setStyle()`, 148–149
    - for setting element styles, 148–149
    - social graph, 15–16
    - social networking, 1, 15
    - Speed Dial application. *See also* Facebook applications
      - About page, 279–280
      - adding an invitation form, 278–279
      - building the canvas page
        - adding dial friends, 258–259
        - adding a page footer, 270
        - adding a random quote display, 268–269
      - creating a header, 257
      - deleting dial friends, 262–263
      - getting list of dial friends, 259–260
      - overview, 256
      - previewing the Speed Dial, 260–261
      - processing user actions, 262–263
      - resetting the Speed Dial, 262
      - styling UI elements, 266–267
      - using a two-column list style, 263–265
    - canvas page, rendering, 271
    - canvas page, setting up, 254–255
    - coming up with design goals, 249–250
    - connecting to Facebook, 255–256
    - creating the database, 253–254
    - deciding between iframes and FBML, 250
    - Poison Control
      - assigning friends to, 261–263
      - defined, 250
      - unassigning friends from, 262
    - profile box content
      - rendering canvas content to, 271–273
      - updating with `render_profile_box()`, 273–274
      - viewing in wide and narrow columns, 271–272, 274–275
    - publishing stories conditionally, 277
    - publishing templated actions, 276
    - sending notifications, 275
    - setting up in Facebook
      - base options, 250–251
      - canvas page URL, 250
      - installation options, 251–252
      - integration point options, 252
      - overview, 250
      - viewing in My Applications page, 253

Speed Dial application (*continued*)  
 source code architecture, 254–255  
 source code listings  
 12-1: `index.php`, 280–286  
 12-2: `appinclude.php`, 286–287  
 12-3: `display.php`, 287–290  
 12-4: `db.php`, 290–291  
 12-5: `profile.php`, 292–294  
 12-6: `publish.php`, 294–296  
 12-7: `invite.php`, 296–297  
 speed, optimizing application, 301–302  
 SQL (Structured Query Language) versus  
 FQL, 123–124  
 start tags, 2  
 status message tags in FBML, 99  
 styles, Facebook. *See also* CSS  
 emulating in iframe canvas pages  
 buttons, 224–225, 229–232  
 clearfix utility class and, 225  
 dashboards, 223–226  
 Facebook color palette and, 222  
 including graphics, 225  
 navigation tabs, 226–228  
 overview, 220–221  
 subtitle sections, 228–229  
 text formatting, 221–222  
 two-column lists, 232–234  
 overview, 215, 220–221  
 styling elements using FBJs `setStyle()`,  
 148–149  
 subtitle sections, creating in iframe canvas  
 pages, 228–229  
 Super Wall application, integrating into  
 Facebook, 16–22

## • T •

tables  
 defined, 122  
`fb:user-table` element, 201–204,  
 206–207  
 in FQL queries. *See also* FQL  
 changing field order in results, 127–128  
 combining two fields into one, 139–142  
 indexable fields, 126–127

retrieving subfield data from arrays, 128  
 searchable tables, 124–125  
 placeholders for in FBML, 96  
 records/rows/fields in, 122  
 tags, defined, 2  
 templated actions. *See* feed methods  
 text formatting in Facebook, emulating in  
 iframe canvas pages, 221–222  
 token variables, 240–243  
 Top Friends application, 318  
 Trip Advisor's Local Picks application,  
 182, 314  
 tweening animation. *See* FBJs; animation  
 two-column list style  
 creating in FBML, 263–265  
 emulating in iframe canvas pages,  
 232–234

## • U •

UI elements/widgets. *See also* FBML  
 adding Walls to pages, 96, 114–115  
 benefits of, 112  
 building Walls, 96, 115–116  
 creating discussion boards, 113–114  
 defined, 96, 101–102, 112–113  
`fb:board`, 113–114  
`fb:comments`, 114–115  
`fb:wall`, 96, 101, 115–116  
 styling UI elements with CSS, 266–267  
 UI styles, Web address of Facebook, 263  
 updating  
 applications when platform changes, 324  
 profile action links, 199–201  
 profile boxes with `profile.setFBML`,  
 198–201  
 profile boxes with  
`render_profile_box()`, 273–274  
 user profiles en masse, 219–220  
 user and group tags. *See also* FBML; groups  
 methods; users methods  
 defined, 97–98  
`fb:if-is-app-user`, 108–109  
`fb:if-is-group-member`, 109  
`fb:if-is-user`, 98, 109

- fb:if-user-has-added-app, 108
  - fb:name, 96, 103–105
  - fb:profile-pic, 106
  - fb:pronoun, 105–106
  - fb:user, 98
  - user profiles. *See also* profile boxes; profile pages
    - adding profile action links to, 212–213
    - updating en masse, 219–220
    - what is displayed in, 198
  - user-agent strings, 309–310
  - users
    - getting events of with `events.get`, 62–69
    - getting pages particular users are fans of, 138–139
    - giving choices about integrating applications, 17–18
    - integrating Facebook apps via requests to, 21
    - privacy settings, integrating Facebook apps via, 21–22
    - providing dynamic feedback to via AJAX, 160–165
    - storing session keys of, 306–307
  - users methods. *See also* Facebook API; user and group tags
    - defined, 51
    - overview, 87
    - `users.getInfo`, 88–92
    - `users.getLoggedInUser`, 87
    - `users.hasAppPermissions`, 93
    - `users.isAppAdded`, 93
- **W** •
- Walls. *See also* UI elements/widgets
    - adding attachments to, 14–15
    - adding to pages, 96, 101, 114–115
    - creating post by post, 101, 115–116
    - defined, 14
    - Super Wall application, 16–22
  - Web application architecture, 22–23
  - Web browser user-agent strings, 309–310
  - Web programming languages, 1–2
  - Web site addresses
    - Data Store API, 325
    - Facebook
      - API Test Console, 168
      - application ads info, 324
      - developer links, 27, 198, 263, 325
      - Facebook UI styles, 263
      - FBML Test Console, 170
      - Feed Preview Console, 173
      - home page, 10
      - mobile site, 302
      - My Applications page, 246
      - prohibited practices, 321
      - sample files, 5
    - Firebug FBJS debugger, 176
    - Google Analytics, 309
    - Introplay, 201
    - open source FBJS Animation library, 153
    - PHP/Java/and other client libraries, 24
    - recommended applications
      - Appsaholic, 319
      - Attack!, 315
      - Flixster, 318
      - iLike, 314
      - iRead, 316
      - Local Picks from Tripadvisor, 314
      - Quizzes, 317
      - Runlicious from Introplay, 319
      - Scrabulous, 315
      - Top Friends, 318
      - Where I've Been, 317
  - WHERE clauses. *See also* FQL
    - using FQL functions in, 141–142
    - in FQL queries, 124, 126
  - Where I've Been application, 196–197, 317
  - WHERE operators in FQL queries, 128–129
  - wide profile boxes. *See also* profile boxes
    - adding content with `fb:wide`, 201–202, 204, 211
    - customizing style properties for, 218–219
    - in Speed Dial application, 271–272, 274–275
  - widgets/UI elements. *See also* FBML
    - adding Walls to pages, 96, 114–115
    - benefits of, 112

widgets/UI elements (*continued*)

building Walls, 96, 115–116

creating discussion boards, 113–114

defined, 96, 101–102, 112–113

fb:board, 113–114

fb:comments, 114–115

fb:wall, 96, 101, 115–116

Workout Olympiad app from Introplay

defined, 319

profile box, defined, 196–197

profile box, updating, 201



XML, Asynchronous JavaScript and.

*See* AJAX