

## A

### asset management

- JavaScript bundling for, 101–102
- site configuration for, 92
- using Hugo pipes for, 91
- using Webpack for, 91, 91n1

Axios, using for remote data, 83, 102

## B

baseURL, 4

blog, adding comments with Disqus, 71–75

blogging with Hugo, 55

## C

categories, adding to Hugo, 58

Chocolatey, installing Hugo with, 2

code conventions, xiv

command line conventions, xiv

### config.toml

- adding Disqus configuration to, 73
- adding file to Git configuration, 113
- adding Google Analytics ID to, 36
- adding related content configuration to, 77
- adding remote data sources to, 44
- adding site configuration data to, 35
- adding theme configuration to, 17
- baseURL setting in, 4
- basic configuration values for, 4
- languageCode setting in, 4
- Netlify configuration in, 113
- title setting in, 4
- traditional web server baseURL configuration, 118

### content

- organizing with taxonomies, 58
- rendering as JSON, 50–52
- syndicating with RSS, 48–50

### content archetypes

- for blogging, 55–56
- as boilerplate files for content creation, 26
- as configuration for page styling, 26
- as content templates, 8
- as holders of specific content, 25

content blocks and partials, as site building blocks, 20

content blocks, as layout placeholders, 17

content management, using page bundles, 96–99

### content sections

- adding, 25
- as collections of content, 25

conventions, code, xiv

conventions, command line, xiv

### CSS

- file locations for, 21, 22

- fingerprinting, 92–93
- Flexbox for site display, 22
- loading stylesheet for site, 23
- minify stylesheet, 92–93
- styling for pagination, 70–71
- styling theme with, 21
- using Sass preprocessor with, 93–94
- using to help assistive technology, 21

## D

- `_default` directory, as page layout location, 10

### data

- conditional display for, 40–42
- populating page content with, 36–38
- using GitHub API as remote source, 44
- using local data files for, 42–44
- using meta tags to incorporate into site, 35
- using remote sources for, 44–48

- Disqus, configuration for blogging, 71–75

### drafts

- showing pages marked as false, 9
- skipping pages marked as true, 9

## F

- file locations, theme vs site, 23

- Flexbox, using for site display, 22

- front matter, predefined fields for, 38, 38n1

## G

- `GetPage` function, 47

### Git

- adding to `config.toml` configuration, 113
- basic commands for Netlify deployment, 110–111
- configuration and installation for, 110

## H

- home page layout, structure for, 5–6

- Homebrew, installing Hugo with, 2

- Hugo, extended version, 1

- Hugo, installation, *see* installing Hugo

- Hugo, migration from other platforms, 123

- `hugo --cleanDestinationDir` command, 12

- `hugo --cleanDestinationDir --minify` command, 12

- `hugo` command, 11

- `hugo new` command, 9

- `hugo new site` command, 3

- `hugo new theme` command, 16

- `hugo server` command, 7

- Hugo server, stopping with `Ctrl+C`, 8

- `hugo version` command, 2

## I

- `_index.md` file, as content holder, 7

## image management

- Markdown syntax to include, 96
- relURL function for image paths, 95
- site configuration options for, 95
- using page bundles for, 97–98
- using shortcodes for, 98–101

## installing Hugo

- on Linux, 1
- on macOS, 1
- using package managers for, 1, 2
- on Windows, 2

## J

### JavaScript

- Axios library for remote data, 83
  - Lunr library for search, 79–80
  - using asset management for, 101–102
  - using minify for bundling, 101–102
- JSON, scripting site configuration with, 103

## L

### layouts

- for blogging, 56–58
- as page structure, 5, 29–31, 56–58

layouts directory, creating, 10, 56

### lists

- adding content to list pages, 31–32
- adding pagination for blog entries, 68–71
- creating a template for, 28
- customize using the theme, 32
- customizing for blog entries, 65–67
- default layout location for, 27–28

Lunr, using as search tool, 79–80, 102

## M

minify (clean) site, 12

## N

Netlify, build configuration settings for, 111–112

npm, using as site package manager, 102–103, 104

## P

pagination, adding for lists, 68–71

### partials

- as content placeholders, 17–18
- editing to add content, 18
- file locations for, 17–19
- for header and footer content, 18, 19
- as navigation location configuration, 18

post content type, for blogging, 55

### public directory

- removing, 12
- as site layout, 11

as static site container, 12

## R

reading time, configuring for blog posts, 57–58

related blog content, display configuration, 75–77

relative URLs, configuring with relURL function, 23

## S

Sass, using as CSS preprocessor, 93–94

search

- creating document collection for, 80–81

- form interface creation for, 82–86

- improvements for, 86–87

- issues with large indexes, 82

- stemming behavior for, 87, 87n5

- tools available for, 79, 79nn1–3

- using client-side JavaScript, 79

- using Lunr, 79–80

shortcodes, for image management, 98–101

site configuration file, *see* config.toml

site deployment

- AWS account and access key required for, 114–115

- AWS configuration for, 114, 115–117

- cloud storage options for, 114, 117

- Netlify configuration for, 110

- options available for, 109

- using a traditional web server for, 118–120

- using Git to manage files for, 110

- web server configuration requirements for, 118–119

site directory structure, explanation of, 3

syntax highlighting, configuring, 59

## T

tags, adding to Hugo, 58

taxonomies

- disabling in config.toml, 63

- organizing content with, 58

theme

- adding to config.toml, 17

- building, 15

- files needed for, 15

- generator for creating a, 16

- hugo command to create, 16

- list of available, 24n3

- list of files created for, 16

title (site title), 4

## U

URLs

- configuring permalinks for, 63–65

- customizing for blog entries, 62–65

## V

version, show, 2

## W

### Webpack

- as front-end manager for Hugo, 102–104

- plugins for minify and fingerprinter, 104

- using npm run-all module for, 106–107

- using npm to run, 105–106