

PART 2 BUILT-IN CLASSES AND MODULES 189

7 *Built-in essentials* 191

- 7.1 Ruby's literal constructors 192
- 7.2 Recurrent syntactic sugar 193
 - Defining operators by defining methods* 194
 - Customizing unary operators* 196
- 7.3 Bang (!) methods and “danger” 197
 - Destructive (receiver-changing) effects as danger* 197
 - Destructiveness and “danger” vary independently* 198
- 7.4 Built-in and custom to_* (conversion) methods 200
 - String conversion: to_s* 200 ▪ *Array conversion with to_a and the * operator* 202 ▪ *Numerical conversion with to_i and to_f* 203
 - Role-playing to_* methods* 205
- 7.5 Boolean states, Boolean objects, and nil 207
 - True and false as states* 207 ▪ *true and false as objects* 209
 - The special object nil* 211
- 7.6 Comparing two objects 212
 - Equality tests* 212 ▪ *Comparisons and the Comparable module* 213
- 7.7 Inspecting object capabilities 215
 - Listing an object's methods* 215 ▪ *Querying class and module objects* 216 ▪ *Filtered and selected method lists* 217
- 7.8 Summary 217

8 *Strings, symbols, and other scalar objects* 219

- 8.1 Working with strings 220
 - String notation* 220 ▪ *Basic string manipulation* 224
 - Querying strings* 228 ▪ *String comparison and ordering* 230
 - String transformation* 231 ▪ *String conversions* 234
 - String encoding: A brief introduction* 235
- 8.2 Symbols and their uses 237
 - Chief characteristics of symbols* 237 ▪ *Symbols and identifiers* 238 ▪ *Symbols in practice* 240 ▪ *Strings and symbols in comparison* 242
- 8.3 Numerical objects 243
 - Numerical classes* 244 ▪ *Performing arithmetic operations* 244