

# R

- race conditions 630, 650
- random() 111
- Random class 111
- random number generators 19, 111
- Reader 594
- Read Eval Print Loop. *See* REPL
- Ready-Bake Code
  - GameHelper class 152–153
  - simple Chat server 606
  - Songs class 398–399
- records 699–700
- reference variables 49
  - accessing in arrays 83
  - in arrays 61–62
  - assignment 264
  - avoiding dot operator exposure of 80
  - calling methods by 215
  - casting 218
  - comparing 86
  - equality 348
  - garbage-collection heap 57–58
  - HeadFirst interview 56
  - on heap 240
  - life and scope of 260–267
  - memory space allotted for 241
  - nulling 265
  - null reference 58
  - and objects of type Object 213–215, 217
  - polymorphism 187–188
  - size of 56–57
- regions, BorderLayout 514–517
- remote interface. *See* RMI
- repainting objects, GUI 492–495
- replaceAll 694
- REPL (Read Eval Print Loop) 684
- reserved words 53, 328

- return keyword, lambda expression 390
- return types 75
  - ignoring value of 78
  - overloaded methods 193
  - polymorphic 189–190, 192
- return values, autoboxing with 292
- reverse domain package names 685
- RMI (Remote Method Invocation) 553
- run() 612, 613
- Runnable interface 612–617
- runnable thread state 616
- RuntimeExceptions 430
- runtime versus checked exceptions 430
- Ryan and Monica concurrency scenario 641–643, 646, 655–656

# S

- SAM (Single Abstract Method) 341, 389, 394
- scope, variable 260–267
- scrolling (JScrollPane) 524, 527
- security
  - and final classes 191
  - from package organization in Java library 156
- semicolon (;) 12
- Sequence 446–447
- Sequencer class 424–427, 444, 446–447
- Serializable interface 547–550
- serialization 540, 541–550
  - game characters example 554–555
  - process 544–546
  - versioning 556–557
  - writing object to file with stream 542–543
- serialVersionUID 557
- server application, networking 601–603
- server-client relationship 589–593
- server, socket 601

- ServerSocketChannel 601–602
- Set interface 345, 349
- setLayout(null) 522
- setLocationCells() 102
- Set.of() 357
- setter methods 80–82
- Setters. *See* Getters and Setters
- short circuit operators (&&, ||) 151
- ShortMessage instance 450
- ShortMessage.setMessage() 450, 498
- short primitive type 51, 53
- shutdown() 629
- Simple Startup Game 98–124
- SimpleStartupGame class 108–111, 126–130
- SimpleStartupTestDrive class 103–106
- Single Abstract Method. *See* SAM
- Sink a Startup game 96–97, 140–153
- skip(), Stream 375
- sleep() 622–624
- Socket 596–602
- SocketAddress 594, 595
- SocketChannel 591, 594, 595, 601–602
- Song class 316
- Song object 316–319
- sort() 318–319, 325–330
- sorted() 375, 381, 390
- sorting
  - Comparable interface 325–329
  - with Comparator 331–338
  - List 311–313
  - TreeSet 352–354
- source, event 466
- source files, structure of 7
- spillage, variable values 52
- split() 570
- stack 238–241
  - calling methods from 239
  - superclass constructor 254–256
  - threads 610–614
  - variable declarations 238
- stack frame 239
- stack variables. *See* local variables
- StartupBust class 141–148
- Startup class 138–139, 141, 150
- Startup objects 143
- statements 12
- state of an object. *See* instance variables
- static final variables 275, 284, 696
- static helper method 408
- static imports 303
- static initializer 284
- static methods 396
  - event handling 498–499
  - and object locking 649
  - and wrappers 294
- static variables
  - initializing 283
  - versus instance variables 304–305
  - and non-static methods 286
  - and serialization 553
- stream pipeline
  - creating 379–381
  - map operation 405–407
- streams (I/O)
  - reading text files 566
  - receiving messages 594
  - in serialization of object 542–543
  - socket connections 596
- streams (Streams API) 369, 373, 375–420
  - building blocks 379–381
  - collecting results 409–410
  - filtering 400–403
  - getting results from 378
  - inability to reuse 384

- lambda expressions 388–397
- parallel streams 695
- stream() 376, 384
- String arrays 19
- StringBuilder 688
- String class 191
- String.format() method 296
- Strings 62
  - immutability 688
  - percent sign (%) in 297–300
  - reusing 688
  - saving data to text file 559–571
  - sorting in mock Songs class 311–313
  - split() 570
  - wrapping and unwrapping 294–295
- subclasses 31, 168–194
  - conditions for making 229
  - depth of, best practices 191
  - as extensions of superclass 179–183
  - as instantiators under abstract classes 203
  - limitations on 191
  - method implementation designs for 174, 175
  - and polymorphism 190
  - relationship to superclass 182
  - and superclass constructors 256
- super() 256, 258
- superclass 31, 168–194
  - contract rules 192
  - in exception declaration and catch 436–437
  - Graphics 474
  - invoking superclass version of method 230
  - and multiple inheritance 224–225
  - and no reverse inheritance 182–186
  - overriding methods from 192
  - reference type as 188
  - relationship to subclasses 179, 184
- superclass constructor 252–259
- super keyword 182
- Swing 462, 464, 509–532
  - components 510, 523–527

- GUI for Beatbox 528–533
- layout managers 511–522
- switch statement 697–698
- synchronization for concurrency in accessing objects 646–654
- synchronized block or methods, threads 647–649
- synchronized keyword 646–649
- syntax 12, 14
- System.out.print() 15
- System.out.println() 15

## T

- TDD (Test-Driven Development) 101–103
- temporarily not-runnable, thread as 617–619
- terminal operations 377
  - collection query options 410–412
  - as eager 382–383
  - stacking 380
  - in stream operations 379
- test code 99, 145
- Test-Driven Development. *See* TDD
- tester class 36
- testing code, annotations in 692
- text area (JTextArea) 524–525
- text field (JTextField) 523
- text file
  - reading from 566–571
  - saving data to 540
  - writing data to 541, 559–564
- this() 258
- Thread class 609–610
- Thread constructor 614
- thread of execution 609–610
- thread pools 626–629
- threads. *See* multithreading
- thread-safe data structure 664–666

## the index

- thread scheduler 617–619
- throw clause, exceptions 426, 429–432, 436
- toString() 316
- Track 446
- transient keyword 550
- transient variables 549, 553
- TreeSet 352–354
- try/catch blocks 427–430
  - catching multiple exceptions 435, 438
  - exception handling role 443
  - flow control 432–433
  - order of multiple 437–440
  - as polymorphic 436–440
  - wrapping call in 444
- TWR (try-with-resources) statement 576–577
- type inference 312, 698
- type modifiers, number formats 301
- type parameters
  - ArrayList 323
  - generic methods 324, 362
  - not defined in class declaration 324
  - String in angle brackets as 137
- types
  - for lambda expression 394
  - parameters and methods 78
  - variables 50

## U

- underscores for large numbers, formatting 296
- unwrapping a value 290

## V

- values, variable
  - object reference as 55
  - passing when calling a method 74
  - static variables 283
  - wrapping and unwrapping 290

- varargs (variable argument lists) 302, 691
- variables
  - in arrays 59–62
  - assigning values to 52
  - comparing types 86
  - concrete type with var 698
  - declaring 54, 84, 85, 116, 144
  - descriptive naming best practice 698
  - final 275, 284–286
  - of generic types 321
  - instance. *See* instance variables
  - local. *See* local variables
  - naming 50–52, 61
  - primitive. *See* primitive variables
  - reference. *See* reference variables
  - static. *See* static variables
  - syntax 12
  - types 41
  - var for local 698

- version ID, serialization 556–557

- vertical scrollbar 527

- virtual method invocation 177

## W

- while loops 13, 115, 566

- widgets 471
  - drawing 2D graphics 471
  - images on 473

- wrapper constructor 290

- wrappers
  - immutability 688
  - Optional as 410–414
  - for primitive types 290–294

- writeObject() 543

- Writer 595



O'REILLY®

**Learn from experts.  
Become one yourself.**

Books | Live online courses  
Instant Answers | Virtual events  
Videos | Interactive learning

**Get started at [oreilly.com](https://oreilly.com).**