

# Index

---

## Symbols

---

- & (ampersand) as And operator, 159, 160–161
- ' (apostrophe) with character literals, 98
- \* (asterisk). *See also* wildcards
  - for comments, 79–80
  - in compound assignment operator, 129–130
  - as multiplication operator, 120, 124
  - replacing string vowels with, 368–369
- \ (backslash)
  - in escape sequences, 99, 362
  - in regular expressions, 370, 505, 507–508
- { } (braces)
  - for do-while loop, 177
  - in Hello, World! program, 70, 71–72
  - for if statements, 151, 156
  - marking blocks of code, 14, 77
  - pairing of, 71–72
  - in regular expression quantifiers, 504
  - in TextPad, 41, 42
- ^ (caret)
  - for negation, 503
  - as Xor operator, 159, 162–163
- «» (double-arrow symbols)
  - in class diagrams, 254
- = (equal sign)
  - as assignment operator, 82, 128–129
  - in compound assignment operators, 129–130
  - in relational operators, 148–149, 165–166
- ! (exclamation mark)
  - as logical operator, 159–160
  - in relational operator, 148
- > (greater-than sign)
  - for redirecting standard output, 111
  - in relational operators, 148–149
- < (less-than sign) in relational operators, 148–149
- (minus sign)
  - in compound assignment operator, 129–130
  - as decrement operator, 120, 124, 126–127
  - as subtraction operator, 120, 124
  - as UML visibility indicator, 253
  - as unary operator, 125
- ( ) (parentheses) in regular expressions, 505–507
- % (percent sign)
  - in compound assignment operator, 129–130
  - as remainder operator, 120, 121–123, 124
  - as SQL wildcard, 738–739
- | (pipe symbol)
  - as Or operators, 159, 161–162, 177
  - in regular expressions, 506–507
- + (plus sign)
  - as addition operator, 120, 124
  - in compound assignment operator, 129–130
  - as concatenation operator, 103, 362
- as increment operator, 120, 124, 126–127
- as UML visibility indicator, 253
- as unary operator, 125
- “ (quotation marks) with string literals, 102
- ; (semicolon) ending statements, 14, 75–76, 733
- / (slash)
  - for comments, 79–80
  - in compound assignment operator, 129–130
  - as division operator, 120–121, 122–123, 124
- 0 (zero)
  - dividing by, catching exceptions, 227, 233–234
  - dividing by, issues for, 144–145
  - floating-point, 98, 145
  - indexes numbered from, 377, 410, 424
  - number minus itself not always equal to, 146

---

## A

---

- abs method, 132, 133, 134–135
- abstract classes, 299–301
- Abstract Factory pattern, 300
- abstract methods, 299, 303
- accept method, 703
- Access (Microsoft), ODBC setup for, 747–749
- Accessor pattern, 674
- accessors, 260–262
- ActionEvent class, 542, 546

- ActionListener
  - interface. *See also*
  - actionPerformed method
  - implementing, 307, 308, 310, 545
  - in TickTockAnonymous program, 344
  - in TickTockInner program, 338, 339
- actionPerformed method
  - for check boxes, 571, 572
  - in ClickMe3 program, 556
  - coding example, 545
  - for combo boxes, 588, 589
  - data validation, 564–565
  - described, 308, 543
  - for Exit button, 551–552
  - for lists, 593
  - for numeric entry, 563
  - in Pizza program, 580
  - for sliders, 583
  - for text areas, 567
  - for text fields, 558, 561, 563, 564–565
  - in TickTockAnonymous program, 344
  - in TickTockInner program, 339
  - in Timer class, 307
- adapter classes, 553–554
- add method
  - ArrayList class, 407, 409–410, 430
  - ButtonGroup class, 573
  - DefaultListModel class, 594
  - LinkedList class, 419, 422–424
- addActionListener method
  - ActionEvent class, 546
  - JComboBox class, 586, 589
- addChangeListener method, 584
- addChoosableFile
  - Filter method, 700, 702, 704
- addElement method, 594
- addFirst method, 419, 423
- addItem method, 437, 439
- addition operator, 120, 124
- addLast method, 420, 423
- adjectives for interface
  - names, 302–303
- ampersand (&) as And operator, 159, 160–161
- And operator, 159, 160–161
- animation
  - Ball class for, 839–841
  - bouncing ball, 837–839
  - collidable balls, 842–843
  - double buffering for, 837
  - moving ball, 834–837
  - pong, 844–850
  - random balls, 841–842
  - sprites for, 833–834
- AnonClass class, 341–342
- anonymous classes, 341–344
- API. *See* Java API
- apostrophe (') with character literals, 98
- append method, 566, 567
- applet viewer, 35, 45, 637–638
- applets
  - BallRoom, 834–837
  - defined, 13, 631
  - example, 633–636
  - HTML page for, 637
  - image and sound support, 821
  - ImageIcon objects with, 825
  - JApplet class for, 632–633
  - NotPong, 845–850
  - running from TextPad, 45
  - support issues for, 632
- Swing applications versus, 631–632
- Swing components in, 632
- testing, 637–638
- archives. *See* JAR files
- arcs, creating, 806–807
- arithmetic operators
  - algebra versus Java usage, 121
  - combining, 124–125
  - double type with, 120–121
  - int type with, 120, 121
  - overview, 119–122
  - as separators, 122
  - table summarizing, 120
- ArithmeticException, 224, 227, 233–234
- array lists
  - accessing elements, 410–411, 412–413
  - adding elements, 409–410, 418
  - array problems solved by, 406
  - capacity, 409, 410
  - changing objects
    - retrieved from, 414
  - creating, 408–409
  - defined, 406
  - deleting elements, 406, 415–416, 418
  - element insertion easy for, 406
  - element type, 409
  - for loops with, 410–411
  - indexes, 410
  - iterators with, 412–413
  - printing contents of, 411–412
  - resized automatically, 406, 409, 410, 418
  - updating elements, 413–415
- ArrayList class, 406–409, 418–419

arrays  
 array list advantages over, 406  
 ArrayList class use of, 406  
 Arrays class for, 398–403  
 chessboard example, 391–398  
 comparing, 402  
 converting to strings, 402–403  
 copying, 400–401  
 creating, 378–379  
 declaring variables for, 378–379  
 defined, 377  
 elements, defined, 377  
 enhanced for loop for, 383–384  
 filling, 379–380, 399  
 finding length of, 381  
 for loops with, 380–384, 390  
 indexes for, 377–378  
 initializing, 379–380  
 jagged, 388–390  
 length of, 378, 379, 380, 381  
 linked lists versus, 417, 418–419  
 methods with, 384–385  
 multi-dimensional, 390–391  
 as objects, 378  
 printing contents of, 380, 385  
 Quicksort technique for, 517–524  
 as reference types, 95  
 searching, 401–402  
 solving problems with, 381–383  
 sorting, 401  
 splitting strings into, 369–371  
 two-dimensional, 385–390  
 types for elements, 377

Arrays class, 398–399  
 arrows in class diagrams, 254  
 assignment operator  
 compound assignment operators, 129–130  
 for creating objects from classes, 82  
 for initializing arrays, 379  
 relational operator versus, 149  
 using, 128–129  
 assignment statements, 92–93, 342  
 association in class diagrams, 254  
 asterisk (\*). *See also* wildcards  
 for comments, 79–80  
 in compound assignment operator, 129–130  
 as multiplication operator, 120, 124  
 replacing string vowels with, 368–369  
 attributes  
 DTD, 768–769, 776–777  
 XML, 765  
 AudioClip interface, 831–832  
 AutoExec.bat file, 28

## B

backslash (\)  
 in escape sequences, 99, 362  
 in regular expressions, 370, 505, 507–508  
 ball animations. *See* animation  
 Ball class  
 creating, 839–841  
 handling collisions, 842–844  
 BallRoom applet, 834–837

BartServer program  
 BartClient program, 487–490  
 BartQuote class, 482–483  
 BartServer program, 483–486  
 BartServer 2.0 program, 490–493  
 port for, 481  
 protocol for, 481  
 telnet command in, 481–482  
 base class or superclass, 249, 280, 292–294  
 batch files, 44–45  
 beans. *See* JavaBeans  
*Beginning Programming with Java For Dummies* (Wiley publication), 9  
 behavior of objects, 12, 80, 247–248  
 binary numbers, converting to decimals, 143–144  
 binary operators, 121, 148  
 binary streams  
 defined, 706  
 reading, 718–724  
 writing, 724–729  
 binarySearch method, 398, 401–402  
 blank spaces. *See* white space  
 blocks of code  
 defined, 77, 108  
 for if statements, 151  
 for if-else statements, 152–153  
 kinds of, 108  
 not created by indenting, 151  
 overview, 77  
 as scope boundaries, 107–108  
 BoboAndTheLockers program, 381–383  
 bold type in this book, 3

BookCart JavaBean, 685–688

boolean expressions  
 basic form for, 148  
 with conditional operators, 164–165  
 defined, 147  
 logical operators for, 158–164  
 relational operators for, 148  
 side effects of, 161  
 simple examples, 148–149

boolean operators. *See* logical operators

boolean type  
 boxing and unboxing, 107  
 converting strings to, 105  
 described, 95  
 overview, 99–100  
 wrapper class for, 100

boolean variables in do-while loop, 178–179

Border layout, 608, 611–612

BorderFactory class, 575–576

BorderLayout class, 611

borders for components, 574–576

bounds checking, 17

Box class, 612–613

Box layout, 608, 612–614

boxing, 107

braces ( { } )  
 for do-while loop, 177  
 in Hello, World! program, 70, 71–72  
 for if statements, 151, 156  
 marking blocks of code, 14, 77  
 pairing of, 71–72  
 in regular expression quantifiers, 504  
 in TextPad, 41, 42

break statements  
 for avoiding infinite loops, 171–172  
 continue statement versus, 173–174  
 do-while loop with, 176  
 for loop with, 188  
 omitting from switch statement, 201–203  
 switch statement with, 197  
 while loop with, 170

breakpoints for debugging, 62, 64–65

BufferedInputStream class, 718

BufferedOutputStream class, 724, 725

BufferedReader class, 707, 708–709

BufferedWriter class, 712

Burd, Barry, *Eclipse For Dummies*, 48

ButtonGroup class, 572, 573

buttons, 538–539, 550–552, 787–788

BuyMyBook.jsp page, 684–685

byte type, 95, 96

bytecode, 10, 34

## C

C language versus Java, 9, 13–15

C++ language, 9, 17

callbacks, 307–310

calling  
 BorderFactory class methods, 576  
 constructors to create object instances, 264–265

constructors within constructors, 266–268  
 methods, 205

capitalization or case. *See also* case sensitivity  
 for class identifiers, 256  
 converting for strings, 366  
 for package names, 347  
 variable conventions, 88

Card layout, 608

caret (^)  
 for negation, 503  
 as Xor operator, 159, 162–163

CarWashApp program, 201–202

CarWashApp2 program, 202–203

case sensitivity  
 error messages due to, 32–33, 73–74  
 of identifiers, 78  
 of Java, 32–33  
 of keywords, 73–74  
 SQL not sensitive, 733

casting. *See also* converting  
 cast operator for, 106  
 conversion versus, 106  
 downcasting, 290–291  
 operands for division, 121  
 rounding during, 107  
 upcasting, 290

catching exceptions. *See also* exceptions  
 catch blocks for, 226–227, 231–232

ClassNotFoundException  
 Exception, 751  
 defined, 224  
 displaying exception messages, 232  
 from dividing by zero, 227, 233–234

EOFException, 720–721

examples, 227–229

- Exception class for, 231–232
- FileNotFoundException
  - Exception, 236–237
- from invalid integer, 228–229
- SQLException, 751
- try statement for, 226–229, 231–232
- cbirt method, 132, 133, 134–135
- ceil method, 138–139
- ChangeListener
  - interface, 584
- ChangeParameters
  - program, 219
- char type
  - assigning values to, 98–99
  - boxing and unboxing, 107
  - described, 95
  - escape sequences for
    - character constants, 99
  - overview, 98–99
  - with switch statements, 200–201
  - wrapper class for, 100
- character classes
  - custom, 501–503
  - predefined, 499–501
- character literals, 98–99
- character streams
  - classes for, 706–707
  - defined, 706
  - reading, 706–712
  - writing, 712–718
- charAt method, 363, 366–367
- CharSequence interface, 375–376
- check boxes, 569–572
- checked exceptions. *See also* exceptions
  - catching FileNotFoundException
    - Exception, 236–237
  - catch-or-throw compiler error, 235–236
  - catch-or-throw rule for, 235
  - defined, 224
  - handling, 235–239
  - ignoring, 238–239
  - thrown by nextInt
    - method, 224–226, 228–231
  - throws clause in main
    - for, 238
- chessboard array example, 391–398
- child classes. *See* subclasses or derived classes
- CircleAreaApp program, 131–132
- Class class, 333–334
- class constants, 94
- .class filename extension, 32
- class keyword, 70
- class variables or fields
  - accessors for, 260–262
  - adding to interfaces, 305
  - BorderLayout class, 611
  - declaring, 89
  - defined, 89, 259
  - GridBagConstraints
    - class, 618–619
  - initialized to empty strings, 103
  - JScrollPane class, 568–569
  - Node interface, 775
  - overview, 259
  - private keyword for, 259
  - public keyword for, 247, 259
  - scope of, 107–109
  - shadowed, 109–110
  - state of objects described
    - by, 247
  - static, 248, 271–275
  - SystemColor class, 794
- ClassCastException, 291
- classes. *See also* JavaBeans; subclasses or derived classes
  - abstract, 299–301
  - Abstract Factory, 300
  - adapter, 553–554
  - adding to Eclipse project, 56–59
  - association, 254
  - base or superclass, 249, 280, 292–294
  - body of, 256–257
  - for character streams, 706–707
  - compiling from command line, 28–31
  - creating objects from, 82, 248
  - declaring class variables
    - in, 89
  - declaring classes, 70, 255–258
  - declaring instance variables in, 90
  - defined, 12, 80
  - disassembling, 33–35
  - for events, 542
  - for exceptions, 223–224
  - filename extension for, 32
  - final keyword for, 289–290
  - hierarchies of, 282
  - identifier guidelines, 256
  - implementing interfaces, 303–304
  - importing, 86
  - inner or nested, 257, 258, 335–344
  - interfaces in, 257
  - in Java API, 12
  - JSPs with, 668–672
  - loading, 248
  - members, 259–260, 264
  - objects as instances of, 81
  - objects versus, 81
  - outer, 335

- classes (*continued*)
  - preventing instantiation, 277
  - public declaration for, 70, 255, 257–258
  - reference types based on, 100–101
  - running from command line, 31–33
  - servlets using, 654–658
  - for shapes, 803–804
  - singletons, 276
  - source files for, 257–258
  - Tomcat setup for, 668
  - UML diagrams for, 251–254
- classes directory for Tomcat, 642
- ClassNotFoundException
  - Exception, 224, 750, 751
- ClassPath environment variable, 352–353
- .classpath file (Eclipse), 51
- clear method
  - ArrayList class, 407, 415, 416
  - LinkedList class, 420, 427
- ClickMe program, 546–548
- ClickMe2 program, 548–550
- ClickMe3 program, 554–556
- client programs
  - BartClient example, 487–490
  - defined, 471
  - examples, 472
- clone method (ArrayList), 407, 432
- clone method (Object)
  - cloning overview, 323
  - creating a shallow copy, 326–327
  - creating deep copies, 327–332
  - described, 313, 314
  - implementing, 323–326
  - Object type returned, 324
  - as protected, 323
  - super.clone() call for, 327
- Cloneable interface, 306
- CloneNotSupportedException
  - Exception, 306
- CloneTest program, 324–326
- CloneTest2 program, 328–332
- collection classes, 405. *See also* generics
  - ArrayList class, 406–409, 418–419
  - LinkedList class, 418–422
- Color class, 792–793
- ColorChooser program, 796–797
- colors
  - color chooser for, 795–797
  - for components, 795
  - creating, 792–793
  - filling shapes with, 801, 809–812
  - for shapes, 801
  - system, 793–794
  - in TextPad, 41
- column functions (SQL), 739–740
- combining
  - arithmetic operators, 124–125
  - logical operators, 163–164
  - strings (concatenating), 103, 165, 362, 372
- combo boxes, 585–589, 590
- comma-delimited text files, 706
- command-line tools. *See also specific tools*
  - applet viewer, 35, 45, 637–638
  - compiling program using, 28–31
  - disassembler, 33–35
  - file parameters, 697–698
  - jar, 35, 350–354
  - javadoc, 35, 357–358
  - list online, 35
  - MySQL, 733
  - obtaining, 23
  - path setting for, 26–28
  - running program using, 31–33
- comments, 41, 79–80, 83
- compiling
  - catch-or-throw compiler error, 235–236
  - error messages during, 29
  - javac command for, 28–31
  - platform independence issues for compilers, 10
  - program from command prompt, 28–31
  - in TextPad, 42–43
- complement (Not) operator, 159–160
- Component class, 528, 529
- compound assignment operators, 129–130
- compound expressions, 158–159
- concatenating strings
  - concatenation operator, 103, 362
  - conditional operator, 165
  - String class limitations, 372
- Conditional And operator, 159, 160–161
- conditional operator, 121, 164–165
- Conditional Or operator, 159, 161–162, 177
- Connection class, 751, 752, 759–760
- console, defined, 69
- const keyword, 73

- constants
  - Color class, 792
  - final variables, 94
  - for float and double classes, 145–146
  - for initializing arrays, 379–380
  - Math class, 131–132
- constructors by class
  - ArrayList, 407, 409
  - BorderLayout, 611
  - BufferedOutputStream, 725
  - BufferedReader, 707
  - BufferedWriter, 712
  - DataOutputStream, 725
  - DefaultListModel, 594
  - DefaultMutableTreeNode, 599
  - Dimension, 613
  - File, 691, 693
  - FileOutputStream, 725
  - FileReader, 707
  - FileWriter, 713, 714
  - FlowLayout, 610
  - Font, 785–786
  - GradientPaint, 810
  - GridLayout, 615
  - ImageIcon, 823
  - JApplet, 633
  - JButton, 538, 823
  - JCheckBox, 569
  - JComboBox, 586
  - JFileChooser, 699
  - JFrame, 530
  - JLabel, 537, 823
  - JList, 591
  - JPanel, 535
  - JRadioButton, 573
  - JScrollPane, 568
  - JSlider, 581
  - JTextArea, 566
  - JTextField, 558
  - JTree, 601
  - LinkedList, 419, 422
  - PrintWriter, 712
  - ServerSocket, 480
  - shape classes, 803–804
  - Socket, 479
  - Thread, 445–446
- constructors in general
  - anonymous classes lacking, 342
  - basic form for, 264
  - calling other
    - constructors, 266–268
  - creating instances by calling, 264–265
  - default, 265–266
  - defined, 101, 248, 256, 264
  - exceptions throwable by, 264
  - inheritance with, 287–288
  - initial values provided by, 264
  - for JavaBeans, empty, 673
  - methods versus, 264
  - overloading, 265
  - preventing instantiation, 277
  - public declaration for, 264
  - for spinners, 595, 596
  - this restrictions for calls, 267–268
- Container class, 528, 529
- context.xml file for Tomcat, 641–642
- continue statements, 173–174, 176, 188–189
- converting. *See also* casting
  - arrays to strings, 402–403
  - automatic for primitive types, 105–106
  - binary numbers to decimals, 143–144
  - case of strings, 366
  - primitives to strings, 104, 362–363
  - strings to primitives, 104–105
- copying
  - arrays, 400–401
  - cloning objects, 323–332
  - copyOf method, 398, 400–401
  - copyOfRange method, 398, 400, 401
- CountBothWays program, 186
- CountDown program, 184–185
- CountDownApp program
  - with aborts, 465–469
  - version 1, 450–452
  - version 2, 453–456
- CountDownAppNew class, 457–458
- CountDownClock class
  - creating threads, 459–460
  - locks for threads, 462
  - version 1, 446–447
- CountDownClockLocked class, 462–463
- CountDownClockNew class, 456–457
- CountDownClockSync class, 460–461
- counting
  - backwards with for loop, 184–185
  - class instances created, 274–275
  - counter variable scope, 182–183
  - even numbers with for loop, 183
  - for loops designed for, 180–181
  - multiple counter variables for, 186–187
  - vowels in string, 367
  - while loops for, 168–169
- CountTestApp program, 274–275
- CountToTen program, 180–182
- CountVowels program, 367
- CountWords program, 370–371

CrazyWithZeros  
 program, 233–235  
 create statement (SQL),  
 733, 735  
 createNewFile method,  
 692, 694  
 createStatement  
 method, 752, 753, 759

## D

data types  
 for array elements, 377  
 for array list elements, 409  
 automatic conversions for  
 primitives, 105–106  
 boxing and unboxing, 107  
 casting, 106–107  
 converting between  
 strings and primitives,  
 104–105  
 defined, 94  
 downcasting, 290–291  
 for generics, 431–432  
 interface use as, 304–305  
 for linked list items, 422,  
 424  
 method return type, 207  
 Object class as, 312–313  
 for objects, 246, 291–292  
 primitive, 94–100  
 primitive versus  
 reference, 95  
 reference, 100–102, 361  
 strongly typed languages,  
 94–95  
 type checking, 15–17  
 upcasting, 290  
 user-defined, 16  
 wildcard type parameters,  
 436–437  
 database connector setup,  
 749  
 database layer, 251  
 databases. *See* relational  
 databases

DataInputStream class,  
 718–721  
 DataOutputStream class,  
 724–727  
 debugging  
 compiler errors, 29, 43  
 in Eclipse, 61–65  
 examining variables,  
 63–64  
 kinds of bugs, 61  
 setting breakpoints, 62,  
 64–65  
 stepping through  
 programs, 61–63  
 in TextPad, 43  
 decimal data  
 converting binary  
 numbers to, 143–144  
 direct support lacking for,  
 19  
 declarations (JSP), 660,  
 666–667  
 declaring. *See also*  
 declaring variables  
 abstract classes, 300  
 abstract method, 299  
 declaration statements  
 for, 75  
 inner classes, 335  
 methods, 71, 81, 206–210  
 parameters for methods,  
 217–218  
 public classes, 70, 255–258  
 public keyword for, 70,  
 71  
 return type for methods,  
 210–211  
 static keyword for, 71,  
 81  
 strings, 102  
 synchronized methods,  
 458–462  
 two-dimensional arrays,  
 386  
 declaring variables  
 for arrays, 378–379  
 basic form for, 88

for check boxes, 570  
 class variables, 89  
 explicit declaration  
 required, 87–88  
 final variables, 94  
 instance variables, 90  
 for linked lists, 422  
 local variables, 90–91  
 multiple variables in one  
 statement, 88  
 shadowing begun by, 110  
 strings, 102  
 decrement operator, 120,  
 124, 126–127  
 deep copy of object,  
 creating, 328–332  
 deepEquals method, 399,  
 402  
 default constructors,  
 265–266  
 default package, 346–347  
 DefaultListModel class,  
 594–595  
 DefaultMutableTree  
 Node class, 598–600  
 Delegation pattern, 283  
 delete method, 692, 696  
 delete statement (SQL),  
 733, 752–753  
 deleteFile method,  
 696–697  
 deleting or removing  
 array list elements, 406,  
 415–416, 418  
 breakpoints in Eclipse, 65  
 combo box items, 588  
 database rows, 742–743,  
 761  
 directories, 696–697  
 files, 696–697  
 linked list items, 418,  
 426–427  
 white space from strings,  
 366  
 delimiters for text files, 706  
 deprecated methods, 30  
 dequeue method, 437, 438

- derived classes. *See*
  - subclasses or derived classes
- Design Pattern icon, 6
- design patterns
  - Abstract Factory, 300
  - Accessor, 261, 674
  - Delegation, 283
  - described, 6
  - Facade, 476
  - Factory, 576
  - icon in book margins, 6
  - Immutable, 362
  - Iterator, 412
  - Marker Interface, 306
  - Singleton, 276
- DialogApp class, 116–117
- DiceApp program, 136–137
- Dimension class, 613
- directives (JSP), 660, 661
- directories
  - defined, 695
  - deleting, 696–697
  - getting contents of, 695
  - in JDK root folder, 25–26
  - for packages, 347–348
  - recursion for listing, 514–517
  - for Tomcat, 642
- DirList program, 514–517
- disassembling classes, 33–35
- displaying. *See also*
  - printing to console
  - dialog boxes, 116–117
  - error messages from
    - do-while loop, 178
  - exception messages, 232
  - JavaDoc pages, 357–358
  - Open dialog box, 700–701
  - Save dialog box, 701
- DivideByZero class, 227
- division operator
  - dividing by zero, catching
    - exceptions, 227, 233–234
  - dividing by zero, issues
    - for, 144–145
  - dividing integers with, 122–123
  - double type with, 120–121
  - int type with, 120–121
  - precedence of, 124
- Document interface, 774
- Document Type Definition (DTD), 767–769
- documentation online. *See*
  - Internet resources
- DocumentEvent class, 542
- DocumentListener interface, 543
- doGet method, 645, 646, 649
- DOM (Document Object Model)
  - described, 769
  - document builder factory
    - for, 771–772
  - document builder for, 772–773
  - getDocument method
    - for, 773
  - getting attribute values, 776–777
  - getting child element values, 777–778
  - interfaces for, 774–775
  - processing elements, 775–776
  - reading documents, 771–773
  - reading nodes, 773–778
  - SAX versus, 769–780
- Domain Name System (DNS), 473
- doPost method, 645, 649
- double buffering, 837
- double type. *See also*
  - floating-point types
  - arithmetic operators with, 120–121
  - described, 95
  - formatting numbers, 139–142
  - overview, 97–98
  - precision of, 97
  - in random number
    - generation, 135–137
  - special constants for, 145–146
- double-arrow symbols (↔)
  - in class diagrams, 254
- do-while loops
  - basic form for, 175
  - boolean variable with, 178–179
  - Conditional Or operator
    - with, 177
  - displaying error messages, 178
  - flowchart for, 175–176
  - guessing game, 190–192
  - nested, 190–192
  - tips for, 176–177
  - validating user input, 177–179
  - while loops versus, 175
- downcasting, 290–291
- downloading. *See* Internet resources
- draw method, 800
- drawImage method, 827
- drawing. *See also* shapes
  - on component, by user, 814–820
  - Image object, 827–828
  - shapes, 800–803, 809
  - text, 814
- DrawingBoard program, 816–820
- drawString method, 814
- drop statement (SQL), 733, 734–735
- DTD (Document Type Definition), 767–769
- Duodecaphobia program, 170, 171–172
- Duodecaphobia2 program, 174

**E**

- 
- E constant, 131
  - Eclipse development environment
    - adding a class file, 56–59
    - buttons for debugging, 63
    - creating simple projects, 51–56
    - debugging programs in, 61–65
    - described, 47
    - downloading, 47
    - editors, 48
    - features (plug-ins), 48
    - further information, 48
    - opening projects, 51
    - perspectives, 49–50
    - projects overview, 50–51
    - refactoring code in, 65–66
    - running programs in, 55–56, 59–60
    - terminology, 48–50
    - views, 48–49
    - Web site, 47
    - Workbench, 48, 49
  - Eclipse For Dummies* (Burd), 48
  - editions of Java, 21
  - Element interface, 774, 775
  - element method, 420, 425
  - elements
    - of array lists, 406, 409–416, 418
    - of arrays, 377, 387–388
    - of DTDs, 767–768, 775–776
    - of trees, 598
  - ellipses, creating, 806
  - else-if statements
    - flowchart for, 158
    - overly complex, 193–194
    - overview, 154–155, 156–158
    - rule for pairing else and if keywords, 155–156
  - end condition for recursion, 512
  - end-of-line comments, 79
  - enhanced for loop, 383–384
  - enqueue method, 437, 438
  - ensureCapacity method, 407, 410
  - EOFException, 720–721
  - equal sign (=)
    - as assignment operator, 82, 128–129
    - in compound assignment operators, 129–130
    - in relational operators, 148–149, 165–166
  - equals method
    - Arrays class, 399, 402
    - Object class, 313, 314, 318–322
    - String class, 166, 363
  - equalsIgnoreCase method, 166, 363
  - erasing. *See* deleting or removing
  - Error class, 296
  - error messages. *See also* debugging; exceptions
    - for case errors, 32–33, 73–74
    - displaying from do-while loop, 178
    - for instance variables in static methods, 90
    - from java command, 32–33
    - from javac command, 29
    - for local variable declarations, 91
    - in TextPad text editor, 43
    - for uninitialized variables, 92
  - escape sequences
    - for character literals, 99
    - for regular expressions, 505
    - for string literals, 507–508
    - in strings, 362
  - EvenCounter program, 168–169
  - EvenCounter2 program, 175
  - event listeners. *See also* actionPerformed method
    - adapter classes for, 553–554
    - for check boxes, 570–571
    - ClickMe2 program, 548–550
    - defined, 307, 541, 543
    - inner classes for, 548–550
    - interface implementation for, 545
    - methods (table), 543–544
    - mouse methods, 814–815
    - registering with event source, 546
    - for sliders, 584
    - for trees, 603–604
    - uses for, 541
    - for window events, 552–554
  - event source, 307, 543, 546
  - events
    - classes for, 542
    - ClickMe programs, 546–550, 554–556
    - for combo boxes, 589
    - components generating, 544–545
    - defined, 541, 543
    - Exit button, 550–552
    - interfaces for, 542
    - low-level, 543
    - semantic, 543
    - steps for handling, 544–546
  - Exception class, 231–232, 296–297
  - exceptions. *See also specific exceptions*
    - approach in Java, 17–18
    - catching, defined, 224

- catching EOFException, 720–721
  - catching with `try` statement, 226–229, 231–232
  - catch-or-throw compiler error, 235–236
  - checked, example, 225–226
  - checked, handling, 235–239
  - checked versus unchecked, 224
  - classes for, 223–224
  - creating custom, 296–297
  - displaying messages for, 232
  - from dividing by zero, 144, 227, 233–234
  - Exception class for, 231–232, 296–297
  - exception objects, 18, 223
  - finally block for, 227, 233–235
  - hierarchy of classes, 295–296
  - importing classes for, 229
  - from invalid integer, 228–229
  - overview, 223–224
  - preventing, 229–231
  - program termination from, 224
  - stack trace for, 225–226
  - throwing, defined, 224
  - throwing custom, 297–298
  - throwing `FileNotFoundException` Exception, 237–238
  - throwing with `throw` statement, 239–240
  - thrown by constructors, 264
  - thrown by `ServerSocket` class, 481
  - thrown by `Socket` class, 480
  - throws clause in `main` for, 238
  - typical examples, 224
  - exclamation mark (!) as logical operator, 159–160
  - in relational operator, 148
  - Exclusive Or (Xor) operator, 159, 162–163
  - `executeQuery` method, 752, 753
  - `executeUpdate` method, 752, 758
  - `exists` method, 692, 694
  - Exit button, 550–552
  - `exit` method, 338, 550
  - `exp` method, 132, 133, 134–135
  - exponential notation, 97, 98
  - expression statements, 75
  - `ExpressionGanging` program, 186–187
  - expressions. *See also* boolean expressions
    - as array indexes, 378
    - for array length, 379
    - in assignment statements, 92–93
    - compound, 158–159
    - in initializers, 93
    - in JSPs, 660, 662–663
    - side effects of, 161
  - extending
    - Exception class, 296–297
    - `HttpServlet` class, 645–646
    - interfaces, 305–307
    - `JavaFilter` class, 703, 704
    - `JPanel` class, 536
    - `Thread` class, 446–447
    - `WindowAdapter` class, 553–554
  - extracting
    - character from string, 366–367
    - local variables in Eclipse, 66
    - methods in Eclipse, 66
    - substrings from string, 367–369
- 
- ## F
- 
- Facade pattern, 476
  - factorial
    - non-recursive solution, 511–512
    - recursive solution, 512–513
  - Factory pattern, 576
  - fields. *See* class variables or fields
  - File class
    - for binary streams, 718
    - for character streams, 706
    - constructor, 691, 693
    - creating a file, 694
    - creating an object, 693
    - deleting files and directories, 696–697
    - described, 691
    - determining existence of file, 694
    - getting directory contents, 695
    - getting information about files, 694–695
    - methods (table), 691–693
    - package for, 691
  - `FileException1` program, 236
  - `FileException2` program, 236–237
  - `FileInputStream` class
    - catching `FileNotFoundException` Exception, 236–237
    - catch-or-throw compiler error, 235–236

- FileInputStream class
  - (continued)
  - constructors, 718
  - described, 718
  - ignoring FileNotFoundException
    - Exception, 238–239
  - throwing FileNotFoundException
    - Exception, 237–238
- FileNotFoundException, 235–239
- FileOutputStream class, 724, 725
- FileReader class, 706, 707
- files. *See also* directories
  - binary, 706
  - binary streams for, 706, 718–729
  - character streams for, 706–718
  - choosing in Swing
    - applications, 698–704
  - command-line parameters for, 697–698
  - creating on disk, 694
  - deleting, 696–697
  - determining existence of, 694
  - File class for, 691–697
  - FileFilter class for, 702–704
  - filters for, 702–704
  - getting information about, 694–695
  - in JDK root folder, 26
  - JFileChooser class for, 698–702
  - Open dialog box for, 700–702
  - renaming, 696
  - Save dialog box for, 701
  - text, 706–718
- FileWriter class, 712, 713, 714
- fill method
  - Arrays class, 399
  - Graphics2D class, 801
- filling arrays, 379–380, 399
- filling shapes, 801, 809–812
- filters for files, 702–704
- final keyword, 288–290, 301
- final variables (constants), 94
- finalize method, 313, 314
- finally block, 227, 233–235
- finding. *See also* searching
  - exceptions with stack trace, 225–226
  - length of arrays, 381
  - type of objects, 291–292
- float type, 95, 97–98.
  - See also* floating-point types
- floating-point types
  - automatic conversions for, 106–107
  - boxing and unboxing, 107
  - casting, 106–107
  - converting strings to, 105
  - converting to decimal numbers, 143–144
  - converting to strings, 104
  - dividing by zero, issues for, 144–145
  - exponential notation for, 97, 98
  - formatting numbers, 139–142
  - overview, 97–98
  - precision of, 97
  - special constants for, 145–146
  - table summarizing, 95
  - using decimal points with, 97–98
  - wrapper classes for, 100
  - zero, positive and negative, 98, 145
- floor method, 138–139
- Flow layout, 540, 607, 609–610
- FlowLayout class, 540, 610
- FocusEvent class, 542
- FocusListener interface
  - methods, 544
- folders. *See* directories
- Font class, 785–786
- fonts
  - for components, 787–788
  - Font class for, 785–786
  - listing available, 788
  - names, 786
  - program, 788–791
  - styles, 786–787
- for loops
  - with array lists, 410–411
  - with arrays, 380–384, 390
  - bodiless, 185, 186–187
  - break statement with, 188
  - continue statement with, 188–189
  - counter variable scope, 182–183
  - counting backwards, 184–185
  - counting even numbers, 183
  - CountToTen program, 180–182
  - enhanced, 383–384
  - factorial calculation with, 511–512
  - flowchart for, 181
  - formal format for, 180–182
  - infinite, 188
  - with linked lists, 424–425
  - multiple counter variables for, 186–187
  - nested, 189
  - omitting expressions, 187–188
  - overview, 179–180
  - in partition method, 521

ForDuodecaphobia program, 188  
 ForDuodecaphobia2 program, 188–189  
 ForEvenCounter program, 183  
 formal type parameters, 431–432  
 format method, 140, 141, 142  
 formatting numbers, 139–142  
 forms with servlets, 651–653  
 forName method, 750–751  
 Frame class, 528, 529  
 frames. *See also* layout managers  
   adding panels to, 535–536, 537–538  
   creating, 531  
   described, 530  
   Exit button for, 550–552  
   exiting applications, 533  
   Hello, World! program, 531–533  
   JFrame class for, 530–531  
   making visible, 531, 533  
   positioning on-screen, 534

## G

garbage collector, 17, 248  
 generics  
   formal type parameters, 431–432  
   as integral to collections, 406  
   queue class, 437–440  
   stack class, 432–436  
   technical nature of, 429  
   uses for, 430  
   wildcard type parameters, 436–437  
 GenQueue class, 437–440  
 GenQueueTest program, 439–440  
 GenStack class, 433–436  
 GenStackTest program, 434–436  
 get accessors, 260–262  
 get method  
   ArrayList class, 407, 410–411, 430  
   ResultSet interface, 754–755  
 GetABet program, 177–178  
 GetABet2 program, 178  
 GetABet3 program, 178–179  
 getAttribute method, 775, 776–777  
 getCanonicalPath method, 692, 695  
 getChildNodes method, 774, 776  
 getClass method, 313, 314, 333–334  
 getConnection method, 758, 759  
 getCurrencyInstance method, 140, 141–142  
 getDefaultToolkit method, 534  
 getDescription method, 703  
 getDocument method, 771, 773  
 getFirst method, 420, 425  
 getFirstChild method  
   DefaultMutableTreeNode class, 599  
   Node interface, 774, 776  
 getImage method, 826, 827  
 getInstanceCount method, 275  
 GetInteger program, 228–229  
 GetInteger2 program, 230–231  
 getLast method, 420, 425  
 getLocalGraphicsEnvironment method, 788  
 getMessage method, 232  
 getName method  
   Class class, 333, 334  
   File class, 692, 694  
 getNextSibling method, 599  
 getNumberInstance method, 140, 141  
 getPercentInstance method, 140, 141  
 getScaledInstance method, 826, 827  
 getScreenSize method, 534  
 getSelectedFile method, 700, 702  
 getSelectedIndex method  
   JComboBox class, 586, 589  
   JList class, 591, 593  
 getSelectedItem method, 586, 588  
 getSelectedValue method, 591, 592  
 getSelectedValues method, 591, 592–593  
 getSelectionMode method, 602  
 getSuperclass method, 333  
 getters, 260–262  
 getText method  
   JTextArea class, 566, 567  
   JTextField class, 558  
 glue, 614  
 Good Morning program, 559–561  
 goto keyword, 73  
 gradient fills, 810–812  
 GradientPaint class, 810–811  
 Graphical User Interface (GUI), 527–528. *See also* Swing

Graphics class methods, 826  
 graphics context for shapes, 799–800  
 Graphics Interchange Format (GIF), 822  
 GraphicsEnvironment class, 788  
 Graphics2D class, 800  
 greater-than sign (>)  
   for redirecting standard output, 111  
   in relational operators, 148–149  
 Greeter class, 84–85  
 Grid layout, 608, 614–615  
 GridBag layout  
   adding components, 618–619  
   described, 608  
   example, 621–624  
   GridBagConstraints for, 618–621  
   plan for, 616–617  
   special features, 616  
 GridBagConstraints class, 618–621  
 GridLayout class, 615  
 Group layout, 608, 624–627  
 GuessingGame program, 190–192  
 GuessingGameMethod program, 208–210  
 GuessingGameMethod2 program, 214–216  
 GuessingGameMethod3 program, 220–222, 226

## H

hash code for objects, 245  
 hashCode method, 313, 314  
 hasItems method  
   GenQueue class, 437, 439  
   GenStack class, 433, 434  
 hasNextInt method, 230

hasNextMethod, 412, 413  
 Hello, World! programs  
   C language compared to Java for, 13–15  
   Eclipse example, 51–60  
   line-by-line discussion, 69–72  
   object-oriented versus procedural version, 84–85  
   servlet examples, 646–648, 649–651  
   Swing version, 531–533  
   TextPad example, 40–45  
   using an object, 82–85  
 HelloApp2.java class, 82–84  
 HelloSayer class, 56–59  
 HelloWorldMethod class, 207–208  
 hierarchies of classes  
   exception classes, 295–296  
   overview, 282  
   Swing classes, 528–529  
 host names, 473  
 HostLookup program, 475–478  
 HTML  
   for applets, 637  
   basic tags, 648  
   for JSPs, 663, 664–665, 666–667  
   printing to Web page, 646  
   for servlets, 646–648, 651–652  
 HTTP (HyperText Transfer Protocol), 639–640  
 HttpServlet class, 645–646  
 HttpServletRequest object, 646  
 HttpServletResponse object, 646  
 hypot method, 132, 133, 134–135

## I

icons  
   ImageIcon class, 822–825  
   in this book, 5–6  
 IDE (Integrated Development Environment), 23  
 identifiers and names  
   avoiding dollar signs for, 78  
   for classes, 256  
   common uses for, 78  
   for constructors, 256  
   defined, 70, 78  
   for fonts, 786  
   host names, 473  
   identifiers versus names, 78  
   for interfaces, 302–303  
   for methods, 207  
   other terms for, 70  
   for packages, 347  
   renaming in Eclipse, 66  
   rules for, 78  
   for source files, 257–258  
 if statements  
   basic form for, 150  
   blocks for, 151  
   boolean expressions with, 147  
   combining logical operators in, 163–164  
   comparing strings, 165–166  
   conditional operator eliminating, 164–165  
   else-if, 154–155, 156–158, 193–194  
   flowcharts for, 150, 153, 158  
   if-else, 152–153  
   indenting, 151  
   inner versus outer, 154  
   instanceof operator in, 292

- local variables in, 91
- logical operators for, 158–164
- nesting, 153–158
- in partition method, 521
- return statements in, 212–214
- rule for pairing `else` and `if` keywords, 155–156
- simple, 150–152
- in switch statements, 199–200
- syntax, 77
- `if-else` statements
  - basic form for, 152
  - blocks for, 152–153
  - `else-if` statements with, 154–155, 156–158
  - flowchart for, 153
  - nesting, 153–158
  - overview, 152–153
  - rule for pairing `else` and `if` keywords, 155–156
- `IllegalArgumentException`
  - `Exception`, 224
- `Image` class, 825–831
- `ImageFilter` class, 831
- `ImageIcon` class, 822–825
- images
  - in applets, 821
  - defined, 822
  - formats supported, 822
  - `Image` class, 825–831
  - `ImageIcon` class, 822–825
- immutable objects, 362, 365
- Immutable pattern, 362
- implicit objects (JSP), 662
- `import` page directive (JSP), 661
- `import` statements
  - for `ArrayList` class, 409
  - for exception classes, 229
  - for `File` class, 691
  - for `Iterator` interface, 412
  - for `JOptionPane` class, 86, 116
  - for packages, 346–347
  - rules for, 86
  - for `Scanner` class, 113
  - for servlets, 645
  - for Swing packages, 529
  - uses for, 86
- increment operator, 120, 124, 126–127
- indenting code, 41, 76, 151, 154
- indexes
  - for array lists, 410
  - for arrays, 377, 378, 390
  - for linked lists, 424
- `IndexOutOfBoundsException`
  - `Exception`, 410, 424
- `InetAddress` class
  - Facade pattern of, 476
  - methods (table), 475
  - overview, 474–475
  - program using, 475–478
- infinite loops, 170–173, 188
- Infinity output, 145
- infix operators, 121
- inheritance
  - by anonymous classes, 342
  - business example, 281–282
  - in class diagrams, 254
  - constructors with, 287–288
  - defined, 280
  - Delegation pattern versus, 283
  - determining type of objects, 291–292
  - downcasting, 290–291
  - game example, 281
  - hierarchies, 282
  - by inner classes, 336
  - interface implementation with, 304
  - interfaces versus, 250, 302
- is-a-type-of relationships
  - best for, 280
- non-computer definitions, 279
- from `Object` class, 311–312
- overview, 250
- polymorphism, 292–294
- type checking with, 16–17
- upcasting, 290
- vehicle example, 280–281
- visibility for inherited members, 284
- initializers
  - for arrays, 379–380
  - general form, 93
  - initializing variables with, 93, 269–270
- instance, 256
- order of execution for, 270
- static, 248, 256, 270, 278
- initializing
  - arrays, 379–380
  - for loop counter variable, 180
  - static fields, 273
  - strings, 102–103
  - two-dimensional arrays, 388
  - variables, 92–93, 269–270
- inlining methods in Eclipse, 66
- inner classes
  - anonymous, 341–344
  - basic structure, 335
  - declaring, 335
  - defined, 257, 258
  - for event listeners, 548–550
  - inheritance by, 336
  - references, 336
  - static, 336, 339–340
  - `TickTockAnonymous` program, 342–344
  - `TickTockInner` program, 336–339
  - `TickTockStatic` program, 339–340
  - visibility of, 335

- input, user. *See* user input
- InputMismatch
  - Exception
  - catching, 228–229, 231–232
  - defined, 224
  - preventing, 229–231
  - thrown by `nextInt` method, 225–226
- InputServlet servlet, 652–653
- InputServlet.html file, 651–652
- insert method, 566, 567
- insert statement (SQL), 733, 740
- installing
  - JDK, 24–25
  - TextPad, 39
  - Tomcat, 641–642
- instance initializers, 256
- instance variables
  - declaring, 90
  - defined, 90
  - initialized to empty strings, 103
  - in JavaBeans, 673
  - `this` keyword for, 268–269
- instanceof operator, 292
- instances, defined, 245
- instantiation
  - by calling constructors, 264–265
  - not allowed for abstract classes, 300
  - not allowed for `Math` class, 277
  - preventing, 277
- int type. *See also* integer types
  - arithmetic operators with, 120, 121
  - described, 95
  - integer overflow issues, 142–143
  - other integer types versus, 96
  - in random number generation, 135–137
- integer types
  - automatic conversions for, 106–107
  - boxing and unboxing, 107
  - casting, 106–107
  - converting strings to, 105
  - converting to strings, 104
  - dividing by zero, 144, 227, 233–234
  - dividing integers, 122–123
  - formatting numbers, 139–142
  - overflow issues, 142–143
  - overview, 96–97
  - promoting, 95
  - table summarizing, 95
  - wrapper classes for, 100
- Integrated Development Environment (IDE), 23
- interfaces. *See also specific interfaces*
  - adding fields, 305
  - for callbacks, 307–310
  - in class diagrams, 253–254
  - in classes, 257
  - creating, 302–303
  - defined, 248, 250
  - for DOM, 774–775
  - for event handling, 543–544
  - for events, 542
  - extending, 305–307
  - implementing, 303–304
  - inheritance versus, 250, 302
  - Marker Interface pattern, 306
  - methods public and abstract in, 303
  - naming, 302–303
  - using as types, 304–305
- Internet resources
  - bonus chapters for this book, 5
  - code listings for this book, 9, 69, 443
  - command-line tools, 23, 35
  - Eclipse download, 47
  - Java API documentation, 12, 36–37
  - Java documentation, complete, 35
  - Java Language Specification, 37
  - Java plug-in downloads, 632
  - Java site, 23
  - javadoc command
    - documentation, 357
  - JDK download, 24
  - MySQL connector driver, 749
  - MySQL download, 733
  - regular expressions
    - information, 496
  - site for this book, 5
  - TextPad download, 39
  - Tomcat, 640
- interpreted languages, 11
- interrupt method, 446, 463, 464
- interrupted static method, 464
- InterruptedException, 447, 464–465
- interrupting threads
  - checking for
    - interruptions, 464–465
  - countdown abort
    - example, 465–469
  - interrupt method for, 463
- IntValidationResult class, 564

- I/O
- binary streams for, 706, 718–729
  - character streams for, 706–718
  - standard I/O streams, 110–111, 705
- IOException, 224, 231–232, 297–298, 720
- IP addresses, 472, 473, 474–475
- is-a-type-of relationships, 280
- isDirectory method, 692, 694, 695
- isFile method, 692, 694
- isSelected method, 570
- ItemEvent class, 542
- ItemListener interface, 543
- itemStateChanged method, 543
- Iterator interface, 412–413
- iterator method
- ArrayList class, 408, 412, 413
  - LinkedList class, 420, 425
- Iterator pattern, 412
- iterators for array lists, 412–413
- 
- J
- 
- jagged arrays, 388–390
- JApplet class, 632–633
- jar command
- for archiving packages, 351–352
  - basic format, 350
  - command-line options, 350–351
  - described, 35
  - running programs from archives, 353–354
- JAR files
- adding to ClassPath, 352–353
  - archiving packages in, 351–352
  - executable, 353
  - jar command-line options for, 350–351
  - for Java API, 350
  - for MySQL connector, 749
  - overview, 350
  - running programs from archives, 353–354
- Java API
- abstract classes in, 301
  - documentation, 12, 36–37
  - JAR file for, 350
  - overview, 12
  - routines for sorting, 517
  - weaknesses of, 18
- java command
- error messages, 32–33
  - options, 33
  - running program using, 31–32, 72
- Java editions, 21
- Java EE (Java Enterprise Edition), 21
- .java filename extension, 40
- Java For Dummies* (Wiley publication), 9
- Java Language Specification, 37
- Java Runtime Environment (JRE), 10–11, 18, 21, 31
- Java SE (Java Standard Edition), 21
- Java Server Pages. *See* JSP
- Java 2 Standard Edition (J2SE), 21
- Java 2D, 799. *See also* shapes
- Java versions, 19–20, 30
- Java Virtual Machine. *See* JVM
- JavaBeans
- Accessor pattern for, 674
  - creating instances, 677–678
  - defined, 673
  - example, 674–676
  - getting property values, 678–679
  - in JSP pages, 677–678
  - properties, 673–674
  - rules for, 673–674
  - scope of, 682
  - setting property values, 679–680
  - shopping cart application, 682–688
- javac command, 28–31. *See also* compiling
- avadoc command, 35, 357–358
- JavaDoc comments
- adding to source files, 354–357
  - defined, 80
  - doc tags, 354–355
  - example, 355–357
- JavaFilter class, 703–704
- javap command, 33–35
- JButton class, 538–539, 787–788, 823
- JCheckBox class, 569–570
- JColorChooser class, 795–796
- JComboBox class, 586
- JComponent class, 529
- JDBC (Java DataBase Connectivity)
- connecting to database, 750–751
  - described, 731, 747
  - driver setup for, 747–749
- Movie Listing program, 755–758
- MySQL connector for, 749
- ODBC data source for, 747–749

JDBC (*continued*)  
 queries using, 751–758  
 select statement with, 752–753  
 updating SQL data with, 758–762

JDK (Java Development Kit)  
 defined, 20  
 downloading, 24  
 files in root folder, 26  
 folders in root folder, 25–26  
 installing, 24–25  
 license agreement, 25, 26  
 path setting for, 26–28

JFileChooser class, 699–701, 828

JFrame class, 528, 529, 530–531. *See also* frames

JLabel class, 529, 537, 787, 795, 823

JList class, 591

JOptionPane class  
 getting input with, 116–117  
 in Good Morning program, 559, 561  
 Greeter class using, 85  
 importing, 86, 116  
 in TickTock program, 308, 310  
 in TickTockInner program, 338

JPanel class, 529, 535–536, 787

JPEG (Joint Photographic Experts Group) format, 822

JRadioButton class, 572–573

JRE (Java Runtime Environment), 10–11, 18, 21, 31

JScrollPane class, 565, 568–569

JSlider class, 581–582

JSP (Java Server Pages)  
 classes with, 668–672  
 declarations, 660, 666–667  
 described, 659  
 directives, 660  
 expressions, 660, 662–663  
 HTML for, 663, 664–665, 666–667  
 implicit objects, 662  
 JavaBeans in, 677–678  
 overview, 659–660  
 page directives, 661  
 scriptlets, 660, 664–666

jsp:getProperty tag, 677, 678–679, 683, 685

JSpinner class, 595

jsp:setProperty tags, 677, 678, 679–680, 681

jsp:useBean tag, 677–678, 681, 682, 684

JTextArea class, 565–567

JTextField class, 558

JTree class, 597, 601–602

J2SE (Java 2 Standard Edition), 21

JVM (Java Virtual Machine)  
 defined, 21  
 exit method shutting down, 338  
 garbage collector, 17  
 loaded by java command, 72  
 platform independence with, 10–11  
 as stack oriented, 11

## K

KeyEvent class, 542

KeyListener interface, 544

keywords. *See also specific keywords*  
 case sensitivity of, 73–74  
 defined, 73  
 table listing, 73  
 in TextPad, 41

KnightMoves program, 391–398

## L

labels, 537–538, 787–788, 795

late binding, 294

LaunchEvent class, 449–450

LaunchEventNew class, 457

launching. *See* opening; running programs

layout managers  
 Border, 608, 611–612  
 Box, 608, 612–614  
 Card, 608  
 challenges with, 607  
 combining, 608  
 default, 609  
 Flow, 607, 609–610  
 Grid, 608, 614–615  
 GridBag, 608, 616–624  
 Group, 608, 624–627  
 illustrated, 609  
 setting for panel or frame, 608  
 Spring, 608  
 tips for layout, 540

length method, 364, 365

length of string, finding, 365

less-than sign (<) in relational operators, 148–149

license agreement (JDK), 25, 26

life cycle of objects, 248

- lines, creating, 801, 804–805
- linked lists
- adding items, 422–424
  - arrays versus, 417, 418–419
  - changing objects
    - retrieved from, 426
  - creating, 422
  - declaring variables for, 422
  - defined, 417
  - for loops with, 424–425
  - inserting items in, 418
  - pointers in, 417–418
  - queues, 418, 437–440
  - removing items, 418, 426–427
  - retrieving items, 424–426
  - setting type for items, 422, 424
  - stacks, 418, 432–436
  - updating items, 426
- LinkedList class, 418–422
- listener object, 307
- listFiles method, 692, 695
- ListMovies servlet class, 654, 657–658
- ListMovies.jsp page, 669–670
- ListMoviesXML program, 755–758
- lists, 590–595
- listSelectionModel class, 592
- literals. *See also* string literals
- character, 98–99
  - as reserved words, 73
- loading classes, 248
- local variables
- declaring, 90–91
  - defined, 90
  - extracting in Eclipse, 66
  - initializing to empty string, 103
  - scope of, 107–109
  - shadowing by, 109–110
  - static, not allowed, 273
- lock object, 463
- locks for threads, 462–463
- log method, 132, 133, 134–135
- logic layer, 251
- logical font names, 786
- logical operators
- And, 159, 160–161
  - combining, 163–164
  - compound expressions
    - using, 158–159
  - defined, 158
  - Not (!), 159–160
  - Or, 159, 161–162, 177
  - overview, 158–159
  - table summarizing, 159
  - Xor, 159, 162–163
- log10 method, 133, 134–135
- long type. *See also* integer types
- described, 95
  - for factorials, 511, 512
  - integer overflow issues, 142–143
  - other integer types
    - versus, 96
- loops. *See also* for loops
- break statement for, 170–172, 176, 188
  - continue statement for, 173–174, 176, 188–189
  - do-while statement for, 175–179, 190–192
  - enhanced for loops, 383–384
  - guessing game example, 190–192
  - infinite, 170–173
  - nesting, 189–192
  - overview, 167
  - while statement for, 168–174, 190–192
- lowercase. *See* capitalization or case; case sensitivity
- low-level events, 543
- 
- ## M
- 
- main method
- HelloApp.java
    - program, 55, 59
    - static declaration
      - required for, 71, 81, 273
      - throws clause in, 238
- MarblesApp program, 122–123
- Marker Interface pattern, 306
- MarkVowels program, 368–369
- Matcher object, 509–510
- Math class
- constants, 131–132
  - creating random numbers
    - with, 135–137
  - instantiation not allowed
    - for, 277
  - mathematical methods, 132–135
  - overview, 130–131
  - rounding methods, 138–139
- MathFunctionsApp program, 134–135
- max method, 133, 134–135
- MediaTracker class, 821
- members of classes
- constructors not
    - considered as, 264
  - overview, 259–260
  - protecting, 285–286
- memory
- for linked lists versus
    - arrays, 419
  - management in Java, 17
  - object locations in, 245
- messages. *See* error messages

## methods by class

Arrays, 398–399  
 BorderFactory, 575  
 Box, 612–613  
 BufferedReader, 707  
 Connection, 752, 760  
 DataInputStream, 719  
 DataOutputStream, 725  
 DefaultListModel, 594  
 DefaultMutableTree  
   Node, 599  
 described, 542  
 Exception, 232  
 FileReader, 707  
 GenQueue, 437  
 GenStack, 433  
 Graphics, 826  
 HttpServlet, 645  
 Image, 826  
 ImageIcon, 823  
 InetAddress, 475  
 JApplet, 633  
 JavaFilter, 703  
 JButton, 538–539  
 JCheckBox, 570  
 JComboBox, 586  
 JFileChooser, 700  
 JFrame, 530–531  
 JLabel, 537  
 JList, 591  
 JPanel, 535  
 JRadioButton, 573  
 JSlider, 582  
 JSpinner, 596  
 JTextArea, 566–567  
 JTextField, 558  
 JTree, 602  
 LinkedList, 419–422  
 Math, 132–135, 138–139  
 NumberFormat, 140–142  
 Object, 313–314  
 PrintWriter, 713  
 ResultSet, 753, 754–755  
 Scanner, 114, 115  
 ServerSocket, 480–481  
 Socket, 479  
 String, 363–365

StringBuilder, 373–374  
 Thread, 446  
 Toolkit, 534, 826  
 methods by interface  
   AudioClip, 831  
   CharSequence, 376  
   Element, 775  
   event listeners, 543–544  
   Iterator, 412, 413  
   Node, 774–775  
   Statement, 752  
 methods in general. *See*  
   *also* static methods  
   abstract, 299, 303  
   accessors, 260–262  
   arrays with, 384–385  
   body of, 207  
   calling, defined, 205  
   constructors versus, 264  
   declaring, 71, 81, 90–91,  
     206–210  
   defined, 205  
   deprecated, 50  
   extracting in Eclipse, 66  
   final keyword for, 289  
   inlining in Eclipse, 66  
   in interfaces, 303  
   JavaBean requirements,  
     673–674  
   late binding, 294  
   for mouse events and  
     listeners, 814–815  
   naming, 207  
   object behavior provided  
     by, 248  
   order in source file, 208  
   overloading, 263  
   overriding, 284–285,  
     316–317, 319–321,  
     645–646  
   parameter lists, 207,  
     217–222  
   private, 260  
   public, 71, 206–207, 259  
   return type for, 207,  
     210–216

signatures, 263  
 synchronized, 458–462  
 this keyword with, 269  
 uses for, 205–206  
 Microsoft Access, ODBC  
   setup for, 747–749  
 min method, 133, 134–135  
 minus sign (-)  
   in compound assignment  
     operator, 129–130  
   as decrement operator,  
     120, 124, 126–127  
   as subtraction operator,  
     120, 124  
   as UML visibility indicator,  
     253  
   as unary operator, 125  
 modulus (remainder)  
   operator, 120, 121–123,  
     124  
 mouse events and listeners,  
   814–815  
 MouseClicker program,  
   832  
 MouseEvent class, 542  
 MouseListener interface,  
   544  
 Movie class  
   JSP example, 670  
   servlet example, 654, 655  
 Movie Listing program  
   JDBC version, 755–758  
   JSP version, 669–670  
   servlet version, 654,  
     657–658  
   XML version, 778–781  
 MovieIO class  
   JSP example, 671–672  
   servlet example, 654,  
     656–657  
 movies.dat file  
   described, 719  
   reading, 721–724  
   writing, 727–729

movies.txt file  
   BufferedReader object  
     for, 708  
   contents, 707–708  
   reading, 709–712  
   writing, 715–718  
 movies.xml file  
   contents, 766  
   determining oldest movie,  
     776–777  
   DTD file for, 767, 769  
   Movie Listing program,  
     778–781  
   processing elements in,  
     775–776  
   reading, 773  
   validating, 772  
 multi-dimensional arrays,  
   390–391. *See also* two-  
   dimensional arrays  
 multiplication operator,  
   120, 124  
 music and sounds, 821,  
   831–832  
 MySQL, 733–735, 749. *See*  
   also SQL (Structured  
   Query Language)

## N

Namer frame class, 559–561  
 names. *See* identifiers and  
   names  
 NaN constant, 145–146  
 NEGATIVE\_INFINITY  
   constant, 145  
 nested if statements  
   braces for clarifying, 156  
   defined, 153  
   else-if statements for,  
     154–155, 156–158  
   general form, 153–154  
   if-else statements for,  
     153–156  
   indenting, 154  
   inner versus outer  
     statements, 154  
   rule for pairing else and  
     if keywords, 155–156  
 NestedLoop program, 189  
 nesting. *See also* inner  
   classes  
   arrays, 385–391  
   comments (not allowed),  
     80  
   defined, 153  
   if statements, 153–158  
   loops, 189–192  
 network programming  
   BartServer example,  
     481–490  
   BartServer 2.0 example,  
     490–493  
   client programs, 471–472  
   creating server appli-  
     cations, 478–481  
   DNS, 473  
   getting information about  
     Internet hosts, 474–478  
   host names, 473  
   IP addresses, 472  
   ports, 472–473  
   server programs, 471–472  
   ServerSocket class for,  
     480–481  
   Socket class for, 479–480  
   telnet, 473–474  
   URLs, 473  
 newAudioClip method,  
   831, 832  
 next method, 412, 413  
 nextInt method, 224–226,  
   228–231  
 Node interface, 774–775  
 NodeList interface, 774  
 nodes of trees, 598  
 Not operator, 159–160  
 notify method, 313, 462  
 notifyAll method, 313,  
   462  
 NotPong applet, 845–850  
 nouns for class names, 256  
 NumberFormat class,  
   140–142, 512  
 NumberPhobia program,  
   172  
 NumberPhobia2 program,  
   173  
 numbers. *See also* floating-  
   point types; integer  
   types  
   arithmetic operators for,  
     119–127  
   assignment operators for,  
     128–130  
   changing sign of, 125  
   character class for (`\d`),  
     499, 500  
   converting strings to,  
     104–105, 562  
   dividing by zero, 144–145,  
     227, 233–234  
   dividing integers, 122–123  
   floating-point issues,  
     143–144  
   formatting, 139–142  
   incrementing and  
     decrementing, 126–127  
   integer overflow issues,  
     142–143  
   Math class for, 130–139  
   random, generating,  
     135–137  
   regular expression  
     quantifiers, 504  
   rounding, 107, 138–139  
   special constants for,  
     145–146  
   text fields for entering,  
     562–563  
   validating user input,  
     563–565

## O

- 
- Object class
    - methods, 313–314, 463
    - as mother of all classes, 311–312
    - not inherited by
      - primitives, 314–315
    - in Swing hierarchy, 528, 529
    - using as type, 312–313
  - object-oriented
    - programming
      - based on real-world simulation, 243–244
    - C language versus Java, 14–15
    - defined, 243
    - designing with objects, 250–251
    - HelloApp program
      - example, 82–85
    - overview, 11–12, 80–85
    - three-layered approach, 250–251
  - objects
    - arrays as, 378
    - behavior, 12, 80, 247–248
    - as class instances, 81
    - classes versus, 81
    - creating by calling
      - constructors, 264–265
    - creating from classes, 82, 248
    - defined, 11, 80
    - designing with, 250–251
    - hash code for, 245
    - identity, 245
    - immutable, 362, 365
    - implicit (JSP), 662
    - instances, defined, 245
    - layers or tiers of, 250–251
    - life cycle, 248
    - memory locations distinct for, 245
    - program example using, 82–85
    - referenced by multiple variables, 101–102
    - simulation of real-world, 243–244
    - state of, 12, 80, 246–247
    - types, 246, 291–292
  - ODBC, 747–749
  - offer method, 420, 424
  - offerFirst method, 420, 424
  - offerLast method, 420, 424
  - Olivetti Programma 101 computer, 11
  - Open dialog box, 700–702
  - openFile method, 236–239
  - opening. *See also* running programs
    - last used workspace in TextPad, 42
    - projects in Eclipse, 51
  - operands, 119, 121
  - operators
    - arithmetic, 119–127
    - assignment, 82, 128–129
    - binary, 121, 148
    - cast, 106
    - compound assignment, 129–130
    - concatenation, 103
    - conditional, 121, 164–165
    - defined, 119
    - infix, 121
    - instanceof, 292
    - logical, 158–164
    - postfix versus prefix, 121, 126
    - precedence of, 124–125
    - relational, 148–149, 165–166
    - as separators, 122
    - ternary, 121, 164
    - unary, 121, 125, 159
    - white space optional with, 122
  - Or operator, 159, 161–162
  - out method (System), 14, 715
  - outer classes, 335
  - overflow issues for integers, 142–143
  - overloading
    - constructors, 265
    - methods, 263
  - overriding methods
    - equals method, 319–321
    - of HttpServlet class, 645–646
    - overview, 284–285
    - toString method, 316–317
- 
- ## P
- 
- package statement, 348
  - packages. *See also* import statements
    - archiving in JAR files, 351–352
    - creating, 347–348
    - default, 346–347
    - defined, 345
    - directories for, 347–348
    - for event handling, 542
    - example, 348–350
    - importance of, 345
    - importing, 346–347
    - package statement for
      - source files, 348
      - for servlets, 645
      - for Swing features, 529
    - page directives (JSP), 661
    - paint method, 799–800, 801, 834
    - paintComponent method, 825
    - PaintSurface class
      - for bouncing ball, 838–839
      - for random balls, 841

- panels, 535–538, 787, 814–820. *See also* layout managers
- parameter lists
  - in `Hello, World!` program, 71
  - for methods, 207, 217–222
- parameters for methods
  - basic form for, 207
  - declaring, 217–218
  - defined, 217
  - `GuessingGameMethod3` program, 220–222
  - names not part of signature, 263
  - pass-by-value for, 219
  - scope of, 218–219
- `ParameterScope` program, 218–219
- parent class or superclass, 249, 280, 292–294
- parentheses `[()]` in regular expressions, 505–507
- parse methods, 104–105, 562
- partition method, 520–521
- Path environment variable
  - for JDK, 26–28
- Pattern class, 509–510
- patterns. *See* design patterns
- `PatternSyntaxException`, 509
- peek method
  - `GenStack` class, 433, 434
  - `LinkedList` class, 421, 425
- `peekFirst` method, 421, 425
- `peekLast` method, 421, 425
- percent sign `(%)`
  - in compound assignment operator, 129–130
  - as remainder operator, 120, 121–123, 124
  - as SQL wildcard, 738–739
- perspectives (Eclipse), 49–50
- PI constant, 131–132
- `PictureFrame` program, 828–831
- pipe symbol `(|)`
  - as Or operators, 159, 161–162, 177
  - in regular expressions, 506–507
- `Pizza` program, 577–581, 621–624
- `PizzaApplet` program, 633–636
- platform independence, 10–11
- plus sign `(+)`
  - as addition operator, 120, 124
  - in compound assignment operator, 129–130
  - as concatenation operator, 103, 362
  - as increment operator, 120, 124, 126–127
  - as UML visibility indicator, 253
  - as unary operator, 125
- PNG (Portable Network Graphics), 822
- pointers, 95, 417–418
- `poll` method, 421, 425, 426
- `pollFirst` method, 421, 425
- `pollLast` method, 421, 426
- polymorphism, 292–294
- pong game, 844–850
- `pop` method
  - `GenStack` class, 433, 434
  - `LinkedList` class, 421, 425
- ports
  - for `BartServer` program, 481
  - numbers for `telnet` command, 474
  - overview, 472–473
  - for Tomcat, 643
- positioning frames
  - on-screen, 534
- `POSITIVE_INFINITY` constant, 145
- postfix operators, 121, 126
- `pow` method, 133, 134–135
- precedence of operators, 124–125
- prefix operators, 121, 126, 159
- presentation layer, 250
- primitive types
  - automatic conversions
    - for, 105–106
  - `boolean`, 95, 99–100
  - boxing and unboxing, 107
  - casting, 106–107
  - `char`, 95, 98–99
  - converting strings to, 104–105, 562
  - converting to strings, 104, 362–363
  - defined, 95
  - floating-point, 95, 97–98
  - integer, 95, 96–97
  - `Object` class not inherited by, 314–315
  - reference types versus, 95
  - table summarizing, 95
  - wrapper classes for, 100
- `print` method, 111, 112, 715
- printing to console
  - array contents, 380, 385
  - array list contents, 411–412
  - `println` method for, 14, 72, 103
  - standard I/O streams for, 110–111, 705
  - `System.out` and `System.err` for, 111–112
- printing to Web page, 646

println method  
 concatenation with, 103, 112  
 in Hello, World! program, 14, 72  
 overloading of, 263  
 overview, 111  
 print method versus, 112  
 writing to character stream, 715

printStackTrace method, 232

PrintStream object, 14

PrintWriter class, 646, 712–714

private keyword  
 for fields, 259  
 for inner classes, 335  
 for methods, 260

ProductDataException class, 296–298

ProductDDB program, 297–298

.project file (Eclipse), 51

promoting integer types, 95

properties (JavaBean)  
 defined, 673–674  
 getting values, 678–679  
 methods required for, 673–674  
 setting values, 679–680

protected visibility, 284, 285–286, 323, 335

protocols, 478, 481

pseudorandom numbers, 135

public keyword  
 for classes, 70, 255, 257–258  
 for constructors, 264  
 for fields, 247, 259  
 in Hello, World! program, 70, 71  
 for inner classes, 335  
 for methods, 71, 206–207, 259

push method, 433, 434

Pythagorean Theorem, 132, 134, 674, 676

## Q

queries (SQL)  
 basic select statement for, 736–737  
 column functions for, 739–740  
 defined, 736  
 eliminating duplicates, 741–742  
 excluding rows, 738  
 with JDBC, 751–758  
 from multiple tables, 740–741  
 narrowing, 737  
 singleton select, 738  
 wildcards for, 738–739

queues, 418, 437–440. *See also* linked lists

Quicksort technique  
 basic steps, 518  
 overview, 517–519  
 partition method, 520–521  
 partitioning, 518–519  
 pivot point, 518  
 QuickSortApp program, 521–524  
 sort method, 519

quotation marks (") with string literals, 102

## R

radio buttons, 572–574

random method, 133, 134–137

random number generation, 135–137

reading  
 binary streams, 718–724  
 character streams, 706–712

DOM documents, 771–773

DOM nodes, 774–778

readLine method, 707, 708

rectangles, creating, 805–806

recursion  
 defined, 511  
 directory listing example, 514–517  
 end condition, 512  
 factorial example, 511–513  
 Quicksort example, 517–524

redirecting standard output, 111

ReentrantLock class, 463

refactoring code, 65–66

reference types, 95, 100–102, 361

Reg program, 496–498

regular expressions  
 backslash (\) issues in Java, 507–508  
 captured groups, 506  
 custom character classes for, 501–503  
 defined, 495  
 escape sequences for, 505  
 further information, 496  
 grouping characters with parentheses, 505–507  
 in Java programs, 507–510  
 Matcher object for, 509–510  
 matching multiple characters, 503–504  
 matching single characters, 499

negation in, 503

overview, 495–496, 498–499

Pattern class for, 509–510

pipe symbol (|) in, 506–507

predefined character classes for, 499–501

quantifiers, 503–504

- ranges in, 502–503
  - regex term for, 496
  - with `split` method, 370
  - `String` class with, 508–509
  - test program for, 496–498
  - uses for, 495
  - relational databases.
    - See also* JDBC (Java DataBase Connectivity); SQL (Structured Query Language)
    - creating, 733–735
    - deleting rows, 742–743, 761
    - inserting rows, 762
    - JDBC connection, 750–751
    - overview, 731–732
    - querying, 736–742
    - updating rows, 744–745, 761–762
  - relational operators
    - for comparing strings, avoiding, 149, 165–166
    - object identity tested by, 245
    - overview, 148–149
  - remainder operator, 120, 121–123, 124
  - Remember icon, 6
  - remove method
    - `ArrayList` class, 408, 415–416
    - `Iterator` interface, 412, 413
    - `LinkedList` class, 421, 425, 426, 427
  - `removeAll` method, 408, 416
  - `removeAllItems` method, 586, 588
  - `removeFirst` method, 421, 425, 426
  - `removeItem` method, 586, 588
  - `removeItemAt` method, 586, 588
  - `removeLast` method, 421, 426
  - `removeRange` method, 408, 416
  - removing. *See* deleting or removing
  - `renameTo` method, 693, 696
  - `repaint` method, 799, 815
  - `replace` method, 567
  - `replaceAll` method, 364, 371
  - `replaceFirst` method, 364, 371
  - replacing
    - parts of strings, 371
    - string vowels with asterisks, 368–369
  - `ResultSet` interface
    - described, 751
    - get methods, 754–755
    - navigation methods, 753
    - update methods, 761–762
  - `retainAll` method, 408, 416
  - return statements
    - in `if` statements, 212–214
    - overview, 211–212
  - return type for methods
    - arrays as return values, 384–385
    - declaring, 210–211
    - not part of signature, 263
    - overview, 207
    - return statements for, 211–214
    - using, 212
  - rigid areas, 614
  - `rint` method, 138–139
  - rotating shapes, 812–813
  - `round` method, 138–139
  - rounding
    - during casting, 107
    - `Math` class methods for, 138–139
  - `RoundingApp` program, 138–139
  - `rt.java` JAR file, 350
  - run method
    - `CountDownLatch` class, 447
    - `LaunchEvent` class, 449, 450
    - Runnable object, 448
    - `Thread` class, 446
  - Runnable interface
    - described, 448
    - implementing, 449–450
    - using, 448–449
  - running programs
    - applets with `TextPad`, 45
    - from archives, 353–354
    - with Eclipse, 55–56, 59–60
    - with `java` command, 31–33, 72
    - servlets, 649
    - with `TextPad`, 44–45
    - Tomcat, 643
  - Runtime Environment, 10–11, 18, 21, 31
  - `RuntimeException` class, 296
- 
- ## S
- 
- Save dialog box, displaying, 701
  - SAX (Simple API for XML), 770
  - `Scanner` class
    - alternative for older Java versions, 112, 116–117
    - declaring and instantiating, 113–114
    - getting input with, 112–113, 114–115
    - importing, 113
    - marble divvying program with, 122–123
    - methods that check input validity, 115
    - methods that get input values, 114
    - program example using, 113

- ScannerApp program, 113
- ScheduledThreadPool
  - Executor class, 458
- scientific notation, 97, 98
- scope of JavaBeans, 682
- scope of variables
  - blocks as boundaries of, 107–108
  - counter variable in for loops, 182–183
  - defined, 107
  - parameters for methods, 218–219
  - program demonstrating, 108–109
  - shadowed variables, 109–110
  - in try and catch blocks, 227
- ScopeApp class, 108–109
- scriptlets (JSP), 660, 664–666
- scroll bars, 565, 568–569
- SDK (Software Development Kit), 21
- searching
  - arrays, 401–402
  - in TextPad, 42
- select statement (SQL)
  - basic, 736–737
  - column functions, 739–740
  - described, 733
  - eliminating duplicates, 741–742
  - with JDBC, 752–753
  - for multiple tables, 740–741
  - verifying changes, 743, 744, 745
  - where clause, 737–739
- semantic events, 543
- semicolon (;) ending statements, 14, 75–76, 733
- separators, operators as, 122
- server applications
  - BartServer example, 483–486
  - creating, 478–481
  - defined, 471
  - examples, 472
  - ServerSocket class for, 480–481
  - Socket class for, 479–480
  - ServerSocket class, 480–481
- servlets. *See also* Tomcat server
  - defined, 13
  - extending HttpServlet class, 645–646
  - forms with, 651–653
  - getting user input, 651–653
  - HelloServlet example, 649–651
  - HelloWorld example, 646–648
  - HTML for, 646–648, 651–652
  - packages for, 645
  - printing to Web page, 646
  - running, 649
  - using classes in, 654–658
  - Web servers with, 639–640
- set accessors, 260–262
- set method
  - ArrayList class, 408, 413–414
  - LinkedList class, 422, 426
- setAcceptAllFile
  - FilterUsed method, 700, 704
- setBorderPainted
  - method, 538, 539
- setColor method, 801
- setContentAreaFilled
  - method, 539
- setDefaultClose
  - Operation method, 530, 533, 552, 556
- setDialogTitle method, 700, 701
- setEditable method, 586, 587
- setFileHidingEnabled
  - method, 700, 701
- setFileSelectionMode
  - method, 700, 701
- setFont method, 787–788
- setLayout method, 535, 540
- setLineWrap method, 567
- setLocation method, 531, 534
- setLocationRelative
  - To method, 531, 534
- setMajorTickSpacing
  - method, 582, 583
- setMaximumFraction
  - Digits method, 140, 141
- setMinimumFraction
  - Digits method, 140, 141
- setMinorTickSpacing
  - method, 582, 583
- setMultiSelection
  - Enabled method, 700, 701
- setPaintLabels method, 582, 583
- setPaintTicks method, 583
- setResizable method, 531, 540
- setSelected method, 570
- setSelectionMode
  - method, 592, 602
- setSize method, 531, 533
- setStroke method, 801
- setters, 260–262
- setText method
  - JButton class, 539
  - JLabel class, 537
- setTitle method, 533
- setVisible method
  - JButton class, 539
  - JFrame class, 531, 533
- setVisibleRowCount
  - method, 602

- setWrapStyleWord method, 567
- ShadowApp class, 109–110
- shadowed variables, 109–110
- shallow copy of object, creating, 326–327
- ShapeMaker program, 807–809
- shapes
  - basic, classes for, 803–804
  - creating arcs, 806–807
  - creating ellipses, 806
  - creating lines, 804–805
  - creating rectangles, 805–806
  - drawing, 800–803, 809
  - filling, 801, 809–812
  - graphics context for, 799–800
  - letting user draw on component, 814–820
  - paint method for, 799–800, 801
  - rotating, 812–813
  - ShapeMaker program for, 807–809
  - SimpleShape program, 801–802
  - translating, 812
  - transparent, 809
- shopping cart JavaBean program, 682–688
- short type, 95, 96, 142–143. *See also* integer types
- showOpenDialog method, 700–702
- showSaveDialog method, 700, 701
- shutting down Tomcat, 644
- side effects of expressions, 161
- sign operators, 124, 125
- signatures of methods, 263
- signum method, 133, 134–135
- Simple API for XML (SAX), 770
- SimpleShape program, 801–802
- simulation of real-world objects, 243–244
- singleton classes, 276
- size method
  - ArrayList class, 408, 410
  - GenQueue class, 437, 439
  - GenStack class, 433, 434
- slash (/)
  - for comments, 79–80
  - in compound assignment operator, 129–130
  - as division operator, 120–121, 122–123, 124
- sleep method
  - CountDownLatch class, 447
  - Thread class, 446, 449, 450, 464
- sliders, 581–584
- Socket class, 479–480
- sockets, defined, 479
- Software Development Kit (SDK), 21
- sort method
  - Arrays class, 399, 401
  - Quicksort technique, 519
- sorting, 401, 517–524
- sounds and music, 821, 831–832
- source files
  - adding JavaDoc comments, 354–357
  - for classes, 257–258
  - defined, 23
  - editing in TextPad, 40–42
  - identifiers for, 257–258
  - package statement for, 348
- spaces. *See* white space
- SpinnerListModel class, 596, 597
- SpinnerModel interface, 595, 596
- SpinnerNumberModel class, 596–597
- spinners, 595–597
- SpinOff program, 604–606
- splitting strings, 369–371, 708
- Spring layout, 608
- sprites, 833–834. *See also* animation
- SQL For Dummies* (Taylor), 731
- SQL (Structured Query Language)
  - common statements, 732–733
  - create statement, 733, 735
  - creating a database, 733–735
  - delete statement, 733, 752–753
  - described, 731, 732
  - drop statement, 733, 734–735
  - further information, 731
  - insert statement, 733, 740
  - not case sensitive, 733
  - pronunciation, 732
  - select statement, 733, 736–742
  - update statement, 733, 744–745
  - use statement, 733, 735
- SQLException, 750, 751
- sqrt method, 133, 134–135
- stack trace
  - finding exceptions with, 225–226
  - printing to standard error stream, 232
- stacks, 418, 432–436. *See also* linked lists

- standard error stream, 111–112, 232
- standard input stream, 110–111
- standard output stream, 14, 111–112
- start method
  - CountDownLatch class, 448
  - Thread class, 446
  - Timer class, 308, 309
- starting. *See* opening; running programs
- state of objects, 12, 80, 246–247
- Statement interface, 751, 752
- statements (Java), 14, 75–76, 77
- statements (SQL)
  - column functions, 739–740
  - common, 732–733
  - create, 733, 735
  - delete, 733, 752–753
  - drop, 733, 734–735
  - insert, 733, 740
  - select, 733, 736–742
  - update, 733, 744–745
  - use, 733, 735
  - where clause, 737–739
- static, defined, 271
- static fields, 248, 272–275
- static initializers, 248, 256, 270, 278
- static keyword. *See also* static methods
  - for class variables, 89
  - for fields, 248, 272–275
  - for inner classes, 336, 339–340
  - not allowed for instance variables, 90
  - not allowed for local variables, 91
  - required for main method, 71, 81
- static methods
  - accessing, 273–274
  - declaring, 206–210
  - defined, 273
  - described, 81
  - instance variables not allowed in, 90
  - main method as, 71, 81, 273
  - parameters with, 217–222
  - returning values, 211–217
  - static keyword for, 81, 207
  - uses for, 272
- stopping Tomcat, 644
- streams
  - binary, 706, 718–729
  - character, 706–718
  - defined, 705
  - standard I/O, 14, 110–111, 232
- String class
  - CharSequence interface for, 375–376
  - limitations of, 371–372
  - methods (table), 363–365
  - objects as immutable, 365
  - regular expressions with, 508–509
- string literals
  - assigning to variables, 361
  - concatenating with string variables, 103
  - escape sequences for, 362
  - quotation marks enclosing, 102
  - as regular expressions, 499
  - regular expressions in, 507–508
  - in TextPad, 41
- StringBuffer class, 372, 375. *See also* String Builder class
- StringBuilder class
  - CharSequence interface for, 375–376
  - creating an object, 372–373
  - example, 375
  - methods (table), 373–374
  - not safe with threads, 372
  - StringBuffer class as mirror of, 372
- StringBuilderApp program, 375
- StringIndexOutOfBoundsException, 366
- strings
  - assigning literals to variables, 361
  - comparing, 149, 165–166, 363
  - concatenating, 103, 165, 362, 372
  - converting arrays to, 402–403
  - converting case, 366
  - converting primitives to, 104, 362–363
  - converting to numbers, 562
  - converting to primitives, 104–105
  - counting vowels in, 367
  - declaring, 102
  - defined, 102
  - drawing text in, 814
  - extracting character from, 366–367
  - extracting substrings, 367–369
  - finding length of, 365
  - as immutable objects, 365
  - initializing, 102–103
  - as reference types, 102, 361
  - relational operators for, avoiding, 149, 165–166

removing white space, 366  
 replacing parts, 371  
 replacing vowels with asterisks, 368–369  
 splitting, 369–371, 708  
 strongly typed languages, 94  
 Structured Query Language. *See* SQL  
 struts, 613  
 styles for fonts, 786–787  
 subclasses or derived classes  
   of abstract classes, 301  
   basic form for, 282  
   creating, 282–284  
   defined, 249, 280  
   inheritance by, 249, 280, 283–284  
   other terms for, 249, 280  
   polymorphism, 292–294  
   super keyword in, 286–287  
   this keyword in, 286  
 subinterfaces, 305  
 substring method, 365, 367–369  
 subtraction operator, 120, 124, 146  
 super keyword, 286–287  
 superclass or base class, 249, 280, 292–294  
 super.clone() call for clone method, 327  
 superinterfaces, 305  
 Swing. *See also specific classes and components*  
   applets versus, 631–632  
   choosing files in applications, 698–704  
   class hierarchy, 528–529  
   components in applets, 632  
   described, 527

double buffering by, 837  
 Hello, World!  
   program, 531–533  
 ImageIcon objects with, 823–825  
 layout tips, 540  
 letting user draw on component, 814–820  
 packages for, 529  
 setting component color, 795  
 setting component font style, 787–788  
 window display in Eclipse, 60  
 switch statements  
   basic form for, 196–197  
   break statement with, 197  
   business example, 197–199  
   char variables with, 200–201  
   default group, 197  
   flowchart for, 198  
   if statements in, 199–200  
   omitting break statement, 201–203  
   for Voting Machine Error Decoder, 195–196  
 symbols. *See* identifiers and names  
 synchronized declaration for methods, 458–462  
 synchronized keyword, 458, 460  
 System class  
   exit method, 338, 550  
   out method, 14, 715  
   standard I/O streams available via, 111  
 SystemColor class, 793–794

---

## T

---

tab-delimited text files, 706.  
   *See also* movies.txt file  
 tags  
   HTML, 648  
   JavaDoc, 354–355  
   JSP, 677  
   XML, 764–765  
 Taylor, Allen G. (*SQL For Dummies*), 731  
 Technical Stuff icon, 5  
 telnet, 473–474, 481–482  
 ternary operators, 121, 164  
 TerseCoder program, 185  
 TestEquality1 program, 317–318  
 TestEquality2 program, 321–322  
 TestToString program, 315–317  
 text areas, 565–569  
 text fields, 557–565  
 text files  
   delimiters for, 706  
   reading character streams, 706–712  
   writing character streams, 712–718  
 Text interface, 774  
 TextPad text editor  
   compiling in, 42–43  
   configuring, 39–40, 42  
   described, 23, 39  
   downloading, 39  
   editing source files, 40–42  
   error messages in, 43  
   features for Java files, 41–42  
   installing, 39  
   running program using, 44–45  
   workspaces, 42

- this keyword, 267–269, 286
  - Thread class, 445–447
  - threads
    - concurrency issues, 458
    - creating and starting, 447–448
    - defined, 443
    - enabling communication between, 452–456
    - executor for, 456–458
    - interrupting, 463–469
    - locks for, 462–463
    - main thread, 444
    - overview, 443–444
    - Runnable interface for, 448–450
    - stepping through, in Eclipse, 61–63
    - StringBuilder class unsafe for, 372
    - synchronized method declaration, 458–462
    - Thread class for, 445–447
    - uses for, 444
  - Throwable class, 295
  - throwing exceptions. *See also* exceptions
    - custom exceptions, 297–298
    - defined, 224
    - FileNotFoundException
      - Exception, 237
    - from main method, 238
    - throw statement for, 239–240
  - throws clause, 237, 238
  - TickTock program, 308–310
  - TickTockAnonymous program, 342–344
  - TickTockInner program, 336–339
  - TickTockStatic program, 339–340
  - TimeMonitor interface, 455
  - Timer class, 307, 308, 309
  - Tip icon, 5
  - toLowerCase method, 365, 366
  - Tomcat server
    - described, 640
    - installing and configuring, 641–642
    - setup for classes, 668
    - starting, 643
    - testing, 644
    - Web site, 640
  - Toolkit class, 534, 826
  - toString method
    - ArrayList class, 411–412
    - Arrays class, 399, 402–403
    - called by combo boxes, 587
    - Exception class, 232
    - File class, 693, 694
    - Object class, 313, 314, 315–317
    - primitive wrapper classes, 104
  - toUpperCase method, 365, 366
  - translating shapes, 812
  - transparent shapes, 809
  - trees, 597–606
  - TreeSelectionListener interface, 603
  - Triangle JavaBean, 675–676
  - Triangle.jsp page, 680–681
  - trim method, 365, 366
  - try statement, 226, 227–229. *See also* exceptions
  - tryToChangeNumber method, 219
  - two-dimensional arrays
    - accessing elements, 387–388
    - analogies for, 386
    - as array of arrays, 385
    - creating, 386
    - initializing, 388
    - jagged, 388–390
    - uses for, 385
  - type casting. *See* casting
  - type checking, 15–17
  - typeface conventions in this book, 3
  - types. *See* data types
- 
- ## U
- 
- UML (Unified Modeling Language), 251–254
  - unary operators, 121, 125, 159
  - unboxing, 107
  - unchecked exceptions, 224. *See also* exceptions
  - Uniform Resource Locators (URLs), 473, 750
  - upcasting, 290
  - update statement (SQL), 733, 744–745
  - updating
    - array list elements, 413–415
    - database rows, 744–745, 761–762
    - database with JDBC, 758–762
    - linked list items, 426
  - uppercase. *See* capitalization or case; case sensitivity
  - use statement (SQL), 733, 735
  - user input. *See also* validating user input
    - borders for components, 574–576
    - check boxes for, 569–572
    - combo boxes for, 585–589, 590
    - drawing on component, 814–820

JOptionPane class for, 116–117  
 lists for, 590–595  
 opening files, 700–702  
 pizza-ordering program, 577–581  
 radio buttons for, 572–574  
 Scanner class for, 112–115  
 for servlets, 651–653  
 sliders for, 581–584  
 spinners for, 595–597  
 standard input stream for, 110–111  
 terminating infinite loops, 172–173  
 text areas for, 565–569  
 text fields for, 557–565  
 trees for, 597–606  
 UTFDataFormat  
   Exception, 720

---

## V

validating data, setters for, 262  
 validating user input  
   catching invalid integer exceptions, 228–229  
   do-while loop for, 177–179  
   integer data in text fields, 563–565  
   Scanner methods for, 115  
 Validation class, 563–565  
 value types. *See* primitive types  
 variables. *See also*  
   declaring variables  
   as array indexes, 378  
   for array length, 379  
   in class diagrams, 253  
   declaring for arrays, 378–379

examining values in  
   Eclipse, 63–64  
 final (constants), 94  
 final keyword for, 288  
 initializing, 92–93, 269–270  
 local, extracting in  
   Eclipse, 66  
 polymorphism, 292–294  
 rules for naming, 88  
 scope, 107–109, 182–183  
 shadowed, 109–110  
 type checking, 15–17  
 versions of Java, 19–20, 30  
 views (Eclipse), 48–49  
 visibility. *See also* private keyword; public keyword  
   of buttons, 539  
   of frames, 531, 533  
   for inherited members, 284  
   of inner classes, 335  
   overview, 260  
   protected, 284, 285–286  
   UML indicators for, 253  
 void keyword, 15, 71  
 VoterApp program, 193–194  
 VoterApp2 program, 195–196

---

## W

wait method, 313, 462  
 Warning! icon, 6  
 Web page, servlet printing to, 646  
 Web resources. *See* Internet resources  
 where clause (SQL), 737–739  
 while loops  
   basic form for, 168  
   break statement with, 170

continue statement  
   with, 174  
 counting loop using, 168–169  
 in do-while loops, 175–179  
 do-while loops versus, 175  
 flowchart for, 168  
 guessing game example, 190–192  
 infinite loops with, 170–173  
 nested, 190–192  
 overview, 168  
 in partition method, 521  
 white space  
   character classes for, 370, 499, 500–501  
   in code, 76  
   inserting during  
     concatenation, 103  
   optional with operators, 122  
   removing from strings, 366  
 wildcards  
   character class, 499  
   with ClassPath, 352–353  
   generics feature, 436–437  
   with import command, 86, 346  
   with javac command, 29  
   for SQL queries, 738–739  
 Window class, 528, 529  
 WindowAdapter class, extending, 553–554  
 WindowEvent class, 542  
 WindowListener  
   interface, 543–544, 553  
 wrapper classes, 100, 104–105, 107, 562  
 writing streams  
   binary, 724–729  
   character, 712–718

**X**

---

XML

- attributes, 765
- benefits of, 763
- described, 764
- DOM versus SAX for, 769–780
- DTD files for, 767–769
- Movie Listing program, 778–781
- movies.xml file for, 766
- reading DOM documents, 771–773

- reading DOM nodes, 774–778
- tags, 764–765
- Xor (Exclusive Or) operator, 159, 162–163

**Y**

---

yield method

- CountDownLatch class, 447
- Thread class, 446, 464

**Z**

---

zero (0)

- dividing by, catching exceptions, 227, 233–234

- dividing by, issues for, 144–145

- floating-point, 98, 145

- indexes numbered from, 377, 410, 424

- number minus itself not always equal to, 146