



# Index

## A

### **abrupt completion**

*See* completion, abrupt

### **abstract modifier**

*See also* declarations; modifiers  
and super method invocation, 472  
classes

- anonymous are never, 429
- binary compatibility considerations, 340
- declaration of, 173
- definition and characteristics, 173
- direct superinterface relationship to, 184
- enum types must not be, 176, 250

### methods

- binary compatibility considerations, 352
- classes, 215
- declaration examples, 209, 266
- interfaces, 267
- overloading, 268, 269
- overriding, 269
- semicolon as body of, 223
- and super method invocation, 472

### **access**

*See also* scope

- accessibility
  - determining, 138
  - term definition, 138
- array, 289
- expression evaluation order, 483
- conflicting, 563
- constructor, binary compatibility considerations, 344
- of fields, expression evaluation, 435
- inheritance of class members, example default, 192

- private, 193
- protected, 193
- public, 193

interface member names, 263  
method, binary compatibility considerations, 350

non-`public` class instances, through `public` superclasses and superinterfaces, 194

overridden methods, using `super` keyword, 225

### qualified

*See also* field access expressions; method invocation expressions  
term definition, 138

### **access control**

*See also* security

- classes, example, 141
- constructors
  - default, example, 142
  - private, example, 144
  - protected, example, 143
  - public, example, 143
- fields
  - default, example, 142
  - private, example, 144
  - protected, example, 143
  - public, example, 143
- methods
  - default, example, 142
  - private, example, 144
  - protected, example, 143
  - public, example, 143
- package names, limited significance of, 154
- protected, details of, 139
- and qualified names, 138
- term definition, 138

- accessible**, 443
  - annotation type, 283
  - class type, 184
  - default for top-level types, 166
  - member type, 44
  - package, term definition, 138
  - single-static import must name a type that is, 164
  - single-type import must name a type that is, 161
  - type may not be named by static-import-on-demand if not, 165
  - type or package may not be named by type-import-on-demand if not, 163
  - types, members and constructors, term definition, 138
- acquire**
  - term definition, 561
- action**
  - committing, 568
  - external, 559, 571
  - final, 561
  - freeze, 575
  - hang, 571
  - inter-thread, 558
  - lock, 558
  - observable, 560, 572
  - read, 558
  - thread divergence, 559
  - unlock, 558
- algebraic identities**
  - See also* expressions; mathematical functions
  - limits on use of, 417
- alphabet**
  - See also* syntax
  - components
    - See* characters
  - data types
    - See* primitive types
  - term definition, 9
- ambiguity**
  - See also* names, ambiguous; scope
  - avoiding, in fields with multiple inheritance, 209
  - avoiding, in types with multiple inheritance, 238, 270
- America, Pierre**, 55
- annotation(s)**, 410
  - element(s)
    - names, 129
  - marker, 282, 285
  - meta, 158, 175, 214, 241, 264, 272, 278, 364
    - term definition, 284
  - names, 128
  - normal, 282, 285
  - of package or top-level type, 410
  - on annotation types, 272
  - on classes, 175
  - on constructors, 241
  - on enum(s), 250
  - on interface fields, 264
  - on interface methods, 267
  - on interfaces, 261
  - on local variables, 364
  - on methods, 214
  - on packages, 158
  - single-element, 282, 285
  - term definition, 281
- annotation.Annotation**, 271, 272, 273
- annotation.ElementType**, 278
- annotation.ElementType.ANNOTATION\_TYP**  
E, 272
- annotation.ElementType.CONSTRUCTOR**,  
241
- annotation.ElementType.FIELD**, 250, 264
- annotation.ElementType.LOCAL\_VARIABLE**  
, 364
- annotation.ElementType.METHOD**, 214, 267
- annotation.ElementType.PACKAGE**, 158
- annotation.ElementType.TYPE**, 261, 272
  - and array subtyping, 175
- annotation.Inherited**, 279
- annotation.Retention**, 278
- annotation.RetentionPolicy**
  - .CLASS, 278
  - .RUNTIME, 278
  - .SOURCE, 278
- annotation.Target**, 158, 175, 214, 241, 250,  
261, 264, 267, 272, 278, 364
- anonymous class**, 173
  - See also* inner class
  - definite assignment before, 548
  - determining immediately enclosing instance  
when instantiating, 426

## INDEX

- example(s), 204, 245
- exceptions in instance initializer, 202, 238, 301
- in an explicit constructor invocation, 245
- implicitly defined by enum constant, 250
- interaction with shadowing of parameter names, 212
- in an interface variable initializer, 265
- protected accessibility of superclass constructor, 140
- supertypes classified as type names, 128
- term definition, 424
- applicable**
  - by method invocation conversion, 442
    - term definition, 446
  - by subtyping, 442
    - term definition, 445
  - constructor, 451
  - potentially, 442, 446
  - variable arity method, 442
  - variable-arity method
    - term definition, 447
- argument**
  - See also* parameters
  - lists, evaluation order, 418
  - values, method invocation conversion context, 99
- arithmetic**
  - See also* floating point; integers; numbers; operators; primitive types
  - integer division, `ArithmeticException`, 414
  - operators, numeric promotion, specification, 108
- arrays**
  - See also* classes; data structures (chapter), 287
  - access, 289
    - expression evaluation order, 483
  - assignment, expression evaluation order, 513
  - character, distinguished from strings, 294
  - `Class`
    - objects, 293
    - obtaining, example, 74
  - `Cloneable` interface implemented by, 292
  - components
    - See also* variables
    - assignment, run-time testing, 295, 412, 514, 519
    - default values, 71
    - initialization, 291
      - as a kind of variable, 69
      - type declaration, 291
    - creation, 45, 289
      - expression evaluation, 432
      - order, 432
      - out-of-memory detection, example, 434
    - elements
      - are variables in memory model, 558
    - example, 288
    - exceptions, `ArrayStoreException`, 295
    - indexing of, 289
    - initialization, 289, 290
    - initializer(s), 283
    - length
      - and data races, 563
    - members, 125, 292
    - names, fully qualified, 145
    - `Object` as superclass of, 293
    - origin, 290
    - variables
      - declaration, 288
      - initialization, 288
  - ASCII characters**
    - See also* characters; Unicode character set
    - Unicode character set relationship to, 14
  - assert**
    - keyword, 21
    - statement, 373
      - can complete normally, 403
      - definite assignment, 541
  - assertion(s)**
    - enabling
      - during initialization, 321
    - term definition, 373
  - `AssertionError`**, 376
  - assignment**
    - See also* fields; initialization
    - array, expression evaluation order, 513, 519
    - assignable to, term definition, 95
    - compatible, 283, 410
      - term definition, 95
    - compound
      - evaluation order, 415
      - operators, evaluation, 518
    - conversion context, specification, 78, 93
    - definite, (chapter), 527

**assignment** (*continued*)

- expressions
  - boolean, definite assignment, 533
  - definite assignment, 533
  - as statements, 371
- operation, to change value of primitive value, 35
- simple operators, evaluation, 513

**B****backslash** (\)

- escape sequence, 30
- Unicode escape use, 15

**backspace**

- escape sequence, 30

**backtrace**

- of an exception, 401

**Bartleby**

- Project (Columbia University), xxii

**base**

- See also* numbers
- permitted in integer literals, 22

**Beta**, 7**biblical quotations**

- I Corinthians 14:40, 414
- John 3:30, 488
- Matthew 6:29, 287

**bibliographic references**

- Dynamic Class Loading in the Java Virtual Machine, 34, 313
- On Variance-Based Subtyping for Parametric Types, 54, 92
- Polling Efficiently on Stock Hardware, 304
- Release-to-Release Binary Compatibility in SOM, 333
- Types and Programming Languages, 91
- Unifying Genericity, 55
- Wild FJ, 54, 92

**binary**

- compatibility
  - See also* code generation; compile-time errors; exceptions
  - (chapter), 333
  - changes that do not break, 334
  - compatible with, term definition, 339
  - contrasted with source compatibility, 340
  - file format, required properties, 335

## name

- See* name, binary
- numeric promotion, specification, 110
- representation, verification of classes and interfaces, 312

**binary representation**, 278**blank final**

## field

- class variable must be assigned by static initializer, 199
  - definite [un]assignment of, 538
  - definite assignment of, 527
  - instance variable must be assigned by every constructor, 199
  - notation for definite assignment, 532
  - restrictions in an inner class, 182
- variable
- decrementation of, 488
  - definite unassignment of, 527
  - incrementation of, 488
  - notation for definite unassignment, 532

**blocks**

- See also* control flow; statements
- (chapter), 359
- definite assignment, 538
- enclosing, 182
- in scope of exception handler parameters, 118, 397
- in scope of local class, 118, 362
- in scope of local variable, 118, 364
- specification and execution of, 361

**Bobrow, Daniel G.**, 6**body**

- See also* declarations
- class
  - declarations in, 189
  - term definition, 189
- constructor, 242
  - binary compatibility considerations, 354
- interface, declarations, 263
- method, 223
  - binary compatibility considerations, 354

**Boolean**, 5**boolean**

- See also* numbers
- literals, term definition and specification, 26
- operators, 43
- types and values, term definition and specification, 43

## INDEX

**bootstrap loader**

*See* class loaders, bootstrap

**bound(s)**, 89, 466

lower, 90

is a direct supertype of type variable, 64

type variable(s), 118, 179, 220, 242, 261

binary compatibility, 342, 351

in erasure, 56

term definition, 50

upper, 90, 441

wildcard

explicit, 53

upper, 53

**boxing**

*See see* conversion, boxing

**brackets ([])**

array type declaration, 44, 287, 364

**break statement**

*See also* control flow

definite assignment, 545

as reason for abrupt completion, 360

specification, 388

**Burke, Edmund**, 359**Burroughs, Edgar Rice**, 392**Burton, Robert**, xxiii**but not phrase**

grammar notation use, 12

**byte type**

*See also* integral types; numbers; primitive types

value range, 35

**C**

**C**, 1, 2, 7

**C++**, 1, 2, 6

**Caesar, Julius**, 493

**caller**

of a statement, term definition, 302

**carriage return (CR)**

escape sequence, 30

handling in a

character literal, 27

string literal, 28

as a line terminator, not input character, 16

**casting**

*See also* conversion

and data races, 563

**boolean**, 43

checked, 102

compiler-generated, 412

conversion context, 79

specification, 101

floating-point types, 40

integral types, 36

reference types, 102

run-time testing, 412

statically correct, 102

unchecked, 102

to **void**, not permitted, 371

**catch clause**, 394

*See also* control flow; try statement

exception handling role of, 297

exception idiom defined with, 306

scope of parameters, 118, 397

**causality**, 571

requirements of memory model, 569

**Cervantes, Miguel de**, 101**characters**

*See also* numbers; primitive types; strings

array of, distinguished from strings, 294

char type, 33

*See also* integral types; numbers; primitive types

value range, 35

line terminators, 28

literals

escape sequences for, 30

term definition and specification, 26

Unicode character set

composite, contrasted with the Unicode

decomposed characters, 20

handling in package names, 156

lexical grammar use as terminal symbols, 9

relationship to ASCII, 14

**Chase, Lincoln**, 152**checking**

*See also* exceptions, checked; throw

for exception handlers, at compile-time, 299

**Christie, Agatha**, 360, 372, 442**circular declaration**

of a class

compile-time error caused by, 185

example involving a local class, 362

link-time error caused by, 186

of an interface, compile-time error caused

by, 262

**circular declaration** (*continued*)

of types, in different compilation units,  
legality of, 157

**Class class**, 48, 283

allowed in annotation type, 273

**class literal**, 128

class or interface name in a, 114  
is a primary expression, 420  
term definition, 421

**class loaders**

bootstrap, 330  
and class unloading, 330  
defining  
and enabling assertions, 374

**class(es)**

*See also* fields; inheritance; interfaces;  
methods; packages; subclasses;  
superclasses; superinterfaces

(chapter), 173

**abstract**, 176

as array component types, 288  
binary compatibility considerations, 340  
overriding abstract methods in, 215  
uses, 177

accessibility, 138

annotation(s), 281

anonymous

binary name, 335  
*See* anonymous class

binary compatibility considerations, 340

binary representation

binary file format requirements, 339  
verification of, 310

body

declarations, binary compatibility  
considerations, 343

term definition and declarations in, 189

.class suffix, as name for compiled files, 156

Class objects, associated with arrays, 293

constructors, binary compatibility

considerations, 350

declarations, 175

generic, 49

names in extends clause, 128

names in implements clause, 128

specifying direct superclasses in, 184

specifying direct superinterfaces in, 186

term definition, 175

as declared entity, 113

enclosing

lexically, 422, 441

top-level is initialized due to assert, 374

exceptions

ClassCastException, 105

Error, 301, 306

Exception, 306

RuntimeException, 301, 306

Throwable, 297, 299, 306

final

binary compatibility considerations, 341

declaration of, 178

finalization of, 341

FP-strict, 411

generic, 173, 184

term definition, 178

inaccessible, accessing members of, 194

inaccessible, accessing members of,

example, 194

initialization, 316

detailed procedure, 319

example, 311

instances

*See* instance(s)

instantiation, preventing, 240

linking

initialization, 307, 310, 319

preparation, 307, 315, 320

process description, 315

resolution, 310

at virtual machine startup, 310

loading, 312

errors

ClassCircularityError, 313

ClassFormatError, 313

NoClassDefFoundError, 313

process description, 313

at virtual machine startup, 310

local

binary name, 335

*See* local class

member

*See* member class

members, 123

declarations, 189

binary compatibility considerations, 343

methods

class, 216

interface, 266

## INDEX

- non-static, 216
  - static, 216
- named, 173
- names, fully qualified, 145
- naming conventions, 147
- nested
  - See* nested classes
- non-public, in example of qualified names
  - and access control, 141
- normal, 113, 176
- preparation, 315
  - at virtual machine startup, 310
- preventing instantiation of, 177
- public
  - access control in, 138
  - binary compatibility considerations, 341
  - in example of qualified names and access control, 141
- references to, binary file format
  - requirements, 335
- resolution
  - exceptions
    - `IllegalAccessError`, 315
    - `IncompatibleClassChangeError`, 315
    - `InstantiationException`, 316
    - `NoSuchFieldError`, 316
    - `NoSuchMethodError`, 316
  - process description, 315
  - at virtual machine startup, 310
- scope of, 117, 161
- in scope of an imported type, 117, 161
- static initializers, 239
  - binary compatibility considerations, 356
- that depend on themselves
  - See* circular declaration, of a class
- that directly depend on a reference type
  - See* circular declaration, of a class
- top-level, 317
- top-level
  - See* top-level class
- type declarations, as members of packages, 166
- unloading of, 330
- variables
  - default values, 71
  - specification, 69, 198
- verification, at virtual machine startup, 310
- wrapper, 2, 497

**ClassCastException**, 95, 100

**classification**

- reclassification of contextually ambiguous names, 129
- syntactic, of a name according to context, 127

**ClassLoader class**, 313

**Cloneable interface**, 85, 98

- and array subtyping, 64

**code generation**

- See also* binary, compatibility; compile-time errors; exceptions; optimization
- asynchronous exceptions, implications for, 303
- initialization, implications for, 321
- symbolic reference resolution, implications of, 315

**comments**

- term definition and specification, 18

**Common Lisp**, 6

**compatibility**

- See* binary, compatibility

**compilation**

- See also* compile-time errors; exceptions; virtual machine

- CompilationUnit* goal symbol, syntactic grammar use, 10

## conditional

- binary compatibility considerations, 348
- if statement reachability handling to support, 405

- exception handlers checked for during, 299
- unit, 444

- components, package declarations, 158
- and default accessibility of top-level types, 166

- implicitly starts with `import`

- `java.lang.*`, 165

- importing types into, 161

- and name of top-level type, 166

- overview, 153

- and package membership, 122, 154

- and package observability, 160

- and scope of an import declaration, 117, 161

- term definition and characteristics, 157

**compile**, 439**compile-time constant**

- See* constants

- compile-time error**, 439
  - ambiguous type in `new`, 425
  - annotation names non-annotation type, 283
  - annotation type name same as enclosing types, 272
  - annotation type with bad `Target`, 272
  - declare an enum type abstract, 176, 250
  - declaring a finalizer in an enum, 251
  - disallowed return type in annotation type, 273
  - enum
    - constant
      - to declare abstract method, 176, 250
    - enum constant with bad `Target`, 158, 175, 198, 211, 214, 241, 250, 261, 264, 267, 364
    - enum in `new`, 424
    - enum type to contain abstract method, 176, 250
  - explicitly declaring an enum final, 250
  - in `?:`, 510
  - in `super.Identifier`, 441
  - inaccessible annotation type, 283
  - inaccessible type in `new`, 424
  - instantiating an enum type, 249
  - interface attempts to override final method in `Object`, 125, 263
  - malformed annotation, 281
  - malformed type declaration specifier, 44
  - malnamed element in annotation, 137, 283
  - method annotated with `Override` does not in fact override, 279
  - modifier repeated in interface declaration, 260
  - no applicable method, 447
  - non-integral type in array index, 432
  - non-override equivalent signature, 273
  - non-reifiable type in array creation, 289, 431
  - nonsubstitutable return type, 268
  - referencing a type parameter of an interface from its fields or type members, 261
  - referencing non-constant static field in an enum, 252
  - repeated annotations of same type, 281
  - self reference in an annotation type, 274
  - subclass of `Enum`, 184
  - Target**, 278
  - two override-equivalent methods, 267
  - void method in conditional expression, 510
  - wildcards in `new`, 424
- compile-time errors**
  - See* binary, compatibility; errors; exceptions
- completion**, 360
  - See also* control flow, exceptions
  - abrupt
    - `break` statement, 388
    - `continue` statement, 390
    - disallowed for static initializers, 239
    - during expression evaluation, 413
    - `for` statement, 384
    - labeled statements, 371
    - reasons for, 360, 413
    - `return` statement, 392
    - synchronized statement, 395
    - `throw` statement, 393
    - `try` statement, 396
    - `try-catch` statement, 398
    - `try-catch-finally` statement, 399
  - normal
    - during expression evaluation, 413
    - during statement execution, 360
    - required for instance initializers, 239
    - required for static initializers, 239
- component(s)**
  - See also* arrays, components
  - type
    - arrays
      - in variable arity methods, 447
      - of reifiable array type, 56
    - type, arrays, 288
- conflicts**
  - memory accesses, 563
  - name
    - See also* hiding; obscuring; scope; shadowing
    - avoiding through use of syntactic context, 127
    - throws clause, 225, 268
- Conner, Michael**, 333
- consistency**
  - happens-before, 566
  - sequential, 560, 563, 578
  - term definition, 560
- constants**, 249
  - See also* fields; class; `final`; fields, interface; literals



## INDEX

- are always FP-strict, 411
- boolean
  - and definite assignment, 535
- characteristics and binary compatibility
  - considerations, 347
- compile-time
  - in enums, 252
- compile-time narrowing of
  - by assignment conversion, implications, 94
  - not permitted by method invocation
    - conversion, 100
- enum, 114, 249
- expressions, 94, 283, 336, 348, 525
- may be declared in inner class, 181
- named
  - See* fields; class; `final`
- variable(s), 317, 336, 526
  - term definition, 71
- constraint(s)**
  - type inference
    - initial, 445, 446, 447, 448, 452, 466
  - type inference
    - initial, 448, 449
- constructor**, 451
- constructors**
  - See also* class(es); execution; initialization; interfaces; methods
  - access, 138
    - binary compatibility considerations, 344
  - control
    - default, example, 142
    - `private`, example, 144
    - `protected`, example, 143
    - `public`, example, 143
    - hidden fields, `this` keyword use for, 367
  - accessibility, 139
  - allow field access through `super`, 438
  - annotation(s), 281
  - anonymous
    - term definition, 429
  - anonymous class
    - cannot have explicitly declared, 429
    - has anonymous constructor, 429
  - applicability, 451
  - are not members, 191
  - body, 242
    - binary compatibility considerations, 354
    - cause a freeze action, 575
    - completion of happens-before finalize, 325
    - as components of a class body, 189
    - declaration(s), 240, 410
      - binary compatibility considerations, 350
      - generic, 49
    - default, 247
    - definite assignment and unassignment
      - within, 550
    - definite assignment of variables before, 538
    - deleting, binary compatibility
      - considerations, 350
    - determining arguments to, 427
    - each must declare exceptions from instance
      - initializers, 202, 238, 301
    - enum
      - constant must not refer to itself, 252
      - default, 250
      - must not reference non-constant static fields, 252
    - explicit invocation statements, 244
    - explicit invocations, 138
    - FP-strict, 411
    - generic, 127, 242
    - in an enum, 251
    - invocation, during instance creation, 322
    - modifiers, 241
    - must assign all blank final instance variables, 199
    - must assign blank final fields, 199
    - names of, 115, 240
    - as non-members of a class, 123
    - of a wildcard, 55
    - of `Object` must be invoked for an object to be finalizable, 327
    - of parameterized types, 55
    - overloading, 246
      - binary compatibility considerations, 355
    - parameters, 240
      - See also* arguments
      - assignment during instance creation, 322
      - binary compatibility considerations, 352
      - definite [un]assignment of, 547
      - must not be assigned if `final`, 211
      - shadowing, 119, 212
      - specification, 72, 240
    - `private`, preventing instantiation with, 178, 248

**constructors** (*continued*)

- protected, accessibility of, 139, 140
- signature, 241
  - binary file format requirements, 338
- throws clause, binary compatibility considerations, 354
- and unreachable statements, 402
- when reachable, 403

**containment**

- type argument, 55

**context(s)**

- See also* inheritance; scope
- conversion, 78
  - assignment, specification, 93
  - casting, specification, 101
  - method invocation, specification, 99
- final field safe, 577
- reclassification of contextually ambiguous names, 129
- role in determining the meaning of a name, 113, 126
- single-threaded, 557
- static
  - See* static context
- syntactic classification of a name according to, 127

**context-free grammars**

- See* grammars, context-free

**continue statement**

- See also* control flow
- definite assignment, 545
- as reason for abrupt completion, 360
- specification, 390

**contract**

- See also* binary, compatibility
- term definition, 339

**control flow**

- See also* completion; definite assignment; exceptions; expressions; statements; statements, unreachable; threads
- boolean expressions use for, 43
- break statement, specification, 388
- continue statement, specification, 390
- deadlock avoidance, multiple lock use for, 396
- do statement, specification, 382
- for statement, specification, 384
- if statement, dangling else handling, 368

- if-then statement, specification, 372
- if-then-else statement, specification, 372
- local variable declaration and switch statement, 70
- return statement, specification, 392
- switch statement, specification, 377
- synchronized statement, specification, 395
- throw statement, specification, 393
- try statement, specification, 396
- try-catch statement, specification, 398
- try-catch-finally statement, specification, 399
- while statement, specification, 380

**conventions**

- naming, 146
  - impact on obscuring of names, 122

**conversion**

- See also* casting; numbers; promotion (chapter), 77
- assignment, 466
- boxing, 78, 93, 95, 99, 101, 421, 486, 488, 511
  - in conditional expression, 512
  - in overload resolution, 442
  - may cause `OutOfMemoryError`, 414
  - term definition, 86
- capture, 56, 78, 89, 101, 135, 136, 436, 450, 482, 511
  - in assignment, 513
- casting, 101
- contexts
  - assignment, specification, 93
  - casting, specification, 101
  - method invocation, specification, 99
- forbidden, specification, 92
- identity, 90, 108
  - in assignment conversion context, 93
  - in casting conversion context, 101
  - in method invocation conversion context, 99
- specification, 80
- kinds of, 80
- method invocation, 452
  - in determining applicability, 442, 446, 447
- in method invocations, 99
- narrowing
  - primitive
    - in assignment conversion context, 94

## INDEX

- in casting conversion context, 101
  - not allowed in method invocation conversion context, reasons for, 95, 100
  - reference, in casting conversion context, 101
- narrowing primitive, 486, 488
- reference
  - narrowing, 85
  - widening, 85
- string, 101, 497
  - in assert statements, 375
  - specification, 87
- term definition, 77
- unboxing, 36, 41, 78, 93, 95, 99, 101, 108, 110, 372, 375, 379, 381, 382, 385, 486, 488, 490, 492, 500, 503, 506, 507, 508, 509, 510, 512
  - in conditional expression, 512
  - in overload resolution, 442
  - term definition, 88
- unchecked, 78, 99, 445, 450
  - term definition, 89
- value set, 92, 486, 488, 489, 500, 503
  - in assignment conversion, 94
  - in binary numeric promotion, 110
  - in casting conversion, 101
  - in method invocation conversion, 99
  - in overview, 78
  - term definition, 92
  - in unary numeric promotion, 108
- value-set, 506
- widening, 108
  - primitive, 93
    - in assignment conversion context, 93
    - in binary numeric promotion context, 110
    - in casting conversion context, 101
    - in method invocation conversion context, 99
    - in unary numeric promotion context, 108
  - reference, 85, 93
    - in assignment conversion context, 93
    - in casting conversion context, 101
    - in method invocation conversion context, 99
- widening primitive, 108

**convertible**

- to a numeric type, 110, 485, 486, 487, 488, 506
  - in conditional expression, 511
  - term definition, 89
- to a primitive integral type, 508
- to a primitive numeric type, 489, 490, 491, 496, 502, 503
- to an integral type, 432
  - term definition, 89
- to numeric type, 506

**creation**

- See also* declaration; initialization
- array, 45, 289
  - expression evaluation
    - example, 433
    - order, 432
    - out-of-memory detection, example, 434
- instance, 45
  - expression evaluation order, 428
  - expressions as statements, 371
  - invocation of initializers for instance
    - variables during, 324
  - method dispatching during, 324
  - specification and procedure, 322
- object, 45
  - of an enum constant, 250
    - term definition, 251

**Creatore, Luigi, 32****D****Danforth, Scott, 333****dangling else**

- See also* control flow
- handling of, 368

**data**

- See also* constants; fields; variables
- structures
  - See* arrays; classes; interfaces; primitive types; vectors
- types
  - See* types
- values
  - See* values

**database**

- storing packages in, 153

- deadlock**, 554
  - avoidance, multiple lock use for, 396
- decimal**
  - See also* numbers
  - base, permitted in integer literals, 22
  - numerals, specification, 22
- declarations**
  - See also* body; execution; methods; parameters; statements
  - class
    - (chapter), 173
    - body, 189
    - member, 189
    - term definition and specification, 173
  - constructor, 240
  - enum, 249
  - field(s), 196
    - constants in interfaces, 264
    - examples of, 205
    - interface, examples, 265
  - identifiers in, kept separate from those in labeled statements, 116
  - import
    - example, 165
    - single-type, 161
    - term definition, 160
    - type-import-on-demand, 163
  - instance variable, 69
  - interface, 260
    - (chapter), 259
    - body, 263
    - members, 263
  - local variable, definite assignment, 539
  - method(s), 209
    - examples of, 230
    - interface, abstract, 267
  - modifiers
    - See*
      - abstract modifier;
      - final modifier;
      - native modifier;
      - private modifier;
      - protected modifier;
      - public modifier;
      - static modifier;
      - synchronized modifier;
      - transient modifier;
      - volatile modifier
  - overriding, example, 230
  - package, in compilation units, 157
  - scope of, 117
  - shadowed, term definition, 119
  - shadowing, 119
  - subclass, 184
  - superclass, 184
  - superinterface, 186, 261
  - term definition, 114
  - type
    - as members of packages, 166
    - need not appear before use of the type, 119
    - usage in, 65
  - variable
    - array, 288
    - local, 363
    - execution of, 367
- declarator**
  - in scope of local variable, 118, 365, 385, 387
- default**
  - access, 224
  - constructor, 247
    - enum, 250
  - keyword
    - in annotation types, 274
  - value
    - for annotations, 274
    - of a variable, 71
    - of array component, 432
    - variable(s), 67
    - write synchronizes with, 561
- definite assignment**
  - See also* exceptions; execution; scope (chapter), 527
  - and anonymous classes, 548
  - of blank final class variables, 199
  - of instance variables, 199
  - and local classes, 539
  - of member types, 549
  - and parameters, 547
  - and qualified class instance creation, 537
  - requirement for locals and parameters before
    - an inner class, 182
  - and static initializers, 549
  - of variables with respect to nested classes, 538
- definite unassignment**, 527
  - See also* definite assignment
  - and incrementing blank finals, 488
  - hypothetical analysis of, 532



## INDEX

- and local classes, 539
- of blank final variables
  - in decrement operator, 488
- and qualified class instance creation, 537
- and static initializers, 549

**DeMichiel, Linda G.**, 6

**denormalized**

- IEEE 754 standard, support required for, 41

**Deprecated**, 280

**dereference chain**, 575

**digits**

- Java, term definition, 19

**direct subtype**

- term definition, 63

**directly depends**

- class, term definition, 185

**discouraged**

- use of raw types, 59

**division**

- See also* arithmetic; numbers
- integer divide by zero, indicated by
  - ArithmeticException, 37, 493

**do statement**

- See also* control flow
- definite assignment, 543
- specification, 382

**double**

- variables
  - non-atomic treatment, 579

**double quote**

- escape sequence, 30
- in string literals, 28

**double type**

- floating-point literal specification, 24

**Duff's device**, 378

**Dylan**, 6

## E

**Eco, Umberto**, 152

**Eisenhower, Dwight D.**, 507

**element type**

- arrays, term definition, 288

**Ellis, Margaret A.**, 6

**Ellison, Shirley**, 152

**entity**

- declared, list of, 114

**enum**

- constant
  - all provide implementation for abstract
    - method, 176, 250
  - term definition, 249
- constants, 283
  - annotations(s), 281
- constructor
  - default, 250
- Enum class, 251
- keyword, 21
- type, 249
- type(s)
  - static when nested, 250

**Enum class**

- subclassing, 184

**Epictetus**, 264

**erasure**, 57, 102, 241, 342, 347, 351, 352, 449, 450, 463

- and qualifying type, 337, 338
- changes cause warning, 60
- of qualifying type, 336
- restrictions required due to, 227
- signature, 213
- term definition, 56

**Ernst, Erik**, 54, 92

**errors**

- class variable initializers may not refer to
  - instance variables, 202
- Error class, unchecked exceptions as
  - subclass of, 222
- linking, 307, 314
  - resolution, 315
  - verification, 314
- loading, 307, 313
- reasons for, 298
- semantic, exceptions as mechanism for
  - signaling, 297
- unchecked, reasons for, 301
- virtual machine, 307

**escapes**

- sequences, for character and string literals, 30
- Unicode, specification, 15

**evaluation**

- See also* execution; initialization; scope
- evaluated, term definition, 409
- expressions



**evaluation** (*continued*)

- additive operators, 496
- additive operators for numeric types, 500
- array access, 482
- array assignment, 513
- array creation, 432
- assignment operators, 512
- bitwise binary operators, 508
- bitwise complement, 490
- boolean equality operators, 507
- boolean logical operators, 508
- cast, 490
- compound assignment operators, 518
- conditional operator, 510
- conditional-and operator, 509
- conditional-or operators, 509
- division, 493
- equality operators, 505
- field access, 435
- instance creation, 428
- integer bitwise operators, 508
- logical binary operators, 508
- logical complement, 490
- method invocation, 440
- method invocation, order of, 473
- multiplication, 492
- multiplicative operators, 491
- numeric comparison, 503
- numeric equality operators, 506
- parenthesized, 422
- postfix, 486
- pre-decrement, 488
- pre-increment, 487
- primary, 420
- reference equality operators, 507
- relational operators, 503
- remainder, 495
- shift operators, 502
- simple assignment operator, 513
- string concatenation, 497
- superclass access, 438
- type comparison, 504
- unary minus, 489
- unary operators, 487
- unary plus, 489

literals, 420

order

- arguments left-to-right, 418
- binary operators, 415

- compound assignment, 415
- left-hand operand first, 415
- left-to-right, 414
- operands evaluated before operations, 416
- parentheses and precedence respected, 417
- result of, term definition, 409

**evolution**

*See also* binary, compatibility; reuse

- of classes, binary compatibility
  - considerations, 340
- of interface, binary compatibility
  - considerations, 356
- of packages, binary compatibility
  - considerations, 340

**examples**

- access control, 140
  - fields, methods, and constructors
    - default, 142
    - private, 144
    - protected, 143
    - public, 143
- arrays, 290
- capture conversion, 90
- classes, `public` and `non-public`, 141
- declarations
  - fields, 205
  - import, 165
  - methods, 230
- default-access methods, constructors, and
  - fields, 142
- exceptions, 304
- fields
  - interface, ambiguous inherited, 265
  - interface, multiply inherited, 266
  - multiply inherited, 207
  - re-inheritance of, 209
- hiding, 231
  - vs. overriding, 232
- of variables
  - class, 205
  - instance, 206
- illegal self-reference in annotation, 274
- inheritance
  - accessing members of inaccessible classes, 194
  - class members, 192
    - default access, 192
    - public, protected, and private access, 193

## INDEX

- multiple, with superinterfaces, 188
  - with default access, 192
  - with private access, 193
  - with protected access, 193
  - with public access, 193
- methods
  - abstract declarations, 269
  - invocation of hidden class, 234
  - private, 144
  - protected, 143
  - public, 143
- overloading, 231
- overriding, 230
  - vs. hiding, 232
  - incorrect, 231
    - because of throws, 236
  - large example, 234
- wildcards, 52
  - as components of array types, 53
  - bounded, 53
- exceptions**
  - See also* binary, compatibility; compile-time errors; errors;
  - (chapter), 297
  - asynchronous, causes and handling of, 303
  - caller, determination of, 302
  - causes of, 298
  - checked
    - constructors, declaring with throws clause
      - in method declarations, 221
    - defining new exception classes as, 306
    - Exception, 306
    - in an instance initializer, 202, 238, 301
    - methods, declaring with throws clause in
      - method declarations, 221
    - in a static initializer, 239
    - UnsatisfiedLinkException, 316
  - classes
    - Error, 299
    - Exception, 301, 306, 398
    - RuntimeException, 301
      - unchecked exceptions found in, 222
    - Throwable, 297, 299
  - errors
    - AbstractMethodError, 352, 353
    - ClassCircularityError, 186, 341
      - meaning, 313
    - ClassFormatError, meaning, 313
    - Error, 306
    - ExceptionInInitializerError, 321, 394
    - IllegalAccessError, 341, 356, 357, 476
      - meaning, 315
    - IncompatibleClassChangeError, 348, 354, 357, 475
      - meaning, 315
    - InstantiationError, meaning, 316
    - LinkageError, 313, 314
    - loading and linkage, 307
    - NoClassDefFoundError, 320
      - meaning, 313
    - NoSuchFieldError, 347
      - meaning, 316
    - NoSuchMethodError, 350, 357, 475
      - meaning, 316
    - OutOfMemoryError, 222, 313, 321, 322, 413, 428, 432, 434
    - StackOverflowError, 477
    - UnsatisfiedLinkError
      - meaning, 316
    - VerifyError, 353, 357
      - meaning, 314
    - VirtualMachineError, 307
  - example, 304
  - handlers
    - compile-time checking, 299
    - how established, 297
    - in try statements, 397
  - handling of, 302
    - asynchronous, 303
    - run-time testing, 412
  - hierarchy, 306
  - NegativeArraySizeException, 413, 432
  - never thrown for
    - assignment conversions, 95
    - information loss due to narrowing
      - primitive conversions, 84
    - information loss due to widening primitive
      - conversions, 81
    - widening reference conversions, 85
  - NullPointerException, 245
  - parameters
    - See also* variables
    - declaration, 397
    - description, 70
    - initial value, 72
    - scope, 118, 397
  - precise, 303

**exceptions** (*continued*)

- RuntimeException, 515
- synchronization integrated with mechanism for handling, 298
- thrown for, narrowing reference conversions, 85
- uncaughtException method, when invoked, 298
- unchecked, 306
- unchecked runtime
  - ArithmeticException, 416, 495
  - ArrayIndexOutOfBoundsException, 514, 519
  - ArrayStoreException, 295, 413, 514, 518
  - ClassCastException, 412, 491, 514
    - casting conversion requirements that can result in, 105
  - IndexOutOfBoundsException, 290, 413, 483
  - NullPointerException, 291, 302, 413, 476, 479, 483, 484, 513, 514, 519
  - RuntimeException, 306, 515

**execution**

- See also* declarations; evaluation; initialization; linking; loading
- (chapter), 309
- of statements for their effect, 359

**execution(s)**

- described by a tuple, 567
- happens-before edges in, 562
- legal, 562
- non-terminating, 560
- order, 560
- well formed, 568

**exit**

- virtual machine, criteria for, 331

**exponent**

- See* value set

**expressions**

- See also* fields; methods; statements
- (chapter), 409
- abrupt completion of, as reason for abrupt statement completion, 361
- additive operators
  - evaluation, 496
  - for numeric types, evaluation, 500
- array
  - access, evaluation, 482

- assignment, evaluation, 513, 519
- creation, evaluation of, 431
- assignment
  - conversion, 93
  - definite assignment, 527, 533
  - operators, evaluation, 512
    - as statements, 371
- bitwise
  - binary operators, evaluation, 508
  - complement, evaluation, 490
- boolean
  - operators
    - !, definite assignment, 534
    - &&, definite assignment, 533
    - ?, definite assignment, 534
    - ||, definite assignment, 534
    - assignment, definite assignment, 535
    - constant, definite assignment, 533
    - equality, evaluation, 507
    - logical, evaluation, 508
- cast, evaluation, 490
- compound, assignment operators, evaluation, 518
- conditional
  - and operator &&, evaluation, 509
  - operator ? :
    - definite assignment, 535
    - evaluation, 510
  - or operator ||, evaluation, 509
- definite assignment and, 533
- division, evaluation, 493
- equality operators, evaluation, 505
- field access, evaluation, 435
- instance creation
  - evaluation of, 423
  - as statements, 371
- integer bitwise operators, evaluation, 508
- logical
  - comparison operators, evaluation, 508
  - complement !, evaluation, 490
- method invocation
  - evaluation, 440
  - evaluation order, 473
  - as statements, 371
- multiplication \*, evaluation, 492
- multiplicative operators \*, /, %, evaluation, 491
- names
  - context in which a name is classified as, 128



## INDEX

- qualified, meaning of, 135
  - simple, meaning of, 134
  - numeric
    - comparison, evaluation, 503
    - equality operators, evaluation, 506
  - operators
    - ++, definite assignment, 536
    - , definite assignment, 536
    - precedence, evaluation, 417
  - parenthesized
    - evaluation of, 422
    - evaluation of, precedence effect of, 417
  - post-decrement --
    - evaluation of, 486
    - as statements, 371
  - postfix, evaluation, 485
  - post-increment ++
    - evaluation of, 485
    - as statements, 371
  - pre-decrement --
    - evaluation of, 488
    - as statements, 371
  - pre-increment ++
    - evaluation of, 487
    - as statements, 371
  - primary, evaluation of, 420
    - See also*
      - arrays, access expressions;
      - arrays, creation;
      - expressions, parenthesized;
      - fields, access expressions;
      - instance, creation;
      - literals;
      - methods, invocations;
      - this keyword
  - reference equality operators ==, !=,
    - evaluation, 507
  - relational operators <, >, <=, >=, evaluation, 503
  - remainder operator %, evaluation, 495
  - run-time checks of, 411
  - semantics and evaluation rules, (chapter), 409
  - shift operators <<, >>, >>>, evaluation, 502
  - simple assignment operator =, evaluation, 513
  - statements
    - definite assignment, 538
    - specification, 368
  - string concatenation +, evaluation, 497
  - superclass access, evaluation, 438
  - type
    - vs. class of object, 73
    - comparison instanceof, evaluation, 504
    - how determined, 73
    - usage in, 65
  - unary
    - minus -, evaluation, 489
    - operators, evaluation, 487
    - plus +, evaluation, 489
    - values, variable as, 410
  - extends clause**
    - See also* classes; implements; object-oriented concepts
    - in class declaration, specifying direct superclasses with, 184
    - in interface declaration, 261
- ## F
- Feeley, Mark, 304**
- fields**
- access control
    - default, example, 142
    - private, example, 144
    - protected, example, 143
    - public, example, 143
  - access expressions, evaluation, 435
  - annotations, 281
  - are variables in memory model, 558
  - of an array, 125, 292
  - of a class
    - binary compatibility considerations, 345
    - declarations, 196
    - examples of, 205
  - final, 199
    - binary compatibility considerations, 347
    - volatile declaration not permitted for, 201
  - multiply inherited, example of, 207
  - non-static
    - default values, 71
    - explicit constructors not permitted to use, 244
    - hiding example, 206
    - initializers for, 202
    - initialization of, 201

**fields** (*continued*)

- invocation of initializers during instance creation, 323
- specification, 69
- re-inheritance of, example, 209
- static**
  - binary compatibility considerations, 349, 356
  - and **final**, binary compatibility considerations, 347
  - hiding of, example, 206
  - initialization of, 201, 239
  - initialization of, during the preparation phase of linking, 315
  - specification, 69
  - term definition and declaration, 198
- transient**, 199
  - binary compatibility considerations, 350
- volatile**, 199
- declarations**
  - binary compatibility considerations, 358
  - as declared entity, 113
- final**
  - semantics of, 575
- hidden**
  - accessing with
    - super** keyword, 438, 439
    - this** keyword, 367
- initializer**, 410
- of an interface**
  - ambiguous inherited, example, 265
  - binary compatibility considerations, 358
  - declarations, 264
    - examples, 265
  - initialization, 265
    - during the preparation phase of linking, 315
    - multiply inherited example, 266
    - public** by default, 264
  - names, naming conventions, 150
  - public**, by default in interface declaration, 263
  - references to, binary file format requirements, 336
  - references, active use, 318
  - shadowing, 119
  - write-protected, 578
- fifth dimension**, 433

**files**

- binary, format, 334
- systems, storing packages in, 153

**Finagle's Law**, 297**final modifier**

- See also* declarations; modifiers
- anonymous class is always implicitly, 429
- classes**
  - binary compatibility considerations, 341
  - declaration of, 178
- enum type is implicitly, 250
- exception parameter
  - may not be assigned, 398
- fields**
  - binary compatibility considerations, 347
  - declaration and use, 199, 264
  - final** by default in, interface declaration, 264
  - not allowed with **volatile**, 201
- formal parameters, 210
- methods**
  - binary compatibility considerations, 353
  - declaration and use, 217
  - not permitted in interface method declarations, 267
- reasons why constructors cannot be, 241

**finalization**

- See also* exceptions; linking
- enum instances may not undergo, 251
- finalizable, as object attribute, 326
- finalize** method, as member of **Object** class, 48
- finalized, as object attribute, 327
- finalizer method calls, unordered nature of, 329
- finalizer-reachable, as object attribute, 326
- finalizers**
  - may not be declared in enums, 251
- finally** clause, exception handling use, 303
- implementing, 326
- of instances, 325
  - implementation procedures, 326

**Float type**, 37

- See also* floating-point

**floating-point**

- See also* arithmetic; numbers; types
- algebraic identities, limits on use of, 417
- float** type, floating-point literal specification, 24, 35

## INDEX

- literals
    - largest and smallest, 26
    - term definition and specification, 26
  - operations, 40
  - required behavior, 41
  - types and values, term definition and specification, 37
  - flow analysis**
    - See also* security
    - conservative, required for definite assignment of local variables, 527
  - Foote, Samuel**, 396
  - for statement**
    - basic, 384
    - can complete normally, 404
    - definite assignment, 543
    - enhanced, 384
      - can complete normally, 404
      - full description, 387
    - ForInit* part
      - initialization by, 385
      - scope of local variable declared in, 118
    - header, local variable declaration in, 366
    - in scope of a local variable, 118, 385, 387
    - specification, 384
  - form feed**
    - escape sequence, 30
  - Forman, Ira**, 333
  - forward reference**
    - compile-time error in
      - initializers, 203
    - to types allowed before declaration, 119
  - FP-strict**
    - actual arguments, 212
    - and addition, 501
    - annotation element values are always, 274
    - and casts, 491
    - classes, 178
    - classes, interfaces, methods, constructors and initializers, 411
    - compile-time constant is always, 526
    - constructors, 241
    - and division, 494
    - expression
      - term definition, 411
    - interfaces, 261
    - methods, 218
      - overriding, 225
    - and multiplication, 492
    - and return statement, 393
    - value set conversion within, 92
  - Franklin, Benjamin**, 309
  - Frost, Robert**, 259, 402, 407
- ## G
- Gabriel, Richard P.**, 6
  - Geisel, Theodore**, 169
  - generic(s)**
    - annotation types cannot be, 271
    - class(es), 49
      - term definition, 178
    - constructor(s), 49
      - term definition, 242
    - declaration(s), 2
    - interface(s), 49, 189
    - method(s), 49, 445, 446, 448, 450
      - term definition, 220
    - method(s) or constructor(s), 127
  - getClass()**
    - treated specially, 48
  - Gilbert, W. S.**, 257
  - glb**, 90, 466
  - goal symbols**
    - CompilationUnit*, 157
      - syntactic grammar use, 10
    - Input*, lexical grammar use, 9
  - Goldberg, Adele**, 7
  - goto statement**
    - See* labeled statements
  - gradual underflow**
    - See also* exceptions; floating-point; IEEE 754 standard; numbers
    - support required for, 41
  - grammars**
    - See also* languages; lexical; semantics (chapter), 9
    - context-free, term definition, 9
    - difficulties with, as given in body of Java Language Specification, 585
    - if statement, dangling else handling, 368
    - lexical
      - (chapter), 13
      - term definition, 9
    - notation, 10
    - syntactic, term definition, 10

**H****Hammerstein, Oscar, II**, 171**Hansen, Christian Plesner**, 54, 92**happens-before**, 561

constructor completion, 325

**Harbison, Samuel**, 7**Harding, Warren G.**, 398**heap**

memory, 558

**heap pollution**, 67, 68, 95, 100, 410**hexadecimal**  *See also* numbers

base, permitted in integer literals, 22

numerals, specification, 22

**hiding**  *See also* scope

by class methods, 225

  of field declarations, in superclasses and  
  superinterfaces, 196

of fields, 174

bypassing using super, 439

term definition, 196

  hidden class methods, invocation of,  
  example, 234  of interface field declarations, in  
  superinterfaces, 264

of interface fields, term definition, 264

of interface member type declarations, 270

  of interface member types, term definition,  
  270

of interfaces, 259

of member types, term definition, 237

method

example, 231

impact on checked exceptions, 222

requirements, 225

of methods, 174

vs. overriding, example, 232

of types, by member types, 174

of variables

class, example of, 205

instance, example of, 206

**hierarchy**

exception, 306

**Higginson, Thomas Wentworth**, 264**Hoare, C. A. R.**, 1, 7**horizontal tab**

escape sequence, 30

**host**  and creation, storage and observability of  
  packages and compilation units, 155  
  may not restrict packages in a database, 167  
  and observable compilation units, 157  
  optional restriction on packages in a file  
  system, 167**hypothetical analysis**

of definite unassignment, 532

**I****identifiers**  *See also* fields; names; scope; variables  characteristics of those that are not names,  
  115

in type declaration specifier, 44

  in labeled statements, distinguished from  
  those in declarations, 116

term definition and specification, 19

**identity**

conversion

in assignment conversion context, 93

in casting conversion context, 101

    in method invocation conversion context,  
    99

specification, 80

**IEEE 754 standard**, 7  *See also* numbers

floating-point

conversion of numbers to, 25

types conformance to, 37

**if statements**  *See also* statements  dangling `else`, handling of, 369

definite assignment, 541

if-then statement, specification, 372

if-then-else statement, specification, 372

specification, 372

**Igarashi, Atsushi**, 54, 55, 92**implement**  *See also* classes; extends clause; interfaces

implements clause, 261

class declaration, specifying direct

superinterfaces with, 186

term definition, 124, 187

**implicit**

cast(s), 412

## INDEX

**import**

*See also* packages; scope  
 automatic, 165  
 declarations  
   example, 165  
   single-type, 161  
   term definition, 160  
   type-import-on-demand, 163  
 imported types  
   as declared entity, 114  
   scope of, 117  
 as part of a compilation unit, 157  
 single-static  
   and shadowing, 120, 164  
 single-type  
   may not declare top-level type in same  
     compilation unit, 168  
   names in, 130  
   scope, 117, 161  
   and shadowing, 119, 161  
 static  
   on-demand, 127, 137, 161, 444  
     and shadowing, 120, 165  
     names, 130  
   single, 127, 130, 137, 160, 444  
     scope, 117, 161  
 static-import-on-demand  
   scope, 117, 161  
 type-import-on-demand, 129  
   scope, 117, 161  
   and shadowing, 120, 163  
 type-on-demand, 160  
   names in, 130

**indexing**

of arrays, 289

**inexact results**

*See also* numbers  
 rounding behavior, 41

**infinity**

*See also* numbers  
 representation of, 26  
 signed, produced by floating-point overflow,  
 41

**inheritance**

*See also* object-oriented concepts; scope  
 in class  
   examples of, 192  
   of members, 190

of members, with `public`, `protected`,  
 and `private` access, examples, 193  
 with default access, example, 192  
 of fields  
   in class declarations  
     multiply inherited from interfaces,  
     example of, 207  
     reinheritance of, example of, 209  
   in interface declarations  
     ambiguous inherited, example, 265  
     multiply inherited, example, 266  
 of members, 123  
 of methods  
   in class declarations, 224  
   in interface declarations, 267  
 multiple  
   *See also* superinterfaces  
   example, 188  
 term definition, 123

**initialization**

*See also* control flow; linking  
 of arrays  
   in creation, to default value, 432, 69  
   using array initializers, 290  
 of classes, 316  
   detailed procedure, 319  
 of classes and interfaces, when it occurs, 317  
 complete  
   term definition, 573  
 detailed procedure, 319  
 of fields  
   in classes, 202  
   in interfaces, 265  
 of `for` statement, 384  
 of interfaces, 316  
   detailed procedure, 319  
 of classes  
   creating enum constants, 250  
   for variables, array, 290

**initializers**  
*See also* creation; execution  
 array  
   in array creation expression, 432  
 arrays, 289  
 executing, in `Test` class example  
   initialization, 311  
 for fields, 410  
   in class, 202

**initializers** (*continued*)

- in interface, 265

- instance

- See* instance initializer

- enum

- constant must not refer to itself, 252

- FP-strict, 411

- instance variable, 246

- lacking in a blank final, 71

- local variable

- in scope of a local variable, 118, 385, 387

- static, 201, 239, 410

- binary compatibility considerations, 356

- static

- FP-strict, 411

- variable

- FP-strict, 411

- in scope of local variable, 118, 364

- for variables

- class, 201

- during class or interface initialization, 201

- instance, 202

- during instance creation, 323

**inner class**

- See also* nested class

- anonymous class is always, 429

- of another class, 182

- direct, 182

- every local class is an, 361

- in a qualified class instance creation

- expression, 425

- and qualified **super** method invocation, 472

- qualified superclass constructor invocation of, 244

- as superclass of an anonymous class, 427

- term definition, 181

**input**

- See also* files

- elements, term definition, 17

- Input* goal symbol, lexical grammar use, 9

- tokens, 17

**instance initializer**, 174

- allows field access through **super**, 438

- anonymous class, 430

- as components of a class body, 189

- containing a **throw** statement, 394

- definite assignment of variables before, 538

- enum

- must not reference non-constant static fields, 252

- execution during constructor invocation, 246

- is not a member, 191

- may not contain a **return** statement, 392

- must be able to complete normally, 239

- overview, 174

- term definition, 238

- and unreachable statements, 402

- when reachable, 403

**instance(s)**

- See also* classes; interfaces; object-oriented concepts

- creation, 45, 322

- constructor

- invocation by creation expressions, 240

- parameter assignment during, 322

- use in, 240

- expression evaluation, 423

- order, 428

- expressions as statements, 371

- expressions cannot use wildcards, 56

- invocation of initializers for instance variables during, 323

- method dispatching during, 324

- enclosing

- determining immediate, 425

- with respect to superclass, 245

- with respect to superclass of an anonymous class, 426

- immediate as first argument to constructor, 428

- immediately, 182

- with respect to a class, 182

- n*th lexical used when evaluating method invocation, 473

- n*th lexically, 182, 422, 473

- restrictions on references to, 245

- finalization of, 325

- implementation procedures, 326

- initializer

- See* instance initializer

- instanceof** operator

- testing expression types with, 412

- testing expressions with, 412

## INDEX

- instantiation
  - preventing, with `private` constructors, 178, 248
- methods
  - See* methods, non-static
- variables
  - See* fields, class, non-static
- instanceof operator**
  - testing expression types with, 412
- instantiation**
  - term definition, 424
- integers**
  - See also* arithmetic; integral types; numbers
  - converting to boolean values, 43
  - literals
    - longest permitted, 23
    - term definition and specification, 21
  - operations, 36
- integral types**
  - See also* arithmetic; numbers; types
  - byte type, 35
  - char type, 35
  - int type, 35
  - long type, 35
  - short type, 35
  - values and operations, 35
- interfaces**
  - See also* class(es); fields; methods; packages; subclasses; superclasses; superinterfaces
  - abstract methods, 266
  - accessibility, 138
  - annotation(s), 281
  - as array element types, 288
  - binary compatibility considerations, 356
  - binary representation
    - binary file format requirements, 334
    - verification, 314
  - body, declarations, 263
  - and checked exceptions in initializers, 301
  - `Cloneable`
    - implemented by arrays, 292
  - declarations, 260
    - (chapter), 259
    - generic, 49
    - names in `extends` clause, 128
    - normal, 260
  - as declared entity, 113
  - dependent on a reference type
    - term definition, 262
  - directly dependent on a reference type
    - term definition, 262
  - fields
    - declarations, 264
    - examples, 265
  - inheritance
    - ambiguous, 265
    - multiply, 266
    - initialization, 265
  - FP-strict, 411
  - generic, 189
    - term definition, 261
  - initialization, 316
    - detailed procedure, 319
  - `java.io.Serializable`
    - implemented by arrays, 292
  - linking, process description, 314
  - loading, 312
    - process description, 313
  - members, 124
    - binary compatibility considerations, 357
    - declarations, 263
    - inheritance from superinterfaces, 124, 263
    - names, access to, 263
  - methods
    - declarations, examples, 269
    - overloading, 268
      - example, 269
    - overriding, 267
      - example, 269
  - names
    - fully qualified, 145
    - naming conventions, 146
  - normal, 113, 259
  - preparation of, 315
  - `public`
    - binary compatibility considerations, 356
    - declaration, 260
  - references to, binary file format
    - requirements, 338
  - in scope of imported type, 117, 161
  - `Serializable`, implemented by arrays, 292
  - superinterfaces
    - binary compatibility considerations, 341
    - of a class, 186
    - declaration of, 186

**interfaces** (*continued*)

- direct, 262
- that depend on themselves, 262
- unloading of, 330

**interruption**

- of threads, 582

**intersection types**

- See types, intersection

**invocation**

- alternate constructor, 244
- constructor
  - determining arguments to, 427
  - expression evaluation, 428
  - expression evaluation, order, 430
  - language constructs that result in, 240
- generic, 184
- of hidden class methods, example, 234
- method
  - conversion, 99
  - expression evaluation, 440
  - order, 473
  - how chosen, 411
- of a generic interface, 189
- superclass constructor, 244

**iteration**

- See also control structures
- continue statement, specification, 390
- do statement, specification, 382
- for statement, specification, 384
- while statement, specification, 380

**J****Java**

- digits, term definition, 19
- .java suffix, as name for source files, 156
- java package is always in scope, 160
- java.lang
  - example, 155
  - may be named in a type-import-on-demand, 163
  - public type names automatically imported from, 153, 158
- java.lang package
  - public type names automatically imported from, 165
  - public types defined in, list of, 165
- letters, term definition, 19

**Java programming language**

- See also grammars; languages; lexical; semantics; syntax

**java.io.Serializable interface**, 85, 98

- and array subtyping, 64

**java.lang.ref**, 328**Johnson, Samuel**, 13**K****Keats, John**, 295**Keene, Sonya E.**, 6**Kelvin, Lord (William Thompson)**, 409**Kernighan, Brian W.**, 7**keywords**

- default
  - in annotation types, 274
- list of, 21
- as token, 10

**Kiczales, Gregor**, 6**L****labeled statements**

- identifiers in, kept separate from those in declarations, 116
- specification, 370

**language**

- See also grammars; lexical; semantics; syntax
- Beta, 7
- C, 1, 2, 7
- C++, 1, 2, 7
- Common Lisp, 6
- Dylan, 6
- Mesa, 5, 7
- Modula-3, 3, 7
- POOL, 55
- Smalltalk-80, 7
- term definition, 9

**Lao-Tsu**, 113**left-hand side**

- term definition, 9

**length**

- of array, 291
- not part of type, 288

**letters**

- See also Unicode character set



## INDEX

- Java, term definition, 19
- lexical**
  - See also* semantics; syntax
  - grammar, term definition, 9
  - structure (chapter), 13
  - translations, steps involved in, 14
- Liang, Sheng**, 34, 313
- line terminators, term definition**, 16
- linefeed (LF)**
  - escape sequence, 30
  - handling in a
    - character literal, 27
    - string literal, 28
  - as a line terminator, not input character, 28
- linking**, 314
  - See also* exceptions; execution; initialization; loading; run-time
  - classes, process description, 314
  - errors, 307
  - interfaces, process description, 314
  - in `Test` class example, at virtual machine startup, 310
- literals**
  - See also* constants; fields; variables
  - boolean, term definition and specification, 26
  - character
    - escape sequences for, 30
    - term definition and specification, 26
  - class, 283
  - evaluation of, 420
  - floating-point
    - largest and smallest permitted, 26
    - term definition and specification, 24
  - hexadecimal, 23
  - integer
    - largest and smallest permitted, 23
    - term definition and specification, 22
  - null, term definition and specification, 30
  - string
    - escape sequences for, 30
    - term definition and specification, 28
    - term definition and specification, 21
    - as token, 10
- loading**
  - See also* `ClassLoader` class; execution; linking
  - classes, 312
  - errors, 307
  - interfaces, 312
  - process, 313
  - in `Test` class example, at virtual machine startup, 310
- local class**, 150, 173
  - declaration is part of a block, 361
  - and definite [un]assignment, 539
  - determining immediately enclosing instance when instantiating, 426
  - example(s), 182, 362
  - interaction with shadowing of parameters, 212
  - requirements for normal completion, 403
  - scope, 118, 362
  - as superclass of anonymous class being instantiated, 427
  - term definition, 361
- local variables**
  - See also* scope; variables
  - declarations
    - definite assignment, 539
    - statements, 363
  - declarators and types, 364
  - definite assignment, required for, 527
  - initial value, 72
  - naming conventions, 151
  - restrictions on use in an inner class, 182
  - scope, 118, 364
  - shadowing, 119
  - specification, 70
- locks**
  - See also* monitors; synchronization; threads (chapter), 553
  - acquisition and release
    - by synchronized networks, 218
    - by synchronized statement, 395
- long type**, 35
  - See also* integral types; numbers
  - integer literal specification, 24
  - value range, 35
  - variables
    - non-atomic treatment, 579
- lub**, 511
- lvalue**
  - See* variables, value of expression

## M

Madsen, Ole Lehrmann, 7

**magnitude**

loss of information about, possibility of in narrowing primitive conversion, 82

**Marx, Chico**, 535, 545

**Marx, Groucho**, 190, 307, 535, 536, 538

**mathematical functions**

*See* arithmetic; numbers

**Matthew, St.**, 287

**Maybury, William**, 7

**member**, 237

**member class**, 114, 146, 173, 174

*See also* member type

can have same name as a method, 210

canonical name of, 146

determining immediately enclosing instance when instantiating, 426

example(s), 181, 183, 245, 247, 362

fully qualified name of, 145

and `public` modifier, 175

as superclass of anonymous class being instantiated, 427

term definition, 237

**member interface**, 114, 145, 146, 174, 237

*See also* member type

can have same name as a method, 210

canonical name of, 146

example(s), 181

fully qualified name of, 145

implicitly `static`, 181, 238

inner class may not declare, 181

term definition, 237

**member type**, 130, 132, 237, 259

*See also* nested type

binary name of, 335

and definite assignment, 549

in interfaces, 270

inherited by interfaces, 261

and the meaning of a qualified type name, 132

**members**

*See also* class(es); fields; interfaces; methods

accessibility, 139

arrays, 125, 292

classes, 123, 190

binary compatibility considerations, 343

declarations, 189

inaccessible, accessing, example, 194

inheritance, examples, 192

private, inheritance examples, 193

inheritance and, 123

interfaces, 124, 263

binary compatibility considerations, 357

declarations, 263

inheritance, 124, 263

names, access to, 263

Object class, 48

of a package, 122, 153

of a type variable, 50

of a wildcard, 55

of an enum, 251

of an intersection type, 62

of parameterized types, 55

of a package, 122, 154

protected accessibility of, 139

reference type, as declared entity, 113

static, restricted in inner classes, 181

of superclasses, accessing with `super`, 438

term definition, 122

type, 44

**memory**

chain, 575

exceptions

`OutOfMemoryError`, 321, 322

heap, 558

`OutOfMemoryError`, 222

shared, 558

**memory model**

and finalization, 328

causality requirements of, 569

hardware, 553

Java programming language, 553

term definition, 557

**Mesa**, 5, 7

**meta-annotation(s)**

*See* annotation(s), meta

**method(s)**, 209

abstract

in enum constant, 176, 250

abstract

binary compatibility considerations, 352

in classes, 214

in interfaces, 266

overloading, 268

access control

default, example, 142

private, example, 144

protected, example, 143

public, example, 143

## INDEX

- access, binary compatibility considerations, 344
- accessible, and method invocation, 442
- annotation(s), 281
- applicable, 450
- arity
  - fixed
    - see method(s), fixed arity, 443
  - variable
    - see method(s), variable arity, 442
- of array, 125, 292
- body, 223
  - binary compatibility considerations, 354
- class
  - See* methods, *static*
- classes, 209
  - abstract, 214
  - binary compatibility considerations, 350
  - body
    - declarations, 223
    - with no executable code, 223
  - constructors compared with, 240
  - declarations, 209
    - examples of, 230
  - final, 217
  - hidden, invocation of, 234
  - hiding by, 225
  - modifiers, 214
  - native
    - declarations, 218
    - semicolon as body of, 223
  - private, as implicitly final, 217
  - signature, 212
  - strictfp, declaration, 218
  - synchronized, declaration, 218
  - void, expression return statement not permitted in, 223
  - with same signature, inheriting, 228
- declarations, 209
  - binary compatibility considerations, 350
  - generic, 49
- as declared entity, 113
- definite [un]assignment of blank final fields before, 538
- definite assignment of variables before, 538
- descriptor, 442
- dispatching during instance creation, 324
- enum
  - implicitly declared, 251
- equals, as member of Object class, 48
- final, 217
  - binary compatibility considerations, 353
- finalize, as member of Object class, 48
- fixed arity
  - evaluating arguments of, 474
  - term definition, 212
  - when more specific, 448
- FP-strict, 411
- generic, 127, 220, 446
  - in an interface, 267
  - term definition, 220
- getClass, as member of Object class, 48
- hashCode, as member of Object class, 48
- hiding of, 225
- inheritance
  - in arrays, 125, 292
  - in classes, 224
  - in interfaces, 124, 263
- instance
  - See* method(s), *non-static*
- of interfaces
  - declarations, 266, 269
  - overloading, 268, 269
  - overriding, 267, 269
  - semicolon as body of, 223
- invocation
  - See also* access control
  - conversion, 99
  - expression evaluation, 440
  - order, 440
  - expressions as statements, 371
  - how chosen, 411
  - qualifying type of
    - See* qualifying type, of a method invocation
- as members of a class, 123
- method table creation during preparation
  - phase of linking, 310
- most-specific, 447
- names
  - context in which a name is classified as, 127
  - naming conventions, 149
  - qualified, meaning of, 137
  - simple, meaning of, 137
- native, 218
  - binary compatibility considerations, 354
- non-static
  - instance method as name for, 216
  - overriding by, 224

**method(s)** (*continued*)

- `notify`, as member of `Object` class, 48
- `notifyAll`, as member of `Object` class, 48
- overloading, 229
  - binary compatibility considerations, 355
  - example, 231
- overriding, 225
  - binary compatibility considerations, 356
  - examples, 230, 231, 232, 234, 269
- parameters
  - See also* arguments; variables
  - binary compatibility considerations, 352
  - definite [un]assignment of, 547
  - description, 69
  - formal
    - unaffected by memory model, 558
  - initial value, 72
  - is assigned when `final`, 211
  - must not be assigned if `final`, 211
  - shadowing, 119, 212
  - specification of, 210
- `private`, in example of qualified names and access control, 144
- `public`, interfaces, declarations, 267
- references to, binary file format requirements, 336
- result type, binary compatibility considerations, 352
- scope of formal parameters, 118, 212
- shadowing, 119
- signatures, 212
- `static`, binary compatibility considerations, 354
- `stop`, as asynchronous exception cause, 298
- `synchronized`, 554
- `synchronized`, 218
  - binary compatibility considerations, 354
- throws clause, binary compatibility considerations, 354
- `toString`, as member of `Object` class, 48
- `uncaughtException`
  - exception handling use, 303
  - when invoked, 298
- and unreachable statements, 402
- variable arity, 442, 443, 446
  - evaluating arguments of, 474
  - term definition, 212
  - when more specific, 448

- virtual
  - invocations
    - semantics not effected by data races, 563
- `void`
  - relationship with conditional expressions, 511
- `wait`
  - as member of `Object` class, 48, 580
  - when reachable, 403

**Mitchell, James G.**, 7**modifiers**

- See also* declarations
- class, 175
- constructor, 241
- declaration
  - See*
    - `abstract` modifier;
    - `final` modifier;
    - `private` modifier;
    - `protected` modifier;
    - `public` modifier;
    - `static` modifier;
    - `strictfp` modifier;
    - `synchronized` modifier;
    - `transient` modifier;
- field, 197
- interface, 260
- method, 214

**Modula-3**, 3, 7**Molière**, 9**Møller-Pedersen, Birger**, 7**monitor**, 554

- lock action on, 558

**Montaigne, Michael de**, 209**Moon, David A.**, 6**Morlay, Bernard de**, 152**N****named class**

- exceptions in instance initializer, 202, 238, 301

**named type**, 260**names**

- See also* identifiers; scope; Unicode character set
- (chapter), 113

## INDEX

- ambiguous
    - handling of, 126, 129
    - reclassification of, 129
  - binary
    - implementation keeps track of types using, 167
    - term definition, 335
  - canonical, 114
    - is binary name of top-level type, 335
    - and single-static imports, 164
    - and single-type imports, 161
    - and static-import-on-demand, 165
    - term definition, 146
    - and type-import-on-demand, 163
  - class
    - naming conventions, 147
  - conflicts
    - See also* hiding
    - avoiding
      - in fields with multiple inheritance, 207
      - through context use, 127
      - through naming conventions, 146
  - constants, naming conventions, 150
  - conventions, 146
    - impact on name obscuring, 122
  - expression
    - context in which a name is classified as, 130
    - qualified, meaning of, 135
    - simple, meaning of, 134
  - fields, naming conventions, 150
  - fully qualified
    - of a top-level type, 166
  - interface
    - member, access to, 263
    - naming conventions, 147
  - meaning of, determining, 126
  - method, 210
    - context in which a name is classified as, 129
    - naming conventions, 149
    - qualified, meaning of, 137
    - simple, meaning of, 137
  - package
    - in compilation units, 157
    - context in which a name is classified as, 130
    - limited significance for access control, 154
    - naming conventions, 147
    - qualified, meaning of, 132
    - scope, 160
    - simple, meaning of, 131
    - unique, 169
      - importance of, 170
  - parameters, naming conventions, 151
  - qualified
    - access control and, 138
    - fully, 145
    - term definition, 113
  - resolving references to, during resolution
    - phase of linking, 310
  - restrictions, types and subpackages, 154
  - rules on being the same
    - class members, 123
    - fields, 123
    - interface fields, 125
  - simple, 44, 335
    - class cannot have same as enclosing class, 175
    - interface cannot have same as enclosing class, 260, 272
    - of an element, 137, 283
    - term definition, 115
  - as subset of identifiers, 115
  - syntactic
    - categories, 126
    - classification of according to context, 127
  - term definition, 115
  - type
    - qualified, meaning of, 132
    - simple, meaning of, 132
  - variables, local, naming conventions, 151
- NaN (Not-a-Number)**
- See also* numbers
  - and addition, 500
  - comparison results, 41
  - and division, 493
  - and multiplication, 492
  - predefined constants representing, 26
  - preventing use of algebraic identities in
    - expression evaluation, 417
  - term definition and specification, 37
- narrowing**
- See also* conversion; numbers
  - primitive conversions
    - in assignment conversion context, 94

**narrowing** (*continued*)

- in casting conversion context, 101
- not allowed in method invocation
  - conversion context, reasons for, 100
- reference conversions, specification, 85

**native modifier**

*See also* declarations; modifiers

## methods

- binary compatibility considerations, 354
- declaration, 218
- semicolon as body of, 223

reasons why constructors cannot be, 241

**nested class**, 173, 174

*See also* nested type

- and definite [un]assignment of variables, 538
- inner class is a, 181
- interaction with shadowing of parameters, 212
- local class is a, 361
- may declare `static` members if non-inner, 181
- name hides same name in enclosing scopes, 190

term definition, 173

**nested interface**, 174, 259

*See also* nested type, 174

term definition, 259

**nested type**

## declaration

- in scope of a class member, 118, 190

## declarations

- included in scope of interface members, 118, 263

- in scope of an interface member, 118, 263

## enum

- implicitly static, 250

**Newton, Sir Isaac**, 1**non-public**

- classes, in example of qualified names and access control, 141

**nonterminal symbols**

*See also* grammars

- definition of, notation for, 10
- term definition, 9

**NoSuchFieldError**

- and enum binary compatibility, 356

**notation**

- See also* names, naming conventions
- grammar, 10

**null**, 283

- literal, term definition and specification, 30
- qualifying a superclass constructor
  - invocation, 245

## type

- in conditional expression, 511
- literal as source code representation of the value of, 21
- term definition and specification, 34

**NullPointerException**, 95**numbers**

*See also* arithmetic; precision; types

## conversions of and to

*See* conversions

## errors in handling

*See* exceptions

## manipulating

*See* arithmetic operators

## numeric promotion

- binary, 110
- specification, 108
- term definition, 78

## unary, 108

## primitive types

*See*

byte type;

char type;

double type;

float type;

floating-point;

int type;

integers;

integral type;

long type;

short type

## related types

*See* boolean; characters; strings

**Nygaard, Kristen**, 7**O****Object class**, 47, 98, 125, 263, 273, 293, 326, 466

- and array subtyping, 64

- cannot have `extends` clause, 184

**object-oriented concepts**

- See* class(es); encapsulation; fields; inheritance; method(s); objects

## INDEX

**objects**

- Class, array, 293
- fields that reference, final modifier effect
  - on changes to, 71, 199
- finalization attributes, 326
- operations on references to, 46
- reachable, finalization implication, 326
- reference
  - See references
- state
  - impact of changes, 46
  - transient fields not part of, 199
- term definition and specification, 45
- unreachable, finalization implications, 327

**obscured**

- declaration, term definition, 122
- label, 117
- by labels, 370
- of labels, 370

**observable**

- all compilation units in java and java.lang
  - are, 158
- compilation unit
  - determined by host, 155
  - term definition, 157
- package
  - and meaning of qualified package name, 132
  - term definition, 160

**Occam, William of, 492****octal**

- See also numbers
- base, permitted in integer literals, 22
- numerals, specification, 23

**one of phrase**

- grammar notation use, 12

**operators**

- See also constructors; expressions;
  - initialization; methods; numbers
- arithmetic, numeric promotion, specification, 108
- assignment, as only way to change the value
  - of a primitive variable, 35
- binary, evaluation order, 414
- boolean, 43
- compound assignment, evaluation order, 415
- floating-point, 40
  - signed infinity produced by overflow, 41
  - signed zero produced by underflow, 41

**integer**

- divide, divide by zero exception thrown by, 37
  - overflow not indicated by, 36
  - underflow not indicated by, 36
- integral, 36
- list of, 31
  - operands evaluated before, 416
  - precedence, in expression evaluation, 417
  - on reference types, 46
  - remainder, divide by zero exception thrown
    - by, 37
  - string concatenation
    - constructor invocation by, 240
    - creating new instances with, 45

**opt subscript**

- grammar notation use, 10

**optimization**

- during
  - preparation phase of linking, 315
  - resolution phase of linking, 315
- final method inlining, 217
- finalization, 326
  - reachable objects reduction, 326
- linking
  - initialization phase of, 321
  - resolution strategies, 314

**optional symbol**

- term definition, 10

**order**

- evaluation
  - argument lists left-to-right, 418
  - binary operators, 415
  - compound assignment, 415
  - left-hand operand first, 415
  - operands evaluated before operations, 416
  - of other expressions, 419
  - parentheses and precedence respected, 417
- execution, 560
- field initialization, at run-time, 265
- finalizer invocations, implications of no
  - ordering imposition, 329
- of floating-point values, 39
- happens-before, 561
- of occurrence, of initialization of types, 316
- partial, 568, 575
  - restrictions of, 555
- program, 558, 560, 561, 562, 567
- synchronization, 561, 567
- total, 560, 567

**origin**

of arrays, 290

**OutOfMemoryError**, 95, 314, 430

**output**

*See* files

**overflow**

*See also* arithmetic; exceptions; numbers  
floating-point operators, signed infinity  
produced by, 41

not indicated by integer operators, 36

**overloading**

*See also* object-oriented concepts;  
inheritance; methods

of abstract interface methods

declarations, 268

example, 269

of constructors, 246

binary compatibility considerations, 355

of methods, 229

binary compatibility considerations, 355

example, 231

term definition, 124, 229

**Override annotation**, 279**overriding**

*See also* object-oriented concepts; scope  
examples, 231, 232, 234, 269

incorrect, 231

incorrect because of throws, example, 236

large example, 234

methods

abstract

in abstract classes, 215

of non-abstract instance methods, 215

binary compatibility considerations, 356

in interfaces, 267

example, 269

instance, 224

requirements in, 225

term definition, 124

**P****packages**

(chapter), 153

and overriding, 224

annotation(s), 281, 410

binary compatibility considerations, 340

canonical name of, 146

declarations

in compilation units, 158

shadowing, 160

as declared entity, 113

host support for, 155

importing all public types from a, 163

members of, 122, 154

named, in compilation units, 158

names

context in which a name is classified as,  
127

fully qualified, 145

naming conventions, 147

qualified, meaning of, 132

simple, meaning of, 131

unique, importance of, 169

observable, 160

scope of, 117

shadowing, 119

storing in a

database, 157

file system, 155

term definition, 154

unnamed

compilation units belonging to, 157

in compilation units, uses and cautions,  
159

and fully qualified name of top-level type,  
166

**Paine, Thomas**, 396

**parameterized type**, 57

**parameters**

*See also* modifiers; variable

annotation(s), 281

constructor, 240

assignment during instance creation, 322

binary compatibility considerations, 352

description, 69

initial value, 72

as declared entity, 113

definite [un]assignment of, 547

exception

description, 70

initial value, 72

scope, 118, 397

exception handler

unaffected by memory model, 558



## INDEX

- method
  - binary compatibility considerations, 352
  - description, 69
  - initial value, 72
  - specification of, 210
- must not be assigned if `final`, 211
- names, naming conventions, 151
- scope, 118, 212
- shadowing, 119, 212
- to an external action, 559
- type
  - formal, 89, 240
  - of a constructor, 242
- Partridge, Eric**, 173
- Pavic, Milorad**, xxvii
- Peirce, Charles**, 527
- Peretti, Hugo E.**, 32
- performance**
  - See* optimization
- Perlis, Alan**, xxv
- Pierce, Benjamin**, 91
- platform-dependent**
  - native methods are, 218
- pointers**
  - See* references
- polling**
  - for asynchronous exceptions, 304
- precedence**
  - See also* expressions
  - of operators, in expression evaluation, 417
- precise**
  - term definition, 303
- precision**
  - See also* numbers
  - possible loss of
    - in narrowing primitive conversions, 82
    - in widening primitive conversions, 81
- preparation**
  - of classes, 315
  - of interfaces, 315
  - in `Test` class example, at virtual machine startup, 310
- preventing**
  - instantiation, 248
    - with `private` constructors, 178
- primitive**
  - See also* numbers
  - conversion
    - narrowing
      - in assignment conversion context, 94
      - in casting conversion context, 101
      - not allowed in method invocation conversion context, reasons for, 100
    - widening
      - in assignment conversion context, 93
      - in binary numeric promotion context, 110
      - in casting conversion context, 101
      - in method invocation conversion context, 99
      - in unary numeric promotion context, 108
  - types
    - as array element types, 288
    - changing value of variables, 35
    - fully qualified name for, 145
    - literal as source code representation of the value of, 21
    - term definition and specification, 34
    - variables of, specification, 67
- private modifier**
  - See also* declarations; modifiers
  - access
    - determining, 139
    - inheritance of class members, example, 193
  - cannot be used for local classes, 362
  - class, pertains only to member, 175
  - constructors
    - access control, example, 144
    - preventing instantiation with, 178
  - default constructor of an enum is, 250
  - fields, access control, example, 144
  - inheritance with, example, 193
  - and interfaces, 260
  - members not inherited, 190
  - methods
    - abstract not permitted with, 214
    - access control, example, 144
    - in example of qualified names and access control, 144
    - as implicitly final, 217
    - overriding and hiding not possible with, 228
  - preventing instantiation by declaring constructors as, 248
  - top-level type may not use, 169

**productions**

term definition, 9

**promotion**

*See also* numbers

(chapter), 77

binary, 486, 488, 491, 500, 502, 503, 506,  
508, 511

numeric, 108

binary, 110

specification, 110

specification, 108

term definition, 78

unary, 108

specification, 108

unary, 489, 490, 502

in array index, 432, 482

**protected modifier**

*See also* declarations; modifiers

access, 139

cannot be use for local classes, 362

class, pertains only to member, 175

constructors, access control, example, 143

fields, access control, example, 143

inheritance with, example, 193

and interfaces, 260

methods, access control, example, 143

top-level type may not use, 169

**provably distinct**

parameterized types, 52

type arguments, 54

**public modifier**

*See also* declarations; modifiers

access, 138

inheritance of class members, example,  
193

cannot be used for local classes, 362

classes

access control in, 138

binary compatibility considerations, 341

in example of qualified names and access  
control, 141

pertains only to top-level and member, 175

constructors, access control, example, 143

fields

access control, example, 143

interface, implicit in declaration, 264

implicit for interface member types, 270

inheritance with, example, 193

interfaces

binary compatibility considerations, 356

declarations, 260

methods

access control, example, 143

in interfaces, implicit in declaration, 267

superclasses, accessing instances of non-  
public subclasses through, 194

superinterfaces, accessing instances of non-  
public subclasses through, 194

**Q****qualified**

*See also* access; scope

access, term definition, 138

class instance creation, 424

and definite [un]assignment, 537

class instance creation expression, 116

protected accessibility of constructor,  
140

protected accessibility of superclass  
constructor, 140

names

access control and, 138

expression, meaning of, 135

fully, 145

method, meaning of, 137

not permitted with parameters, 212

package, meaning of, 132

term definition, 113

type, meaning of, 132

super, 244

superclass constructor invocation, 244

and protected accessibility, 140

**qualifying type**

of a constructor invocation

term definition, 337

of a field reference

term definition, 336

of a method invocation

as compile-time information stored for use  
at run-time, 472

term definition, 336

**Quayle, J. Danforth**, 298

## INDEX

**R****race(s)**

- data, 556, 560, 573
- and happens-before relation, 562
- term definition, 563

**Raper, Larry**, 333**raw type**

- term definition, 57

**reachable**

- See also* scope
- objects
  - finalization implications, 326
- term definition, 326

**recursive**

- term definition, 10

**redundant**

- type-import-on-demand, 163

**references**

- See also* expressions; types
- conversions
  - narrowing, 85
  - widening, 85
    - in assignment conversion context, 93
    - in casting conversion context, 101
    - in method invocation conversion context, 99
- to fields, active use, 318
- forward
  - permitted in
    - class type usage, 169
    - interface type usage, 260
- object, value of array variables as, 288
- operations, 46
- symbolic
  - binary file format requirements, 335
  - resolution, 315
- term definition, 45
- types
  - array, 288
  - class
    - See* class
  - criteria for determining when two are the same, 49
  - interface
    - See* interface
  - member name, scope of, 118
  - member, as declared entity, 113

- specification, 44
  - in class declaration, 175
  - in interface declaration, 259
- variables, specification, 67

**reflection**, 317**reifiable**

- array component type must be, 291
- type
  - term definition, 56

**reifiable type**

- term definition, 56

**release**

- term definition, 561

**release-to-release compatibility**

- See* binary, compatibility

**representation**

- binary, classes and interfaces, verification of, 314

**resolution**

- late, in class and interface linking, 314
- lazy, 311
- name conflicts
  - avoiding with multiply inherited fields, 207
  - avoiding with multiply inherited types, 237, 270
  - context use in avoiding, 127
- static, in class and interface linking, 314
- symbolic references, 315
- in Test class example, at virtual machine startup, 310

**return statement**

- definite assignment, 545
- instance initializer, 239
- as reason for abrupt completion, 360
- specification, 392

**return-type-substitutable**, 268

- term definition, 220

**reuse, software**

- See* binary, compatibility

**right-hand side**

- term definition, 9

**Ritchie, Dennis M.**, 7**Robson, David**, 7**rounding**

- IEEE 754 standard default mode, support required for, 41
- round to nearest, term definition, 41
- round toward zero, term definition, 41

**run-time**

- checks of expressions, 411
- errors
  - See* exceptions
- RuntimeException class
  - unchecked exceptions found in, 222
- state, linking binary objects into, 314
- type, 73
- validity checks, casting conversion
  - requirements, 104

**Runtime class, 331****S****scope**

- See also* declarations; inheritance; names; object-oriented concepts
- of declarations, 117
  - formal parameter, 212, 118, 212
  - of a local class, 118, 362
  - of local variable declared by `for` statement, 118, 385, 387
  - member, 118, 190, 263
  - overview, 113
  - package, 160
  - parameters
    - formal, 212
  - shadowing rules, 119
  - of a top-level type, 117, 166
  - types declared by an `import` declaration, 117, 161
- exception parameters, 397
- in, term definition, 117
- initializer execution relative to issues of, 202, 203, 239
- of a local class, 362
- in method invocation expressions, 440
- of names
  - of exception handler parameters, 118, 397
  - local variables, 70, 118, 364
  - parameters
    - formal, 118, 212
- of a class' type parameter, 118, 179, 220, 242, 261
- of names
  - nested classes, variables and definite [un]assignment, 538

- term definition, 117
- type initialization order restrictions, 317

**Sebastian, John, 372****security**

- See also* access control; exceptions; types; virtual machine
- internal symbolic reference table, role in maintaining, 312
- type systems, verification importance to, 342

**SecurityManager class, 313****semantics**

- See also* grammars; lexical; syntax
- errors in, exceptions as mechanism for signaling at run-time, 297
- expressions, (chapter), 409
- names
  - context role, 127
  - determining, 126
  - expressions, 134
  - methods, 137
  - package, 131
  - types, 132
- verifying during linking process, 314

**semicolon (;)**

- as method body, 223

**separators**

- list of, 31
- as token, 10

**sequences**

- escape, for character and string literals, 30

**Seuss, Dr., 169****shadowed, 444****shadowing**

- absence of by statement labels, 370
- of declarations, 119
  - by package declarations, 160
- of exception parameters, 397
- of labels, 370
- of local classes, 362
- by local variables, 367
- of local variables, 365
- by member types, 237
- of members by local variables, 366
- of methods, 119
- by a nested class, 190
- package, 160
- of packages, 119
- parameters, 212

## INDEX

- by single-static import, 120, 164
- by single-type import, 119, 161
- by static-import-on-demand, 120, 165
- by type-import-on-demand, 120, 163
- of types, 119
- of variables, 119
- Shakespeare, William**, 32, 77, 152, 331
- Shelley, Percy Bysshe**, 495
- short type**
  - See also* integral types; numbers
  - value range, 35
- side effects**
  - See also* expressions
  - from expression evaluation, reasons for, 409
- signature(s)**
  - of constructors, 241
  - of methods, 212, 269, 352, 442
  - methods with same, inheriting, 228
  - override-equivalent, 124, 212, 267, 273, 449
  - term definition, 214
- simple names**
  - See also* identifiers; names
  - expression, meaning of, 134
  - method, meaning of, 137
  - package, meaning of, 131
  - term definition, 115
  - type, meaning of, 132
- single quote(')**
  - escape sequence, 30
- sleep**
  - thread, 583
- Smalltalk-80**, 7
- sorting**
  - Unicode, reference for details on, 20
- source code**
  - compatibility, compared with binary compatibility, 340
  - transformations that preserve binary compatibility, 335
- Southey, Robert**, 490
- square brackets ([])**
  - in array
    - type declaration, 288
    - variable declaration, 289
- stack trace**
  - of an exception, 401
- startup**
  - virtual machine, 309
- state**
  - not shared by primitive values, 35
- object**
  - impact of changes, 46
  - transient fields not part of persistent, 199
- statements**
  - See also* constructors; control flow; expressions; initializers; methods
  - (chapter), 359
- assert**
  - specification, 373
- break**
  - definite assignment, 545
  - specification, 388
- completion of
  - abrupt, reasons for, 360
  - normal, reasons for, 360
- continue**
  - definite assignment, 545
  - specification, 390
- definite assignment, 538
- do**
  - definite assignment, 543
  - specification, 382
- empty**
  - definite assignment, 538
  - specification, 370
- expression**
  - definite assignment, 540
  - specification, 371
- for**
  - definite assignment, 543
  - specification, 384
- general specification, 368
- if**
  - dangling else handling, 369
  - definite assignment, 541
- if-then, specification, 372
- if-then-else, specification, 372
- labeled
  - definite assignment, 540
  - specification, 370
- local variable declarations
  - execution of, 367
  - specification, 364
- return**
  - definite assignment, 545
  - specification, 392

**statements** (*continued*)

- switch
  - definite assignment, 541
  - specification, 377
- synchronized
  - definite assignment, 545
  - specification, 395
- throw
  - definite assignment, 545
  - specification, 393
- try
  - definite assignment, 545
- try-catch, specification, 398
- try-catch-finally, specification, 399
- unreachable, conservative flow analysis
  - detection of, 402
- while
  - definite assignment, 542
  - specification, 380

**static**

- restrictions in inner classes, 181

**static context**, 181, 471

- and method invocation, 472
- and qualified superclass constructor
  - invocation, 245

**static initializer**

- definite [un]assignment within, 549
- definite assignment of variables before, 538
- inner class may not declare, 181
- may not contain a return statement, 392
- must assign all blank final class variables, 199
- not member, 191
- overview, 174
- and unreachable statements, 402
- when reachable, 403
- and checked exceptions, 301

**static modifier**

- See also* declarations; modifiers
- anonymous class is never, 429
- class, pertains only to member, 175
- for fields, 198
  - declaring class variables using, 198
  - cannot be used for local classes, 362
  - initialization of, 201
  - binary compatibility considerations, 349
  - creating during the preparation phase of linking, 315

- default initialization of during the preparation phase of linking, 315
- implicit in interface member declarations, 264

- initialization of, 239
- initializers, 239

## for methods

- declaring class methods with, 216
- hiding by, 225
- not permitted in interface method declarations, 267

## implicit for interface member types, 270

## initializers

- as components of a class body, 189
- and interfaces, 260
- on nested enums, 250
- reason why constructors do not use, 241
- top-level type may not use, 169

**Stein, Gertrude**, 152, 489, 585**store**

- array, exception, 294

**storing**

- packages
  - in a database, 157
  - in a file system, 245

**strictfp modifier**

- constructors cannot be declared, 241
- example(s), 418
- for classes, 175
  - semantics, 178
- and FP-strict expressions, 411
- has no effect on method override, 225
- interfaces, 260
  - methods may not be, 267
  - semantics, 261
- methods, 214
  - semantics, 218
- and widening conversion, 81

**string(s)**

- See also* characters; numbers; primitive types
- allowed in annotation type, 273
- character arrays are not the same as, 294
- concatenation operator (+)
  - constructor invocation by, 240
  - creating new instances with, 45
- conversion
  - context specification, 101
  - specification, 92

## INDEX

- literals
  - escape sequences for, 30
  - interning of, 29
  - term definition and specification, 28
- String class, 283
  - creating instances, with string concatenation operator, 46
  - literal as source code representation of the value of, 21
  - specification, 48
  - string literal as reference to an instance of, 28
- Stroustrup, Bjarne**, 6
- subclasses**
  - See also* class(es); interfaces; superclasses; superinterfaces
  - declaration, 184
  - direct, extending classes with, 184
  - relationship to direct subclass, 185
- subpackage**, 153
  - as package member, 122, 154
  - and package observability, 160
- subsignature**, 224, 225, 268
  - term definition, 213
- subtype**
  - class, of two different parameterizations of the same interface, 189
- subtyping**
  - in determining applicability, 442
  - in determining applicability, 445
- sufficient**
  - set of synchronization edges
    - term definition, 562
- super keyword**
  - accessing
    - overridden methods with, 225
    - superclass members with, in expression evaluation, 438
  - binary compatibility considerations, 341
  - not permitted in
    - class variable initialization, 202
    - explicit constructor invocation, 243
    - interface field declarations, 265
  - permitted
    - in instance initializers, 239
  - permitted in instance variable initialization, 202
- superclass(es)**
  - See also* classes; fields; interfaces; methods; packages; subclasses; superinterfaces
  - accessing fields with `super`, expression evaluation, 438
  - of array is `Object`, 293
  - binary compatibility considerations, 341
  - direct
    - declaration, with `extends` clause, 184
    - of an enum, 251
    - term definition, 184
  - extending classes with, 184
  - `public`, accessing instances of non-public subclasses through, 194
- superinterface(s)**
  - See also* classes; fields; interfaces; methods; packages; subclasses
  - binary compatibility considerations, 341, 357
  - direct
    - declaration, with `implements` clause, 186
    - term definition, 187
  - of interface, declaration, 261
  - `public`, accessing instances of non-public subclasses through, 194
  - term definition, 186
- supertype**, 63
- SuppressWarnings**, 89, 280
- Sweet, Richard**, 7
- switch**
  - block, term definition, 377
- switch statement**
  - See also* statements
  - definite assignment, 541
  - local variable declaration specification, impact on, 70
  - specification, 377
- symbolic references**
  - binary file format requirements, 336
  - resolution, 315
- symbols**
  - See also* identifiers; name; variables
  - goal
    - `CompilationUnit`, 10
    - `Input`, 9
    - term definition, 9
  - nonterminal
    - notation for definition of, 10
    - term definition, 9
  - optional, term definition, 10
  - terminal, term definition, 9
- synchronization**, 554
  - See also* locks; threads

**synchronization** (*continued*)

- correct, 563
- exception mechanism integration with, 298
- initialization implications of, 319
- locks
  - use by synchronized methods, 218
  - use by synchronized statement, 395
- order, 561
- sufficient
  - set of edges, 562
- synchronized modifier
  - See also* modifiers
  - in method declarations, 218
  - methods, binary compatibility considerations, 354
  - reason why constructors do not use, 241
- synchronized statement, 554
  - definite assignment, 545
  - specification, 395
- volatile fields used for, with threads, 199

**synchronized-with**

- term definition, 561

**syntactic**

- See also* lexical; semantics
- classification, of a name according to context, 127

**System class**, 331

- err field is write protected, 578
- in field is write protected, 578
- out field is write protected, 578

**T****term definition**

- abrupt completion, 360, 414
- access, 113
  - accessible, 138
  - control, 138
  - qualified, 138
- acquire, 561
- activation frame, 477
- alphabet, 9
- ambiguous member, 265
- annotation, 281
  - type
    - element, 273
- annotation type, 270
- anonymous class, 424

- applicable, 442
- applicable by method invocation conversion, 446
- applicable by subtyping, 445
- applicable variable-arity method, 447
- array, 287
  - components, 69, 287
  - element type, 287
  - type, 287
- element, 287
- empty, 287
- initializer, 290
- length of, 287
- type, 287
- assertion, 373
- assignable to, 95
- assignment, 93, 515
  - compatible with, 95
- associated labels, 377
- binary
  - compatibility, 339
  - numeric promotion, 110
- blank final, 71
- block, 361
  - enclosing, 182
- body
  - of class, 189
  - of constructor, 242
  - of interface, 263
  - of method, 223
- bound
  - of a type variable, 50
- boxing conversion, 86
- break binary compatibility with, 339
- break target, 388
- caller, 302
- can complete normally, 402
- capture conversion, 89
- cast operator, 101, 487
- catch clause, 297, 396
- caught, 297, 393
- checked exceptions, 299
- class(es), 173, 175
  - abstract, 176
  - body, 189
  - declaration, 175
  - depends on a reference type, 185
  - direct inner, 181
  - directly depends on a reference type, 185



## INDEX

**term definition** (*continued*)

- error, 301
- exception, 301
- final, 178
- initialization, 316, 317
- inner, 181
- inner of another class, 182
- local, 361
- modifiers, 175
- of object, 73
- runtime exception, 301
- same, 49
- strictfp, 178
- unloading, 330
- variables, 198
- commensurate, 283
- comments, 18
  - end-of-line, 18
  - traditional, 18
- compatibility, binary, 339
- compilation unit, 157
  - observable, 157
- compile-time
  - declaration, 471
  - type, 67
- compile-time type(s)
  - same, 49
- complete
  - abruptly, 360, 413
  - normally, 360, 402, 413
- completely initialized object, 573
- component, 287
  - of array, 69, 287
  - type of array, 287
- constant
  - expression
    - compile-time, 525
- constructor(s), 240
  - anonymous, 429
  - default, 247
  - parameter, 69, 240
  - signature, 241
- contain, 359
  - immediately, 359
- context
  - static, 181
- context of conversion, 78
- continue target, 391
- contract, 339
- control
  - access, 138
- conversion
  - contexts, 78
- conversion(s), 77
  - assignment, 93
  - casting, 101
  - identity, 80
  - method invocation, 99
  - primitive
    - narrowing, 82
    - widening, 80
  - reference
    - narrowing, 85
    - widening, 85
  - string, 497
  - value set, 92
- convertible to a numeric type, 89
- convertible to an integral type, 89
- correctly synchronized, 563
- creating
  - enum
    - constant, 251
- data race, 563
- declaration(s), 114
  - array variable, 288
- class, 173
  - variable, 69
- compile-time, 471
- constructor, 240
- field, 196, 264
- import, 160
- instance variable, 69
- interface, 263
- local variable, 70
  - statement, 363
- method, 209
- package, 158
- parameter
  - exception, 70
  - method, 69
- shadowed, 119
- single-static import, 164
- single-type import, 161
- static-import-on-demand, 165
- top-level type, 166
- type-import-on-demand, 163

- declarator, 364
- default
  - constructor, 247
  - value, 71
- definite assignment, 527
- definitely assigned
  - after, 530
    - when false, 530
    - when true, 530
  - before, 530
- definitely unassigned after, 530
- definitely unassigned after when false, 530
- definitely unassigned after when true, 530
- definitely unassigned before, 530
- direct
  - extension, 259
  - subclass, 184
  - subtype, 63
  - superclass, 184
  - superinterface, 186, 261
- directly implement, 259
- dynamic method lookup, 476
- dynamically enclosed, 302
- element, 287
  - type, 287
- empty
  - array, 287
  - statement, 370
- enclosed, dynamically, 302
- erasure, 56
- error classes, 301
- escape
  - sequence, character and string, 30
  - Unicode, 15
- evaluation
  - evaluated, 409
  - expression, result of, 409
  - order, 414
- exception
  - caller of, 302
  - caught, 297
  - checked, 299
  - classes, 299
    - checked, 299
    - runtime, 301
    - unchecked, 299
  - handler, 397
    - parameter, 70
    - polling for, 304
  - precise, 303
  - thrown, 297
  - unchecked, 301
- executed, 409
- exit of virtual machine, 331
- expression, 409
  - constant, 525
  - statement, 371
- extension, direct, 259
- field, 196, 264
  - declaration, 196
  - final, 199, 264
  - non-static, 198
  - static, 198, 264
  - transient, 199
  - volatile, 199
- finalizable, 327
- finalization
  - object, 325
- finalized, 327
- finalizer, 325
- finalize-reachable, 327
- fixed arity method, 212
- formal parameter, 210, 240
- FP-strict expression, 411
- frame activation, 477
- generic
  - interface, 261
- goal symbol, 9
- gradual underflow, 41
- grammar
  - context-free, 9
  - lexical, 9
- handler of exception, 397
- handles, 302
- hide, 225
  - a field, 196
  - an interface field, 264
  - an interface member type, 270
  - a member type, 237
- identifier, 19
- immediately contain, 359
- implement, 124, 187, 224
  - directly, 259
- implemented, 206
- import
  - on demand, 163, 165

## INDEX

**term definition** (*continued*)

- single static, 164
- single type, 161
- in scope, 117
- inexact, 41
- infinities, 37
- inherited, 123, 206
- inherits, 224
- initializer
  - array, 290
  - static, 239
  - variable, 201
- input elements, 17
- instance
  - of class, 45
  - immediately enclosing, 182
  - immediately enclosing with respect to a
    - class, 182
  - initializer, 238
  - lexically enclosing, 182
  - method, 216
  - variables, 198
- instantiation, 424
- interface(s)
  - abstract, 261
  - body, 263
  - initialization, 316
  - same, 49
  - strictfp, 261
  - top-level, 259
- intersection type, 62
- invocation
  - alternate constructor, 244
  - superclass constructor, 244
- iteration statements, 390
- Java
  - digits, 19
  - letters, 19
- keyword, 21
- label, 370
  - associated, 377
- language, 9
- left of, to the, 17
- left-hand side, 9
- length of array, 287
- line terminator, 16
- linking, 314
- literal, 21
  - boolean, 26
  - character, 26
  - floating-point, 24
  - integer, 22
  - null, 30
  - string, 28
- loading, 312
- local variable, 70
  - declaration statement, 363
- lookup
  - dynamic method, 476
- loss of precision, 81
- maximally specific, 449
- member class, 237
- member interface, 237
- members, 122
  - ambiguous, 265
  - dynamic lookup, 476
- memory model, 557
- meta-annotation, 284
- method(s), 209
  - abstract, 214, 267
  - applicable, 442
  - body, 223
  - class, 216
  - final, 217
  - hidden, 225
  - instance, 216
  - interface, 267
  - lookup, dynamic, 476
  - more specific, 449
  - most specific, 442, 447, 449
  - native, 218
  - non-static, 216
  - overloaded, 229
  - parameter, 69
  - static, 216
  - synchronized, 218
- modifiers
  - class, 175
  - constructor, 241
  - field, 197
  - interface, 260
  - method, 214
- more specific, 448
- most specific, 447, 449
- name(s), 115
  - ambiguous, 126

**term definition** (*continued*)

- canonical, 146
- contextually ambiguous, 129
- expression, 129, 130
- method, 129
- package, 130
- qualified, 115
- simple, 115
- type, 130
- unique package, 169
- NaN (Not-a-Number), 37
- narrowing
  - primitive conversion, 82
  - reference conversion, 85
- nested class, 173
- non-static
  - field, 198
  - method, 216
- nonterminal, 9
- normal completion, 360, 413, 414
- numeric promotion, 108
- object(s), 45
  - class of, 73
  - target, 476
- obscured declaration, 122
- observable
  - package, 160
- operators, 31
  - additive, 496
  - assignment, 512
  - bitwise, 508
  - cast, 491
  - equality, 505
  - logical, 508
  - multiplicative, 491
  - relational, 503
  - shift, 502
  - unary, 487
- order of evaluation, 414
- ordered, 39
- overload, 229, 269
- override-equivalent, 214
- package(s)
  - accessible, 138
  - unique names, 169
- parameter, 210
  - constructor, 69, 240
  - exception handler, 70, 397
  - formal, 210, 240
  - method, 69, 211
- pointer, 45
- polling for exceptions, 304
- precise exception, 303
- precision, loss of, 81
- preparation, 315
- primitive conversion
  - narrowing, 82
  - widening, 80
- productions, 9
- promotion
  - numeric, 108
    - binary, 110
    - unary, 108
- qualified
  - access, 138
  - name, 115
- qualifying type
  - of a field, 336
  - of a method invocation, 336
- raw type, 57
- reachable, 326, 359, 402
- reason, 298, 360, 413
- recursive, 10
- reference, 45
  - conversion
    - narrowing, 85
    - widening, 85
  - target, 473
  - types, 44
- reifiable type, 56
- release, 561
- resolution, 315
  - late, 314
  - lazy, 314
- result, 409
- return-type-substitutable, 220
- right of, to the, 17
- right-hand side, 9
- round
  - to nearest, 41
  - toward zero, 41
- runtime exception classes, 301
- run-time type(s)
  - same, 49
- same
  - class, 49
  - compile-time type, 49
  - interface, 49

## INDEX

- run-time class, 49
- run-time interface, 49
- run-time type, 49
- scope, 117
- separator, 31
- sequentially consistent, 560
- signature, 241
- simple name, 115
- specific
  - maximally, 449
  - more
    - strictly, 449
  - most, 447, 449
- statements, 359
  - empty, 370
  - expression, 371
  - iteration, 390
  - labeled, 370
  - local variable declaration, 363
  - unreachable, 402
- static
  - field, 198
  - import
    - on demand, 165
  - initializers, 239
  - method, 216
  - resolution, 314
- strictly more specific, 449
- strongly typed, 33
- subclass, 185
  - direct, 184
- subinterface, 262
- subsignature, 213
- sufficient
  - set of synchronization edges, 562
- superclass, 185
  - direct, 184
- superinterfaces, 187, 262
  - direct, 186, 261
- symbol
  - goal, 9
  - nonterminal, 9
  - optional, 10
  - terminal, 9
- synchronized-with, 561
- target
  - break, 389
  - continue, 391
  - object, 476
  - reference, 473
- terminal symbol, 9
- thead returns normally, 580
- thread, 553
- thrown, 297, 298
- throws clause, 221
- token, 9, 17
- top-level class, 173
- type declaration specifier, 44
- type parameters, 178
- type(s), 34, 73
  - boolean, 35
  - compile-time, 67
  - floating-point, 35, 37
  - import
    - on demand, 163
  - integral, 35
  - null, 34
  - numeric, 35
  - primitive, 34
  - reference, 44
- typed
  - strongly, 33
- unary numeric promotion, 108
- unboxing conversion, 88
- unchecked conversion, 89
- unchecked exception, 299
  - classes, 299
- underflow
  - gradual, 41
- unfinalized, 327
- Unicode escapes, 15
- unique package name, 169
- unloading of classes, 330
- unordered, 40
- unreachable, 402
- unreachable object, 327
- value
  - default, 71
  - of expression, 410
- value set
  - double, 38
  - double-extended-exponent, 38
  - float, 38
  - float-extended-exponent, 38
- variable arity method, 212
- variable(s), 67
  - class, 69, 198
  - final, 71

**term definition** (*continued*)

- instance, 69, 198
- local, 70
- verification, 314
- virtual machine exit, 331
- visible, 120
- white space, 18
- widening
  - primitive conversion, 80

**terminal symbol**

- term definition, 9

**terminators**

- line
  - carriage return and linefeed characters as, 28
  - term definition, 28

**Test**

- program, how to run, 5

**this keyword**

- accessing hidden fields with, 367
- evaluation of, 421, 422
  - during instance creation, 322
- not permitted
  - in class variable initialization, 202
  - in explicit constructor calls, 242
  - in interface field declarations, 265
- permitted
  - in instance initializers, 239
  - in instance variable initialization, 202
- qualified, 114

**Thompson, William (Lord Kelvin), 409****Thorup, Kresten Krab, 55****threads**

- See also* synchronization
- (chapter), 553
- action, inter-, 558
- blocked, 571
- divergence
  - action, 559
- initialization implications of multiple, 319
- interruption of, 582
- locks acquisition and release, by
  - synchronized statement, 395
- return normally
  - term definition, 580
- semantics
  - intra-, 557
- sleep, 583

- synchronized modifier, methods,
  - declaration, 218
- termination and virtual machine exit, 331
- volatile fields use with, 199
- yield, 583

**throw**

- See also* control flow; exceptions
- throw statement
  - definite assignment, 545
  - as reason for abrupt completion, 360
  - specification, 393

**Throwable class, 179****throws clause, 239**

- of an anonymous constructor, 430
- checked exception classes named in, 299
- constructors, 242
  - binary compatibility considerations, 354
- incorrect overriding because of, example, 236
- methods, 221
  - binary compatibility considerations, 354
  - relation to instance initializers, 202, 238, 301

**tokens**

- See also* grammars
- term definition, 9, 17

**top-level class, 130, 145, 146**

- accessibility of, 138
- canonical name of, 146
- enables or disables assertions, 374
- as package member, 122, 154
- and private access, 139, 144
- and public modifier, 175
- term definition, 173

**top-level interface, 145**

- canonical name of, 146
- as package member, 122, 154
- term definition, 259

**top-level type**

- binary name of, 335
- fully qualified name of, 166
- as part of a compilation unit, 157
- scope, 117, 166

**Torgersen, Mads, 54, 55, 92****transient modifier**

- See also* declarations; modifiers
- fields, 199
  - binary compatibility considerations, 350

**translations**

- lexical, steps involved in, 14

## INDEX

**try statements**

*See also* control flow; exceptions; statements  
 definite assignment, 545  
 exception handling role of, 302  
 scope of parameters to exception handlers,  
 118, 397  
 specification, 396  
 try-catch statement, specification, 398  
 try-catch-finally statement,  
 specification, 399

**types**

*See also* arrays; classes; interfaces; primitive,  
 types  
 (chapter), 33  
 annotation(s), 4, 113, 158, 175, 241, 259,  
 264, 267, 283, 358, 364  
 declaration(s), 260  
 element(s)  
   names, 129  
   term definition, 273  
   term definition, 270  
 argument values, method invocation  
   conversion context, 99  
 argument(s), 44, 424  
   containment, 55  
   explicit, 445, 446, 447  
   inference  
     *See* types, inference  
   provably distinct, 54  
 array, 283  
   allowed in annotation types, 273  
   canonical name of, 146  
   erasure of, 56  
   members of, 125  
   syntax and specification, 288  
   when reifiable, 56  
 binary compatibility considerations, 339  
 boolean  
   Boolean literal specification, 26  
   term definition and specification, 43  
 char  
   character literal specification, 26  
 classes  
   members of, 123  
   naming conventions, 147  
   non-parameterized, 85  
 commensurate, 274  
   term definition, 283

compile-time, 73  
 data values relation to, 34  
 declaration(s)  
   as members of packages, 166  
   enclosing, 473  
   enum, 249  
   generic, 89  
   specifier, 44  
   term definition, 73  
 double, floating-point literal specification,  
 24  
 elements, of array, 287  
 enum, 3, 113, 241, 242, 247, 283  
   allowed in annotation type, 273  
   binary compatibility, 356  
   direct superclass of, 184  
   in switch, 377  
   may not be instantiated, 424  
   must not be declared abstract, 176, 250  
   nested  
     implicitly static, 250  
 erasure  
   *See* erasure  
 existential  
   relation to wildcards, 54  
 expression  
   assignment conversion, 93  
   how determined, 73  
 of expressions, 410  
 float, floating-point literal specification, 24  
 floating-point, term definition and  
   specification, 37  
 imported  
   as declared entity, 114  
   scope of, 117, 161  
 importing all public static members from a,  
 165  
 importing, compilation units, 158  
 inference, 220, 242, 445, 446, 447, 448  
   not required for wildcards, 54  
   of unresolved type arguments, 466  
   process defined, 451  
 infinite, 465  
 int, integer literal specification, 22  
 integral, 35, 89  
 interface  
   implications for variables and expressions,  
   73

- types** (*continued*)
- members of, 124
  - naming conventions, 147
  - interfaces
    - non-parameterized, 85
  - intersection, 336, 337
    - direct supertypes, 64
    - glb(), 90
    - term definition, 62
  - local variables, declaration of, 364
  - long, integer literal specification, 22
  - member, 44
  - name(s), 44
    - context in which a name is classified as, 129
    - qualified, meaning of, 132
    - simple, meaning of, 132
  - named, 160
  - nested
    - enum
      - implicitly static, 250
    - erasure of, 56
  - non-reifiable, 412
  - null, 90, 504, 506
    - term definition and specification, 34
  - numeric, 89
    - in conditional expression, 511
    - of a conditional expression, 511
  - of members and constructors of a raw typw, 59
  - parameter(s), 114, 261, 357
    - cannot be referenced in a static method, 216
    - explicit, 444
    - formal, 213
      - binary compatibility, 342, 351
      - of a method, 220
    - names of, 113
    - term definition, 178
  - parameter, method invocation conversion
    - context, 99
  - parameterized, 89, 90, 103, 127, 178, 347
    - and heap pollution, 68
    - casts to, 102
    - direct superinterfaces, 262
    - erasure of, 56
    - in class literals, 421
    - members and constructors, 55
    - reifiable, 56
  - primitive, 88, 283
    - allowed in annotation types, 273
    - are reifiable, 56
    - as array element types, 288
    - do not share state, 35
    - fully qualified name, 145
    - in class literals, 421
    - term definition and specification, 34
    - variables, specification, 67
  - qualifying
    - See qualifying type
  - raw, 57, 68, 89, 99, 103
    - are reifiable, 56
  - reference, 88, 89, 431
    - as array element types, 288
    - criteria for determining when two are the same, 49
    - in conditional expression, 511
  - member
    - as declared entity, 113
    - specifying with
      - class declaration, 173
      - interface declaration, 260
  - term definition and specification, 44
  - variables, specification, 67
- reifiable, 56, 504
  - in arrays, 289, 431
- run-time, 73
- substitution(s), 262
- term definition, 73
- term definition and specification, 34
- top-level
  - annotations, 410
- usage, 65
  - in declarations and expressions, 65
- of variable
  - how determined, 73
- variable(s), 173, 178, 220, 242, 261, 347, 441
  - are declared entities, 114
  - disallowed in catch clauses, 397
  - erasure of, 56
  - fresh, 90
  - in class literals, 421
  - may not be a subtype of two invocations of same interface, 50
  - members, 50
  - names in extends clause, 127
  - term definition, 49
  - where used, 65



## INDEX

**U****unary**

numeric promotion, specification, 108

**unassignment**

definite

*See* definite unassignment

**unboxing**

*See* conversion, unboxing

**unchecked**

cast, 102

conversion

*See* conversion, unchecked

exceptions, 306

warning(s), 412

and type soundness, 67

**underflow**

floating-point operators, signed zero  
produced by, 41

integer operators, not indicated by, 36

**unfinalized**

as object attribute, 326

term definition, 327

**Unicode character set**

*See also* characters

character sequences, represented by  
instances of class `String`, 48

composite characters, contrasted with the  
Unicode decomposed characters, 20

escapes, 14, 15

specification for handling, 15

term definition, 15

use when suitable font not available, 16

handling in package names, 156

lexical grammar use in terminal symbols, 9

sorting, reference for details on, 20

translation of raw stream, 14

writing any character in, using ASCII  
characters in escapes, 15

**unloading**

*See also* linking; loading

of classes and interfaces, 330

of interfaces, 330

**unlock**

term definition, 554

**unqualified**

class instance creation

and definite [un]assignment, 537

class instance creation expression, 128

**unreachable**

*See also* exceptions

objects

finalization implications, 327

term definition, 327

statements, conservative flow analysis

detection of, 402

term definition, 327

**V****value**

return, specifying method with no, 210

**value set**

and contents of local variables, 364

double

term definition, 38

double-extended-exponent

term definition, 38

float

term definition, 38

float-extended-exponent

term definition, 38

and FP-strict expressions, 411

and return statement, 393

**values**

*See also* assignment; initialization; primitive,  
types; variable

(chapter), 33

`boolean`, term definition and specification,  
43

data, relation to types, 34

expressions, variables as, 410

floating-point, term definition and  
specification, 37

integral, term definition and specification, 35

primitive, term definition and specification,  
34

reference

*See* references

relation to types, 34

variables

initial and default, 71

**variables, 409**

*See also* data structures; fields; identifiers;  
scope

(chapter), 33

assignment conversion, 93

**variables** (*continued*)

- constant, 71
- double
  - non-atomic treatment of, 579
- initializer
  - instance, 252
- local, 364
  - annotation(s), 281
    - never retained, 278
  - declaration statements, 363
  - as declared entity, 113
  - definite assignment, declarations, 539
  - description, 70
  - naming conventions, 151
  - scope of, 118, 385, 387
  - shadowing of names by, 367
  - unaffected by memory model, 558
- long
  - non-atomic treatment of, 579
- memory model terminology, 558
- primitive type
  - changing value of, 35
  - specification, 67
- read of, 558
- reference type, specification, 67
- term definition and specification, 67
- type of, how determined, 73
- values
  - of expressions, 410
  - held by different types, summary of, 33
  - initial and default, 71
- volatile
  - read of, 558
  - write of, 558
- write of, 558
- verification**
  - See also* security
  - of binary representation, of classes and interfaces, 314
  - type safety dependence on existence and use of, 342
- Viroli, Mirko**, 54, 55, 92
- virtual machine**
  - See also* exceptions; optimization
  - class file format specified in, 334, 335
  - errors, 307
  - exception handling by, 297
  - exit, criteria for, 331
  - startup, 309

**visible**, 117

- method
  - and meaning of method invocation, 440
  - term definition, 120

**void keyword**

- See also* methods
- casting to, not permitted, 371
- methods
  - expression return statement not permitted in, 223
  - in conditional expression, 510
  - no return value specified by, 210

**volatile**

- read, 558, 568

**volatile modifier**

- fields, 199
- final declaration not permitted for, 201

**W****wait**, 580**warning(s)**

- deprecated, 280
  - suppression of, 280
- mandatory, 89
- unchecked, 60, 67, 89, 222, 225, 268, 412
  - suppression of, 280

**Webb, Jim**, 476**Weiss, George David**, 32**while statement**

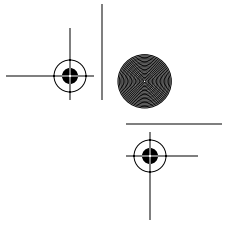
- See also* control flow
- definite assignment, 542
- specification, 380

**white space**

- term definition, 18

**Whitman, Walt**, 33, 111, 240, 358, 399**widening**

- See also* conversions
- primitive conversion
  - in assignment conversion context, 93
  - in binary numeric promotion context, 110
  - in casting conversion context, 101
  - in method invocation conversion context, 99
  - in unary numeric promotion context, 108
- reference conversion, 85
- in assignment conversion context, 93



*INDEX*

in casting conversion context, 101  
in method invocation conversion context,  
99  
**wildcard(s)**, 424  
capture, 90  
type of members, 55  
unbounded, 89, 102  
in reifiable types, 56  
**word tearing**, 578  
**Wordsworth, William**, 75, 286, 413, 553  
**write**  
of a field, 327

**Y**

**yield**  
thread, 583

**Z**

**zero**  
*See also* exceptions; numbers  
divide by, exceptions thrown by integer  
divide and remainder operators, 37

