



BRUCE ECKEL



THINKING IN JAVA

4TH
EDITION

The definitive introduction to object-oriented programming
in the language of the world wide web

FOR
JAVA
SE5/6



Software Development Jolt Award & Productivity Award
Java World Editor's Choice, Reader's Choice
Java Developer's Journal Editor's Choice, Reader's Choice



CODE & SUPPLEMENTS AT WWW.MINDVIEW.NET



What's Inside

Preface	1
Java SE5 and SE6.....	2
Java SE6.....	3
The 4 th edition	3
Changes	4
Note on the cover design	6
Acknowledgements	7
Introduction	13
Prerequisites	14
Learning Java.....	14
Goals.....	15
Teaching from this book	16
JDK HTML documentation	17
Exercises.....	17
Foundations for Java	18
Source code	18
Coding standards	21
Errors	21
Introduction to Objects	23
The progress of abstraction	24
An object has an interface.....	26
An object provides services	29
The hidden implementation.....	30
Reusing the implementation.....	32
Inheritance	33
Is-a vs. is-like-a relationships	37
Interchangeable objects with polymorphism	38
The singly rooted hierarchy.....	43
Containers	44
Parameterized types (Generics) ...	45
Object creation & lifetime ...	46
Exception handling: dealing with errors	49
Concurrent programming ...	50
Java and the Internet	51
What is the Web?.....	51
Client-side programming	53
Server-side programming	59
Summary	60
Everything Is an Object	61
You manipulate objects with references	61
You must create all the objects.....	63
Where storage lives	63
Special case: primitive types	65
Arrays in Java	66
You never need to destroy an object	67
Scoping	67
Scope of objects	68
Creating new data types: class	69
Fields and methods	70
Methods, arguments, and return values	72
The argument list	73

Building a Java program.....	74
Name visibility	74
Using other components.....	75
The static keyword.....	76
Your first Java program.....	78
Compiling and running.....	80
Comments and embedded documentation	81
Comment documentation	82
Syntax	83
Embedded HTML	84
Some example tags.....	85
Documentation example.....	87
Coding style	88
Summary	89
Exercises.....	89

Operators 93

Simpler print statements	93
Using Java operators	94
Precedence	95
Assignment.....	95
Aliasing during method calls	97
Mathematical operators.....	98
Unary minus and plus operators	101
Auto increment and decrement.....	101
Relational operators.....	103
Testing object equivalence.....	103
Logical operators.....	105
Short-circuiting	106
Literals.....	108
Exponential notation	109
Bitwise operators	111
Shift operators.....	112
Ternary if-else operator ...	116
String operator + and +=	118

Common pitfalls when using operators	119
Casting operators	120
Truncation and rounding	121
Promotion	122
Java has no “sizeof”	122
A compendium of operators	123
Summary	133

Controlling Execution 135

true and false	135
if-else	135
Iteration.....	137
do-while	138
for	138
The comma operator	140
Foreach syntax	140
return	143
break and continue	144
The infamous “goto”	146
switch	151
Summary	154

Initialization & Cleanup 155

Guaranteed initialization with the constructor	155
Method overloading	158
Distinguishing overloaded methods	160
Overloading with primitives.....	161
Overloading on return values.....	165
Default constructors.....	166
The this keyword.....	167
Calling constructors from constructors	170
The meaning of static	172
Cleanup: finalization and garbage collection	173
What is finalize() for?	174

You must perform cleanup	175	Delegation	246
The termination condition.....	176	Combining composition	
How a garbage collector works.....	178	and inheritance	249
Member initialization	181	Guaranteeing proper cleanup	251
Specifying initialization	183	Name hiding	255
Constructor initialization.....	185	Choosing composition	
Order of initialization	185	vs. inheritance	256
static data initialization.....	186	protected	258
Explicit static initialization	190	Upcasting.....	260
Non- static		Why “upcasting”?	261
instance initialization	191	Composition vs. inheritance	
Array initialization	193	revisited	261
Variable argument lists.....	198	The final keyword	262
Enumerated types	204	final data.....	262
Summary	207	final methods	267
Access Control	209	final classes	270
package :		final caution	271
the library unit	210	Initialization	
Code organization	212	and class loading	272
Creating unique		Initialization with inheritance....	272
package names.....	213	Summary	274
A custom tool library	217	Polymorphism	277
Using imports		Upcasting revisited.....	278
to change behavior.....	220	Forgetting the object type	279
Package caveat	220	The twist	281
Java access specifiers	221	Method-call binding.....	281
Package access	221	Producing the right behavior	282
public : interface access	222	Extensibility	286
private : you can't touch that!...	224	Pitfall: “overriding”	
protected : inheritance access..	225	private methods.....	290
Interface		Pitfall: fields	
and implementation.....	228	and static methods.....	290
Class access	229	Constructors and	
Summary	233	polymorphism	293
Reusing Classes	237	Order of constructor calls.....	293
Composition syntax	237	Inheritance and cleanup.....	295
Inheritance syntax	241	Behavior of polymorphic	
Initializing the base class.....	244	methods inside constructors	301
		Covariant return types	303

Designing with inheritance	304	Reaching outward from a multiply nested class	368
Substitution vs. extension.....	306	Why inner classes?	369
Downcasting and runtime type information	308	Closures & callbacks	372
Summary	310	Inner classes & control frameworks	375
Interfaces	311	Inheriting from inner classes	382
Abstract classes and methods.....	311	Can inner classes be overridden?.....	383
Interfaces.....	316	Local inner classes	385
Complete decoupling	320	Inner-class identifiers	387
“Multiple inheritance” in Java	326	Summary	388
Extending an interface with inheritance	329	Holding Your Objects	389
Name collisions when combining interfaces	330	Generics and type-safe containers	390
Adapting to an interface	331	Basic concepts	394
Fields in interfaces	335	Adding groups of elements	396
Initializing fields in interfaces ...	335	Printing containers	398
Nesting interfaces	336	List	401
Interfaces and factories.....	339	Iterator	406
Summary	343	ListIterator	409
Inner Classes	345	LinkedList	410
Creating inner classes	345	Stack	412
The link to the outer class	347	Set	415
Using .this and .new	350	Map	419
Inner classes and upcasting	352	Queue	423
Inner classes in methods and scopes	354	PriorityQueue	425
Anonymous inner classes	356	Collection vs. Iterator ..	427
Factory Method revisited.....	361	Foreach and iterators	431
Nested classes	364	The <i>Adapter Method</i> idiom.....	434
Classes inside interfaces	366	Summary	437
		Error Handling with Exceptions	443
		Concepts	444
		Basic exceptions	445
		Exception arguments.....	446

Catching an exception.....	447
The try block.....	447
Exception handlers	448
Creating your own exceptions.....	449
Exceptions and logging.....	452
The exception specification	457
Catching any exception	458
The stack trace	460
Rethrowing an exception	461
Exception chaining	464
Standard Java exceptions.....	468
Special case:	
RuntimeException	469
Performing cleanup with finally	471
What's finally for?	473
Using finally during return	476
Pitfall: the lost exception	477
Exception restrictions	479
Constructors.....	483
Exception matching	489
Alternative approaches	490
History.....	492
Perspectives	494
Passing exceptions	
to the console	497
Converting checked	
to unchecked exceptions.....	497
Exception guidelines.....	500
Summary	501

Strings 503

Immutable Strings	503
Overloading '+' vs.	
StringBuilder	504
Unintended recursion.....	509
Operations on Strings	511

Formatting output.....514

printf()	514
System.out.format()	514
The Formatter class.....	515
Format specifiers	516
Formatter conversions	518
String.format()	521
Regular expressions	523
Basics	524
Creating regular expressions.....	527
Quantifiers.....	529
Pattern and Matcher	531
split()	540
Replace operations	541
reset()	544
Regular expressions	
and Java I/O	544

Scanning input

Scanner delimiters	549
Scanning with	
regular expressions.....	550
StringTokenizer	551
Summary	552

Type Information 553

The need for RTTI	553
The Class object	556
Class literals.....	562
Generic class references	565
New cast syntax	568
Checking before a cast.....	569
Using class literals.....	576
A dynamic instanceof	578
Counting recursively.....	580
Registered factories.....	582
instanceof vs. Class	
equivalence.....	586
Reflection: runtime	
class information.....	588
A class method extractor	590

Dynamic proxies	593
Null Objects.....	598
Mock Objects & Stubs	606
Interfaces and type information	607
Summary	613

Generics 617

Comparison with C++	618
Simple generics	619
A tuple library	621
A stack class	625
RandomList	626
Generic interfaces	627
Generic methods	631
Leveraging type argument inference	633
Varargs and generic methods	635
A generic method to use with Generators	636
A general-purpose Generator .	637
Simplifying tuple use	639
A Set utility	641
Anonymous inner classes	645
Building complex models	647
The mystery of erasure	650
The C++ approach	652
Migration compatibility	655
The problem with erasure.....	656
The action at the boundaries	658
Compensating for erasure	662
Creating instances of types	664
Arrays of generics.....	667
Bounds.....	673
Wildcards	677
How smart is the compiler?	680
Contravariance.....	682

Unbounded wildcards	686
Capture conversion	692
Issues	694
No primitives as type parameters	694
Implementing parameterized interfaces.....	696
Casting and warnings	697
Overloading	699
Base class hijacks an interface ..	700
Self-bounded types.....	701
Curiously-recurring generics.....	701
Self-bounding	703
Argument covariance	706
Dynamic type safety	710
Exceptions	711
Mixins	713
Mixins in C++	714
Mixing with interfaces.....	715
Using the Decorator pattern	717
Mixins with dynamic proxies	719
Latent typing	721
Compensating for the lack of latent typing.....	726
Reflection.....	726
Applying a method to a sequence	728
When you don't happen to have the right interface	731
Simulating latent typing with adapters	733
Using function objects as strategies	737
Summary: Is casting really so bad?.....	743
Further reading	746

Arrays 747

Why arrays are special	747
------------------------------	-----