

# activejava

*Object-Oriented Programming for*     *the World Wide Web*

**Adam Freeman**

*Knowledge Media Institute  
The Open University*

**& Darrel Ince**

*Department of Computing  
The Open University*



Addison-Wesley Publishing Company

Harlow, England • Reading, Massachusetts • Menlo Park, California  
New York • Don Mills, Ontario • Amsterdam • Bonn • Sydney • Singapore  
Tokyo • Madrid • San Juan • Milan • Mexico City • Seoul • Taipei

---

# CONTENTS

Preface	vii
1 JAVA AND THE INTERNET	1
1.1 Introduction	1
1.2 The Java programming language	2
1.2.1 The language	2
1.2.2 A simple Java program	4
1.2.3 The system	5
1.2.4 The current state of the system	5
1.3 The evolution of browsers	6
1.3.1 Browsing and the Internet	6
1.3.2 The current state of browsers	7
1.4 This book	8
1.4.1 Summary	8
1.4.2 Prerequisites and entry points	8
1.5 What you will learn from this book	9
2 OBJECTS	10
2.1 Introduction	10
2.2 Some examples of object-based systems	11
2.2.1 A simple air traffic control system	11
2.2.2 A more complicated air traffic control system	14
2.2.3 A print spooler	15
2.3 Summary	16
3 CLASSES	19
3.1 Introduction	19
3.2 Some concepts	23
3.2.1 A robot	24
3.2.2 Messages and classes	28
3.3 Inheritance	29
3.3.1 The augmented set	29
3.3.2 Invoices	38
3.3.3 Employee records	41
3.4 Class libraries	44
3.5 Summary	46
4 DATA TYPES AND CONTROL STRUCTURES	47
4.1 Introduction	47
4.2 Data types	48

4.2.1 Primitive data types	48
4.2.2 First class objects	53
4.3 Control structures	55
4.3.1 Conditional processing	55
4.3.2 Repetitive processing	60
4.3.3 Strings as objects	66
4.3.4 Comments	73
4.4 Summary	74
5 CLASSES IN JAVA	75
5.1 Introduction	75
5.2 Objects	76
5.3 Classes	77
5.3.1 Defining classes in Java	77
5.3.2 Access mechanisms	80
5.3.3 Static variables and static methods	81
5.3.4 Inheritance in Java	86
5.3.5 Some examples of Java classes	89
5.3.6 General classes	102
5.3.7 Exceptions and other mechanisms	107
5.3.8 Threads	108
5.3.9 Java programs	110
5.4 Summary	110
6 LIBRARIES	112
6.1 Introduction	114
6.2 Classes	114
6.2.1 The <code>Vector</code> class	120
6.3 I/O class library	120
6.4 Summary	120
7 THE AWT LIBRARY	121
7.1 Introduction	121
7.2 Overview of the AWT	122
7.3 The AWT at large	123
7.3.1 Containers and layouts	123
7.3.2 Control elements	129
7.4 The event system	135
7.4.1 Approaches to handling events	135
7.5 Graphics	139
7.5.1 Using Graphics in a basic applet	139
7.5.2 Double buffering	140
7.6 Fonts and colours	142
7.6.1 Font support	142
7.6.2 Colour support	143
8 THE <code>java.net</code> LIBRARY	145
8.1 Introduction	145
8.2 A tour of the package	145
8.2.1 The common protocol	146
8.2.2 Software and servers	146

8.2.3	Some simple examples	147
8.2.4	Possible applications	147
8.3	Writing a server	147
8.3.1	The cost of using Java for servers	148
8.3.2	Introducing the first iteration of the server	148
8.3.3	Extending the server	151
8.3.4	Handling multiple clients	152
8.3.5	Other tasks with sockets	156
8.3.6	Summary of server applications	157
8.4	Writing an applet	158
8.4.1	Applet security	158
8.4.2	Our first look at a client	158
8.4.3	Making constrained connections	161
8.4.4	Guidelines for applets and applications for client-server applications	163
8.5	General considerations	163
8.5.1	Network bandwidth constraints	163
8.5.2	Good behaviour	164
8.6	Summary	164
9	THE JAVA DEVELOPMENT KIT	166
9.1	Introduction	166
9.2	First steps	167
9.2.1	Compiling a Java source file	167
9.2.2	Running Java code	170
9.3	Other JDK tools	173
9.3.1	Native method support – Java	173
9.3.2	The Java disassembler	174
9.3.3	API documentation	177
9.4	HTML tags	180
9.4.1	Non-Java browsers	181
9.5	Summary	182
10	BUILDING AN APPLLET	183
10.1	Introduction	183
10.1.1	Hello World revisited again	184
10.2	Introducing some essential applet methods	184
10.2.1	Methods to control the life of an applet	185
10.2.2	Adding life-cycle methods to the HelloWorld class	185
10.3	Drawing and handling events	186
10.3.1	Drawing to the screen	186
10.3.2	Adding drawing methods to the HelloWorld class	187
10.3.3	Improving drawing performance	188
10.3.4	Handling applet events	189
10.3.5	Adding event handling to HelloWorld	189
10.4	Using threads in applets	194
10.4.1	Threads are our friends	194
10.4.2	Making sure your threads die with your applet	194
10.4.3	Scheduling threads within applets	195
10.4.4	Adding thread support to the HelloWorld Class	196
10.4.5	Summary of thread use in applets	197
10.5	Adding an interface to an applet	197

10.5.1 Adding a <code>TextField</code>	198
10.5.2 Adding the start button	200
10.5.3 Adding the selector	201
11 BUILDING AN APPLICATION	203
11.1 Introduction	203
11.1.1 Introducing an application	204
11.2 Applications life cycle	206
11.2.1 Using threads in an application	207
11.2.2 Dual-purpose code	208
11.3 Differences between applets and applications	210
11.3.1 Loading images	211
11.3.2 Applet security limitations	211
11.3.3 Creating servers	213
11.3.4 Summary of differences	213
11.4 Application limitations	213
11.5 Adding user interface elements to applets	214
11.6 Summary	215
12 JAVA INTERNALS	217
12.1 Introduction	217
12.2 The compilation process	217
12.2.1 Byte-codes	218
12.2.2 Just-in-time compilers	218
12.3 The Java virtual machine	219
12.3.1 The basics of a virtual machine	219
12.3.2 The downside of a virtual machine	219
12.3.3 Using other programming languages to create byte-codes	221
12.4 Memory management	221
12.5 Applets vs applications	222
12.5.1 What an applet is good for	222
12.5.2 What an application is good for	223
12.6 Java security	223
12.6.1 The Java language and compiler	223
12.6.2 Byte-code verification	224
12.6.3 The class loader	224
12.6.4 Local system protection	224
12.6.5 The cost of security	225
12.7 The future of Java	225
12.7.1 Java's secret weapons	226
12.7.2 Competition to Java	229
12.8 The value of testing	229
12.8.1 Interface design	230
12.8.2 Cross-platform testing	230
12.8.3 Debugging code	230
12.8.4 Irritating behaviour	231
12.9 Summary	231
Index	232