

# Inhaltsverzeichnis

<b>1</b>	<b>Die Sprache Java</b>	<b>13</b>
1.1	Warum Java?	14
1.1.1	Programme im WWW?	14
1.1.2	Plattformunabhängigkeit	17
1.1.3	Compiler versus Interpreter	18
1.1.4	Die Lösung des Dilemmas	20
1.1.5	Standardisierung von Java	22
1.1.6	Versionen	22
1.1.7	Unterschiede zwischen den Versionen	23
1.1.8	Unterschiede zwischen 1.1 und 2	26
1.2	Objektorientierte Programmierung	27
1.2.1	Objekt	27
1.2.2	Methode	28
1.2.3	Kapselung von Eigenschaften	29
1.2.4	Nachrichten	30
1.2.5	Klasse	31
1.2.6	Vererbung	32
1.2.7	Redefinition von Methoden und Eigenschaften	34
1.2.8	Überladung von Methoden	36
1.2.9	Mehrfachvererbung	37
1.3	Das erste Applet	38
1.3.1	Der Quelltext	38
1.3.2	Die erste Java-Web-Seite	40
1.3.3	Das Grundgerüst eines Applets	41
1.3.4	Zusammenspiel mit dem Browser	44
1.3.5	Bibliotheken und Import	46
1.3.6	Parameter und Variablen	47
1.3.7	Schleifen	59
1.3.8	Statische Methoden	62
1.4	Der allgemeine Aufbau eines Java-Applets	64
1.4.1	Eine Vorlage für Applets	64
1.4.2	Stop and go...	68
1.4.3	Interaktionen	69
1.4.4	Wir veranschaulichen die Funktionsweise des Applets	70
1.4.5	Die Anzeige aktualisieren	73
1.4.6	Java 1.1	75



1.4.7	Ein Quiz	80
1.5	Applet versus Applikation	93
1.5.1	Sicherheitsaspekte	96
1.6	CGI-Programme in Java	97
1.6.1	WWW-Formulare	97
1.6.2	CGI-Programme	100
1.6.3	Java als CGI-Sprache	108
1.6.4	Eine Umfrage	114
1.7	Das »Abstract Window Toolkit« AWT	122
1.7.1	Knöpfe und Labels	126
1.7.2	Der Layout Manager	127
1.7.3	Panel	129
1.7.4	...uuuuund action	129
1.7.5	Interaktionselemente und Layout Manager	130
<b>2</b>	<b>Events</b>	<b>135</b>
2.1	Quellen und Zuhörer	137
2.2	Innere Klassen	140
2.2.1	Übersicht: Innere Klassen	146
2.3	Standard-Events behandeln	147
2.3.1	ActionEvent	150
2.3.2	ItemEvent	152
2.3.3	AdjustmentEvent	153
2.3.4	MouseEvent (MouseMotionListener)	153
2.3.5	FocusEvent	155
2.3.6	TextEvent	155
2.3.7	MouseEvent (MouseListener)	156
2.3.8	KeyEvent	156
2.3.9	ContainerEvent	157
2.3.10	ComponentEvent	158
2.3.11	WindowEvent	158
2.4	AWT-Events in Komponenten	159
<b>3</b>	<b>Servlets – Die Alternative zu CGI</b>	<b>163</b>
3.1	Was sind Servlets?	164
3.2	Vorteile von Servlets gegenüber CGI-Scripts	164
3.3	Wie werden Servlets programmiert?	165
3.4	Kommunikation leichtgemacht – der Java-Chat	170
3.5	Cookies – Informative Krümel	178
3.6	Sessions – Behalten Sie Ihre Kunden im Auge	184
<b>4</b>	<b>Drag and Drop</b>	<b>195</b>
<b>5</b>	<b>Collections</b>	<b>203</b>
5.1	Java-Implementationen der Mengenklassen	206
5.2	Arbeitersparnis durch Collections	211
5.3	Synchronisation von Collections	212



5.4	Das Interface »Iterator«	213
5.5	Suchen, Sortieren und andere Algorithmen	214
<b>6</b>	<b>RMI – Verteilte Programmierung unter JAVA</b>	<b>221</b>
6.1	Programmierung der Server-Objekte	223
6.2	Programmierung eines Client-Objekts	226
6.3	Aktivierbare Objekte	228
<b>7</b>	<b>Serialisierung</b>	<b>233</b>
7.1	Objekte (de)serialisieren	234
7.2	Was wird serialisiert?	236
7.3	Benutzerdefinierte Serialisierung	237
7.3.1	Zu serialisierende Felder angeben	237
7.3.2	Lesen und Schreiben mit eigenen Methoden	238
7.3.3	Alles unter Kontrolle	239
<b>8</b>	<b>Das Reflection-API</b>	<b>241</b>
8.1	Einblick zur Laufzeit	243
8.2	Weitere Anregungen	252
<b>9</b>	<b>Swing</b>	<b>255</b>
9.1	Der erste Einblick	257
9.1.1	Packages	257
9.1.2	Basisklassen	258
9.2	Einführung mit Beispielen	260
9.2.1	Die erste Swing-Anwendung	260
9.2.2	Crashkurs: Swing-Features	264
9.3	Das Projekt: Ein Texteditor	281
9.3.1	Version 1: Eine kleine Änderung	281
9.3.2	Version 2: Menüs mit Action	282
9.3.3	Version 3: Eine Symbolleiste	293
9.3.4	Version 4: plaf	304
9.3.5	Version 5: Desktops und Dialoge	309
9.3.6	Version 6: Dateien öffnen und speichern	343
9.3.7	Version 7: Editierbefehle	361
9.3.8	Dynamische Toolbars	372
9.3.9	Version 8: Undo!	377
9.3.10	Fazit – Was hat es genützt?	383
9.4	Kurzer Überblick: Der Rest von Swing	385
9.4.1	JSplitPane	385
9.4.2	Tab-Register	387
9.4.3	Buttons	388
9.4.4	Auswahl von Farben	390
9.4.5	Combo-Boxen	391
9.4.6	Fortschrittsanzeigen und Timer	392
9.4.7	Slider	394
9.4.8	Tabellen	395



9.4.9	Paßwort-Eingabe	398
9.4.10	BoxLayout	400
9.4.11	Thread-Sicherheit	401
<b>10</b>	<b>Komponentenentwicklung</b>	<b>403</b>
10.1	Grundlagen	404
10.1.1	Package-Organisation	404
10.1.2	Allgemeingültigkeit	406
10.1.3	Noch eine kleine Regel	407
10.2	Der einfache Anfang: Aggregation	408
10.2.1	Version 1: Textfield + Scrollbar = ?	408
10.2.2	Version 2: Eingabe mit der Scrollbar	410
10.2.3	Version 3: Eingabe mit den Cursortasten	412
10.2.4	Version 4: Den Entwickler nicht vergessen	415
10.2.5	Version 5: Selbst Events auslösen	416
10.2.6	Version 6 und Schluß: Eingabe mit der Tastatur	419
10.2.7	Nach fünf Betas die Endversion	430
10.3	Vererbung einer vorhandenen Komponente	431
10.3.1	Version 1: Ein Verzeichnisbaum	431
10.3.2	Version 2: Eine bessere Optik	436
10.3.3	Version 3: Stamm-/Root-Verzeichnisse anzeigen	441
10.3.4	Version 4: Nicht nur Ordnersymbole	444
10.3.5	Version 5: Fit für Wechsel(-medien)	448
10.3.6	Version 6: Fit für die Beanbox	461
10.4	»Neuschreiben« einer Komponente	475
10.4.1	Version 1: Eine Lightweight-Komponente	475
10.4.2	Version 2: Ein schöneres Erscheinungsbild	478
10.4.3	Version 3: Reaktion auf die Maus	480
10.4.4	Version 4: Auslösen eines Events	483
<b>11</b>	<b>Beans</b>	<b>487</b>
11.1	Beans sind Komponenten	488
11.1.1	Methoden	489
11.1.2	Properties	490
11.1.3	Übersicht: Einfache und indizierte Properties	492
11.1.4	Events	505
11.1.5	Serialisierung	506
11.2	Von der Komponente zur Bean	506
11.2.1	Schritt 1: Methodennamen anpassen	506
11.2.2	Schritt 2: Bound Properties hinzufügen	507
11.2.3	Schritt 3: Serialisierung ermöglichen	508
11.2.4	Schritt 4: Die Bean verpacken	508
11.3	Entwicklungshilfe	511
11.3.1	Ein Schritt zurück	511
11.3.2	Ab in die Beanbox!	512
11.3.3	Informationen für die Beanbox	516
11.3.4	Bearbeitung zur Entwurfszeit	525
11.4	Vergleich: Beans contra ActiveX	531
11.4.1	Kurze Vorstellung von ActiveX	531
11.4.2	Plattformen	532



11.4.3	Systemkonfiguration	532
11.4.4	Sicherheit	533
11.4.5	API und Programmiersprache	533
11.4.6	Programmierkonzept	534
11.4.7	Komponenten und Entwurf	534
11.4.8	Anspruch an den Entwickler	535
11.4.9	Geschwindigkeit und Größe	535
11.4.10	Fazit	536
<b>12</b>	<b>Grafik mit dem 2D-API</b>	<b>537</b>
12.1	Koordinatensysteme	538
12.2	Einfache Formen zeichnen	540
12.3	Verwenden der neuen 2D-API-Zeichenfunktionen	543
12.4	Figur + Figur = Constructive Area Geometry	545
12.5	Komplexere Zeichenobjekte – Kurven und General Paths	548
12.6	Transformationen	552
12.7	Fonts	555
12.8	Linien zeichnen – das Interface »Stroke«	557
12.9	Bilder, Farbverläufe – Formen füllen	561
12.10	Bildbearbeitung	566
12.11	Transparenz und die Klasse »AlphaComposite«	570
12.12	Bildbearbeitung mit der Klasse »BufferedImage«	572
12.13	Filter	575
12.14	Animationen unter Java	581
12.15	Clipping	584
12.16	Drucken	586
<b>13</b>	<b>Dreidimensionale Grafiken mit dem 3D-API</b>	<b>591</b>
13.1	Licht aus – Spot an! Noch mehr Beleuchtung	597
13.2	Weitere Objekte – die Interfaces »Shape3D« und »Primitive«	598
13.3	Das Aussehen von Objekten festlegen – Appearance	603
13.4	Jetzt kommt Bewegung ins Spiel – Transformation und Behavior	611
13.5	Die Klasse »Interpolator«	622
13.6	Interaktion	626
<b>14</b>	<b>Sound in Java</b>	<b>631</b>
14.1	Einbindung von Audio-Dateien	632
14.2	Abspielen, Anhalten und Wiederholen einer Audiodatei	632
<b>15</b>	<b>Internet-Programmierung in Java</b>	<b>635</b>
15.1	Das erste Netzwerkprogramm	637
15.2	Unser Client lernt »sprechen«	639
15.3	Unser erster Internet-Server...	642
15.4	Quasseln im Netz – der Chat-Server	647
15.5	Die Klassen URL und URLConnection	655



<b>16</b>	<b>Datenbanken und JDBC</b>	<b>659</b>
16.1	Eine Datenbank erstellen	660
16.1.1	Datenbank unter ODBC registrieren	662
16.1.2	Die Abfragemaske	662
16.2	Das CGI-Programm	663
16.3	In Datenbanken schreiben	670
<b>17</b>	<b>Java Virtual Machine Debugger Interface</b>	<b>675</b>
17.1	Anforderung und Freigabe von Speicher	679
17.2	Funktionen für Klassen	680
17.2.1	Status einer Klasse	680
17.2.2	Dateiname und Signatur einer Klasse	681
17.2.3	Zugriffsflags einer Klasse	681
17.2.4	Ermittlung von Variablen und Methoden einer Klasse	682
17.2.5	Interfaces einer Klasse	682
17.2.6	Weitere Informationen über Klassen	682
17.3	Funktionen für Variablen	683
17.3.1	Name und Signatur einer Variablen	683
17.3.2	Zugriffsflags einer Variablen	683
17.3.3	Weitere Informationen über Variablen	684
17.4	Funktionen für Methoden	685
17.4.1	Name und Signatur einer Methode	685
17.4.2	Zugriffsflags einer Methode	685
17.4.3	Startadresse der Methode ermitteln	686
17.4.4	Lokale Variablen einer Methode	686
17.4.5	Größe der lokalen Variablen und Methodenargumente	687
17.4.6	Quelltextzeilen ermitteln	687
17.4.7	Bytecode der Methode ermitteln	688
17.4.8	Exceptions	688
17.4.9	Weitere Informationen über Methoden	689
17.5	Breakpoints	689
17.6	Überwachung von Variablen	690
17.6.1	Überwachung des Zugriffs auf Variablen	690
17.6.2	Überwachung der Modifikation von Variablen	691
17.7	Threads	691
17.7.1	Status eines Threads	691
17.7.2	Alle Threads ermitteln	692
17.7.3	Steuerung von Threads	692
17.7.4	Informationen über einen Thread	693
17.8	Events	693
17.8.1	Aktivieren und Deaktivieren von Events	695
17.8.2	Behandlung von Events	695
17.9	Weitere Funktionen	700
17.9.1	Ermittlung aller geladener Klassen	700
17.9.2	JVMDI-Versionsnummer ermitteln	700
17.9.3	Eigenschaften des JVMDI ermitteln	700



<b>18</b>	<b>Java Native Interface</b>	<b>703</b>
18.1	Erstellen von Java-Klassen mit nativen Methoden	704
18.1.1	Schreiben der Java-Klasse	705
18.1.2	Erzeugen der Header-Datei	706
18.1.3	Implementierung der nativen Methoden	707
18.2	Mapping zwischen Java und nativen Datentypen	708
18.2.1	Basisdatentypen	708
18.2.2	Java-Klassen	709
18.3	Zugriff auf Java-Objekte in nativen Methoden	710
18.3.1	Java-Strings	710
18.3.2	Zugriff auf Java-Arrays	712
18.3.3	Aufruf von Java-Methoden	715
18.3.4	Zugriff auf Java-Variablen	720
18.3.5	Exceptions	721
<b>19</b>	<b>Sicherheit in Java</b>	<b>723</b>
19.1	Das Sandbox-Modell	725
19.2	Das neue Security-API im JDK 2	726
19.2.1	Das Security Package	726
19.2.2	Der Bytecode-Verifizierer	726
19.2.3	Der Class Loader	726
19.2.4	Der Access Controller	727
19.2.5	Der Security Manager	727
19.2.6	Die Schlüsseldatenbank (Keystore)	727
19.3	Der Security Manager	727
19.3.1	Zugriff auf den Security Manager	728
19.3.2	Implementierung von Security Managern	728
19.4	Der Access Controller	730
19.4.1	Permissions	731
19.4.2	Die Klasse CodeSource	735
19.4.3	Die Klasse Policy	736
19.4.4	Die Klasse ProtectionDomain	738
19.4.5	Die Klasse AccessController	738
<b>20</b>	<b>Tips und Tricks</b>	<b>741</b>
20.1	Sprache	742
20.1.1	Der Operator + für Strings	742
20.1.2	Die booleschen Operatoren	743
20.1.3	for-Schleife wie in C	744
20.1.4	Überschreiben von Methoden mit Exceptions	745
20.2	Exceptions in der main-Methode	746
20.3	Klassenbibliothek	746
20.3.1	Debugging mit »System.out.println«	746
20.3.2	Objekte kopieren	747
20.3.3	Vorsicht beim Vergleichen von Dateien!	747
20.4	Fehler und Fehlermeldungen	748
20.4.1	javac: »Can't make static reference to ...«	748



20.5	JDK	750
20.5.1	Behalten Sie das alte JDK!	750
20.5.2	Schrittweise Ausführung im jdb	751
20.5.3	Testen einer Klasse in einem Package	751
20.5.4	Arbeiten mit einem anderen CLASSPATH	751
20.5.5	API-Spezifikationen	752
20.6	Schneller Start unter Windows	753
20.7	Allgemeines	754
20.7.1	Helfen Sie, Bugs zu jagen!	754
<b>21</b>	<b>Kurzreferenz: Die Tools von Java</b>	<b>757</b>
21.1	appletviewer – Betrachter für Applets	758
21.2	extcheck – Verifizierung installierter Erweiterungen	759
21.3	jar – Das Java-Archivtool	759
21.4	jarsigner – Signierer für jar-Dateien	762
21.5	java – Der Java-Interpreter	763
21.6	javac – Der Java-Compiler	764
21.7	javadoc – Der Dokumentationsgenerator	767
21.8	javah – Generator für C-Header- und Stub-Dateien	772
21.9	javap – Der Java-Disassembler	774
21.10	jdb – Der Java-Debugger	776
21.11	keytool – Verwalten von Schlüsseln und Zertifikaten	779
21.12	native2ascii – Konvertierung von Quelldateien	782
21.13	policytool – Das Policy-Tool	782
21.14	rmic – Der RMI-Compiler	783
21.15	rmid – Der RMI-Dämon	785
21.16	rmiregistry – Registry für Remote-Objekte	785
21.17	serialver – Ermittlung der »serialVersionUID«	786
21.18	tnameserv – Der Transient Name Server	787
	<b>Stichwortverzeichnis</b>	<b>789</b>