

Objekttechnologie

Reihenherausgeber

Martin Schader, Mannheim

*Springer-Verlag
Berlin
Heidelberg
GmbH*

Objekttechnologie

Martin Schader und Michael Rundshagen
Objektorientierte Systemanalyse
2. Auflage
IX, 241 Seiten. 1996

Martin Schader
Objektorientierte Datenbanken
Die C++-Anbindung des
ODMG-Standards
X, 219 Seiten. 1997

Günther Vinek
Objektorientierte Softwareentwicklung
mit Smalltalk
XII, 440 Seiten. 1997

Martin Schader und Stefan Kuhlins
Programmieren in C++
Einführung in den Sprachstandard
5. Auflage
XII, 386 Seiten. 1998

Stefan Kuhlins und Martin Schader
Die C++-Standardbibliothek
Einführung und Nachschlagewerk
XIV, 384 Seiten. 1999

Martin Schader · Lars Schmidt-Thieme

JavaTM

Eine Einführung

Zweite, aktualisierte
und verbesserte Auflage

Mit 56 Abbildungen
und 22 Tabellen



Springer

Prof. Dr. Martin Schader
Universität Mannheim
Lehrstuhl für Wirtschaftsinformatik III
Schloß
D-68131 Mannheim

Dr. Lars Schmidt-Thieme
Universität Karlsruhe
Institut für Entscheidungstheorie und Unternehmensforschung
Kollegium am Schloß
D-76131 Karlsruhe

Additional material to this book can be downloaded from <http://extras.springer.com>

ISBN 978-3-540-65716-3 ISBN 978-3-662-00456-2 (eBook)
DOI 10.1007/978-3-662-00456-2

Die Deutsche Bibliothek – CIP-Einheitsaufnahme
Java™ [Medienkombination]: eine Einführung / Martin Schader; Lars Schmidt-Thieme. – Berlin; Heidelberg; New York; Barcelona; Hongkong; London; Mailand; Paris; Singapur; Tokio: Springer
(Objekttechnologie)

Buch. – 2., aktualisierte und verb. Aufl. – 1999

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland vom 9. September 1965 in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.

Der Springer-Verlag ist nicht Urheber der Daten und Programme. Weder der Springer-Verlag noch der Autor übernehmen Haftung für die CD-ROM und das Buch, einschließlich Qualität, Handels- oder Anwendungseignung. In keinem Fall übernehmen der Springer-Verlag oder der Autor Haftung für direkte, indirekte, zufällige oder Folgeschäden, die sich aus der Nutzung der CD-ROM oder des Buches ergeben.

© Springer-Verlag Berlin Heidelberg 1999
Ursprünglich erschienen bei Springer-Verlag Berlin Heidelberg 1999

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, daß solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürfen.

Umschlaggestaltung: Struwe & Partner, Heidelberg

SPIN 10720864

43/2202-5 4 3 2 1 0 – Gedruckt auf säurefreiem Papier

Vorwort zur zweiten Auflage

In die zweite Auflage haben wir einen Abschnitt über die neuen Java-Collection-Klassen aufgenommen. Entsprechend wurden alle Vector- und Hashtable-Objekte durch Collection-Objekte (Mengen, Listen usw.) ersetzt. Und zur Iteration über Collections werden nun Iteratoren und keine Enumerations mehr verwendet.

Weitere wesentliche Änderungen betreffen das RMI-Kapitel, das wir so umgeschrieben haben, daß die Beispiele mit den neuen Java 2-Systemen lauffähig sind.

Die Java-Quellen zum Buch sowie die Lösungen der schwierigeren Übungsaufgaben findet man wieder auf der beiliegenden CD-ROM. Sofern Anregungen oder Verbesserungsvorschläge seitens der Leser zu Code-Veränderungen führen, werden diese für *Anonymous ftp* auf unserem Server ([ftp.wifo.uni-mannheim.de](ftp:wifo.uni-mannheim.de)) im Verzeichnis `/pub/buecher` zugänglich gemacht.

Dem Springer-Verlag, insbesondere Frau Dr. Bihn und ihren Mitarbeitern, gilt unser Dank für die gute Zusammenarbeit.

Mannheim, März 1999

Martin Schader, Lars Schmidt-Thieme

Vorwort zur ersten Auflage

Seit der Verfügbarkeit des ersten Java Development Kits (JDK) im Jahr 1995 hat die Programmiersprache Java einen Erfolg erlebt, wie er auf dem Gebiet der Softwaretechnologie bisher ohnegleichen ist.

Es werden nicht nur ständig neue, komfortable Java-Entwicklungsumgebungen vorgestellt, auch die Codegenerierung der wichtigsten CASE-Tools wird auf Java umgestellt, die Hersteller objektorientierter Datenbankmanagementsysteme bieten Java-Anbindungen (meist nach dem ODMG-Standard) an, bei der Entwicklung verteilter Systeme, insbesondere unter Benutzung von Object Request Brokern, führt an Java kein Weg vorbei, und auch auf dem Gebiet der Komponententechnologie ist mit JavaBeans eine ernstzunehmende Alternative zum bisher Vorhandenen entstanden.

Einige der in diesem Zusammenhang immer wieder genannten Schlagworte sind: Java ist *einfach, objektorientiert, verteilt, robust, sicher, architekturneutral, portabel, interpretiert, leistungsfähig, multithreaded, dynamisch*. Aus unserer Sicht kommt dabei insbesondere der Plattformunabhängigkeit und der "write once, run everywhere"-Zielsetzung eine besondere Bedeutung zu.

Das vorliegende Buch ist aus einem Vorlesungszyklus „Objekttechnologie“, der an der Universität Mannheim gehalten wird, entstanden. Es wendet sich an Leser, die über Grundkenntnisse von Rechnern und ihrer Programmierung verfügen. Die einzelnen Sprachkonstrukte werden anhand vieler kleiner Beispielprogramme und einiger umfangreicherer Anwendungen erklärt. Die Beispiele sind in der Regel bezüglich der Fehlerbehandlung äußerst kurz gehalten. Hier muß von den Leserinnen und Lesern noch eigenständig weiterentwickelt werden.

Alle Beispiele wurden mit den von Sun zur Verfügung gestellten JDKs für Win 95/NT und Solaris getestet, viele auch mit MRJ unter MacOS 8. Diese JDKs findet man im WWW unter <http://www.javasoft.com> bzw. <http://www.apple.com>.

Am Ende der Kapitel ist jeweils eine Reihe von Übungsaufgaben zusammengestellt, mit denen der behandelte Stoff vertieft werden kann. Lösungen zu ausgewählten Aufgaben und die Java-Quellen der Beispiele findet man auf der beiliegenden CD-ROM. Aktuelle, ergänzende Informationen sind mittels *Anonymous ftp* auf unserem Server ([ftp.wifo.uni-mannheim.de](ftp:wifo.uni-mannheim.de)) im Verzeichnis `/pub/buecher` sowie im WWW unter <http://www.wifo.uni-mannheim.de/Java> zugänglich.

Die Beispiele für das Netzwerk-Kapitel (20) und das RMI-Kapitel (21) kann man auf unserem Server über das Internet testen; nähere Beschreibungen erhält man ebenfalls unter dem obigen URL.

Über Anregungen und Verbesserungsvorschläge seitens der Leser an unsere Postanschrift oder an javabuch@wifo.uni-mannheim.de würden wir uns freuen.

Der Fa. COMPAS MEDIA, Bremen danken wir für die Genehmigung, eine Reihe von Bilddateien bereitstellen zu können. Dem Springer-Verlag, Dr. Bihn, Dr. Müller und ihren Mitarbeitern gilt unser Dank für die bewährt sehr gute Zusammenarbeit.

Mannheim, Mai 1998

Martin Schader, Lars Schmidt-Thieme

Inhaltsverzeichnis

1	Einleitung	1
1.1	Erste Beispiele	1
1.2	Die Ausführung von Java-Programmen	9
1.3	Übungsaufgaben	10
2	Lexikalische Konventionen	13
2.1	Lexikalische Elemente	13
2.2	Kommentare	14
2.3	Bezeichner	17
2.4	Schlüsselwörter	17
2.5	Interpunktionszeichen	17
2.6	Operatoren	18
2.7	Syntaxnotation	18
2.8	Übungsaufgabe	19
3	Typen und Werte	21
3.1	Datentypen	21
3.2	Wertebereiche	23
4	Konstanten und Variablen	25
4.1	Ganzzahlige Konstanten	25
4.2	Gleitpunktkonstanten	26

4.3	Zeichenkonstanten	27
4.4	Zeichenketten	27
4.5	Die Nullreferenz	28
4.6	Variablen	28
4.7	Symbolische Konstanten	31
4.8	Die Initialisierung von Variablen	32
4.9	Übungsaufgaben	33
5	Typumwandlungen, Ausdrücke und Operatoren	35
5.1	Typumwandlungen	35
5.1.1	Elementare Typvergrößerungen	37
5.1.2	Elementare Typverkleinerungen	38
5.1.3	Vergrößerungen von Referenztypen	39
5.1.4	Verkleinerungen von Referenztypen	39
5.1.5	Typumwandlungen nach String	40
5.2	Konversionskontexte	40
5.2.1	Zuweisungskonversionen	40
5.2.2	Methodenaufruf-Konversionen	41
5.2.3	String-Konversionen	42
5.2.4	Numerische Typangleichungen	43
5.3	Explizite Casts	44
5.4	Ausdrücke und Operatoren	45
5.4.1	Elementare Ausdrücke und Namen	46
5.4.2	Postfix-Inkrement und Dekrement	47
5.4.3	Einstellige Operatoren	47
5.4.4	Multiplikative Operatoren	48
5.4.5	Additive Operatoren	49
5.4.6	Shift-Operatoren	50

5.4.7	Relationale Operatoren	51
5.4.8	Gleichheitsoperatoren	51
5.4.9	Bit-Operatoren und logische Operatoren	52
5.4.10	Boolesche Operatoren	53
5.4.11	Der Konditional-Operator	54
5.4.12	Zuweisungsoperatoren	55
5.4.13	Konstante Ausdrücke	55
5.4.14	Zuweisungen und Initialisierungen	56
5.5	Übungsaufgaben	57
6	Anweisungen	61
6.1	Lokale Variablen	61
6.1.1	final lokale Variablen	63
6.2	Leeranweisungen	64
6.3	Ausdrucksanweisungen	64
6.4	Auswahanweisungen	64
6.4.1	Die if-Anweisung	65
6.4.2	Die switch-Anweisung	65
6.5	Wiederholungsanweisungen	67
6.5.1	Die while-Anweisung	67
6.5.2	Die do-Anweisung	68
6.5.3	Die for-Anweisung	69
6.6	Markierte Anweisungen	70
6.7	Die break-Anweisung	71
6.8	Die continue-Anweisung	72
6.9	Weitere Java-Anweisungen	73
6.10	Übungsaufgaben	73

7	Felder	77
7.1	Feldvariablen	77
7.2	Die Erzeugung von Feldern	78
7.3	Der Zugriff auf Feldkomponenten	79
7.4	Mehrdimensionale Felder	80
7.5	Felder als Objekte	83
7.6	Typumwandlungen	86
7.7	Felder und Zeichenketten	86
7.8	Übungsaufgaben	87
8	Klassen und Objekte	91
8.1	Klassendeklarationen	92
8.2	Der Zugriff auf Klasselemente	93
8.3	Instanz- und Klassenvariablen	96
8.4	Die Initialisierung von Variablen	97
8.5	Konstante Klasselemente	99
8.6	Methoden	100
8.6.1	Methodendeklaration	100
8.6.2	Methodenaufruf	101
8.6.3	Die return-Anweisung	102
8.7	Instanz- und Klassenmethoden	104
8.8	Überladene Methoden	106
8.9	Die Konstruktion von Objekten	109
8.10	static Initialisierer	112
8.11	Objektzerstörung	113
8.12	Übungsaufgaben	115

9	Subklassen, Superklassen und Vererbung	119
9.1	Vererbung	119
9.2	Verdeckte Variablen	122
9.3	Umwandlungen von Referenztypen	123
9.4	Überschriebene Methoden	127
9.4.1	static Methoden	133
9.4.2	Methodenaufrufe mittels <code>super</code>	134
9.5	final Methoden und Klassen	135
9.6	Die Konstruktion von Objekten	138
9.7	Abstrakte Methoden und Klassen	139
9.8	Typinformationen zur Laufzeit	143
9.9	Übungsaufgaben	145
10	Pakete, Geltungsbereiche und Zugreifbarkeit	149
10.1	Pakete	149
10.2	Geltungsbereiche	152
10.3	Zugriffsrechte	153
10.3.1	Zugriffsrechte und Vererbung	158
10.4	Import-Deklarationen	160
10.5	Die Suche nach <code>class</code> -Dateien	162
10.6	Sinnvolle Konventionen	163
10.7	Übungsaufgaben	166
11	Interfaces	169
11.1	Interfacedeklarationen	170
11.2	Interfaceelemente	171
11.2.1	Interfacemethoden	171
11.2.2	Interfacevariablen	172
11.3	Die Implementation von Interfaces	172

11.4	Sub- und Superinterfaces	175
11.5	Mehrdeutigkeiten	176
11.6	Interfaces aus der Java-Bibliothek	180
11.7	Übungsaufgaben	182
12	Eingebettete Klassendeklarationen	185
12.1	Innere Klassen	185
12.2	Anonyme Klassen	189
12.3	Eingebettete static Klassen	191
12.4	Übungsaufgaben	193
13	Das Abstract Window Toolkit, Applets und Frames (Teil I)	195
13.1	Applets: Deklaration und wichtigste Aktivitäten	196
13.2	Applet-Parameter	199
13.3	Java-Archive	201
13.4	AWT-Komponenten	202
13.5	AWT-Ereignisse	206
13.6	Komponenten für Benutzerschnittstellen (Teil I)	215
13.6.1	Label	215
13.6.2	TextField	215
13.6.3	Checkbox und CheckboxGroup	216
13.6.4	Choice	217
13.6.5	Button	217
13.7	Ereignisverarbeitung	218
13.8	Layout-Manager	221
13.8.1	Das FlowLayout	221
13.8.2	Das BorderLayout	222
13.8.3	Das GridLayout	223
13.8.4	Kombinationen	223

13.9 Von Applets zu Frames	225
13.10 Insets	229
13.11 Focus	230
13.12 Übungsaufgaben	231
14 Grundlegende Klassen	235
14.1 Die Klasse <code>String</code>	235
14.2 Die Klasse <code>StringBuffer</code>	238
14.3 Die Klasse <code>Math</code>	240
14.4 Die Klasse <code>DecimalFormat</code>	241
14.5 Hüllklassen für elementare Datentypen	242
14.6 Die Klassen <code>Date</code> und <code>DateFormat</code>	245
14.7 <code>Calendar</code> -Klassen	248
14.8 Länderspezifische Einstellungen mittels <code>Locale</code>	249
14.8.1 Internationalisierung mit <code>ResourceBundles</code>	254
14.9 <code>Collection</code> -Klassen	260
14.9.1 <code>Set</code>	261
14.9.2 <code>SortedSet</code> und <code>Comparable</code>	263
14.9.3 <code>List</code>	265
14.10 <code>Map</code> -Klassen	268
14.10.1 <code>HashMap</code>	269
14.10.2 <code>TreeMap</code>	271
14.11 Die Klasse <code>System</code>	271
14.12 Übungsaufgaben	273
15 Ausnahmebehandlung	277
15.1 Einleitung	277
15.2 Ausnahmetypen	278
15.3 Die Behandlung von Ausnahmen	279

15.4	Die <code>throws</code> -Klausel	284
15.5	Ausnahmen explizit auswerfen	286
15.6	Überschriebene Methoden und <code>throws</code> -Klauseln	287
15.7	Übungsaufgaben	291
16	Ein- und Ausgabeströme	295
16.1	Einleitung	295
16.2	Byteorientierte Ein- und Ausgabe	297
16.3	Zeichenorientierte Ein- und Ausgabe	299
16.4	Ein- und Ausgabe mit Dateien	305
16.5	Die Klasse <code>File</code>	307
16.6	Das Serialisieren von Objekten	309
16.6.1	Persistenz durch Erreichbarkeit	313
16.6.2	Klassenspezifische Versionen von <code>writeObject</code> und <code>readObject</code>	316
16.7	Übungsaufgaben	318
17	Threads und Prozesse	321
17.1	Threads	321
17.2	Eigenschaften eines <code>Thread</code> -Objekts	327
17.3	<code>Thread</code> -Scheduling, <code>Thread</code> -Prioritäten	330
17.4	<code>Thread</code> gruppen	336
17.5	<code>Thread</code> -Synchronisation	337
17.6	Die Kommunikation zwischen <code>Threads</code>	345
17.7	<code>Threads</code> und Applets	351
17.8	Systemprozesse	355
17.9	Übungsaufgaben	357

18 Das Abstract Window Toolkit, Applets und Frames (Teil II)	361
18.1 Aufbau von Benutzerschnittstellen (Teil II)	361
18.1.1 TextArea	361
18.1.2 List	363
18.1.3 ScrollPane und Scrollbar	365
18.1.4 Component: Rekapitulation und weitere Details	368
18.1.5 Window	371
18.1.6 Frame	372
18.1.7 Dialog	373
18.1.8 FileDialog	375
18.1.9 Menüs	379
18.1.10 Popup-Menüs	385
18.1.11 Panel- und Canvas-Objekte	387
18.2 Die Klasse Cursor	388
18.3 Peers	389
18.4 Übungsaufgaben	390
19 Zeichnen, Image- und Audioverarbeitung	393
19.1 Elementare Grafik	393
19.1.1 Geometrische Figuren	394
19.1.2 Schriften und Fonts	397
19.1.3 Farben	401
19.2 Bilder laden und anzeigen	403
19.2.1 Grundlegende Methoden	403
19.2.2 Details zum Ladevorgang	406
19.3 Animationen	413
19.3.1 Daumenkino-Animationen	413
19.3.2 Das Flackern vermeiden – Doublebuffering	417

19.3.3	Die Beschleunigung von Animationen durch Clipping	423
19.4	Filter	427
19.4.1	Die Verwendung fertiger Filter	429
19.4.2	Farbfilter	432
19.4.3	Allgemeine Filter	434
19.4.4	Zugriffe auf Pixeldaten	437
19.5	Der Ausdruck von Grafiken	438
19.6	Elementare Klangunterstützung für Applets	441
19.7	Transparente Lightweight-Komponenten	442
19.8	Übungsaufgaben	444
20	Netzwerke, Client/Server-Programmierung	447
20.1	Zugriffe auf den DNS	450
20.2	TCP/IP-Verbindungen	451
20.2.1	Client-Sockets	452
20.2.2	Server-Sockets	456
20.2.3	Server mit mehreren Clients	459
20.2.4	Ein einfaches Anwendungsprotokoll über TCP/IP	463
20.3	Datenübertragung mittels UDP/IP	466
20.4	Zugriffe auf Netzressourcen über die Klasse URL	471
20.4.1	Die Bestandteile eines URLs	471
20.4.2	Das Lesen von Ressourcen	473
20.5	Dynamisches HTML und Servlets	477
20.6	Übungsaufgaben	484
21	Methodenaufrufe für entfernte Objekte (RMI)	487
21.1	Methodenaufrufe für entfernte Objekte	487
21.2	Das Starten der Registry innerhalb des Servers	496
21.3	Kopien und Referenzen für entfernte Objekte	497

21.4	Callbacks	503
21.5	Netzwerkmethoden in Applets	507
21.6	Übungsaufgaben	510
Anhang		513
A	Die Java-Syntaxregeln	513
B	ASCII-Tabelle	526
C	Konversionen von Referenztypen	527
D	Operatorprioritäten	528
E	Serialisierbare Klassen	529
F	Locale-Konstanten	530
G	Wichtige HTML-Markierungen	531
Index		537