

Inhaltsverzeichnis

Vorwort	17
Einführung	19
1 Erste Schritte	25
1.1 Das Phänomen Java	26
1.1.1 Wissenswertes zu Java	26
1.1.2 Was kann Java?	28
1.1.3 Wie kann Java mein Leben ändern?	29
1.2 Die Anwendung Hello World!	30
1.2.1 Hello World! für die NetBeans IDE	30
1.2.2 Hello World! für Microsoft Windows	38
1.2.3 Hello World! für Solaris und Linux	43
1.3 Die Anwendung Hello World! unter der Lupe	48
1.3.1 Kommentare im Quellcode	48
1.3.2 Definition der Klasse HelloWorldApp	49
1.3.3 Die Methode main()	49
1.4 Häufige Probleme (und ihre Lösungen)	51
1.4.1 Probleme mit dem Compiler	51
1.4.2 Laufzeitprobleme	53
1.5 Fragen und Übungen: Erste Schritte	55
1.5.1 Fragen	55
1.5.2 Übungen	55
2 Konzepte der objektorientierten Programmierung	57
2.1 Was ist ein Objekt?	58
2.2 Was ist eine Klasse?	60
2.3 Was ist Vererbung?	62
2.4 Was ist eine Schnittstelle?	63
2.5 Was ist ein Paket?	64
2.6 Fragen und Übungen: Konzepte der objektorientierten Programmierung	65
2.6.1 Fragen	65
2.6.2 Übungen	65



3	Sprachgrundlagen	67
3.1	Variablen	68
3.1.1	Benennungen	69
3.1.2	Primitive Datentypen	70
3.1.3	Arrays	74
3.1.4	Zusammenfassung zu den Variablen	78
3.1.5	Fragen und Übungen: Variablen	79
3.2	Operatoren	79
3.2.1	Zuweisungs-, Rechen- und unäre Operatoren	81
3.2.2	Gleichheits-, Vergleichs- und Bedingungsoperatoren	84
3.2.3	Bitverknüpfende und Bitverschiebungsoperatoren	87
3.2.4	Zusammenfassung zu den Operatoren	88
3.2.5	Fragen und Übungen: Operatoren	90
3.3	Ausdrücke, Anweisungen und Blöcke	91
3.3.1	Anweisungen	93
3.3.2	Blöcke	93
3.3.3	Fragen und Übungen: Ausdrücke, Anweisungen und Blöcke	94
3.4	Programmablaufanweisungen	94
3.4.1	Die Anweisungen if und if-else	95
3.4.2	Die Anweisung switch	97
3.4.3	Die Anweisungen while und do-while	99
3.4.4	Die Anweisung for	100
3.4.5	Verzweigungsanweisungen	103
3.4.6	Zusammenfassung zu den Programmablaufanweisungen	107
3.4.7	Fragen und Übungen: Programmablaufanweisungen	107
4	Klassen und Objekte	109
4.1	Klassen	110
4.1.1	Klassen deklarieren	111
4.1.2	Member-Variablen deklarieren	112
4.1.3	Methoden definieren	114
4.1.4	Konstruktoren für Ihre Klassen bereitstellen	117
4.1.5	Daten an eine Methode oder einen Konstruktor übergeben	118
4.2	Objekte	122
4.2.1	Objekte erstellen	124
4.2.2	Objekte einsetzen	128
4.3	Mehr Wissenswertes zu Klassen	131
4.3.1	Einen Wert von einer Methode zurückgeben	132
4.3.2	Das Schlüsselwort this verwenden	134
4.3.3	Den Zugriff auf die Members einer Klasse steuern	136



4.3.4	Instanz- und Klassen-Member verstehen	138
4.3.5	Felder initialisieren	142
4.3.6	Zusammenfassung zur Erstellung und Verwendung von Klassen und Objekten.	144
4.3.7	Fragen und Übungen: Klassen	145
4.3.8	Fragen und Übungen: Objekte	146
4.4	Verschachtelte Klassen.	147
4.4.1	Wofür man verschachtelte Klassen benötigt.	148
4.4.2	Statische verschachtelte Klassen.	148
4.4.3	Innere Klassen.	149
4.4.4	Zusammenfassung zu den verschachtelten Klassen	153
4.4.5	Fragen und Übungen: verschachtelte Klassen	154
4.5	Aufzählungstypen	154
4.5.1	Fragen und Übungen: Aufzählungstypen	158
4.6	Annotationen.	158
4.6.1	Dokumentation	159
4.6.2	Vom Compiler verwendete Annotationen	160
4.6.3	Annotationsprozessoren	162
4.6.4	Fragen und Übungen: Annotationen.	162
5	Schnittstellen und Vererbung	165
5.1	Schnittstellen.	166
5.1.1	Schnittstellen in Java.	166
5.1.2	Schnittstellen als APIs	168
5.1.3	Schnittstellen und Mehrfachvererbung.	168
5.1.4	Schnittstelle definieren	168
5.1.5	Schnittstelle implementieren.	169
5.1.6	Schnittstelle als Typ verwenden	171
5.1.7	Schnittstellen umformulieren	173
5.1.8	Zusammenfassung zu den Schnittstellen	173
5.1.9	Fragen und Übungen: Schnittstellen.	174
5.2	Vererbung	174
5.2.1	Die Klassenhierarchie in Java.	175
5.2.2	Was Sie mit einer abgeleiteten Klasse tun können	177
5.2.3	Private Members in einer Basisklasse.	178
5.2.4	Objekt umwandeln	178
5.2.5	Methoden überschreiben und verbergen.	180
5.2.6	Felder verbergen	182
5.2.7	Das Schlüsselwort super verwenden.	182
5.2.8	Object als Basisklasse.	184



- 5.2.9 Finale Klassen und Methoden schreiben 188
- 5.2.10 Abstrakte Methoden und Klassen. 189
- 5.2.11 Zusammenfassung zur Vererbung 192
- 5.2.12 Fragen und Übungen: Vererbung. 193

- 6 Generics** 195
 - 6.1 Einleitung 196
 - 6.1.1 Eine einfache Klasse Box. 196
 - 6.2 Generische Typen. 198
 - 6.2.1 Konventionen zu Typparametern. 200
 - 6.3 Generische Methoden und Konstruktoren 201
 - 6.4 Eingeschränkte Typparameter (Bounds) 202
 - 6.5 Zuweisungskompatibilität 204
 - 6.6 Platzhalter. 206
 - 6.7 Type-Erasure 207
 - 6.8 Zusammenfassung zu den Generics 209
 - 6.8.1 Fragen und Übungen: Generics 210

- 7 Packages**. 213
 - 7.1 Packages erstellen und verwenden. 214
 - 7.1.1 Package erstellen 215
 - 7.1.2 Package benennen 217
 - 7.1.3 Package-Members verwenden 218
 - 7.1.4 Quell- und Klassendateien verwalten. 222
 - 7.1.5 Zusammenfassung zur Erstellung und Verwendung von Packages. 224
 - 7.1.6 Fragen und Übungen: Packages erstellen und verwenden 225

- 8 Zahlen und Strings**. 227
 - 8.1 Zahlen. 228
 - 8.1.1 Die Zahlenklassen 228
 - 8.1.2 Numerische Ausgaben formatieren 231
 - 8.1.3 Komplexere Arithmetik 236
 - 8.1.4 Zusammenfassung zu den Zahlen 241
 - 8.1.5 Fragen und Übungen: Zahlen. 242
 - 8.2 Zeichen. 243
 - 8.2.1 Escape-Sequenzen 244
 - 8.3 Strings. 245
 - 8.3.1 Strings erstellen 245
 - 8.3.2 String-Länge. 246
 - 8.3.3 Strings verketteten. 247
 - 8.3.4 Format-Strings erstellen 248



8.3.5	Zwischen Zahlen und Strings konvertieren	249
8.3.6	Zeichen in einem String manipulieren	251
8.3.7	Strings und Teil-Strings vergleichen	257
8.3.8	Die Klasse StringBuilder	259
8.3.9	Zusammenfassung zu Zeichen und Strings	263
8.3.10	Fragen und Übungen: Zeichen und Strings	264
9	Exceptions	267
9.1	Was ist eine Exception?	268
9.2	Das Catch or Specify-Prinzip	269
9.2.1	Die Exception-Typen	270
9.2.2	Catch or Specify umgehen	271
9.3	Exceptions abfangen und behandeln	271
9.3.1	Der try-Block	272
9.3.2	Die catch-Blöcke	273
9.3.3	Der finally-Block	274
9.3.4	Es wächst zusammen, was zusammengehört	276
9.4	Von einer Methode auslösbare Exceptions angeben	279
9.5	Exceptions auslösen	280
9.5.1	Die Anweisung throw	281
9.5.2	Die Klasse throwable und von ihr abgeleitete Klassen	282
9.5.3	Die Klasse Error	282
9.5.4	Die Klasse Exception	283
9.5.5	Verkettete Exceptions	283
9.5.6	Exception-Klassen erstellen	285
9.5.7	Beispiel	285
9.6	Ungeprüfte Exceptions: die Kontroverse	286
9.7	Vorteile von Exceptions	287
9.7.1	Vorteil 1: Trennung des Fehlerbehandlungscodes von »regulärem« Code	287
9.7.2	Vorteil 2: Weitergabe von Fehlern im Aufrufstapel nach oben durchreichen	290
9.7.3	Vorteil 3: Gruppierung und Differenzierung von Fehlertypen	291
9.8	Zusammenfassung	293
9.9	Fragen und Übungen: Exceptions	294
9.9.1	Fragen	294
9.9.2	Übungen	295
10	Grundlagen zur Ein- und Ausgabe	297
10.1	Ein-/Ausgabe-Datenströme	298
10.1.1	Byteströme	299
10.1.2	Zeichenströme	302



10.1.3	Gepufferte Datenströme	305
10.1.4	Scannen und Formatieren	306
10.1.5	Ein- und Ausgabe über die Befehlszeile	313
10.1.6	Datenströme für Daten	316
10.1.7	Objektströme	318
10.2	Ein- und Ausgabe für Dateien	322
10.2.1	Dateiobjekte	323
10.2.2	Dateien mit wahlfreiem Zugriff	326
10.3	Die neuen Ein-/Ausgabe-Packages	328
10.4	Zusammenfassung	328
10.5	Fragen und Übungen: Grundlagen zur Ein- und Ausgabe	329
10.5.1	Fragen	329
10.5.2	Übungen	329
11	Collections	331
11.1	Einführung in die Collections	332
11.1.1	Was ist ein Collection-Framework?	332
11.1.2	Vorteile des Java Collections Framework	333
11.2	Schnittstellen	334
11.2.1	Die Schnittstelle Collection	336
11.2.2	Die Schnittstelle Set	340
11.2.3	Die Schnittstelle List	345
11.2.4	Die Schnittstelle Queue	355
11.2.5	Die Schnittstelle Map	357
11.2.6	Objektsortierung	367
11.2.7	Die Schnittstelle SortedSet	374
11.2.8	Die Schnittstelle SortedMap	377
11.2.9	Zusammenfassung zu den Schnittstellen	379
11.2.10	Fragen und Übungen: Schnittstellen	379
11.3	Implementierungen	380
11.3.1	Set-Implementierungen	383
11.3.2	List-Implementierungen	385
11.3.3	Map-Implementierungen	386
11.3.4	Queue-Implementierungen	387
11.3.5	Wrapper-Implementierungen	389
11.3.6	Convenience-Implementierungen	391
11.3.7	Zusammenfassung zu den Implementierungen	393
11.3.8	Fragen und Übungen: Implementierungen	394
11.4	Algorithmen	394
11.4.1	Sortieren	395



11.4.2	Mischen	397
11.4.3	Routinemäßige Datenmanipulation	397
11.4.4	Suchen	398
11.4.5	Zusammensetzung	398
11.4.6	Extremwerte ermitteln	398
11.5	Benutzerdefinierte Collection-Implementierungen	399
11.5.1	Gründe zum Schreiben einer Implementierung	399
11.5.2	Benutzerdefinierte Implementierungen schreiben	400
11.6	Interoperabilität	402
11.6.1	Kompatibilität	402
11.6.2	API-Aufbau	404
12	Nebenläufigkeit	407
12.1	Prozesse und Threads	408
12.1.1	Prozesse	408
12.1.2	Threads	409
12.2	Thread-Objekte	409
12.2.1	Threads definieren und starten	410
12.2.2	Ausführung mit sleep() unterbrechen	411
12.2.3	Interrupts	412
12.2.4	Joins	413
12.2.5	Das Beispielprogramm SimpleThreads	414
12.3	Synchronisierung	416
12.3.1	Thread-Interferenz	416
12.3.2	Speicherkonsistenzfehler	417
12.3.3	Synchronisierte Methoden	418
12.3.4	Intrinsische Sperren und Synchronisierung	420
12.3.5	Synchronisierte Anweisungen	420
12.3.6	Atomarer Zugriff	422
12.4	Lebendigkeit	423
12.4.1	Deadlocks	423
12.4.2	Aushungern und Livelock	424
12.5	Guarded-Blöcke	425
12.6	Unveränderliche Objekte	429
12.6.1	Beispiel für eine synchronisierte Klasse	430
12.6.2	Eine Strategie zur Definition unveränderlicher Objekte	432
12.7	Objekte höherer Nebenläufigkeit	434
12.7.1	Sperrobjekte	434
12.7.2	Ausführer	437
12.7.3	Nebenläufige Collections	439



12.7.4	Atomare Variablen	440
12.8	Weiterführende Lektüre	441
12.9	Fragen und Übungen: Nebenläufigkeit	442
12.9.1	Fragen.	442
12.9.2	Übungen	442
13	Reguläre Ausdrücke	443
13.1	Einleitung	444
13.1.1	Was sind reguläre Ausdrücke?	444
13.1.2	Wie werden reguläre Ausdrücke in diesem Package dargestellt?	444
13.2	Testaufbau	445
13.3	String-Literale	446
13.3.1	Metazeichen	447
13.4	Zeichenklassen	448
13.4.1	Einfache Klassen	448
13.4.2	Negation	449
13.4.3	Bereiche	449
13.4.4	Unions	450
13.4.5	Schnittmengen	451
13.4.6	Subtraktion	452
13.5	Vordefinierte Zeichenklassen.	452
13.6	Quantoren	455
13.6.1	Vergleiche mit Nulllänge	455
13.6.2	Capturing-Groups und Zeichenklassen mit Quantoren	459
13.6.3	Unterschiede zwischen Greedy-, Reluctant- und Possessive-Quantoren.	460
13.7	Capturing-Groups.	461
13.7.1	Nummerierung	461
13.7.2	Rückwärtsreferenzen	462
13.8	Boundary-Matcher	463
13.9	Die Methoden der Klasse Pattern.	465
13.9.1	Muster mit Flags erstellen	465
13.9.2	Eingebettete Flag-Ausdrücke	467
13.9.3	Die Methode matches(String,CharSequence) verwenden	468
13.9.4	Die Methode split(String) verwenden	468
13.9.5	Weitere Hilfsmethoden.	469
13.9.6	Äquivalente zu Pattern-Methoden in java.lang.String	469
13.10	Die Methoden der Klasse Matcher.	470
13.10.1	Indexmethoden	470
13.10.2	Untersuchungsmethoden	470
13.10.3	Ersetzungsmethoden.	471



13.10.4	Die Methoden start() und end() verwenden	471
13.10.5	Die Methoden matches() und lookingAt() verwenden	472
13.10.6	replaceFirst(String) und replaceAll(String) verwenden	473
13.10.7	Die Methoden appendReplacement() und appendTail()	474
13.10.8	Äquivalente zu den Matcher-Methoden in java.lang.String	475
13.11	Die Methoden der Klasse PatternSyntaxException	475
13.12	Zusammenfassung	477
13.13	Leseempfehlung	478
13.14	Fragen und Übungen: reguläre Ausdrücke	478
13.14.1	Fragen	478
13.14.2	Übungen	479
14	Die Plattformumgebung	481
14.1	Konfigurationshilfsmittel	482
14.1.1	Eigenschaften	482
14.1.2	Befehlszeilenargumente	486
14.1.3	Umgebungsvariablen	488
14.1.4	Weitere Konfigurationshilfsmittel	489
14.2	Systemhilfsmittel	490
14.2.1	Befehlszeilenbasierte Ein-/Ausgabe-Objekte	490
14.2.2	Systemeigenschaften	490
14.2.3	Der Sicherheitsmanager	494
14.2.4	Verschiedene Methoden in System	495
14.3	PATH und CLASSPATH	496
14.3.1	PATH unter Microsoft Windows NT/2000/XP aktualisieren	497
14.3.2	Path unter Solaris und Linux aktualisieren	497
14.3.3	CLASSPATH überprüfen	498
14.4	Fragen und Übungen: Die Plattformumgebung	499
14.4.1	Fragen	499
14.4.2	Übungen	500
15	Swing	501
15.1	Eine kurze Einführung in das Swing-Package	502
15.1.1	Was ist Swing?	502
15.1.2	Swing: eine Demonstration	505
15.2	Eigenschaften von Swing	509
15.2.1	Eine bebilderte Übersicht zu den Swing-Komponenten	510
15.2.2	Pluggable Look and Feel	517
15.2.3	Drag&Drop und Datenübergabe	521
15.2.4	Internationalisierung und Lokalisierung	521



15.2.5	Eingabehilfen	523
15.2.6	Integration mit dem Desktop	524
15.2.7	Unterstützung für Symbole im Tray	525
15.3	Fragen: Grafische Bedienoberflächen	526
15.3.1	Fragen	526
16	Programme in JAR-Dateien packen	527
16.1	JAR-Dateien verwenden: die Grundlagen	529
16.1.1	JAR-Datei erstellen	530
16.1.2	Inhalt einer JAR-Datei anzeigen	534
16.1.3	Inhalt einer JAR-Datei extrahieren	536
16.1.4	JAR-Datei aktualisieren	537
16.1.5	JAR-gepackte Software ausführen	539
16.2	Mit Manifestdateien arbeiten: die Grundlagen	541
16.2.1	Das Standardmanifest verstehen	541
16.2.2	Manifestdatei bearbeiten	542
16.2.3	Eintrittspunkt einer Anwendung einstellen	543
16.2.4	Klassen zum Klassenpfad der JAR-Datei hinzufügen	544
16.2.5	Package-Versionsinformationen einstellen	546
16.2.6	Package-Sealing in einer JAR-Datei	547
16.3	JAR-Dateien signieren und verifizieren	549
16.3.1	Signierung und Verifizierung verstehen	549
16.3.2	JAR-Dateien signieren	552
16.3.3	Signierte JAR-Dateien verifizieren	554
16.4	JAR-spezifische APIs verwenden	555
16.4.1	Die Anwendung JarRunner	556
16.4.2	Die Klasse JarClassLoader	556
16.4.3	Die Klasse JarRunner	560
16.5	Fragen: JAR-Dateien	562
16.5.1	Fragen	562
17	Java Web Start	563
17.1	Java Web Start-Anwendungen ausführen	565
17.1.1	Java Web Start-Anwendung aus einem Browser heraus ausführen	565
17.1.2	Anwendung aus dem Java Cache Viewer heraus starten	566
17.1.3	Java Web Start-Anwendung über den Desktop starten	567
17.2	Java Web Start-Anwendungen bereitstellen	567
17.2.1	Webserver einrichten	567
17.2.2	JNLP-Datei erstellen	568
17.2.3	Anwendung auf dem Webserver ablegen	578



17.2.4	Webseite erstellen	578
17.3	Java Web Start-Anwendungen entwickeln	579
17.3.1	Anwendungen in JAR-Dateien packen	579
17.3.2	Ressourcen in einer JAR-Datei lesen	579
17.3.3	Nicht vertrauenswürdige Anwendungen	580
17.4	Das JNLP-API	580
17.5	Java Web Start und Sicherheit	582
17.5.1	Grundlagen zur Java Web Start-Sicherheit	582
17.5.2	JAR-Dateien für Java Web Start signieren	582
17.5.3	Sicherheit und JNLP-Dateien	583
17.5.4	Dynamisches Herunterladen von HTTPS-Zertifikaten	583
17.6	Häufig auftretende Probleme bei Java Web Start	584
17.7	Fragen und Übungen: Java Web Start	585
17.7.1	Fragen	585
17.7.2	Übungen	585
18	Applets	587
18.1	Einleitung	588
18.1.1	Applets oder Anwendungen?	588
18.1.2	Applets in eine Webseite laden	589
18.2	Erste Schritte mit Applets	589
18.2.1	Anwendungen in Applets konvertieren	590
18.2.2	Applet oder JApplet erweitern	591
18.2.3	Der Lebenszyklus eines Applets	592
18.2.4	Methoden für Meilensteine	595
18.2.5	Methoden zum Zeichnen und zur Ereignisbehandlung	597
18.2.6	Methoden zum Hinzufügen von GUI-Komponenten	599
18.2.7	Was Applets können und was sie nicht können	603
18.3	Das Applet-API optimal einsetzen	604
18.3.1	Datendateien suchen und laden	605
18.3.2	Kurze Status-Strings anzeigen	606
18.3.3	Dokumente im Browser anzeigen	607
18.3.4	Nachrichten an andere Applets senden	609
18.3.5	Klänge wiedergeben	613
18.3.6	Applet-Parameter definieren und verwenden	616
18.3.7	Das <APPLET>-Tag	619
18.4	Praktische Aspekte beim Schreiben von Applets	624
18.4.1	Sicherheitsbeschränkungen	624
18.4.2	Bedienoberflächen erstellen	626
18.4.3	Systemeigenschaften ermitteln	629



18.4.4	Threads in Applets	631
18.4.5	Mit serverseitigen Anwendungen arbeiten	635
18.5	Applet abschließen	639
18.5.1	Bevor Sie ein Applet ausliefern	639
18.5.2	Das optimal abgeschlossene Applet	640
18.6	Applets bereitstellen	641
18.6.1	Allgemeine Gesichtspunkte der Bereitstellung	641
18.6.2	Das <APPLET>-Tag verwenden	641
18.6.3	Das <OBJECT>-Tag verwenden	642
18.6.4	Das <EMBED>-Tag verwenden	644
18.6.5	Applets in einer gemischten Browser-Umgebung bereitstellen	645
18.7	Häufige Probleme beheben	646
18.8	Fragen und Übungen: Java-Applets	648
18.8.1	Fragen	648
18.8.2	Übungen	648
A	Schlüsselwörter in Java	649
B	Vorbereitung auf die Java-Programmiersprachenzertifizierung	651
C	Antworten zu Fragen und Übungen	663
D	Die Begleit-CD	713
	Stichwortverzeichnis	715