

**MASTER
CLASS**

→ **Karsten Samaschke**

Java 6

Einstieg für Anspruchsvolle



ADDISON-WESLEY

[in Kooperation mit]

**PEARSON
Studium**

Inhaltsverzeichnis

Vorwort	15
Kapitel 1 Einleitung	17
1.1 Geschichte von Java	17
1.2 Was ist Java?	18
1.3 Java SE Version 6	20
1.4 Download und Installation.	20
1.5 Entwicklungsumgebung.	20
1.5.1 IntelliJ IDEA.	21
1.5.2 Eclipse	21
1.5.3 NetBeans	21
1.5.4 Texteditoren.	21
1.6 Java-Tools	21
1.6.1 java	22
1.6.2 javac	22
1.6.3 Appletviewer	23
1.6.4 javadoc	23
1.6.5 jar	24
1.6.6 jdb	25
1.7 Fazit.	25
Kapitel 2 Java? Java!	27
2.1 Anweisungen	27
2.2 Variablen	28
2.2.1 Variablen deklarieren	29
2.2.2 Konventionen.	29
2.2.3 Mehrere Variablen vom gleichen Typ deklarieren.	30
2.2.4 Zuweisen von initialen Werten.	30
2.2.5 Lebensdauer und Sichtbarkeit von Variablen	30
2.3 Datentypen.	31
2.3.1 Datentypobjekte	32
2.3.2 valueOf-Zuweisungen.	32
2.3.3 Typkonvertierung.	33
2.4 Konstanten und konstante Variablen	34

2.5	Kommentare	34
2.6	Ausdrücke und Operatoren	35
2.6.1	Arithmetische Operatoren	36
2.6.2	Relationale Operatoren	36
2.6.3	Logische Operatoren	37
2.6.4	Weitere Operatoren	38
2.6.5	Operator-Vorrangregeln	38
2.7	Bedingungen und Verzweigungen	39
2.7.1	if-Anweisung	39
2.7.2	Switch-Anweisung	40
2.8	Schleifen.	42
2.8.1	for-Schleife	42
2.8.2	for-each-Schleife	43
2.8.3	while-Schleife	44
2.8.4	do-while-Schleife	44
2.8.5	break und continue.	45
2.9	Methodendeklaration	47
2.9.1	Aufruf einer Methode	48
2.9.2	Parameter: by-reference oder by-value.	48
2.10	Fazit	49

Kapitel 3 Objektorientierte Programmierung 51

3.1	Klassen und Objektinstanzen	52
3.1.1	Abstraktion	52
3.1.2	Kapselung	53
3.1.3	Wiederverwendbarkeit	54
3.2	Beziehungen zwischen Klassen.	54
3.2.1	Spezialisierung	54
3.2.2	Komposition und Aggregation	56
3.2.3	Assoziation	56
3.3	Polymorphie	56
3.4	Definition von Klassen	58
3.4.1	Eigenschaften/Instanzvariablen	58
3.4.2	Getter und Setter.	59
3.5	Erzeugen von Objektinstanzen	63
3.5.1	Instanz einer inneren Klasse erzeugen	63
3.6	Qualifizierung; Zugriff auf Methoden.	64
3.7	Konstruktoren und Destruktoren.	64
3.7.1	Konstruktor.	65
3.7.2	Destruktor.	66
3.8	Methodenüberladung	66
3.8.1	Konstruktorkaskadierung	69
3.8.2	Variable Argumente (VarArgs)	70

3.9	Zugriffsmodifizierer	72
3.9.1	Standardmodifizierer	73
3.9.2	public	73
3.9.3	protected	73
3.9.4	private	74
3.10	Statische Methoden und Variablen.	75
3.11	Finale Klassen, Methoden und Variablen	77
3.12	Vererbung	78
3.12.1	Ableitung von einer Basisklasse	78
3.12.2	Polymorphie im Einsatz	79
3.12.3	Factory-Entwurfsmuster	81
3.13	Abstrakte Klassen und Methoden	84
3.13.1	Deklaration einer abstrakten Klasse	85
3.13.2	Implementieren der abstrakten Klasse	85
3.13.3	Überschreiben und Implementieren von Methoden einer abstrakten Klasse	86
3.13.4	Definition abstrakter Methoden	87
3.14	Interfaces	88
3.14.1	Ein Interface definieren	89
3.14.2	Implementieren eines Interface	90
3.14.3	Mehrere Interfaces implementieren	91
3.14.4	Interface-Ableitung	94
3.15	Lokale und anonyme Klassen	97
3.15.1	Lokale Klassen in Methoden	100
3.15.2	Anonyme Klassen	101
3.16	Aufzählungen	103
3.16.1	Eindeutig und typsicher	104
3.16.2	Enumerations-Member durchlaufen	105
3.17	Pakete	106
3.17.1	Pakete definieren	107
3.17.2	Pakete verwenden	107
3.17.3	Das import-Statement	108
3.17.4	Das import-Statement mit Platzhaltern	108
3.17.5	Namensraumkonflikte	109
3.18	Zusammenfassung	110
Kapitel 4 Listen		111
4.1	Array	111
4.1.1	Deklaration	111
4.1.2	Instanziierung	112
4.1.3	Zuweisung und Abruf von Werten	112
4.1.4	Mehrdimensionale Arrays	114
4.1.5	Vor- und Nachteile	115

4.2	ArrayList	115
4.2.1	Deklaration und Instanziierung	116
4.2.2	Zuweisen von Werten	117
4.2.3	Abrufen von Werten	119
4.2.4	Ersetzen von Werten	122
4.2.5	Löschen von Werten	123
4.2.6	Überprüfen, ob ein Element in der Liste existiert	123
4.2.7	Vor- und Nachteile der ArrayList	124
4.3	HashMap	124
4.3.1	Deklaration und Instanziierung	124
4.3.2	HashMap und Generics	125
4.3.3	Werte zuweisen	125
4.3.4	Abrufen von Werten	127
4.3.5	Abrufen aller Elemente einer HashMap	128
4.3.6	Löschen von Elementen	130
4.3.7	Überprüfen, ob bestimmte Schlüssel oder Werte in der Liste existieren	131
4.4	Properties	135
4.4.1	Deklaration und Instanziierung	135
4.4.2	Zuweisen und Abrufen von Werten	135
4.4.3	Speichern der Werte in Textdateien	137
4.4.4	Speichern der Werte in XML-Dateien	138
4.4.5	Laden der Werte aus Textdateien	139
4.4.6	Laden der Werte aus XML-Dateien	140
4.4.7	Vor- und Nachteile der Properties-Klasse	141
4.5	Weitere Listen	142
4.5.1	HashSet	142
4.5.2	TreeSet	142
4.5.3	LinkedList	142
4.5.4	TreeMap	142
4.5.5	LinkedHashMap	142
4.5.6	Vector	142
4.6	Wichtige Interfaces	143
4.6.1	Enumeration	143
4.6.2	Iterator	144
4.6.3	Collection	145
4.6.4	Map	147
4.6.5	List	148
4.6.6	Set	149
4.7	Generics	151
4.7.1	Was sind Generics?	151
4.7.2	Was bringen Generics?	151
4.7.3	Wie werden die Datentypen angegeben?	151
4.7.4	Müssen Generics zwingend deklariert werden?	152
4.7.5	Generics in eigenen Klassen und Methoden verwenden	152

4.7.6	Generische Typen einschränken	153
4.7.7	Generics und Performance	155
4.8	Zusammenfassung	155

Kapitel 5 Ausnahmen **157**

5.1	Ausnahmen werfen	157
5.2	Ausnahmen abfangen	159
5.2.1	Mehrere Ausnahmen behandeln	160
5.2.2	Alle Ausnahmen auf einen Schlag abfangen	162
5.2.3	Das finally-Schlüsselwort	162
5.3	Eigene Ausnahmen definieren	164
5.4	Strategien beim Einsatz von Ausnahmen	167
5.4.1	Behandeln Sie Ausnahmen so lokal wie möglich	167
5.4.2	Behandeln Sie nur Ausnahmen, für die Sie zuständig sind.	168
5.4.3	Verwenden Sie spezifische Ausnahmetypen	168
5.5	Fazit	168

Kapitel 6 Threads **171**

6.1	Einen Thread erstellen	171
6.1.1	Eine threadingfähige Klasse	171
6.1.2	Einen Thread starten	173
6.2	Timer statt Thread	174
6.3	Priorität von Threads setzen	177
6.4	Threads beenden	179
6.4.1	Verwendung von interrupt() zum Abbrechen eines Thread.	179
6.4.2	Verwendung einer eigenen Thread-Ableitung	180
6.4.3	StoppableThread statt Thread als Super-Klasse.	181
6.5	Zugriff auf den aktuellen Thread	183
6.6	Ermitteln aller laufenden Threads	183
6.7	Organisation von Threads in Thread-Gruppen	185
6.8	Synchronisierung von Threads.	188
6.8.1	Monitor	189
6.8.2	wait() und notify()	193
6.9	Fazit	197

Kapitel 7 Prozesse und Systemumgebung **199**

7.1	Starten einer externen Applikation	199
7.2	Einlesen der Rückgabe eines Prozesses.	201
7.3	Freien Speicher ermitteln	204
7.3.1	Initialen Arbeitsspeicher festlegen	205
7.4	Garbage Collector erzwingen	206
7.5	Umgebungsinformationen abrufen.	207

7.6	Betriebssystem und Java-Version bestimmen	209
7.7	Informationen zum aktuellen Nutzer ermitteln	210
7.8	Zugesicherte Umgebungsvariablen	211
7.9	System-Umgebungsinformationen abrufen.	212
7.10	Zusammenfassung	214

Kapitel 8 Arbeiten mit der Kommandozeile 215

8.1	Aufbau einer Kommandozeilenapplikation	215
8.2	Parameter einlesen	217
8.2.1	Feste Parameterfolge mit Standardwerten.	218
8.2.2	Feste Parameterfolge mit Ausgabe eines Hinweises.	222
8.2.3	Benannte Parameter	225
8.2.4	Welcher Parametertyp für welchen Zweck?	231
8.2.5	Das CLI-Paket der Apache Foundation	231
8.2.6	Commons-cli oder eigene benannte Parameter?	239
8.3	Einlesen von Informationen	239
8.4	Ausgabe von Informationen	244
8.4.1	Einfache Ausgabe	244
8.4.2	Überschreiben der Methode toString().	244
8.4.3	Sichere Ausgabe von Zeilenumbrüchen	245
8.4.4	Umleiten der Ausgabe in eine Datei	247
8.5	Zusammenfassung	248

Kapitel 9 Swing 251

9.1	Was ist Swing?	251
9.2	Swing oder JFC?	252
9.3	Swing-Basics	252
9.3.1	Komponenten	252
9.3.2	Root-Container	252
9.3.3	Verwendung von JFrame	253
9.3.4	Basisklasse für alle folgenden Beispiele	259
9.4	Mehrzweck-Container.	260
9.4.1	JPanel	260
9.4.2	JScrollPane	261
9.4.3	JSplitPane	263
9.4.4	JTabbedPane	266
9.4.5	JToolBar	267
9.5	Komponenten	269
9.5.1	JLabel	269
9.5.2	JButton	272
9.5.3	JCheckBox	281
9.5.4	JRadioButton	285
9.5.5	JComboBox	289

9.5.6	JTextField	294
9.5.7	JPasswordField	302
9.5.8	JFormattedTextField	303
9.5.9	JTextArea	309
9.5.10	JEditorPane	313
9.5.11	JFileChooser	318
9.5.12	JSlider	322
9.5.13	JProgressBar	325
9.6	Menüs mit JMenu	328
9.6.1	Erzeugen von Menüs	329
9.6.2	Beenden einer Swing-Applikation	332
9.6.3	Anzeigen anderer Menüelemente	333
9.6.4	CheckBoxen in Menüs verwenden	335
9.6.5	JPopupMenu	337
9.7	Layout-Manager	342
9.7.1	FlowLayout	342
9.7.2	GridLayout	343
9.7.3	BorderLayout	343
9.7.4	GridBagLayout	345
9.8	Standarddialoge	349
9.8.1	Message-Dialog	349
9.8.2	Confirm-Dialog	351
9.8.3	Input-Dialog	353
9.8.4	Schließverhalten von Input-Dialogen beeinflussen	355
9.9	Fazit	358

Kapitel 10	Applets	361
10.1	Ein einfaches Applet	361
10.1.1	Anzeigen eines Applet über den Browser	362
10.1.2	Anzeige eines Applet per Applet-Viewer	363
10.2	Lebenszyklus eines Applet	364
10.2.1	init()	364
10.2.2	start()	365
10.2.3	stop()	365
10.2.4	destroy()	365
10.3	Elemente des Benutzer-Interface	365
10.3.1	java.awt.Button	366
10.3.2	java.awt.CheckBox	367
10.3.3	java.awt.TextArea	369
10.3.4	java.awt.TextField	371
10.3.5	Menu	373
10.3.6	java.awt.List	377
10.3.7	java.awt.Choice	380
10.3.8	Komponenten, LayoutManager, Applets ...?	382

10.4	paint() und update() – Grafik mit dem AWT	382
10.4.1	Grafische Elemente zeichnen	382
10.4.2	Text direkt ausgeben.	385
10.4.3	Bilder laden	388
10.5	Threading	389
10.6	Was Applets dürfen und was nicht.	393
10.7	Interaktion mit umgebenden Elementen	394
10.7.1	Übergabe von Parametern an das Applet	394
10.7.2	Das <applet>-Tag und seine Parameter.	397
10.7.3	Beeinflussung des Statustextes des Fensters	398
10.7.4	Kommunikation mit anderen Applets in derselben Seite.	399
10.7.5	Redirects im Browser	403
10.8	Applet – JApplet?.	405
10.9	Fazit	406

Kapitel 11 Streams und Dateibehandlung **407**

11.1	Streams und zugrunde liegende Prinzipien.	407
11.1.1	Byte-Streams	408
11.1.2	Character-Streams.	410
11.1.3	Character- oder Byte-Streams?	412
11.1.4	Brückenklassen.	412
11.2	Input- und OutputStreams	413
11.2.1	ByteArrayInputStream und ByteArrayOutputStream.	414
11.2.2	FileInputStream und FileOutputStream.	414
11.2.3	FilterInputStream und FilterOutputStream	417
11.2.4	BufferedInputStream und BufferedOutputStream	417
11.2.5	DataInputStream und DataOutputStream	418
11.2.6	PrintStream.	423
11.3	Writer und Reader	424
11.3.1	FileReader und FileWriter	425
11.3.2	BufferedReader und BufferedWriter	427
11.4	RandomAccessFile	430
11.5	File	437
11.5.1	Pfadbestandteile ermitteln.	438
11.5.2	Detaillierte Informationen über die File-Instanz	439
11.5.3	Dateiaktionen	439
11.5.4	Verzeichnisaktionen	440
11.6	Fazit	442

Kapitel 12 JDBC **445**

12.1	Das JDBC-Prinzip	445
12.2	MySQL, PostGre-SQL, Cloudscape, Oracle ...?	446
12.2.1	MySQL, der JDBC-Treiber und die Admin-Werkzeuge	447

12.3	Verbindung herstellen	451
12.3.1	Treiber laden	451
12.3.2	Verbindungsparameter übergeben	451
12.4	Datenbankabfragen	452
12.4.1	Daten einfügen oder ändern	453
12.4.2	Abrufen von Daten	455
12.5	PreparedStatement	458
12.5.1	Datensätze einfügen, ändern oder löschen	459
12.5.2	Abrufen von Daten	461
12.6	Tipps und Tricks	462
12.6.1	Sichern Sie Ihre Datenbank!	462
12.6.2	Verwenden Sie PreparedStatements!	463
12.6.3	Geben Sie bei SELECT-Statements immer alle Spalten an!	463
12.6.4	Kapseln Sie Datenbankzugriffe in Datenklassen	464
12.6.5	Auslagerung der ConnectionStrings und der Klassennamen	472
12.7	Hibernate und O/R-Mapping-Mechanismen verwenden	473
12.7.1	Hibernate	474
12.7.2	Installation von Hibernate	475
12.7.3	Mappings	475
12.7.4	Konfiguration	478
12.7.5	Konfigurationsdatei für MySQL	478
12.7.6	Arbeiten mit Hibernate	480
12.7.7	HibernateUtil-Hilfsklasse	481
12.7.8	Implementieren des GuestbookDAO	482
12.7.9	Änderung der GuestbookDAOFactory-Klasse	484
12.8	Fazit	484
Kapitel 13 Netzwerk		487
13.1	Auflösung eines Hostnamens	487
13.2	Netzwerkadapter auflisten	489
13.3	URL	490
13.3.1	Analyse eines URL	492
13.3.2	Verarbeitung des referenzierten Inhalts	493
13.3.3	URLConnection	496
13.4	Client/Server-Kommunikation	497
13.4.1	Sockets	497
13.5	E-Mails senden	504
13.5.1	Was ist Java Mail?	504
13.5.2	Installation von Java Mail	505
13.5.3	Versenden von E-Mails	505
13.6	Fazit	509
J2EE Software License		511
Stichwortverzeichnis		517