

Inhaltsverzeichnis

Vorwort	15
1 Ein Loblied auf Java.....	17
1.1 Das Profil von Java.....	17
1.2 Programmiersprache, Interpreter und Compiler.....	20
1.3 Die virtuelle Maschine und das ultimative Wohnmobil.....	20
2 Entwicklung mit IDEen und ohne	23
2.1 Ein kleines Programmbeispiel als Opfer.....	23
2.2 Java-Programmierung mit dem JDK.....	25
2.3 Eclipse	29
2.3.1 Installation.....	30
2.3.2 Anlegen eines neuen Projekts	31
2.3.3 Übersetzung und Testlauf	33
2.3.4 Erzeugen einer JAR-Datei	34
2.4 Aufgabe: Grüßaugust selbst erstellen	34
3 Aufstieg zum Programmierer	35
3.1 Java-Programm-Skelett.....	35
3.1.1 Ordnende Elemente	36
3.1.2 Einrückung	36
3.1.3 Semikolon	36
3.2 Bezeichnende Namen.....	37
3.3 Kommentare sind nett zu lesen.....	37
3.4 Aufgaben	40
3.5 Variable Attribute	40
3.5.1 Zuweisung.....	41
3.5.2 Auch Variablen wollen getauft sein	43

3.5.3	Variablen und Typen.	44
3.6	Primitive Typen.	44
3.6.1	Ganze Zahlen	45
3.6.2	Casting.	46
3.6.3	Kaputte Zahlen	46
3.6.4	Noch ein paar Primitive	48
3.6.5	Konstanten	49
3.6.6	Aufgabe: Zahlensalat	49
3.7	Kleines Rechenwerk.	51
3.7.1	Ausdruck	51
3.7.2	Inkrementieren und dekrementieren.	53
3.7.3	Punkt vor Strich	54
3.7.4	Ganzzahlig kontra Fließkomma.	55
3.7.5	Der Fluch der 10 Finger	55
3.8	Zeichen	56
3.8.1	Zeichen sind auch nur Zahlen	56
3.8.2	Internationale Abwege.	57
3.8.3	Steuerzeichen	58
3.8.4	Zeichenkettenliterale	59
3.8.5	Mit Zeichen spielen und rechnen	59
3.8.6	Aufgabe: Typenberechnungen.	60
4	Entscheiden und Wiederholen.	61
4.1	Nur unter einer Bedingung	61
4.1.1	if-Anweisung.	62
4.1.2	Operatoren	63
4.1.3	Alternativen heißen Else	64
4.2	Logeleien.	66
4.2.1	Wenn alles gelten muss: UND	66
4.2.2	Wenn mindestens eines gilt: OR	68
4.2.3	Exklusiv ODER	69
4.2.4	Genau das Gegenteil: NOT.	70
4.2.5	Aufgabe: Notengebung	71
4.2.6	Negieren, aber richtig: De Morgan.	71
4.2.7	Umkehrung einer UND-Verknüpfung	72
4.2.8	Umkehrung einer ODER-Verknüpfung.	72
4.2.9	Boolesche Variablen.	73
4.2.10	Das Fragezeichen und der bedingte Ausdruck.	73
4.2.11	Aufgabe: Minimum	74

4.3	Die Fallunterscheidung – switch/case	74
4.4	Endlich lernen wir Schleifen zu binden!	76
4.4.1	Die kopfgesteuerte while-Schleife	77
4.4.2	Zählschleife mit for.	79
4.4.3	Die fußgesteuerte do-while-Schleife	81
4.5	Aufgaben	83
4.6	Wenn Schleifen brechen müssen	85
4.7	Überblick behalten mit Struktogrammen	86
5	Funktionierende Methoden	89
5.1	Salzkartoffeln kochen für Anfänger	90
5.1.1	Methodenkopf	91
5.1.2	Parameterklammern	92
5.1.3	Methodenrumpf	92
5.2	Methoden mit Datenfluss	92
5.2.1	Parameter	94
5.2.2	Lokale Variablen	94
5.2.3	Rückgabe eines Wertes	95
5.3	Aufgaben	95
5.4	Überladen von Methoden	96
5.4.1	Überladene Experimente	97
5.5	Rekursionen	99
5.5.1	Der freie Stab	103
5.5.2	Das fertige Programm	103
6	Klasse Objekte	105
6.1	Ein Beispiel für objektorientierte Programmierung	106
6.2	Eine Klasse im Selbstbau	107
6.2.1	Das Hochregallager für Objekte	108
6.2.2	Ein Objekt wird geboren	108
6.2.3	Aufgabe: Punkte und Linien	109
6.2.4	Die Klassen haben so ihre Methoden	110
6.2.5	Stolperfälle: Kopieren und Vergleichen von Referenzen	111
6.2.6	Eine Objektkopie selbst anfertigen	112
6.2.7	Datumsvergleich	113
6.2.8	Aufgabe: Addieren mit Methode	114
6.2.9	Wie stirbt ein Objekt?	117

6.3	Die Vorteile der Privatsphäre	118
6.3.1	Auch Klassen können privat sein!	120
6.3.2	Eine kleine, private Kiste	121
6.4	Konstruktionsbeding.	122
6.4.1	Konstruktoren ohne und mit Parameter	122
6.4.2	Aufgabe: Konstruierte Linien	124
6.4.3	Der Singleton und der private Konstruktor	124
6.5	Statische Attribute und Methoden	125
6.6	Aufzählungstyp mit enum definieren	127
6.7	Aufgabe: Setter-Methode	128
6.8	Verteilung der Klassen auf Pakete	128
6.8.1	Packages definieren	129
6.8.2	Zugriff auf Packages	129
6.8.3	Packages und Verzeichnisse	131
6.8.4	Öffentlichkeit der Klassenzugriffe	132
7	Arrays	133
7.1	Wir bauen ein Array	135
7.2	Wir spielen Lotto!	137
7.3	Arrays von Referenzen	138
7.4	Ein Array für die Monatstage	140
7.5	Eine spezielle for-Schleife für Arrays	142
7.6	Arrays von Arrays	143
7.7	Die Klasse java.util.Arrays	144
7.7.1	Vergleichen mit equals	144
7.7.2	Füllen mit fill	144
7.7.3	Sortieren mit sort	145
7.7.4	Durchsuchen mit binarySearch	145
7.8	Bermuda: Die Schiffe und das Meer	146
7.8.1	Das Rahmenprogramm für Bermuda	147
7.8.2	Aufgaben	150
8	Die Strings: Viele, viele Buchstaben	151
8.1	Auch Strings haben so ihre Methoden	153
8.1.1	Zusammengesetzt mit Plus und concat	153
8.1.2	Tod der rechteckigen Klammern	154
8.1.3	Strings und seinesgleichen	155
8.1.4	Dieser Text hat Längen	155
8.1.5	Inhaltsangabe eines Strings	156

8.1.6	Geschnittene Strings	156
8.1.7	Wo bist Du?	156
8.1.8	String-Zerlegung	157
8.1.9	Ersatzbeschaffung	157
8.1.10	Regulärer Ausdruck	158
8.1.11	Aufgabe: Reguläre Namensgleichheit	159
8.2	Beispiel Pfadumbau	159
8.3	Aufgaben	161
8.4	Typwandler	162
8.4.1	toString	163
8.4.2	String nach Zahl	163
8.4.3	Aufgabe: Oder einfach selbst parsen.	164
8.5	Das String-Array der Methode main.	165
9	Klassen erweitern	167
9.1	Noch mal Klasse: Der Mensch als Objekt.	167
9.2	Was den Mensch vom Tier unterscheidet	168
9.3	Referenzen auf erweiterte Objekte	170
9.4	Object als der Urahn aller Objekte	171
9.5	Die Mensa der Universität Norgaardholz.	171
9.6	Ein Objekt weiß, was es tut: Polymorphie.	174
9.7	Eine abstrakte Suppe	176
9.8	Die Mahlzeit als vollkommen abstraktes Interface	177
9.9	Das Interface und die Mehrfachvererbung	179
9.10	Der Beruf wird geschützt.	180
9.11	Aufgaben	181
10	Hoffentlich Exception-versichert	185
10.1	Ausnahmsweise Fehler.	185
10.2	Eingabe einer Zahl per Dialogbox.	187
10.3	Der catch-Parameter	191
10.4	Weiterwerfen	193
10.5	Selber werfen	194
10.6	... und schließlich	196
10.7	Der Stammbaum der Exceptions.	196
10.8	Die Katzen-Exception	198
10.8.1	Kann Java helfen?	198
10.8.2	Das Katzenleben	199
10.8.3	Der Mensch.	200

10.8.4	Die Ausnahmen	201
10.8.5	Die Katze	202
10.9	Noch mal clone	203
11	Fertige Bausteine zur Selbstbedienung	205
11.1	Wie der Zufall es will	205
11.1.1	Wir erzeugen Zufälle	206
11.1.2	Lottozahlenpärchen	207
11.1.3	Bermuda: Schiffe verstecken	210
11.2	Das Datum und der Kalender	211
11.2.1	Das Datum ist nur eine Zahl	211
11.2.2	Der alte Kalender: Date!	215
11.3	Die Util-Klasse Arrays	215
11.4	Rechnerisches aus der Math-Klasse	216
11.4.1	Winkelfunktionen	216
11.4.2	Exponenten und Logarithmen	217
11.4.3	Absolut und Rundheraus	218
12	Fenster, Mäuse und Grafik	219
12.1	JFrame: Guck mal durchs Fenster	219
12.2	Ereignisreichtum	221
12.2.1	Das Fenster schließt sich	221
12.2.2	Lauschangriff	222
12.2.3	Mausereignisse	225
12.2.4	Mauseschwänzchen	226
12.3	Zeichnen und Malen	229
12.3.1	Ein kleines Beispiel-Oval	230
12.3.2	Sofort zeichnen per repaint	232
12.3.3	Es geht auch anders: JPanel als Leinwand	232
12.3.4	Linien, Rechtecke und Polygone	234
12.3.5	Kreise, Ellipsen, Ovale und Kreisabschnitte	235
12.3.6	Ein Tortendiagramm	236
12.3.7	Aufgabe: Klick Dir ein Rechteck	238
12.3.8	Texte malen	238
12.3.9	Aufgabe: Eingabe zeichnen	238
12.3.10	Der Umgang mit Bildern und Images	239
12.3.11	Clipping	241
12.4	Modernisierung: Graphics2D	243
12.4.1	Selbsterstellte Figuren	244

12.4.2	Farbverläufe und Strichstärke	246
12.4.3	Aufgabe: Hufeisen	247
13	Das Spiel Bermuda als Swing-Anwendung	249
13.1	Objektverteilung	250
13.2	Auf der Werft: Wir bauen Schiffe!	251
13.3	Platzwart für das Spielfeld	252
13.3.1	Die Suche per Schiffsvergleich	255
13.3.2	Die Achsen	255
13.3.3	Die Diagonalen	255
13.3.4	Richtungen zählen	255
13.3.5	Alternative Lösung: Schiffssuche durch Wanderschaft	257
13.4	Und nun mit Swing	259
13.5	Bermuda mit Bildern	262
13.6	Wir erstellen eine Bermuda-JAR-Datei	264
13.6.1	JAR-Datei generieren	265
13.6.2	Anpassungen für die JAR-Bilder	265
14	Erstellen eines Layouts	267
14.1	Noch mal auf Anfang	268
14.2	Alle bitte nebeneinander aufstellen! FlowLayout	269
14.3	Gleichmäßig vergittert: GridLayout	270
14.4	Die Mitte wird fett: BorderLayout	272
14.5	Kombination der Layouts mit Panels	273
14.6	GridBagLayout	276
14.6.1	Anordnungsbeschreibung im GridBagConstraints	276
14.6.2	Ein Eingabedialog als Beispiel	279
14.6.3	setConstraints	280
14.7	BoxLayout	281
14.8	Null-Diät für Layout-Manager	283
14.9	Aufgaben	283
15	Kontrollelemente	285
15.1	Hochgradiger Simpel: Das Label	285
15.2	Das Auffangen der Ereignisse	287
15.3	Druckvolle Buttons	289
15.4	Ein BMI-Programm als Spielwiese	291
15.4.1	Die Benutzereingaben	291
15.4.2	Abarten des Eingabefelds	294

15.4.3	Fallunterscheidung mit Radiobuttons	294
15.4.4	Radiobuttons im Eigenbau	295
15.4.5	Combobox: Auswahl aus einer Liste	295
15.4.6	JList	300
15.5	Der Button hat einen Haken	300
15.6	Menüs und Dateidialoge	302
15.7	Der Fokus und der Multiliner	305
15.8	Aufgaben	307
16	Android	309
16.1	Alles ist ein wenig anders	309
16.2	Arbeitsumgebung schaffen	310
16.2.1	Das Android Software Development Kit einrichten	311
16.2.2	Eclipse und das Android Development Tool	311
16.3	Schnell mal 'ne App	313
16.3.1	Projekterstellung unter Eclipse	313
16.3.2	Die App startet	314
16.3.3	Wir erstellen den Bildschirm	316
16.3.4	Texte in die String-Datei	319
16.3.5	Die Activity wird aktiv	320
16.3.6	Eingabefelder	322
16.4	Weitere Kontrollelemente unter Android	323
16.4.1	Radiobuttons und ihre Gruppe	324
16.4.2	Fokuswechsel fangen	325
16.4.3	Was könnten Sie nun alles programmieren?	325
16.5	Bermuda auch auf Android?	325
16.5.1	Ein paar Grafikprimitive des Canvas	328
16.5.2	Paint, die Art des Zeichnens	329
16.5.3	Bermuda mit Bitmaps	330
17	Datenelemente sammeln und anordnen	335
17.1	Java Collection Framework	335
17.1.1	Interfaces	336
17.1.2	Klassen	336
17.2	Ein Ausflug in die Generics	337
17.3	Ein eigener generischer Container	339
17.4	Das Interface Collection und seine Methoden	342

17.5	Listig: ArrayList und LinkedList im Vergleich	343
17.5.1	Die Methoden des Interfaces List	343
17.5.2	ArrayList	344
17.5.3	Kapazität	347
17.5.4	Landkreise in der ArrayList	347
17.5.5	Spielen Sie mal wieder Lotto	349
17.5.6	Ein altes flexibles Array: Der Vector	350
17.5.7	In Ketten: Die LinkedList	350
17.5.8	Iterieren über Container	351
17.5.9	Veränderung kann tödlich sein	352
17.5.10	Kreise per Iterator	353
17.5.11	Queue und noch einmal LinkedList	354
17.6	Ausmist: Q-Mist	357
17.7	Implementierung von Set	357
17.7.1	Hash ist toll	358
17.7.2	Keine doppelten Wörter	359
17.7.3	Die Kreise in einem Set	361
17.8	Map-Implementierung: Der assoziative Speicher	363
17.8.1	Maps durchlaufen	364
17.8.2	Eine Zusammenfassung der Methoden von Map	365
17.8.3	Lebt denn der alte Hashtable noch?	366
17.9	Die Algorithmen des Java Collection Framework	367
17.9.1	Unordnung stiften für Anfänger	367
17.9.2	Umdrehen mit reverse	368
17.9.3	Endlich Ordnung!	368
17.9.4	Aufgabe Lottozahlen	369
17.9.5	Wie sortiert man Autos?	369
17.9.6	Das Interface Comparable	370
17.9.7	Nachsortieren mit dem Comparator	371
17.9.8	Suche in der Liste: binarySearch	372
17.9.9	Wrapper-Klassen für Primitive	372
18	Dateien halten die Daten frisch	375
18.1	Serialisierte Klassen	375
18.2	Spielstandsicherung für Bermuda	379
18.3	Sequentielle Dateien	380
18.4	Random-Access-Dateizugriffe	382
18.5	Umgang mit dem Dateisystem	387

19	Datenbankzugriff per JDBC	389
19.1	Eine Datenbank für die Bonbonbude	389
19.1.1	Die Datenbank Derby	390
19.1.2	Einbinden einer jar-Datei	390
19.1.3	Eclipse	390
19.1.4	Kommandozeile	391
19.1.5	Verbindung zu derby.jar	391
19.1.6	Ein Programm erzeugt die Tabelle	392
19.1.7	Verbindung aufnehmen	394
19.1.8	Anlegen der Tabelle	395
19.1.9	Ein Datensatz wird in die Datenbank geschrieben	395
19.1.10	Datenbank auslesen	397
19.1.11	Werte in der Datenbank ändern	399
19.1.12	Ein kurzer Blick auf WHERE	399
19.1.13	Mehrere Inserts und Updates	401
19.1.14	Aufgaben: Löschen von Einträgen	402
19.2	Wenn Datenbanken einmal groß werden	402
19.2.1	Der JDBC-Treiber	403
19.2.2	Verbindung mit der Datenbank	405
20	Parallelverarbeitung mit Threads	407
20.1	Leichtgewichtige Threads	408
20.2	Runnable Thread	410
20.2.1	Alternative: Erweiterung von Thread	411
20.2.2	Wozu?	411
20.3	Computer kaputtmachen!	411
20.4	Synchronschwimmen	415
21	Schluss	419
	Stichwortverzeichnis	421