

# Inhaltsverzeichnis

Über den Autor	15
<b>Einführung</b>	<b>27</b>
Wie Sie an dieses Buch herangehen sollten	27
Konventionen, die in diesem Buch verwendet werden	27
Was Sie nicht lesen müssen	28
Ein paar einfache Annahmen	29
Wie dieses Buch aufgebaut ist	30
Teil I: Los geht's	30
Teil II: Eigene Java-Programme schreiben	30
Teil III: Das große Ganze im Auge behalten: die objektorientierte Programmierung	30
Teil IV: Fortgeschrittene Java-Techniken	31
Teil V: Der Top-Ten-Teil	31
Symbole, die in diesem Buch verwendet werden	31
Anmerkung der Übersetzerin	32
Wie es weitergeht	32
 <b>Teil I</b>	
<b>Los geht's</b>	<b>33</b>
 <b>Kapitel 1</b>	
<b>Alles über Java</b>	<b>35</b>
Was Sie mit Java machen können	36
Warum Sie Java verwenden sollten	37
Einen Überblick erhalten: Wie sich Java einordnen lässt	37
Objektorientierte Programmierung (OOP)	39
Objektorientierte Sprachen	40
Objekte und ihre Klassen	42
Was ist das Besondere an einer objektorientierten Sprache?	42
Ihr Verständnis für Klassen und Objekte vertiefen	45
Wie geht es weiter?	46
 <b>Kapitel 2</b>	
<b>Alles über Software</b>	<b>47</b>
Anleitungen für einen Schnellstart	47
Was Sie auf Ihrem Computer installieren	49
Was ist ein Compiler?	50

Was ist eine Java Virtual Machine?	52
Software entwickeln	58
Was ist eine integrierte Entwicklungsumgebung?	59

## **Kapitel 3**

### **Die grundlegenden Teile verwenden** **63**

Die Sprache Java sprechen	63
Die Grammatik und die gebräuchlichen Bezeichnungen	64
Die Wörter in einem Java-Programm	65
Sich zum ersten Mal mit Java-Code beschäftigen	67
Ein einfaches Java-Programm verstehen	68
Die Java-Klasse	68
Die Java-Methode	69
Die Methode »main« eines Programms	71
Wie Sie dem Computer letztendlich mitteilen, dass er etwas tun soll	72
Geschweifte Klammern	74
Kommentare, Kommentare	76
Dem Code Kommentare hinzufügen	77
Und wie sieht Barrys Ausrede aus?	80
Kommentare verwenden, um mit dem Code zu experimentieren	80

## **Teil II**

### **Eigene Java-Programme schreiben** **83**

#### **Kapitel 4**

#### **Das Optimum aus Variablen und ihren Werten herausholen** **85**

Eine Variable variieren	85
Zuweisungsbefehle	87
Wertetypen, die Variablen annehmen können	88
Text anzeigen	91
Ganzzahlen	92
Deklarationen kombinieren und Variablen initialisieren	93
Die Atome: Javas primitive Typen	94
Der Typ »char«	95
Der Typ »boolean«	97
Die Moleküle und Verbindungen: die Referenztypen	99
Eine Importdeklaration	103
Mit Operatoren neue Werte erstellen	104
Einmal initialisieren, mehrfach zuweisen	107
Inkrement- und Dekrement-Operatoren	108
Zuweisungsoperatoren	112

## **Kapitel 5**

### **Den Programmablauf mit entscheidungsfindenden Befehlen steuern 115**

Entscheidungen fällen (»if«-Befehle in Java)	115
Eine Zahl raten	116
Tastatureingaben kontrollieren	117
Zufallszahlen	119
Der Befehl »if«	120
Das doppelte Gleichheitszeichen	121
Nichts als Klammern	121
Den Code bei »if«-Befehlen einrücken	122
Elselos in Afrika	123
Bedingungen mit Vergleichsoperatoren und mit logischen Operatoren bilden	124
Zahlen miteinander vergleichen: Vergleichsoperatoren	124
Objekte vergleichen	125
Alles auf einen Schlag importieren	128
Javas logische Operatoren	128
Vive les nuls!	131
(Bedingungen in Klammern)	132
Verschachtelungen	134
Unter mehreren Alternativen wählen (Javas »switch«-Befehle)	136
Der »switch«-Befehl	136
To break or not to break	140
Der neue, verbesserte »switch«-Befehl	142

## **Kapitel 6**

### **Den Programmablauf mit Schleifen steuern 145**

Anweisungen mehrfach wiederholen (der Java-Befehl »while«)	145
Eine bestimmte Anzahl an Wiederholungen festlegen (der Java-Befehl »for«)	149
Die Anatomie eines »for«-Befehls	151
Die Weltpremiere von »Al's All Wet«	151
Wiederholen, bis Ihr Wunsch erfüllt wird (der Java-Befehl »do«)	153
Ein einzelnes Zeichen lesen	156
Java und die Behandlung von Dateien	157
Variablendeklarationen und Blöcke	158

## **Teil III**

### **Das große Ganze im Auge behalten: Die objektorientierte Programmierung 161**

## **Kapitel 7**

### **In Begriffen wie Klassen und Objekte denken 163**

Eine Klasse definieren (was es heißt, ein Konto zu sein)	163
Variablen deklarieren und Objekte erstellen	165

Eine Variable initialisieren	168
Die Felder eines Objekts verwenden	168
Ein Programm; mehrere Klassen	168
Öffentliche (»public«) Klassen	169
Eine Methode in einer Klasse definieren (ein Konto anzeigen)	170
Ein Konto, das sich selbst anzeigt	171
Der Kopf der Methode »display«	172
Werte an Methoden senden und von dort erhalten (Zinsen berechnen)	173
Einen Wert an eine Methode übergeben	176
Einen Wert von der Methode »getZinsen« zurückgeben	178
Zahlen gut aussehen lassen	179
Einzelheiten mit Zugriffsmethoden verbergen	183
Gute Programmierung	183
Öffentliches Leben und private Träume: ein Feld sperren	186
Regeln mit Zugriffsmethoden erzwingen	187

## **Kapitel 8**

### **Zeit und Geld sparen: Code wiederverwenden** **189**

Eine Klasse definieren (was es bedeutet, ein Mitarbeiter zu sein)	189
Das letzte Wort an Mitarbeiter	190
Die Klasse gut verwenden	191
Gehalt bezahlen	193
Mit Dateien auf der Festplatte arbeiten (ein kleiner Umweg)	194
Daten in einer Datei ablegen	194
Code kopieren und einfügen	195
Eine Datei auslesen	196
Wer hat die Datei verschoben?	199
Den Dateinamen um den Verzeichnisnamen erweitern	199
Eine Zeile nach der anderen lesen	200
Die Verbindung zu einer Datei auf der Festplatte beenden	202
Unterklassen definieren (was es bedeutet, ein Vollzeit- oder ein Teilzeitmitarbeiter zu sein)	202
Unterklassen erstellen	204
Das Anlegen von Unterklassen kann zur Gewohnheit werden	207
Mit Unterklassen arbeiten	207
Typen passend gestalten	209
Die zweite Hälfte der Geschichte	210
Methoden überschreiben (Zahlungen für einige Mitarbeiter ändern)	211
Eine Java-Annotation	213
Methoden aus Klassen und Unterklassen verwenden	214

## **Kapitel 9**

### **Neue Objekte entwerfen**

**217**

Konstruktoren definieren (was es bedeutet, eine Temperatur zu sein)	217
Was ist eine Temperatur?	218
Was ist eine Temperaturskala? (Der Java-Typ »enum«)	218
Also gut, was ist denn nun eine Temperatur?	219
Was Sie mit einer Temperatur anfangen können	221
Eine andere Temperatur (32.0) aufrufen – eine Fallstudie	224
Einige Dinge ändern sich nie	226
Noch mehr Unterklassen (etwas gegen das Wetter unternehmen)	227
Für bessere Temperaturen sorgen	227
Konstruktoren für Unterklassen	229
Alles anwenden	230
Der Standardkonstruktor	231
Ein Konstruktor, der mehr kann	233
Klassen und Methoden der Java-API	235
Die Annotation »SuppressWarnings«	236

## **Teil IV**

### **Fortgeschrittene Java-Techniken**

**239**

## **Kapitel 10**

### **Variablen und Methoden richtig platzieren**

**241**

Klassen definieren (Was es heißt, ein Baseballspieler zu sein)	241
Ein anderer Weg, um Zahlen zu formatieren	243
Die Klasse »Player« verwenden	243
Neunmal	246
Eine GUI	246
Eine Ausnahme von Methode zu Methode weiterreichen	248
Etwas statisch machen (den Teamdurchschnitt herausfinden)	249
Warum gibt es da so viel Statisches?	250
Das statische Initialisierungsprogramm	251
Den Teamdurchschnitt anzeigen	252
»static« ist ein alter Hut	255
Achtung Statisches – mit Vorsicht genießen!	255
Mit Variablen experimentieren	256
Eine Variable richtig platzieren	257
Einer Variablen aufzeigen, wohin sie gehört	259
Parameter übergeben	262
Übergabe als Wert	262
Ein Ergebnis zurückgeben	264
Übergabe per Referenz	264
Ein Objekt von einer Methode zurückgeben	266
Nachtrag	268

**Kapitel 11****Arrays verwenden, um mit Werten zu jonglieren** **269**

Bereiten Sie sich gut vor	269
Ein Array in zwei einfachen Schritten erstellen	271
Werte speichern	272
Tabulatoren und anderes	274
Einen Array-Initialisierer verwenden	275
Ein Array mit einer erweiterten »for«-Schleife durchlaufen	276
Suchen	277
In eine Datei schreiben	280
Wann eine Datei geschlossen werden muss	281
Arrays aus Objekten	283
Die Klasse »Room« verwenden	284
Und noch ein Weg, um Zahlen schön zu gestalten	287
Der Bedingungsoperator	288
Argumente auf der Befehlszeile	289
Befehlszeilenargumente in einem Java-Programm verwenden	290
Prüfen, ob die richtige Anzahl an Befehlszeilenparametern vorhanden ist	292

**Kapitel 12****Sammlungen und Streams verwenden  
(wenn Arrays nicht mehr ausreichen)** **295**

Hilfe durch Sammelklassen	296
Eine »ArrayList« verwenden	297
Generische Typen verwenden	299
Testen, ob noch mehr Daten vorhanden sind	302
Einen Iterator verwenden	302
Die vielen Sammelklassen Javas	303
Neu in Java 8: funktionale Programmierung	304
Ein Problem auf die altbewährte Weise lösen	307
Streams	308
Lambda-Ausdrücke	309
Eine Klassifizierung von Lambda-Ausdrücken	312
Streams und Lambda-Ausdrücke verwenden	313
Warum sich Sorgen machen?	318
Methodenreferenzen	318

**Kapitel 13****Auch dann noch gut aussehen, wenn die Dinge eine unerwartete  
Wendung nehmen** **321**

Mit Ausnahmen umgehen	322
Der Parameter einer »catch«-Klausel	326
Ausnahmetypen	327

Wer fängt die Ausnahme ein?	329
Die Multi-catch-Klausel	335
Alle Bedenken in den Wind schlagen	336
Etwas Sinnvolles tun	336
Unsere Freunde, die guten Ausnahmen	338
Eine Ausnahme verarbeiten oder den Schwarzen Peter weitergeben	338
Die Aufgabe mit der Klausel »finally« abschließen	343
Ein »try«-Befehl mit Ressourcen	346

## **Kapitel 14**

### **Programmteile gemeinsam nutzen lassen** **349**

Zugriffsmodifizierer	350
Klassen, Zugriff und Multipart-Programmierung	350
Mitglied oder Klassen	351
Zugriffsmodifizierer für Mitglieder	352
Eine Zeichnung auf einem Frame platzieren	354
Die Verzeichnisstruktur	356
Einen Frame erstellen	358
Sich heimlich vom ursprünglichen Code absetzen	359
Standardzugriff	361
Wieder ins Paket zurückkrabbeln	362
Geschützter Zugriff	364
Klassen, die keine Unterklassen sind, in demselben Paket unterbringen	366
Zugriffsmodifizierer für Java-Klassen	367
Öffentliche Klassen	367
Nicht öffentliche Klassen	368

## **Kapitel 15**

### **Auf Tastatureingaben und Mausklicks reagieren** **371**

Machen Sie weiter ... Klicken Sie auf diese Schaltfläche	371
Ereignisse und Ereignisbehandlung	374
Das Java-Interface	374
Threads	376
Das Schlüsselwort »this«	377
Das Innere der Methode »actionPerformed«	378
Die serialVersionUID	378
Auf andere Dinge als auf das Anklicken von Schaltflächen reagieren	380
Innere Klassen erstellen	385

## **Kapitel 16**

### **Java-Applets schreiben** **389**

Applets-Einmaleins	389
Auf einen Aufruf warten	390
Eine öffentliche Klasse	391
(Noch einmal) die Java-API	391

Die Dinge in Bewegung versetzen	392
Die Methoden in einem Applet	394
Was alles in diese Methoden gehört	395
Auf Ereignisse in einem Applet reagieren	396

## **Kapitel 17**

### **Mit Java Datenbankverbindungen aufbauen und nutzen** **399**

JDBC und Java DB	399
Daten erstellen	400
SQL-Befehle verwenden	402
Verbinden und trennen	403
Daten abfragen	404

## **Teil VII**

### **Der Top-Ten-Teil** **407**

## **Kapitel 18**

### **Zehn Wege, um Fehler zu vermeiden** **409**

Großbuchstaben dorthin setzen, wo sie hingehören	409
Aus einer »switch«-Anweisung aussteigen	409
Werte mit einem doppelten Gleichheitszeichen vergleichen	410
Einer GUI Komponenten hinzufügen	410
Listeners für Ereignisbehandlungen hinzufügen	411
Die benötigten Konstruktoren definieren	411
Nicht statische Verweise reparieren	411
Sich in den Grenzen eines Arrays aufhalten	411
»NullPointers« vorhersehen	412
Java bei der Suche nach seinen Dateien helfen	413

## **Kapitel 19**

### **Zehn Websites für Java** **415**

Die Websites dieses Buches	415
Aus erster Hand	415
News, Berichte und Beispielcode	415
Es gibt nicht nur Englisches	416
Jedermanns Lieblingssite	416

### **Stichwortverzeichnis** **417**