

David J. Barnes • Michael Kölling

Java lernen mit **BlueJ**

Objects first - Eine Einführung in Java

6., aktualisierte Auflage



Inhaltsverzeichnis

Vorwort von James Gosling, Erfinder von Java	13
Vorwort für den Lehrenden	14
Danksagungen	25
Projekte, die in diesem Buch detailliert besprochen werden	26
Teil I Objekte und Klassen	29
Kapitel 1 Objekte und Klassen	31
1.1 Objekte und Klassen	31
1.2 Instanzen erzeugen	32
1.3 Methoden aufrufen	33
1.4 Parameter	35
1.5 Datentypen	36
1.6 Eine Klasse, viele Instanzen	37
1.7 Zustand	38
1.8 Das Innenleben eines Objekts	38
1.9 Java-Code	40
1.10 Objektinteraktion	41
1.11 Quelltext	42
1.12 Ein weiteres Beispiel	44
1.13 Aufrufergebnisse	44
1.14 Objekte als Parameter	45
Kapitel 2 Klassendefinitionen	51
2.1 Ticketautomaten	51
2.2 Eine Klassendefinition untersuchen	53
2.3 Der Kopf der Klasse	55
2.4 Datenfelder, Konstruktoren und Methoden	56
2.5 Datenübergabe mit Parametern	62
2.6 Zuweisungen	64
2.7 Methoden	65
2.8 Sondierende und verändernde Methoden	67

2.9 Ausgaben in Methoden	70
2.10 Zusammenfassung der Methoden	72
2.11 Zusammenfassung des naiven Ticketautomaten	72
2.12 Bewertung des Entwurfs des naiven Ticketautomaten	73
2.13 Entscheidungen treffen: die bedingte Anweisung	76
2.14 Ein weiteres Beispiel für eine bedingte Anweisung	78
2.15 Hervorhebung von Sichtbarkeitsbereichen	79
2.16 Lokale Variablen	80
2.17 Datenfelder, Parameter und lokale Variablen	82
2.18 Zusammenfassung des besseren Ticketautomaten	84
2.19 Übungen zur Selbstüberprüfung	84
2.20 Vertrautes neu betrachtet	86
2.21 Methoden aufrufen	88
2.22 Ausdrücke testen: die Direkteingabe	89
Kapitel 3 Objektinteraktion	97
3.1 Das Uhren-Beispiel	97
3.2 Abstraktion und Modularisierung	98
3.3 Abstraktion in Software	99
3.4 Modularisierung im Uhren-Beispiel	100
3.5 Implementierung der Uhrenanzeige	100
3.6 Klassendiagramme und Objektdiagramme	101
3.7 Primitive Typen und Objekttypen	103
3.8 Die Klasse Nummernanzeige	103
3.9 Die Klasse Uhrenanzeige	110
3.10 Objekte erzeugen Objekte	112
3.11 Mehrere Konstruktoren	114
3.12 Methodenaufrufe	114
3.13 Ein weiteres Beispiel für Objektinteraktion	118
3.14 Die Benutzung eines Debuggers	122
3.15 Mehr zu Methodenaufrufen	126
Kapitel 4 Objektsammlungen	131
4.1 Themen aus Kapitel 3 vertiefen	131
4.2 Die Abstraktion Sammlung	132
4.3 Ein Verwaltungssystem für Musikdateien	133
4.4 Eine Bibliotheksklasse verwenden	135
4.5 Objektstrukturen mit Sammlungen	138
4.6 Generische Klassen	139
4.7 Nummerierung in Sammlungen	140
4.8 Musikdateien abspielen	144

4.9	Komplette Sammlungen verarbeiten	146
4.10	Unbestimmte Iteration	152
4.11	Verbesserung der Struktur – die Klasse Track	159
4.12	Der Typ Iterator	163
4.13	Zusammenfassung des Musiksammlung-Beispiels	167
4.14	Ein weiteres Beispiel: ein Auktionssystem	169
Kapitel 5	Funktionale Verarbeitung von Sammlungen (fortgeschrittene Konzepte)	183
5.1	Ein neuer Blick auf die Themen von Kapitel 4	183
5.2	Monitoring von Tierpopulationen	185
5.3	Lambda-Ausdrücke – ein erster Blick	188
5.4	Die forEach-Methode von Sammlungen	190
5.5	Streams	193
Kapitel 6	Bibliotheksklassen nutzen	207
6.1	Die Dokumentation der Bibliotheksklassen	208
6.2	Das Kundendienstsystem	209
6.3	Die Klassendokumentation lesen	215
6.4	Zufälliges Verhalten einbringen	221
6.5	Pakete und Importe	226
6.6	Benutzung von Map-Klassen für Abbildungen	227
6.7	Der Umgang mit Mengen	232
6.8	Zeichenketten zerlegen	233
6.9	Abschluss des Kundendienstsystems	234
6.10	Autoboxing und Wrapper-Klassen	236
6.11	Die Klassendokumentation schreiben	239
6.12	Öffentliche und private Eigenschaften	242
6.13	Klassen über ihre Schnittstelle verstehen	245
6.14	Klassenvariablen und Konstanten	250
6.15	Klassenmethoden	253
6.16	Programmausführung ohne BlueJ	254
6.17	Weitere fortgeschrittene Konzepte	255
Kapitel 7	Sammlungen mit fester Größe – Arrays	263
7.1	Sammlungen fester Größe	263
7.2	Arrays	263
7.3	Die Analyse einer Logdatei	264
7.4	Die for-Schleife	270
7.5	Das Projekt Automat	276
7.6	Arrays mit mehr als einer Dimension (fortgeschritten)	285
7.7	Arrays und Streams (fortgeschritten)	291

Kapitel 8	Klassenentwurf	293
8.1	Einführung	294
8.2	Die Welt von Zuul	296
8.3	Kopplung und Kohäsion	297
8.4	Code-Duplizierung	298
8.5	Erweiterungen für Zuul	301
8.6	Kopplung	303
8.7	Entwurf nach Zuständigkeiten	308
8.8	Änderungen lokal halten	311
8.9	Implizite Kopplung	311
8.10	Vorausdenken	315
8.11	Kohäsion	316
8.12	Refactoring	320
8.13	Refactoring für Sprachunabhängigkeit	324
8.14	Entwurfsregeln	330
Kapitel 9	Fehler vermeiden	335
9.1	Einführung	335
9.2	Testen und Fehlerbeseitigung	336
9.3	Modultests in BlueJ	337
9.4	Tests automatisieren	343
9.5	Refactoring bei Streams einsetzen (fortgeschritten)	351
9.6	Debugging	351
9.7	Kommentierung und Programmierstil	353
9.8	Manuelle Ausführung	354
9.9	Ausgabebeispielen	360
9.10	Debugger	363
9.11	Das Debugging von Streams (fortgeschritten)	365
9.12	Die Wahl der richtigen Teststrategie	366
9.13	Techniken umsetzen	367
Teil II	Anwendungsstrukturen	369
Kapitel 10	Bessere Struktur durch Vererbung	371
10.1	Das Beispiel „Netzwerk“	372
10.2	Einsatz von Vererbung	382
10.3	Vererbungshierarchien	384
10.4	Vererbung in Java	385
10.5	Weitere Einsendungen für Netzwerk	389
10.6	Vorteile durch Vererbung (bis hierher)	391

10.7 Subtyping	391
10.8 Die Klasse Object	397
10.9 Die Hierarchie der Sammlungstypen	398
Kapitel 11 Mehr über Vererbung	403
11.1 Das Problem: die Methode zum Anzeigen	403
11.2 Statischer und dynamischer Typ	405
11.3 Überschreiben von Methoden	408
11.4 Dynamische Methodensuche	410
11.5 super-Aufrufe in Methoden	413
11.6 Methoden-Polymorphie	414
11.7 Methoden aus Object: toString	414
11.8 Objektgleichheit: equals und hashCode	417
11.9 Der Zugriff über protected	420
11.10 Der Operator instanceof	422
11.11 Ein weiteres Beispiel für Vererbung mit Überschreiben	423
Kapitel 12 Weitere Techniken zur Abstraktion	429
12.1 Simulationen	429
12.2 Die Fuechse-und-Hasen-Simulation	431
12.3 Abstrakte Klassen	444
12.4 Weitere abstrakte Methoden	451
12.5 Multiple Vererbung	453
12.6 Interfaces	457
12.7 Ein weiteres Beispiel für ein Interface	465
12.8 Die Klasse Class	467
12.9 Abstrakte Klasse oder Interface?	468
12.10 Ereignisgesteuerte Simulationen	468
12.11 Zusammenfassung der Vererbung	470
Kapitel 13 Grafische Benutzungsoberflächen	475
13.1 Einführung	475
13.2 Komponenten, Layout und Ereignisbehandlung	476
13.3 AWT und Swing	477
13.4 Das Beispiel: ein Bildbetrachter	478
13.5 Bildbetrachter 1.0: die erste komplette Version	490
13.6 Bildbetrachter 2.0: die Programmstruktur verbessern	504
13.7 Bildbetrachter 3.0: weitere GUI-Komponenten	509
13.8 Innere Klassen	513
13.9 Zusätzliche Erweiterungen	518
13.10 Ein weiteres Beispiel: der Musikplayer	520

Kapitel 14 Fehlerbehandlung	527
14.1 Das Adressbuch-Projekt	528
14.2 Defensive Programmierung	531
14.3 Fehlermeldungen durch den Dienstleister	534
14.4 Prinzipien der Exception-Behandlung	539
14.5 Die Behandlung von Exceptions	545
14.6 Neue Exception-Klassen definieren	552
14.7 Die Verwendung von Zusicherungen	553
14.8 Wiederaufsetzen und Fehlervermeidung	557
14.9 Dateibasierte Ein- und Ausgabe	560
Kapitel 15 Entwurf von Anwendungen	573
15.1 Analyse und Entwurf	573
15.2 Klassenentwurf	581
15.3 Dokumentation	583
15.4 Kooperation	584
15.5 Prototyping	585
15.6 Softwarewachstum	586
15.7 Der Einsatz von Entwurfsmustern	588
Kapitel 16 Eine Fallstudie	597
16.1 Die Fallstudie	597
16.2 Analyse und Entwurf	599
16.3 Klassenentwurf	602
16.4 Iterative Entwicklung	608
16.5 Ein weiteres Beispiel	616
16.6 Ein Blick nach vorn	616
Anhang A Arbeiten mit BlueJ-Projekten	617
A.1 BlueJ installieren	617
A.2 Ein Projekt öffnen	617
A.3 Der Debugger in BlueJ	617
A.4 BlueJ konfigurieren	618
A.5 Auf deutsche Schnittstelle umstellen	618
A.6 Einbinden einer lokalen API-Dokumentation	619
A.7 Vorlagen für neue Klassen ändern	619
Anhang B Datentypen in Java	621
B.1 Primitive Typen	621
B.2 Cast-Operator für primitive Typen	622
B.3 Objekttypen	623

B.4 Wrapper-Klassen	623
B.5 Cast-Operator für Objekttypen	624
Anhang C Operatoren	625
C.1 Arithmetische Ausdrücke	625
C.2 Boolesche Ausdrücke	626
C.3 Abkürzungsoperatoren	627
Anhang D Kontrollstrukturen in Java	629
D.1 Kontrollstrukturen	629
D.2 Auswahlanweisungen	629
D.3 Schleifen	632
D.4 Exceptions	634
D.5 Zusicherungen	635
Anhang E Java ohne BlueJ	637
E.1 Java ohne BlueJ ausführen	637
E.2 Konsolenanwendungen und die Problematik der Umlaute	639
E.3 Ausführbare jar-Dateien erzeugen	641
E.4 Entwickeln ohne BlueJ	642
Anhang F Benutzung des Debuggers	643
F.1 Haltepunkte	644
F.2 Die Kontrollknöpfe	644
F.3 Anzeige der Variablen	645
F.4 Die Anzeige der Aufruffolge	646
F.5 Die Thread-Anzeige	646
Anhang G Testwerkzeuge für Modultests mit JUnit	647
G.1 Aktivieren der Test-Funktionalität	647
G.2 Eine Testklasse erzeugen	647
G.3 Eine Testmethode erzeugen	647
G.4 Zusicherungen bei Tests	648
G.5 Tests ausführen	648
G.6 Testgerüste	648
Anhang H Werkzeuge für die Teamarbeit	649
H.1 Server-Einrichtung	649
H.2 Teamarbeit-Funktionalität aktivieren	649
H.3 Ein Projekt zur gemeinsamen Nutzung einrichten	649
H.4 An einem Projekt gemeinsam arbeiten	650
H.5 Aktualisieren und Abgeben	650
H.6 Weitere Informationen	650

Anhang I	Javadoc	651
I.1	Dokumentationskommentare	651
I.2	Unterstützung für Javadoc in BlueJ	654
Anhang J	Quelltextkonventionen	655
J.1	Benennung	655
J.2	Layout	656
J.3	Dokumentation	657
J.4	Restriktionen bei der Sprachbenutzung	658
J.5	Programmiermuster	659
Anhang K	Wichtige Bibliotheksklassen	661
K.1	Das Paket java.lang	661
K.2	Das Paket java.util	662
K.3	Die Pakete java.io und java.nio.file	664
K.4	Das Paket java.util.function	665
K.5	Das Paket java.net	666
K.6	Weitere wichtige Pakete	666
Register		667