

Inhaltsverzeichnis

1. Einleitung	9
1.1 Java - was ist das?	9
1.2 Ein kurzer Überblick über die Geschichte der Programmiersprache	11
1.3 Compiler oder Interpreter - die Besonderheit von Java	12
1.4 Java: hervorragend geeignet für den Einstieg in die Programmierung	15
2. Vorbereitungsmaßnahmen für die Programmierung	17
2.1 Java Development Kit und IDE installieren	17
2.2 Das Java Development Kit für die Anwendung vorbereiten	21
2.3 Einen passenden Texteditor installieren	27
3. Das erste Programm gestalten	30
3.1 Ein Programm mit einer einfachen Ausgabe schreiben	30
3.2 Die verschiedenen Elemente des Programmcodes	32
3.3 Das Programm kompilieren und ausführen	35
3.4 Kommentare für ein leichteres Verständnis des Programms	38
3.5 Übungsaufgabe: Mehrere Zeilen mit einem Java-Programm ausgeben	40
4. Variablen und Operatoren	44
4.1 Was sind Variablen?	44
4.2 Variablen in Java verwenden	46
4.3 Verschiedene Variablentypen	49
4.4 Arrays: mehrere Werte zu größeren Einheiten zusammenführen	55
4.5 Mathematische Operatoren	60
4.6 Weitere Operatoren für Variablen	63
4.7 Den Typ der Daten ändern	65
4.8 Übungsaufgabe: Werte mit Variablen verarbeiten	69
5. if-Abfragen: unverzichtbar für die Programmierung mit Java	73
5.1 So ist eine if-Abfrage in Java aufgebaut	73

5.2 Vergleichsoperatoren und logische Operatoren	74
5.3 Alternativen durch Else-Verzweigungen einfügen	79
5.4 Übung: Programme mit Abfragen und Verzweigungen erstellen	83
6. Schleifen: ein wichtiger Bestandteil vieler Programme	88
6.1 While-Schleifen bieten vielfältige Steuerungsmöglichkeiten	88
6.2 Fußgesteuerte do-while-Schleife	92
6.3 For-Schleifen: ideal für eine feste Anzahl von Durchläufen	94
6.4 For-each-Schleifen: speziell auf Arrays zugeschnitten	95
6.5 Übung: verschiedene Schleifen selbst programmieren.....	98
7. Objektorientierte Programmierung: essenziell für das Verständnis von Java	104
7.1 Grundzüge und Vorteile der objektorientierten Programmierung...	104
7.2 Die Klasse und ihre Attribute	106
7.3 Objekte: Instanzen der Klassen	108
7.4 Methoden: Funktionen für Objekte.....	111
7.5 Vererbung: wichtiges Prinzip der objektorientierten Programmierung	115
7.6 Übung: objektorientierte Programme gestalten	118
8. API: Mit vorgefertigten Routinen arbeiten	122
8.1 Was bedeutet der Begriff API?	122
8.2 Anwendungsbeispiel: Fenster für den Dialog mit dem Anwender ..	123
8.3 Vielfältige weitere Bibliotheken nutzen	126
8.4 Die Dokumentation der APIs.....	128
8.5 Übung: vorgefertigte Methoden nutzen.....	129
9. Exception Handling	135
9.1 Was sind Laufzeitfehler?	135
9.2 Laufzeitfehler durch Ausnahmen abfangen	137
9.3 Wann ist die Verwendung von Ausnahmen sinnvoll?	140

10. Daten in Dateien abspeichern	142
10.1 Daten einlesen	143
10.2 Daten speichern	148
10.3 Einen Eintrag löschen	151
10.4 Übung: Dateien in Programme einbinden	155
11. IDE: hilfreich für die Erstellung von Programmen	158
11.1 Welche Vorteile bietet eine IDE?	158
11.2 IDEs für die Programmierung von Java	159
11.3 Ein neues Projekt mit NetBeans erstellen	160
11.4 Einige hilfreiche Funktionen der IDE	163
12. JavaFX: hilfreich bei der Gestaltung grafischer Benutzeroberflächen 	167
12.1 Verschiedene Techniken für die Erstellung von User Interfaces ..	167
12.2 Das erste User Interface erstellen	169
12.3 Ein einfaches Formular mit JavaFX erstellen	175
12.4 Scene Builder: grafische Benutzeroberflächen erstellen	183
12.5 Übung: ein kleines Rechenprogramm mit grafischem UI.....	189
13. Anwendungsbeispiel: ein kleines Adressbuch.....	195
13.1 Die grundlegende Benutzeroberfläche	197
13.2 Ein Formular für die Eingabe neuer Adressen.....	201
13.3 Vorhandene Adressen abrufen.....	214
13.4 Eine Adresse löschen.....	219
14. Multithreading	231
14.1 Einen Thread in Java erzeugen.....	232
14.2 Die Zustände eines Threads	238
14.3 Die Reihenfolge bei der Ausführung ist nicht vorhersehbar.....	239
14.4 Priorität eines Threads festlegen	242
14.5 Probleme beim Zugriff auf gemeinsam verwendete Variablen	246

14.6 Interferenzen bei der Ausführung vermeiden.....	251
14.7 Übungsaufgabe: Threads selbst erstellen	253
14.8 Ein kleines Anwendungsbeispiel für Multithreading	261
15. Datenbanken: Programmdaten sicher abspeichern	271
15.1 Was ist eine Datenbank.....	271
15.2 Java DB: Einfache Einbindung von Datenbanken unter Java	273
15.3 Eine Datenbank erzeugen und Tabellen, Spalten und Felder einfügen.....	274
15.4 SQL-Befehle für den Zugriff auf die Daten	286
15.5 Ein Java-Programm mit einer Datenbank verbinden	293
15.6 SQL-Befehle in das Java-Programm einbauen.....	297
15.7 Übungsaufgabe: eine einfache Datenbankanwendung gestalten	302
15.8 Ein Anwendungsbeispiel für Datenbanken	304