

Inhaltsübersicht

Aufbau und Gliederung

Der Schnelleinstieg

Basiskonzepte im Detail

Klassen benutzen und bereitstellen

Assoziationen

Einfachvererbung

Mehrfachvererbung und Schnittstellen

Persistenz und Datenhaltung

Generische Datentypen

Funktionale Programmierung

Konstruktive Qualitätssicherung

Analytische Qualitätssicherung

Von OOA zu OOP

Einführung in C++

Einführung in C#

Inhalt

1	Aufbau und Gliederung *	1
2	Der Schnelleinstieg *	7
2.1	Objekte: das Wichtigste *	8
2.2	Klassen: das Wichtigste *	12
2.3	Klassen in Java: das Wichtigste *	17
2.4	Java-Compiler, -Entwicklungsumgebungen und UML- Werkzeuge *	22
2.5	Java-Konstrukturen: das Wichtigste *	23
2.6	Botschaften in Java: das Wichtigste *	27
3	Basiskonzepte im Detail *	33
3.1	Objekte und Klassen in der UML *	34
3.2	Box: Richtlinien und Konventionen für Bezeichner *	37
3.3	Trennung Benutzungsoberfläche – Fachkonzept *	39
3.4	Attribute in Klassen vs. Attribute in Operationen *	41
3.5	Klassenattribute und -operationen *	45
3.6	Objekte als Eingabeparameter in Java *	52
3.7	Objekte als Ergebnisparameter in Java *	56
3.8	Konstrukturen im Detail *	58
3.9	UML-Sequenzdiagramme *	62
3.10	Objekte als Eingabeparameter in Konstrukturen *	64
3.11	Botschaften *	65
3.12	Box: Kreuzworträtsel 1 *	69
4	Klassen benutzen und bereitstellen *	73
4.1	Pakete *	74
4.2	Dokumentationskommentare und Javadoc *	80
4.3	Vorhandene Klassen benutzen *	84
4.4	Die Hüllklassen für einfache Typen *	87
4.5	Die Klasse ArrayList<E> *	91
4.6	Die Klasse Scanner *	97
4.7	Ausnahmebehandlung mit throw *	103
4.8	Die selbstentwickelte Klasse Console *	108
4.9	Die String-Klassen von Java *	110
4.9.1	Die Klasse String *	111
4.9.2	Die Klasse StringBuilder *	118
4.10	Das Paket java.time *	119
4.11	Vom Problem zur Lösung: Teil 1 **	125
5	Assoziationen *	137
5.1	Links: Beziehungen zwischen Objekten *	138
5.2	Assoziationen: Beziehungen zwischen Klassen *	143
5.3	Assoziationsnamen und Rollen *	147
5.4	Assoziationsklassen – wenn die Assoziation zur Klasse wird *	150
5.5	Höherwertige Assoziationen ***	152
5.6	Navigierbarkeit *	153
5.7	Navigierbarkeit und Multiplizitäten in Java *	156

5.8	Container *	162
5.9	Das Singleton-Muster *	165
6	Einfachvererbung *	169
6.1	Generalisieren – entdecke Gemeinsamkeiten *	170
6.2	Vererbung in Java *	174
6.3	Klassen spezialisieren und Methoden redefinieren *	179
6.4	Die Java-Klassenhierarchie und Object *	186
6.5	Identität vs. Gleichheit *	189
6.6	Klonen vs. Kopieren *	191
6.7	Klassenzugehörigkeit eines Objekts *	198
6.8	Die Vererbungsregeln *	200
6.9	Zugriffsrechte und Sichtbarkeit *	202
6.10	Sonderfälle *	206
6.11	Polymorphismus *	208
7	Mehrfachvererbung und Schnittstellen *	213
7.1	Mehrfachvererbung *	213
7.2	Schnittstellen *	215
7.3	Die Java-Syntax und -Semantik für Schnittstellen *	221
7.4	Leere Implementierung von Schnittstellen *	224
7.5	Schnittstellen und Vererbung *	225
7.6	Konstanten in Schnittstellen *	227
7.7	Die Schnittstellen Iterator und Iterable *	229
7.8	Innere und anonyme Klassen *	234
7.9	Aufzählungen mit enum *	237
7.10	Vom Problem zur Lösung: Teil 2 **	239
8	Persistenz und Datenhaltung *	247
8.1	Persistenz und Datenhaltung in Java *	248
8.2	Dateien sequenziell lesen *	251
8.3	Dateien sequenziell schreiben *	255
8.4	Eine einfache Indexverwaltung *	258
8.4.1	Das Konzept einer indexbasierten Dateiorganisation *	259
8.4.2	Direktzugriffsspeicher in Java *	261
8.5	Drei-Schichten-Architektur *	265
8.6	Vom Fachkonzept zur Drei-Schichten-Architektur *	267
8.7	Die Serialisierung von Objekten *	271
8.8	Vernetzte Objekte serialisieren *	279
9	Generische Datentypen *	285
9.1	Einfache Klassenschablonen *	286
9.2	Typeinschränkung bei generischen Datentypen *	292
9.3	Einfache Methodenschablonen **	296
9.4	Mehrere Typparameter *	298
9.5	Vererbung von generischen Typen ***	301
9.6	Generisches Sortieren *	307
9.7	Generisches Sortieren mit mehreren Sortierkriterien *	311
9.8	Kreuzworträtsel 2 *	316

10	Funktionale Programmierung **	319
10.1	Deklarativer Zugriff auf Sammlungen (<i>Collections</i>) **	321
10.2	Deklarative Programmierung von Datenströmen (<i>Streams</i>) **	326
10.3	Eine Sammlung auf einen Wert reduzieren **	328
10.4	Implementierung von Vergleichsoperationen **	331
10.5	Box: Funktionale Sprachkonzepte im Überblick 1 **	337
10.6	Box: Funktionale Sprachkonzepte im Überblick 2 **	344
11	Konstruktive Qualitätssicherung **	349
11.1	Bindung von Methoden **	351
11.2	Bindung von Klassen **	357
11.3	Kopplung von Methoden ***	359
11.4	Kopplung von Klassen **	363
11.5	<i>Refactoring</i> von Klassen und Methoden **	368
12	Analytische Qualitätssicherung **	373
12.1	Testen – Terminologie und Konzepte **	374
12.2	Dynamische Testverfahren **	378
12.3	Funktionale Äquivalenzklassenbildung **	384
12.4	Grenzwertanalyse und Test spezieller Werte **	389
12.5	Kombinierter Funktions- und Strukturtest **	392
12.6	Testen von Unterklassen ***	403
12.7	Testgetriebenes Programmieren **	407
13	Von OOA zu OOP **	415
13.1	Fallst. Aufgabenplaner: Erstes Gespräch **	417
13.2	Fallst. Aufgabenplaner: Das OOA-Modell **	421
13.3	Fallst. Aufgabenplaner: Die Benutzungsoberfläche **	423
13.4	Fallst. Aufgabenplaner: Das OOD-Modell **	431
13.5	Fallst. Aufgabenplaner: OOP – Fachkonzeptklassen **	433
13.6	Fallst. Aufgabenplaner: OOP – UI-Klasse **	437
13.7	Fallst. Aufgabenplaner: OOP – Container **	440
13.8	Fallst. Aufgabenplaner: OOP – Datenhaltung **	441
14	Einführung in C++ *	443
14.1	Das erste C++-Programm *	444
14.2	Klassen in C++ *	446
14.3	Trennung Schnittstelle – Implementierung *	448
14.4	Dynamische und statische Objekte *	451
14.5	Vererbung und Polymorphismus *	456
14.6	Klassenattribute und Klassenoperationen *	464
14.7	Generische Klassen **	466
14.8	Box: Von Java nach C++ *	470
15	Einführung in C# *	471
15.1	Ein Überblick über .NET *	471
15.2	Ein Überblick über C# *	474
15.3	Einfache und strukturierte Typen *	477
15.4	Klassen *	483
15.5	Kreuzworträtsel 3 *	489

Anhang A Java: Syntaxnotation *	493
Anhang B Kreuzwörtertsel 1: Lösung **	499
Anhang C Kreuzwörtertsel 2: Lösung *	501
Anhang D Kreuzwörtertsel 3: Lösung *	503
Glossar	505
Literatur	517
Sachindex	519