

# Inhaltsverzeichnis

<b>1 Einleitung</b>	<b>1</b>
1.1 Programmierung . . . . .	1
1.2 Objektorientierte Programmierung . . . . .	2
1.3 Voraussetzungen und Ziele dieses Buches . . . . .	3
1.4 Aufbau dieses Buches . . . . .	4
<b>2 Das imperative Hamster-Modell</b>	<b>9</b>
2.1 Komponenten des Hamster-Modells . . . . .	9
2.2 Anweisungen und Programme . . . . .	11
2.3 Prozeduren . . . . .	14
2.4 Auswahlanweisungen . . . . .	15
2.5 Wiederholungsanweisungen . . . . .	17
2.6 Boolesche Funktionen . . . . .	19
2.7 Variablen und Ausdrücke . . . . .	21
2.8 int-Funktionen . . . . .	24
2.9 Funktionsparameter . . . . .	26
2.10 Rekursion . . . . .	28
<b>3 Hamster-Objekte</b>	<b>31</b>
3.1 Das objektorientierte Hamster-Modell . . . . .	31
3.2 Hamster-Namen . . . . .	32
3.3 Der Standard-Hamster . . . . .	33
3.4 Aufruf von Hamster-Befehlen . . . . .	33
3.5 Koordinatensystem . . . . .	34
3.6 Erzeugung neuer Hamster . . . . .	34
3.7 Initialisierung erzeugter Hamster . . . . .	35
3.8 Hamster-Befehle . . . . .	36
3.9 Neue Hamster-Befehle . . . . .	38
3.10 Der Datentyp Hamster . . . . .	39
3.11 Beispielprogramme . . . . .	45
3.12 Aufgaben . . . . .	50
<b>4 Die Klasse Hamster</b>	<b>55</b>
4.1 Motivation . . . . .	55
4.2 Attribute . . . . .	57
4.3 Methoden . . . . .	57
4.4 Implementierung von Methoden . . . . .	58
4.5 this . . . . .	59
4.6 Implementierung der Hamster-Methode maulLeer . . . . .	61
4.7 Erzeugung von Hamster-Objekten . . . . .	61
4.8 Initialisierung von Hamster-Objekten . . . . .	62

4.9 Aufgaben . . . . .	63
<b>5 Erweiterte Hamster-Klassen (Grundlagen)</b>	<b>67</b>
5.1 Definition erweiterter Hamster-Klassen . . . . .	68
5.2 Definition von Attributen . . . . .	69
5.3 Definition von Methoden . . . . .	69
5.4 Zugriff auf Attribute . . . . .	70
5.5 Attribute versus methodenlokale Variablen . . . . .	71
5.6 Erzeugung erweiterter Hamster-Objekte . . . . .	72
5.7 Aufruf von Methoden . . . . .	73
5.8 Komplette Programme mit erweiterten Hamster-Klassen . . . . .	74
5.9 Separate Klassen . . . . .	76
5.10 Beispielprogramme . . . . .	78
5.11 Aufgaben . . . . .	83
<b>6 Erweiterte Hamster-Klassen (Fortsetzung)</b>	<b>91</b>
6.1 Überladen von Methoden . . . . .	91
6.2 Rekursive Methoden . . . . .	92
6.3 Konstruktoren . . . . .	93
6.4 Klassenattribute . . . . .	98
6.5 Klassenmethoden . . . . .	101
6.6 Die Klasse Territorium . . . . .	104
6.7 Konstanten . . . . .	107
6.8 Subobjekte . . . . .	108
6.9 Enums . . . . .	111
6.10 Beispielprogramme . . . . .	113
6.11 Aufgaben . . . . .	124
<b>7 Vererbung</b>	<b>131</b>
7.1 Erben der Klasse Hamster . . . . .	131
7.2 Ableiten von erweiterten Hamster-Klassen . . . . .	133
7.3 <b>super</b> . . . . .	135
7.4 Überschreiben von Methoden . . . . .	136
7.5 Gleichnamige Attribute . . . . .	140
7.6 Konstruktoren . . . . .	141
7.7 Klassenattribute . . . . .	145
7.8 <b>final</b> . . . . .	146
7.9 Vorteile der Vererbung . . . . .	147
7.10 Allround-Hamster . . . . .	148
7.11 Beispielprogramme . . . . .	154
7.12 Aufgaben . . . . .	163
<b>8 Arrays</b>	<b>167</b>
8.1 Motivation . . . . .	167
8.2 Definition und Erzeugung von Arrays . . . . .	168
8.3 Zuweisung . . . . .	169
8.4 Zugriff auf Arrays . . . . .	171
8.5 Laufzeitfehler beim Zugriff auf Arrays . . . . .	172

8.6 Initialisierung von Arrays . . . . .	173
8.7 Referenzdatentypen . . . . .	174
8.8 Ermittlung der Größe eines Arrays . . . . .	175
8.9 Arrays als Parameter von Funktionen und Methoden . . . . .	176
8.10 Arrays als Funktions- bzw. Methodenwerte . . . . .	178
8.11 for-Schleife . . . . .	179
8.12 Arrays als Attribute . . . . .	183
8.13 Hamster-Arrays . . . . .	187
8.14 Mehrdimensionale Arrays . . . . .	187
8.15 Beispielprogramme . . . . .	192
8.16 Aufgaben . . . . .	200
<b>9 Verallgemeinerung des Klassen/Objekte-Konzeptes</b>	<b>209</b>
9.1 Klassen und Objekte . . . . .	209
9.2 Vererbung . . . . .	212
9.3 Beispielprogramme . . . . .	214
9.4 Aufgaben . . . . .	223
<b>10 Ein- und Ausgabe</b>	<b>227</b>
10.1 Zeichenketten . . . . .	227
10.2 Ausgabe . . . . .	231
10.3 Eingabe . . . . .	233
10.4 Beispielprogramme . . . . .	234
10.5 Aufgaben . . . . .	242
<b>11 Polymorphie und dynamisches Binden</b>	<b>249</b>
11.1 Polymorphie . . . . .	249
11.2 Einschränkung des Protokolls . . . . .	251
11.3 Polymorphie und Parameter . . . . .	253
11.4 Explizite Typumwandlung . . . . .	254
11.5 Dynamisches Binden von Methoden . . . . .	256
11.6 Erweiterbarkeit . . . . .	261
11.7 Die Klasse <code>Object</code> . . . . .	262
11.8 Wiederverwendbarkeit . . . . .	264
11.9 Autoboxing und Unboxing . . . . .	268
11.10 Weitere Methoden der Klasse <code>Territorium</code> . . . . .	270
11.11 Beispielprogramme . . . . .	272
11.12 Aufgaben . . . . .	280
<b>12 Abstrakte Klassen und Interfaces</b>	<b>287</b>
12.1 Abstrakte Klassen . . . . .	287
12.2 Interfaces . . . . .	297
12.3 Vergleich von abstrakten Klassen und Interfaces . . . . .	308
12.4 Beispielprogramme . . . . .	309
12.5 Aufgaben . . . . .	321
<b>13 Fehlerbehandlung mit Exceptions</b>	<b>329</b>
13.1 Überblick . . . . .	329
13.2 Fehlerklassen . . . . .	334

13.3 Erzeugen und Werfen von Fehlerobjekten . . . . .	335
13.4 Fehlertypdeklaration . . . . .	336
13.5 Auffangen von Fehlerobjekten . . . . .	339
13.6 Unchecked-Exceptions . . . . .	348
13.7 Verwendung von Exceptions . . . . .	354
13.8 Beispielprogramme . . . . .	355
13.9 Aufgaben . . . . .	366
<b>14 Zugriffsrechte und Pakete</b>	<b>373</b>
14.1 Zugriffsrechte . . . . .	373
14.2 Pakete . . . . .	379
14.3 Zusammenspiel von Zugriffsrechten und Paketen . . . . .	390
14.4 Beispielprogramme . . . . .	393
14.5 Aufgaben . . . . .	403
<b>15 Generics</b>	<b>407</b>
15.1 Hamster und ihre Frauen . . . . .	408
15.2 Generische Klassen . . . . .	410
15.3 Bounds . . . . .	414
15.4 Wildcards . . . . .	418
15.5 Generische Methoden . . . . .	422
15.6 Einschränkungen . . . . .	425
15.7 Beispielprogramme . . . . .	428
15.8 Aufgaben . . . . .	440
<b>16 Java-Klassenbibliothek</b>	<b>445</b>
16.1 Das Paket <code>java.lang</code> . . . . .	446
16.2 Utilities . . . . .	448
16.3 Graphische Benutzungsoberflächen . . . . .	465
16.4 Ein- und Ausgabe mit Java . . . . .	475
16.5 Netzwerkprogrammierung . . . . .	479
16.6 Aufgaben . . . . .	484
<b>17 Ausblick</b>	<b>491</b>
17.1 Hamster-Programme versus Java-Programme . . . . .	492
17.2 Objektorientierte Softwareentwicklung . . . . .	493
17.3 Parallele Programmierung . . . . .	495
<b>A Klassen des Java-Hamster-Modells</b>	<b>499</b>
A.1 Exception-Klassen . . . . .	499
A.2 Die Klasse <code>Hamster</code> . . . . .	502
A.3 Die Klasse <code>Territorium</code> . . . . .	508
<b>Glossar</b>	<b>511</b>
<b>Literaturverzeichnis</b>	<b>515</b>
<b>Sachverzeichnis</b>	<b>517</b>