

# Inhalt

<b>Vorwort .....</b>	<b>XVII</b>
<b>Danksagung .....</b>	<b>XIX</b>
<b>1 Einleitung .....</b>	<b>1</b>
1.1 Programmieren lernen in einer Woche .....	1
1.2 Aufbau des Buches .....	2
1.3 Zielgruppe des Buches .....	3
1.4 Programmieren lernen „interaktiv“ .....	4
1.5 Download der Beispiele und Hilfe .....	4
1.6 Anregungen und Kritik .....	5
<b>Teil I Einführung in Python 3 – Für Ein- und Umsteiger .....</b>	<b>7</b>
<b>2 Kommandos und Programme .....</b>	<b>9</b>
2.1 Erste Schritte mit Python .....	9
2.1.1 Linux .....	9
2.1.2 Windows .....	10
2.2 Herkunft und Bedeutung des Begriffes interaktive Shell .....	11
2.2.1 Erste Schritte in der interaktiven Shell .....	11
2.3 Verlassen der Python-Shell .....	13
2.4 Benutzung von Variablen .....	13
2.5 Mehrzellige Anweisungen in der interaktiven Shell .....	14
2.6 Programme schreiben oder schnell mal der Welt “Hallo” sagen .....	14
<b>3 Bytecode und Maschinencode .....</b>	<b>17</b>
3.1 Einführung .....	17
3.2 Unterschied zwischen Programmier- und Skriptsprachen .....	17
3.3 Interpreter- oder Compilersprache .....	17

<b>4</b>	<b>Datentypen und Variablen .....</b>	<b>21</b>
4.1	Einführung.....	21
4.2	Statische und dynamische Typdeklaration.....	22
4.3	Typumwandlung .....	24
4.4	Datentyp ermitteln .....	25
<b>5</b>	<b>Sequentielle Datentypen .....</b>	<b>27</b>
5.1	Übersicht.....	27
5.1.1	Zeichenketten oder Strings .....	27
5.1.2	Listen.....	28
5.1.3	Tupel .....	28
5.1.4	Sequenz von Binärdaten .....	28
5.2	Indizierung von sequentiellen Datentypen .....	28
5.3	Slicing oder Ausschneiden.....	30
5.4	Aufgaben .....	31
<b>6</b>	<b>Dictionaries .....</b>	<b>33</b>
6.1	Dictionaries und assoziative Felder .....	33
6.2	Definition und Benutzung.....	34
6.3	Fehlerfreie Zugriffe auf Dictionaries .....	36
6.4	Zulässige Typen für Schlüssel und Werte .....	37
6.5	Verschachtelte Dictionaries .....	37
6.6	Methoden auf Dictionaries .....	38
6.7	Operatoren.....	41
6.8	Zip.....	42
6.9	Dictionaries aus Listen erzeugen .....	43
6.10	Aufgaben .....	44
<b>7</b>	<b>Mengen .....</b>	<b>47</b>
7.1	Übersicht.....	47
7.2	Mengen in Python .....	47
7.2.1	Sets erzeugen .....	48
7.2.2	Mengen von unveränderlichen Elementen.....	48
7.3	Frozensets.....	49
7.4	Operationen auf „set“-Objekten .....	49
7.4.1	add(element).....	49
7.4.2	clear() .....	50
7.4.3	copy .....	50
7.4.4	difference() .....	50

7.4.5	difference_update()	51
7.4.6	discard(el)	51
7.4.7	remove(el)	52
7.4.8	intersection(s)	52
7.4.9	isdisjoint()	52
7.4.10	issubset()	53
7.4.11	issuperset()	53
7.4.12	pop()	53
<b>8</b>	<b>Verzweigungen</b>	<b>55</b>
8.1	Verzweigungen im Allgemeinen	55
8.2	Bedingte Anweisungen in Python	55
8.3	Beispiel Hundejahre	56
8.4	Wahr oder falsch: Bedingungen in Verzweigungen	57
8.5	Aufgaben	58
<b>9</b>	<b>Schleifen</b>	<b>59</b>
9.1	Übersicht	59
9.2	while-Schleife	60
9.3	die Alternative im Erfolgsfall: else	61
9.4	For-Schleife	62
9.5	Aufgaben	65
<b>10</b>	<b>Dateien lesen und schreiben</b>	<b>69</b>
10.1	Dateien	69
10.2	Text aus einer Datei lesen	69
10.3	Schreiben in eine Datei	71
10.4	In einem Rutsch lesen: readlines und read	71
10.5	Aufgaben	72
<b>11</b>	<b>Listen und Tupel im Detail</b>	<b>75</b>
11.1	Stapelspeicher	75
11.2	Stapelverarbeitung in Python: pop und append	76
11.3	extend	77
11.4	Entfernen eines Wertes	77
11.5	Prüfen, ob ein Element in Liste enthalten ist	78
11.6	Finden der Position eines Elementes	78
11.7	Einfügen von Elementen	79
11.8	Besonderheiten bei Tupel	79

11.8.1	Leere Tupel .....	80
11.8.2	1-Tupel .....	80
11.8.3	Mehrfachzuweisungen, Packing und Unpacking .....	81
11.9	Die veränderliche Unveränderliche .....	81
11.10	Sortieren von Listen .....	82
11.10.1	„sort“ und „sorted“ .....	82
11.10.2	Umkehrung der Sortierreihenfolge .....	83
11.10.3	Eigene Sortierfunktionen .....	83
11.11	Aufgaben .....	86
<b>12</b>	<b>Modularisierung .....</b>	<b>89</b>
12.1	Module .....	89
12.1.1	Namensräume von Modulen .....	90
12.1.2	Namensräume umbenennen .....	91
12.1.3	Modularten .....	91
12.1.4	Suchpfad für Module .....	92
12.1.5	Inhalt eines Modules .....	93
12.1.6	Eigene Module .....	93
12.1.7	Dokumentation für eigene Module .....	94
12.2	Pakete .....	95
<b>13</b>	<b>Flaches und tiefes Kopieren .....</b>	<b>97</b>
13.1	Einführung .....	97
13.2	Kopieren einer Liste .....	99
13.3	Kopie mit Teilbereichsoperator .....	100
13.4	Kopieren mit deepcopy aus dem Modul copy .....	101
13.5	Deepcopy für Dictionaries .....	101
<b>14</b>	<b>Funktionen .....</b>	<b>103</b>
14.1	Allgemein .....	103
14.2	Funktionen in Python .....	104
14.3	Optionale Parameter .....	105
14.4	Docstring .....	106
14.5	Schlüsselwortparameter .....	106
14.6	Rückgabewerte .....	106
14.7	Mehrere Rückgabewerte .....	107
14.8	Lokale und globale Variablen in Funktionen .....	108
14.9	Beliebige Anzahl von Parametern .....	109
14.10	Parameterübergabe .....	110

14.11	Nebeneffekte .....	112
14.12	Kommandozeilenparameter .....	113
14.13	Variable Anzahl von Parametern.....	113
14.14	* in Funktionsaufrufen .....	115
14.15	Beliebige Schlüsselwortparameter .....	116
14.16	Doppeltes Sternchen im Funktionsaufruf .....	116
<b>15</b>	<b>Rekursive Funktionen.....</b>	<b>117</b>
15.1	Definition und Herkunft des Begriffs .....	117
15.2	Definition der Rekursion.....	118
15.3	rekursive Funktionen in Python .....	118
15.4	Die Tücken der Rekursion .....	119
15.5	Fibonacci-Folge in Python.....	120
15.6	Aufgaben .....	124
<b>16</b>	<b>Globale und lokale Variablen .....</b>	<b>127</b>
16.1	Einführung.....	127
16.2	Globale und lokale Variablen in Funktionen .....	128
<b>17</b>	<b>Alles über Strings ... .....</b>	<b>131</b>
17.1	... fast alles .....	131
17.2	Aufspalten von Zeichenketten .....	132
17.2.1	split.....	132
17.2.2	Standardverhalten und „maxsplit“ .....	134
17.2.3	rsplit.....	136
17.2.4	Folge von Trennzeichen .....	137
17.2.5	splitlines .....	138
17.2.6	partition .....	139
17.3	Zusammenfügen von Stringlisten mit join .....	139
17.4	Suchen von Teilstrings .....	140
17.4.1	„in“ oder „not in“ .....	140
17.4.2	s.find(substring[, start[, end]]).....	140
17.4.3	s.rfind(substring[, start[, end]]).....	141
17.4.4	s.index(substring[, start[, end]]) .....	141
17.4.5	s.rindex(substring[, start[, end]]) .....	141
17.4.6	s.count(substring[, start[, end]]) .....	142
17.5	Suchen und Ersetzen .....	142
17.6	Nur noch Kleinbuchstaben oder Großbuchstaben .....	143
17.7	capitalize und title .....	143

17.8	Stripping Strings .....	144
17.9	Strings ausrichten.....	144
17.10	String-Tests .....	145
17.11	Aufgaben .....	147
<b>18</b>	<b>Ausnahmebehandlung.....</b>	<b>149</b>
18.1	Die optionale else-Klausel .....	150
18.2	Fehlerinformationen über sys.exc_info .....	151
18.3	Abfangen mehrerer Exceptions .....	151
18.4	except mit mehrfachen Ausnahmen .....	152
18.5	Exceptions generieren.....	152
18.6	Finalisierungsaktion .....	152
<b>19</b>	<b>Objektorientierte Programmierung .....</b>	<b>155</b>
19.1	Einführung.....	155
19.2	Die Kuchenklasse .....	156
19.3	Objekte .....	156
19.4	Klasse .....	156
19.5	Kapselung von Daten .....	157
19.6	Vererbung .....	158
19.7	Klassen in Python .....	158
19.8	Methoden .....	159
19.9	Konstruktor .....	160
19.10	Destruktor .....	160
19.11	Lauffähige Version der Kontoklasse .....	161
19.12	„Öffentlicher Schönheitsfehler“ oder Public-Attribute .....	162
19.13	Datenkapselung und die Benutzung von Public- Protected- und Private-Attributen .....	163
19.14	Statische Member .....	164
19.14.1	__del__ .....	165
19.15	Properties .....	166
19.16	Dynamische und statische Attribute .....	168
19.17	Vererbung .....	169
19.17.1	Oberbegriffe und Oberklassen .....	169
19.17.2	Vererbung in Python .....	170
19.18	Mehrfachvererbung .....	174
19.18.1	Theorie.....	174
19.18.2	Diamond-Problem oder „deadly diamond of death“ .....	176
19.18.3	Beispiel: CalendarClock .....	177

19.19	Polymorphie .....	185
19.20	Operator-Überladung .....	187
19.21	Standardklassen als Basisklassen .....	188
19.22	Aufgaben .....	189
<b>Teil II Weiterführende Themen .....</b>		<b>193</b>
<b>20</b>	<b>Tests und Fehler .....</b>	<b>195</b>
20.1	Einführung .....	195
20.2	Modultests .....	197
20.3	Modultests unter Benutzung von <code>__name__</code> .....	197
20.4	doctest-Modul .....	200
20.5	Testgetriebene Entwicklung oder „Im Anfang war der Test“ .....	202
20.6	unittest .....	204
20.7	Methoden der Klasse <code>TestCase</code> .....	206
20.8	Aufgaben .....	209
<b>21</b>	<b>Systemprogrammierung .....</b>	<b>211</b>
21.1	Systemprogrammierung .....	211
21.2	Häufig falsch verstanden: Shell .....	211
21.3	os-Modul .....	212
21.3.1	Vorbemerkungen .....	213
21.3.2	Umgebungsvariablen .....	213
21.3.3	Dateiverarbeitung auf niedrigerer Ebene .....	215
21.3.4	Die <code>exec</code> -„Familie“ .....	221
21.3.5	Weitere Funktionen im Überblick .....	226
21.3.6	<code>os.path</code> - Arbeiten mit Pfaden .....	240
21.4	<code>shutil</code> -Modul .....	249
<b>22</b>	<b>Forks .....</b>	<b>255</b>
22.1	Fork .....	255
22.2	Fork in Python .....	255
<b>23</b>	<b>Das Modul „pickle“ .....</b>	<b>259</b>
23.1	Daten „einpökeln“ mit <code>pickle.dump</code> .....	259
23.2	<code>pickle.load</code> .....	260

<b>24</b>	<b>Reguläre Ausdrücke</b>	<b>261</b>
24.1	Ursprünge und Verbreitung	261
24.2	Stringvergleiche	261
24.3	Überlappungen und Teilstrings	263
24.4	Das re-Modul	263
24.5	Matching-Problem	264
24.6	Syntax der regulären Ausdrücke	266
24.6.1	Beliebiges Zeichen	266
24.7	Zeichenauswahl	266
24.8	Endliche Automaten	267
24.9	Vordefinierte Zeichenklassen	268
24.10	Anfang und Ende eines Strings	269
24.11	Optionale Teile	271
24.12	Quantoren	272
24.13	Gruppierungen und Rückwärtsreferenzen	274
24.13.1	Match-Objekte	274
24.14	Umfangreiche Übung	276
24.15	Alles finden mit findall	278
24.16	Alternativen	279
24.17	Kompilierung von regulären Ausdrücken	280
24.18	Aufspalten eines Strings mit oder ohne regulären Ausdruck	282
24.18.1	split-Methode der String-Klasse	282
24.18.2	split-Methode des re-Moduls	283
24.18.3	Wörter filtern	285
24.19	Suchen und Ersetzen mit sub	286
24.20	Aufgaben	286
<b>25</b>	<b>lambda, map, filter und reduce</b>	<b>289</b>
25.1	lambda	289
25.2	map	292
25.3	Filtern von sequentiellen Datentypen mittels „filter“	294
25.4	reduce	294
25.5	Aufgaben	296
<b>26</b>	<b>Listen-Abstraktion/List Comprehension</b>	<b>297</b>
26.1	Die Alternative zu Lambda und Co.	297
26.2	Einführung in die Listen-Abstraktion	297
26.3	Syntax	298
26.4	Weitere Beispiele	299



26.5	Die zugrunde liegende Idee .....	299
26.6	Anspruchsvolleres Beispiel .....	300
26.7	Mengen-Abstraktion .....	301
26.8	Rekursive Funktion zur Berechnung der Primzahlen .....	301
26.9	Generator Comprehension/ Generatoren-Abstraktion.....	302
26.10	Aufgaben .....	303
<b>27</b>	<b>Generatoren und Iteratoren .....</b>	<b>305</b>
27.1	Einführung .....	305
27.2	Iteration in for-Schleifen .....	305
27.3	Generatoren .....	307
27.4	Beispiele.....	309
27.4.1	Permutationen.....	309
27.4.2	Variationen und Kombinationen.....	310
27.5	Generatoren zählen mit firstn und islice.....	312
27.6	send-Methode .....	312
27.7	Generator-Ausdrücke .....	313
27.8	Aufgaben .....	314
<b>28</b>	<b>Memoisation .....</b>	<b>317</b>
28.1	Bedeutung und Herkunft des Begriffes .....	317
28.2	Memoisation mit Dekorateur-Funktionen .....	318
28.3	Memoize in einer Class .....	319
28.4	Dekorateure in Python .....	319
28.5	Überprüfung von Argumenten durch Dekorateure .....	321
<b>29</b>	<b>NumPy .....</b>	<b>323</b>
29.1	Übersicht.....	323
29.2	Arrays in NumPy .....	325
29.3	Arrays flach machen .....	327
29.4	Arrays umdimensionieren .....	328
29.5	Arrays konkatenieren.....	330
29.6	Array, neue Dimension hinzufügen .....	332
29.7	Array mit Nullen und Einsen initialisieren .....	332
29.8	Matrizenarithmetik.....	333
29.9	Vektoraddition und Vektorsubtraktion .....	334
29.10	Matrix-Klasse .....	336
29.10.1	Matrix-Produkt .....	337
29.10.2	Eine einfache praktische Anwendung.....	338

29.11	Inverse Matrix .....	339
29.12	Kreuzprodukt / Vektorprodukt .....	340
29.13	Lineare Gleichungssysteme .....	340
29.14	Polynome .....	342
29.15	Aufgaben .....	343
<b>Teil III</b>	<b>Umfassende Beispiele .....</b>	<b>345</b>
<b>30</b>	<b>Bruchklasse .....</b>	<b>347</b>
30.1	Brüche à la 1001 Nacht .....	347
30.2	Zurück in die Gegenwart .....	348
30.3	Rechenregeln .....	351
30.3.1	Multiplikation von Brüchen .....	351
30.3.2	Division von Brüchen .....	352
30.3.3	Addition von Brüchen .....	353
30.3.4	Subtraktion von Brüchen .....	353
30.3.5	Die Bruchklasse im Überblick .....	354
<b>31</b>	<b>Mastermind .....</b>	<b>357</b>
31.1	Die Ursprünge des Spiels .....	357
31.2	Die Spielregeln .....	358
31.2.1	„Bulls and Cows” .....	358
31.2.2	Mastermind .....	358
31.3	Kombinatorikmodul .....	359
31.4	Mastermind in Python .....	360
<b>32</b>	<b>Textklassifikation .....</b>	<b>365</b>
32.1	Einführung in die Textklassifikation .....	365
32.2	Textklassifikation: Aufgabe .....	367
32.3	Naive-Bayes-Klassifikator .....	367
32.3.1	Definition .....	367
32.3.2	Bayes-Theorem .....	367
32.4	Formale Herleitung der Naive-Bayes-Klassifikation .....	368
32.5	Textklassifikation in Python .....	370
32.5.1	BagOfWords-Klasse .....	370
32.5.2	Document-Klasse .....	371
32.5.3	DocumentClass-Klasse .....	373
32.5.4	Pool-Klasse .....	374

<b>33</b>	<b>Lösungen zu den Aufgaben .....</b>	<b>379</b>
33.1	Lösungen zu Kapitel 5 (Sequentielle Datentypen) .....	379
33.2	Lösungen zu Kapitel 6 (Dictionaries) .....	381
33.3	Lösungen zu Kapitel 8 (Verzweigungen) .....	383
33.4	Lösungen zu Kapitel 9 (Schleifen) .....	385
33.5	Lösungen zu Kapitel 10 (Dateien lesen und schreiben) .....	388
33.6	Lösungen zu Kapitel 11 (Listen und Tupel im Detail) .....	390
33.7	Lösungen zu Kapitel 15 (Rekursive Funktionen) .....	393
33.8	Lösungen zu Kapitel 17 (Alles über Strings ...) .....	397
33.9	Lösungen zu Kapitel 19 (Objektorientierte Programmierung) .....	400
33.10	Lösungen zu Kapitel 20 (Tests und Fehler) .....	405
33.11	Lösungen zu Kapitel 24 (Reguläre Ausdrücke) .....	405
33.12	Lösungen zu Kapitel 25 (lambda, map, filter und reduce) .....	410
33.13	Lösungen zu Kapitel 26 (Listen-Abstraktion/List Comprehension) .....	411
33.14	Lösungen zu Kapitel 27 (Generatoren und Iteratoren) .....	412
33.15	Lösungen zu Kapitel 29 (NumPy) .....	415
	<b>Stichwortverzeichnis .....</b>	<b>417</b>