

# Inhaltsverzeichnis

	<b>Vorwort</b>	<b>11</b>
<b>1</b>	<b>Programmieren für Einsteiger</b>	<b>15</b>
1.1	Was tut eigentlich ein Programmierer?	15
1.2	Wie der Computer mit Daten umgeht	19
1.3	Sprachbarrieren zwischen Mensch und Computer	20
<b>2</b>	<b>Annäherung an Python</b>	<b>23</b>
2.1	Beschaffung und Installation	24
2.2	Werkzeug	25
2.2.1	Die Standard-IDE IDLE	26
2.2.2	Direktaufruf von Editor und Interpreter	27
2.3	Gestaltung eines Python-Programms	29
2.4	Kommentare	30
2.5	Die Last mit den Umlauten	31
<b>3</b>	<b>Speicherwerke für Rechenkünstler</b>	<b>33</b>
3.1	Verschiedene Zahlen	34
3.2	Mathematische Ausdrücke	35
3.3	Programmatisches Geschwätz: Zeichenketten	38
3.4	Alles muss raus: print	39
3.5	Futter für die Programme: input	41
3.6	Konvertierung der Variablentypen	42
3.7	Aufgaben	44
<b>4</b>	<b>Ablaufstrukturen</b>	<b>47</b>
4.1	Abfrage if	47
4.2	Der Umgang mit der Wahrheit	50
4.2.1	Vergleich zwischen Zahlen	50
4.2.2	Vergleich von Zeichenketten	52
4.2.3	Boolesche Verknüpfungen	53
4.3	Programmierte Wiederholung: Schleifen	55
4.3.1	Mit der Bedingung im Kopf: while	55
4.3.2	Prüfung der Benutzereingabe	56
4.3.3	Listen durchschleifen: for	60

4.3.4	Schleifenbruch mit break und continue . . . . .	62
4.3.5	Schleifen können auch anders: else . . . . .	63
4.4	Das Bermuda-Projekt . . . . .	63
4.5	Aufgaben . . . . .	66
<b>5</b>	<b>Funktionen</b> . . . . .	<b>69</b>
5.1	Rückgabewert und Adresse einer Funktion . . . . .	70
5.2	Parameter . . . . .	71
5.2.1	Vorbelegte Parameter . . . . .	73
5.2.2	Parameter per Namen belegen . . . . .	74
5.2.3	Variable Anzahl von Parametern . . . . .	75
5.3	Globale und lokale Variablen . . . . .	76
5.4	Rekursion: Selbstgespräche einer Funktion . . . . .	78
5.5	Die anonyme Funktion lambda . . . . .	81
5.6	Das Bermuda-Projekt . . . . .	81
5.7	Aufgaben . . . . .	83
<b>6</b>	<b>Wir bauen die Welt: Klassen</b> . . . . .	<b>87</b>
6.1	Konstruktor (und Destruktor?) . . . . .	91
6.2	Privat oder Öffentlich? . . . . .	92
6.3	Statisch: Ein Attribut für alle Objekte der Klasse . . . . .	96
6.4	Wenn eine Zuweisung zur Referenz führt . . . . .	98
6.5	Referenzlos im Abfall . . . . .	100
6.6	Ein eigener Datencontainer: Der Binärbaum . . . . .	101
6.6.1	Bäume, Äste und Knoten . . . . .	102
6.6.2	Einbau einer Besucherfunktion . . . . .	105
6.7	Selbst gebaute Operatoren . . . . .	106
6.8	Erweiterte Erbschaftsangelegenheiten . . . . .	109
6.8.1	Vererbung am Beispiel von Supermans Kiosk . . . . .	110
6.8.2	Abstrakte Basisklassen . . . . .	114
6.8.3	Polymorphie? Ein Problem der Anderen . . . . .	116
6.8.4	Mehrfachvererbung . . . . .	117
6.9	Das Bermuda-Projekt . . . . .	119
6.10	Aufgaben . . . . .	121
<b>7</b>	<b>Ausnahmsweise falsch: Die Exception</b> . . . . .	<b>125</b>
7.1	Selbst gemachte Exception basteln . . . . .	128
7.2	Python-Exceptions . . . . .	131
7.3	Eine Zusicherung: Assert . . . . .	132
<b>8</b>	<b>Der Umgang mit Texten</b> . . . . .	<b>133</b>
8.1	Der Umgang mit Zeichenketten und deren Operatoren . . . . .	133
8.2	Die Sequenz und die rechteckigen Klammern . . . . .	134

8.3	Die String-Methoden . . . . .	136
8.4	Textformatierung . . . . .	139
8.4.1	Zeichenketten mit format ausrichten . . . . .	140
8.4.2	Formatierung über das Prozentzeichen . . . . .	141
8.5	Steuerzeichen . . . . .	142
8.6	Internationales Parkett . . . . .	143
8.6.1	Internationale Sonderzeichen im Programm . . . . .	143
8.6.2	Daten umcodieren . . . . .	144
8.6.3	Die nationale Umgebung . . . . .	145
8.7	Mustererkennung mit regulären Ausdrücken . . . . .	149
8.7.1	Die Funktionen für den Vergleich . . . . .	149
8.7.2	Die wichtigsten Platzhalterzeichen . . . . .	152
8.7.3	Beispielhafte Suche . . . . .	152
8.7.4	Verfressene Quantoren . . . . .	155
8.7.5	Noch ein paar Platzhalter . . . . .	155
8.8	Das Bermuda-Projekt . . . . .	156
8.9	Aufgaben . . . . .	157
<b>9</b>	<b>Käfighaltung für Datenrudel . . . . .</b>	<b>159</b>
9.1	Schön eng beieinander: Die Sequenz . . . . .	160
9.2	Heterogene Gemeinsamkeit: Das Tupel . . . . .	160
9.3	Elementweise veränderbar: Die Liste . . . . .	165
9.3.1	Listen anlegen und initialisieren . . . . .	165
9.3.2	Elemente hinzufügen, ändern und löschen . . . . .	166
9.3.3	Umräumen und sortieren . . . . .	168
9.4	Dimensionen . . . . .	169
9.5	Die Reißverschlussfunktion zip . . . . .	171
9.6	Zugriff per Schlüssel: Das Dictionary . . . . .	171
9.7	Einzigartig und unordentlich: Die Menge . . . . .	173
9.8	Das Geheimnis der Container . . . . .	176
9.9	Das Bermuda-Projekt . . . . .	178
9.10	Aufgaben . . . . .	183
<b>10</b>	<b>Grafische Oberfläche (Tkinter) . . . . .</b>	<b>187</b>
10.1	Eine kleine Anwendung . . . . .	189
10.2	Das Programm ist Event-gesteuert . . . . .	190
10.2.1	Kontrollelemente lösen Ereignisse aus . . . . .	190
10.2.2	Ereignisse und Reaktionen . . . . .	192
10.3	Kontrollelemente im Fenster anordnen: Layout . . . . .	194
10.3.1	Pack alles aufeinander! . . . . .	194
10.3.2	Hinter Gitter verteilen: Grid . . . . .	198
10.3.3	Pixelweise positionieren: Place . . . . .	199

10.4	Kontrollelemente und ihre Optionen . . . . .	201
10.4.1	Tkinter-Variablen . . . . .	203
10.4.2	Label: Aufkleber mit Schriftart . . . . .	204
10.4.3	Druckknopf: Button . . . . .	206
10.4.4	Einer unter vielen: Radiobutton . . . . .	207
10.4.5	Abgehakt: Checkbutton . . . . .	209
10.4.6	Der Umgang mit der Listbox . . . . .	210
10.4.7	Ein Scrollbar am Beispiel einer Listbox . . . . .	212
10.4.8	Die Eingabezeile Entry . . . . .	214
10.4.9	Texte anzeigen und editieren . . . . .	217
10.4.10	Schmackhafte Menüs . . . . .	219
10.5	Eine selbst gebaute Tabelle . . . . .	222
10.6	Dialogboxen . . . . .	224
10.6.1	Kurzmeldung: Die MessageBox . . . . .	224
10.6.2	Datei- und Verzeichnisauswahl . . . . .	227
10.6.3	Farbselektor . . . . .	231
10.6.4	Der eigene Dialog: tk_dialog . . . . .	231
10.7	Das Bermuda-Projekt . . . . .	233
10.8	Aufgaben . . . . .	235
<b>11</b>	<b>Kleine Zeichenschule . . . . .</b>	<b>241</b>
11.1	Canvas erzeugen . . . . .	241
11.2	Zeichnen im Canvas . . . . .	242
11.2.1	Linien . . . . .	243
11.2.2	Rechtecke . . . . .	244
11.2.3	Polygone . . . . .	245
11.2.4	Ovale und Kreise . . . . .	246
11.2.5	Bögen und Tortenstücke . . . . .	247
11.2.6	Texte in die Grafik . . . . .	250
11.2.7	Bilder und Symbole . . . . .	251
11.2.8	Die Ereignisse im Canvas . . . . .	253
11.3	Canvas als Widget-Halter . . . . .	256
11.4	Das Bermuda-Projekt . . . . .	257
11.5	Aufgaben . . . . .	262
<b>12</b>	<b>Gut verschnürt: Module, Pakete und Bibliotheken . . . . .</b>	<b>265</b>
12.1	Import . . . . .	265
12.2	Eigene Module . . . . .	267
12.3	Module testen . . . . .	269
12.3.1	Das Hauptprogramm als Selbsttest . . . . .	269
12.3.2	Kommentare testen Funktionen: doctest . . . . .	270
12.3.3	unittest . . . . .	272

12.4	Gebündelt zum Paket . . . . .	274
12.5	Aufgaben . . . . .	276
<b>13</b>	<b>Naturwissenschaftliches . . . . .</b>	<b>277</b>
13.1	Mathematik: math . . . . .	277
13.2	Zufallszahlen: random . . . . .	280
13.3	Dezimalbrüche: decimal . . . . .	280
13.4	Lineare Algebra mit NumPy und SciPy . . . . .	282
13.5	Datumsbibliothek . . . . .	284
	13.5.1 Das Paket datetime . . . . .	284
	13.5.2 Das klassische Modul time . . . . .	286
	13.5.3 Schlafenszeit . . . . .	288
13.6	Aufgaben . . . . .	289
<b>14</b>	<b>Dateten . . . . .</b>	<b>291</b>
14.1	Textdateien . . . . .	291
14.2	Binärdateien . . . . .	293
14.3	Zusammenfassung . . . . .	295
14.4	Exceptions . . . . .	296
14.5	Strukturierte Variablen sichern: JSON . . . . .	297
14.6	Packen und Komprimieren: GZIP, ZIP und TAR . . . . .	298
	14.6.1 Eine Datei komprimieren: gzip . . . . .	299
	14.6.2 ZIP-Dateien und TAR-Archive . . . . .	300
14.7	Aufgaben . . . . .	302
<b>15</b>	<b>Python und Datenbanken . . . . .</b>	<b>305</b>
15.1	Nichts als Tabellen . . . . .	306
15.2	Verbindungsaufbau . . . . .	309
	15.2.1 Eine kleine, lokale Datenbank namens SQLite . . . . .	309
	15.2.2 Die allgegenwärtige Datenbank MySQL . . . . .	310
	15.2.3 Wenn es etwas mehr sein soll: PostgreSQL . . . . .	312
15.3	Tabellen erstellen: CREATE . . . . .	313
15.4	Variationen von INSERT . . . . .	315
15.5	Änderungen mit UPDATE . . . . .	316
15.6	Löschen mit DELETE . . . . .	316
15.7	Ergebnisermittlung durch SELECT . . . . .	317
<b>16</b>	<b>Umgang mit XML-Dateien . . . . .</b>	<b>319</b>
16.1	Eine Speisekarte als XML-Datei . . . . .	321
16.2	XML-Dateien mit ElementTree bearbeiten . . . . .	322
16.3	Der XML-Klassiker DOM . . . . .	326
16.4	Schrittweise durch die XML-Datei: SAX . . . . .	329
16.5	Aufgaben . . . . .	331

<b>17</b>	<b>Netzwerk: Socket, E-Mail und WWW</b>	<b>335</b>
17.1	Der Griff in die Steckdose: Sockets	335
17.2	Versenden von Mails: SMTP	339
17.3	Auslesen von Mails per POP3	343
17.4	Griff ins WWW	347
<b>18</b>	<b>Parallelverarbeitung mit Threads</b>	<b>349</b>
18.1	Das machen wir im Schlaf	350
18.2	Glockengeläut im Threading	350
18.2.1	Die Vereinzelungsanlage	352
18.2.2	Warten auf den Startschuss	354
18.2.3	Ein kleiner Wecker	355
18.3	Die alte Bibliothek thread	356
<b>19</b>	<b>Zusammenarbeit mit dem Betriebssystem</b>	<b>359</b>
19.1	Die Aufrufparameter des Programms	359
19.2	Umgebungsvariablen	360
19.3	Aufruf anderer Programme	361
19.4	Umgang mit dem Dateisystem	363
19.5	Das Ende des Programms	364
	<b>Programmentwicklung und Werkzeuge</b>	<b>367</b>
A	Fehler sind auch nur Menschen: Der Debugger	367
B	Eclipse mit PyDev	368
C	Richtig distribuieren: Das eigene Setup	372
	<b>Stichwortverzeichnis</b>	<b>377</b>