

# Auf einen Blick

## Teil I Einstieg in Python

1	Überblick über Python .....	27
2	Die Arbeit mit Python .....	31
3	Der interaktive Modus .....	39
4	Grundlegendes zu Python-Programmen .....	51
5	Kontrollstrukturen .....	59
6	Das Laufzeitmodell .....	75
7	Basisdatentypen .....	87
8	Dateien .....	197
9	Funktionen .....	209

## Teil II Fortgeschrittene Programmiertechniken

10	Modularisierung .....	257
11	Objektorientierung .....	269
12	Weitere Spracheigenschaften .....	329

## Teil III Die Standardbibliothek

13	Mathematik .....	385
14	Strings .....	415
15	Datum und Zeit .....	451
16	Schnittstelle zum Betriebssystem .....	481
17	Parallele Programmierung .....	525
18	Datenspeicherung .....	549
19	Netzwerkkommunikation .....	607
20	Debugging .....	691

## Teil IV Weiterführende Themen

21	Distribution von Python-Projekten .....	739
22	Optimierung .....	753
23	Grafische Benutzeroberflächen .....	759
24	Wissenschaftliches Rechnen .....	881
25	Anbindung an andere Programmiersprachen .....	903
26	Insiderwissen .....	933
27	Von Python 2 nach Python 3 .....	955

# Inhalt

Über dieses Buch .....	19
------------------------	----

## Teil I Einstieg in Python

---

<b>1 Überblick über Python</b>	27
--------------------------------	----

<b>1.1 Geschichte und Entstehung .....</b>	27
<b>1.2 Grundlegende Konzepte .....</b>	28
<b>1.3 Einsatzmöglichkeiten und Stärken .....</b>	29
<b>1.4 Einsatzbeispiele .....</b>	30

---

<b>2 Die Arbeit mit Python</b>	31
--------------------------------	----

<b>2.1 Die Verwendung von Python .....</b>	31
2.1.1 Windows .....	33
2.1.2 Linux .....	33
2.1.3 Mac OS X .....	33
<b>2.2 Tippen, kompilieren, testen .....</b>	34
2.2.1 Shebang .....	35
2.2.2 Interne Abläufe .....	36

---

<b>3 Der interaktive Modus</b>	39
--------------------------------	----

<b>3.1 Ganze Zahlen .....</b>	40
<b>3.2 Gleitkommazahlen .....</b>	41
<b>3.3 Zeichenketten .....</b>	42
<b>3.4 Listen .....</b>	43
<b>3.5 Variablen .....</b>	43
<b>3.6 Logische Ausdrücke .....</b>	45
<b>3.7 Funktionen und Methoden .....</b>	47

3.7.1	Funktionen .....	47
3.7.2	Methoden .....	48
<b>3.8</b>	<b>Bildschirmausgaben .....</b>	<b>49</b>

---

## **4 Grundlegendes zu Python-Programmen**

---

<b>4.1</b>	<b>Grundstruktur eines Python-Programms .....</b>	<b>51</b>
<b>4.2</b>	<b>Das erste Programm .....</b>	<b>53</b>
<b>4.3</b>	<b>Kommentare .....</b>	<b>55</b>
<b>4.4</b>	<b>Der Fehlerfall .....</b>	<b>56</b>

---

## **5 Kontrollstrukturen**

---

<b>5.1</b>	<b>Fallunterscheidungen .....</b>	<b>59</b>
5.1.1	Die if-Anweisung .....	60
5.1.2	Conditional Expressions .....	63
<b>5.2</b>	<b>Schleifen .....</b>	<b>64</b>
5.2.1	Die while-Schleife .....	65
5.2.2	Vorzeitiger Abbruch einer Schleife .....	65
5.2.3	Erkennen eines Schleifenabbruchs .....	66
5.2.4	Vorzeitiger Abbruch eines Schleifendurchlaufs .....	68
5.2.5	Die for-Schleife .....	70
5.2.6	Die for-Schleife als Zählschleife .....	71
<b>5.3</b>	<b>Die pass-Anweisung .....</b>	<b>73</b>

---

## **6 Das Laufzeitmodell**

---

<b>6.1</b>	<b>Die Struktur von Instanzen .....</b>	<b>77</b>
6.1.1	Datentyp .....	77
6.1.2	Wert .....	78
6.1.3	Identität .....	80
<b>6.2</b>	<b>Referenzen und Instanzen freigeben .....</b>	<b>81</b>
<b>6.3</b>	<b>Mutable vs. immutable Datentypen .....</b>	<b>82</b>
6.3.1	Mutable Datentypen und Seiteneffekte .....	84

<b>7</b>	<b>Basisdatentypen</b>	87
7.1	Operatoren .....	88
7.2	Das Nichts – <code>NoneType</code> .....	92
7.3	Numerische Datentypen .....	93
7.3.1	Arithmetische Operatoren .....	93
7.3.2	Vergleichende Operatoren .....	95
7.3.3	Konvertierung zwischen numerischen Datentypen .....	96
7.3.4	Ganzzahlen – <code>int</code> .....	96
7.3.5	Gleitkommazahlen – <code>float</code> .....	102
7.3.6	Boolesche Werte – <code>bool</code> .....	104
7.3.7	Komplexe Zahlen – <code>complex</code> .....	110
7.4	Parameter von Funktionen und Methoden .....	113
7.4.1	Wiederholung .....	113
7.4.2	Parameter .....	114
7.5	Sequentielle Datentypen .....	115
7.5.1	Operationen auf Instanzen sequentieller Datentypen .....	118
7.5.2	Listen – »list« .....	128
7.5.3	Unveränderliche Listen – <code>tuple</code> .....	139
7.5.4	Strings – <code>str</code> , <code>bytes</code> , <code>bytearray</code> .....	142
7.6	Mappings .....	173
7.6.1	<code>Dictionary</code> – <code>dict</code> .....	174
7.7	Mengen .....	185
7.7.1	Die Datentypen <code>set</code> und <code>frozenset</code> .....	185
7.7.2	Veränderliche Mengen – <code>set</code> .....	193
7.7.3	Unveränderliche Mengen – <code>frozenset</code> .....	195
<b>8</b>	<b>Dateien</b>	197
8.1	Datenströme .....	197
8.2	Daten aus einer Datei auslesen .....	198
8.3	Daten in eine Datei schreiben .....	202
8.4	Verwendung des Dateiobjekts .....	204

<b>9.1</b>	<b>Schreiben einer Funktion</b>	211
<b>9.2</b>	<b>Funktionsparameter</b>	215
9.2.1	Optionale Parameter	215
9.2.2	Schlüsselwortparameter	216
9.2.3	Beliebige Anzahl von Parametern	217
9.2.4	Reine Schlüsselwortparameter	219
9.2.5	Entpacken einer Parameterliste	221
9.2.6	Seiteneffekte	222
<b>9.3</b>	<b>Namensräume</b>	225
9.3.1	Zugriff auf globale Variablen – global	225
9.3.2	Zugriff auf den globalen Namensraum	226
9.3.3	Zugriff auf übergeordnete Namensräume – nonlocal	227
<b>9.4</b>	<b>Lokale Funktionen</b>	229
<b>9.5</b>	<b>Anonyme Funktionen</b>	230
<b>9.6</b>	<b>Rekursion</b>	231
<b>9.7</b>	<b>Eingebaute Funktionen</b>	231

## Teil II Fortgeschrittene Programmietechniken

<b>10.1</b>	<b>Einbinden globaler Module</b>	257
<b>10.2</b>	<b>Lokale Module</b>	260
10.2.1	Namenskonflikte	261
10.2.2	Modulinterne Referenzen	262
<b>10.3</b>	<b>Pakete</b>	262
10.3.1	Absolute und relative Import-Anweisungen	265
10.3.2	Importieren aller Module eines Pakets	266
<b>10.4</b>	<b>Built-in Functions</b>	267

---

<b>11 Objektorientierung</b>	269
<b>11.1 Klassen</b>	274
11.1.1 Definieren von Methoden	276
11.1.2 Der Konstruktor und die Erzeugung von Attributen	277
<b>11.2 Vererbung</b>	280
11.2.1 Technische Grundlagen	281
11.2.2 Die Klasse GirokontoMitTagesumsatz	283
11.2.3 Beispiele möglicher Erweiterungen	289
11.2.4 Ausblick	293
11.2.5 Mehrfachvererbung	294
<b>11.3 Setter und Getter und Property Attributes</b>	295
<b>11.4 Klassenattribute und Klassenmethoden sowie statische Methoden</b>	298
<b>11.5 Magic Methods und Magic Attributes</b>	302
11.5.1 Allgemeine Magic Methods	302
11.5.2 Zugriff auf Attribute anpassen	306
11.5.3 Operatoren überladen	309
11.5.4 Datentypen emulieren	317
<b>11.6 Built-in Functions für Objektorientierung</b>	322
11.6.1 Funktionen für die Verwaltung der Attribute einer Instanz	323
11.6.2 Funktionen für Informationen über die Klassenhierarchie	324
<b>11.7 Objektphilosophie</b>	326
<b>12 Weitere Spracheigenschaften</b>	329
<b>12.1 Exception Handling</b>	329
12.1.1 Eingebaute Exceptions	330
12.1.2 Werfen einer Exception	334
12.1.3 Abfangen einer Exception	335
12.1.4 Eigene Exceptions	340
12.1.5 Erneutes Werfen einer Exception	342
12.1.6 Exception Chaining	344
<b>12.2 Comprehensions</b>	346
12.2.1 List Comprehensions	346
12.2.2 Dict Comprehensions	348
12.2.3 Set Comprehensions	349

<b>12.3</b>	<b>Generatoren</b>	349
<b>12.4</b>	<b>Iteratoren</b>	353
<b>12.5</b>	<b>Docstrings</b>	363
<b>12.6</b>	<b>Interpreter im Interpreter</b>	365
<b>12.7</b>	<b>Die with-Anweisung</b>	368
<b>12.8</b>	<b>Function Annotations</b>	371
<b>12.9</b>	<b>Function Decorator</b>	373
<b>12.10</b>	<b>assert</b>	377
<b>12.11</b>	<b>Weitere Aspekte der Syntax</b>	378
12.11.1	Umbrechen langer Zeilen	378
12.11.2	Zusammenfügen mehrerer Zeilen	379
<b>12.12</b>	<b>Geplante Sprachelemente</b>	380

## Teil III Die Standardbibliothek

---

<b>13</b>	<b>Mathematik</b>	385
<b>13.1</b>	<b>Mathematische Funktionen – math, cmath</b>	385
13.1.1	Zahlentheoretische Funktionen	389
13.1.2	Exponential- und Logarithmusfunktionen	391
13.1.3	Trigonometrische Funktionen	392
13.1.4	Funktionen aus cmath	394
<b>13.2</b>	<b>Zufallszahlengenerator – random</b>	394
13.2.1	Steuerungsfunktionen	396
13.2.2	Funktionen für ganze Zahlen	397
13.2.3	Funktionen für Sequenzen	398
13.2.4	Alternative Generatoren	399
<b>13.3</b>	<b>Präzise Dezimalzahlen – decimal</b>	399
13.3.1	Verwendung des Datentyps	400
13.3.2	Nichtnumerische Werte	403
13.3.3	Das Context-Objekt	404
<b>13.4</b>	<b>Spezielle Generatoren – itertools</b>	405

<b>14 Strings</b>	415
<b>14.1 Reguläre Ausdrücke – re</b>	415
14.1.1 Syntax regulärer Ausdrücke .....	416
14.1.2 Verwendung des Moduls re .....	427
14.1.3 Ein einfaches Beispielprogramm – Searching .....	438
14.1.4 Ein komplexeres Beispielprogramm – Matching .....	439
<b>14.2 Lokalisierung von Programmen – gettext</b>	442
14.2.1 Beispiel für die Verwendung von gettext .....	443
<b>14.3 Hash-Funktionen – hashlib</b>	446
14.3.1 Verwendung des Moduls .....	448
14.3.2 Beispiel .....	450
<b>15 Datum und Zeit</b>	451
<b>15.1 Elementare Zeitfunktionen – time</b>	451
<b>15.2 Komfortable Datumsfunktionen – datetime</b>	460
15.2.1 Attribute und Datentypen (Übersicht) .....	460
15.2.2 date .....	461
15.2.3 datetime.time .....	466
15.2.4 datetime.datetime .....	468
15.2.5 datetime.timedelta .....	477
<b>16 Schnittstelle zum Betriebssystem</b>	481
<b>16.1 Funktionen des Betriebssystems – os</b>	481
16.1.1 Zugriff auf den eigenen Prozess und andere Prozesse .....	482
16.1.2 Zugriff auf das Dateisystem .....	484
<b>16.2 Umgang mit Pfaden – os.path</b>	490
<b>16.3 Zugriff auf das Dateisystem – shutil</b>	495
16.3.1 Verzeichnis- und Dateioperationen .....	497
16.3.2 Archivoperationen .....	499
<b>16.4 Zugriff auf die Laufzeitumgebung – sys</b>	501
16.4.1 Konstanten .....	503
16.4.2 Exceptions .....	506

16.4.3	Hooks .....	507
16.4.4	Sonstige Funktionen .....	509
<b>16.5</b>	<b>Informationen über das System – platform</b> .....	511
<b>16.6</b>	<b>Kommandozeilenparameter – argparse</b> .....	512
16.6.1	Taschenrechner – ein einfaches Beispiel .....	513
16.6.2	Komplexeres Beispiel .....	518
<b>16.7</b>	<b>Kopieren von Instanzen – copy</b> .....	519
<b>16.8</b>	<b>Das Programmende – atexit</b> .....	523

---

## **17 Parallel Programmierung** 525

<b>17.1</b>	<b>Prozesse, Multitasking und Threads</b> .....	525
<b>17.2</b>	<b>Die Thread-Unterstützung in Python</b> .....	528
<b>17.3</b>	<b>Das Modul _thread</b> .....	528
17.3.1	Ein Beispiel für eine rechenintensive Funktion – die Approximation von Pi .....	528
17.3.2	Parallele Berechnung mithilfe von Threads .....	529
17.3.3	Datenaustausch zwischen Threads – locking .....	531
<b>17.4</b>	<b>Das Modul threading</b> .....	536
17.4.1	Locking im threading-Modul .....	539
17.4.2	Worker-Threads und Queues .....	542
17.4.3	Ereignisse definieren – threading.Event .....	546
17.4.4	Barrieren definieren – threading.Barrier .....	546
17.4.5	Eine Funktion zeitlich versetzt ausführen – threading.Timer .....	546

---

## **18 Datenspeicherung** 549

<b>18.1</b>	<b>Komprimierte Dateien lesen und schreiben – gzip</b> .....	549
<b>18.2</b>	<b>XML</b> .....	551
18.2.1	DOM – Document Object Model .....	553
18.2.2	SAX – Simple API for XML .....	565
18.2.3	ElementTree .....	570
<b>18.3</b>	<b>Datenbanken</b> .....	575
18.3.1	Beispieldatenbank .....	577
18.3.2	Pythons eingebaute Datenbank – sqlite3 .....	579

<b>18.4</b>	<b>Serialisierung von Instanzen – pickle</b>	595
<b>18.5</b>	<b>Das Tabellenformat CSV – csv</b>	599
<b>18.6</b>	<b>Temporäre Dateien – tempfile</b>	604

---

## **19 Netzwerkkommunikation**

---

<b>19.1</b>	<b>Socket API</b>	609
19.1.1	Client-Server-Systeme	610
19.1.2	UDP	612
19.1.3	TCP	614
19.1.4	Blockierende und nicht-blockierende Sockets	616
19.1.5	Verwendung des Moduls	618
19.1.6	Die Socket-Klasse	621
19.1.7	Netzwerk-Byte-Order	625
19.1.8	Multiplexende Server – select	627
19.1.9	socketserver	630
<b>19.2</b>	<b>URLs</b>	634
19.2.1	Zugriff auf Ressourcen im Internet – urllib.request	634
19.2.2	Verarbeiten einer URL – urllib.parse	640
<b>19.3</b>	<b>FTP – ftplib</b>	646
<b>19.4</b>	<b>E-Mail</b>	655
19.4.1	SMTP – smtplib	655
19.4.2	POP3 – poplib	660
19.4.3	IMAP4 – imaplib	665
19.4.4	Erstellen komplexer E-Mails – email	671
<b>19.5</b>	<b>Telnet – telnetlib</b>	676
<b>19.6</b>	<b>XML-RPC</b>	680
19.6.1	Der Server	680
19.6.2	Der Client	685
19.6.3	Multicall	687
19.6.4	Einschränkungen	688

---

## **20 Debugging**

---

<b>20.1</b>	<b>Der Debugger</b>	691
<b>20.2</b>	<b>Inspizieren von Instanzen – inspect</b>	694

## Inhalt

20.2.1	Datentypen, Attribute und Methoden .....	696
20.2.2	Quellcode .....	697
20.2.3	Klassen und Funktionen .....	699
<b>20.3</b>	<b>Formatierte Ausgabe von Instanzen – pprint</b> .....	<b>703</b>
<b>20.4</b>	<b>Logdateien – logging</b> .....	<b>706</b>
20.4.1	Das Meldungsformat anpassen .....	709
20.4.2	Logging Handler .....	711
<b>20.5</b>	<b>Automatisiertes Testen</b> .....	<b>713</b>
20.5.1	Testfälle in Docstrings – doctest .....	713
20.5.2	Unit Tests – unittest .....	718
<b>20.6</b>	<b>Traceback-Objekte – traceback</b> .....	<b>722</b>
<b>20.7</b>	<b>Analyse des Laufzeitverhaltens</b> .....	<b>726</b>
20.7.1	Laufzeitmessung – timeit .....	727
20.7.2	Profiling – cProfile .....	730
20.7.3	Tracing – trace .....	734

## Teil IV Weiterführende Themen

---

<b>21</b>	<b>Distribution von Python-Projekten</b> .....	<b>739</b>
<b>21.1</b>	<b>Erstellen von Distributionen – distutils</b> .....	<b>739</b>
21.1.1	Schreiben des Moduls .....	740
21.1.2	Das Installationsscript .....	742
21.1.3	Erstellen einer Quellcodedistribution .....	746
21.1.4	Erstellen einer Binärdistribution .....	748
21.1.5	Distributionen installieren .....	749
<b>21.2</b>	<b>Erstellen von EXE-Dateien – cx_Freeze</b> .....	<b>750</b>

---

<b>22</b>	<b>Optimierung</b> .....	<b>753</b>
<b>22.1</b>	<b>Die Optimize-Option</b> .....	<b>754</b>
<b>22.2</b>	<b>Mutable vs. immutable</b> .....	<b>754</b>
<b>22.3</b>	<b>Schleifen</b> .....	<b>755</b>
<b>22.4</b>	<b>Funktionsaufrufe</b> .....	<b>756</b>

<b>22.5</b>	<b>C</b>	756
<b>22.6</b>	<b>Lookup</b>	757
<b>22.7</b>	<b>Exceptions</b>	757
<b>22.8</b>	<b>Keyword Arguments</b>	758

---

## **23 Grafische Benutzeroberflächen** 759

---

<b>23.1</b>	<b>Toolkits</b>	759
23.1.1	Tkinter	760
23.1.2	PyGObject	760
23.1.3	PyQt	760
23.1.4	PySide	761
23.1.5	wxPython	761
<b>23.2</b>	<b>Einführung in tkinter</b>	762
23.2.1	Ein einfaches Beispiel	762
23.2.2	Steuerelementvariablen	764
23.2.3	Der Packer	767
23.2.4	Events	771
23.2.5	Die Steuerelemente	779
23.2.6	Die Klasse Tk	817
23.2.7	Weitere Module	818
<b>23.3</b>	<b>Einführung in PyQt</b>	826
23.3.1	Installation	826
23.3.2	Grundlegende Konzepte von Qt	827
23.3.3	Entwicklungsprozess	829
<b>23.4</b>	<b>Signale und Slots</b>	836
<b>23.5</b>	<b>Wichtige Widgets</b>	839
23.5.1	QCheckBox	840
23.5.2	QComboBox	841
23.5.3	QDateEdit, QDateTimeEdit, QTimeEdit	842
23.5.4	QDialog	843
23.5.5	QLineEdit	844
23.5.6	QListWidget, QListView	844
23.5.7	QProgressBar	845
23.5.8	QPushButton	846
23.5.9	QRadioButton	846
23.5.10	QSlider, QDial	847

23.5.11	QTextEdit .....	848
23.5.12	QWidget .....	848
<b>23.6</b>	<b>Zeichenfunktionalität .....</b>	<b>850</b>
23.6.1	Werkzeuge .....	851
23.6.2	Koordinatensystem .....	853
23.6.3	Einfache Formen .....	853
23.6.4	Grafiken .....	856
23.6.5	Text .....	857
23.6.6	Eye-Candy .....	859
<b>23.7</b>	<b>Model-View-Architektur .....</b>	<b>863</b>
23.7.1	Beispielprojekt: Ein Adressbuch .....	864
23.7.2	Auswählen von Einträgen .....	874
23.7.3	Bearbeiten von Einträgen .....	875
<b>24</b>	<b>Wissenschaftliches Rechnen</b>	<b>881</b>
<b>24.1</b>	<b>Installation .....</b>	<b>882</b>
<b>24.2</b>	<b>Das Modellprogramm .....</b>	<b>883</b>
24.2.1	Der Import von numpy, scipy und matplotlib .....	884
24.2.2	Vektorisierung und der Datentyp numpy.ndarray .....	885
24.2.3	Visualisieren von Daten mit matplotlib.pyplot .....	889
<b>24.3</b>	<b>Überblicke über die Module numpy und scipy .....</b>	<b>891</b>
24.3.1	Überblick über den Datentyp numpy.ndarray .....	891
24.3.2	Überblick über scipy .....	899
<b>25</b>	<b>Anbindung an andere Programmiersprachen</b>	<b>903</b>
<b>25.1</b>	<b>Dynamisch ladbare Bibliotheken – ctypes .....</b>	<b>904</b>
25.1.1	Ein einfaches Beispiel .....	904
25.1.2	Die eigene Bibliothek .....	905
25.1.3	Schnittstellenbeschreibung .....	909
25.1.4	Verwendung des Moduls .....	911
<b>25.2</b>	<b>Schreiben von Extensions .....</b>	<b>914</b>
25.2.1	Ein einfaches Beispiel .....	914
25.2.2	Exceptions .....	918
25.2.3	Erzeugen der Extension .....	919
25.2.4	Reference Counting .....	921

<b>25.3</b>	<b>Python als eingebettete Scriptssprache</b> .....	922
25.3.1	Ein einfaches Beispiel .....	922
25.3.2	Ein komplexeres Beispiel .....	924
25.3.3	Python-API-Referenz .....	927
<b>26</b>	<b>Insiderwissen</b>	933
<b>26.1</b>	<b>URLs im Standardbrowser öffnen – <code>webbrowser</code></b> .....	933
<b>26.2</b>	<b>Funktionsschnittstellen vereinfachen – <code>functools</code></b> .....	934
26.2.1	Funktionsschnittstellen vereinfachen .....	934
26.2.2	Caches .....	935
26.2.3	Ordnungsrelationen vervollständigen .....	937
<b>26.3</b>	<b>Weitere Datentypen – <code>collections</code></b> .....	938
26.3.1	Zählen von Häufigkeiten .....	938
26.3.2	Dictionarys mit Standardwerten .....	941
26.3.3	Doppelt verkettete Listen .....	942
26.3.4	Benannte Tupel .....	944
26.3.5	Sortierte Dictionarys .....	945
<b>26.4</b>	<b>Interpretieren von Binärdaten – <code>struct</code></b> .....	946
<b>26.5</b>	<b>Versteckte Passworteingaben – <code>getpass</code></b> .....	949
<b>26.6</b>	<b>Kommandozeilen-Interpreter – <code>cmd</code></b> .....	949
<b>26.7</b>	<b>Dateiinterface für Strings – <code>io.StringIO</code></b> .....	952
<b>27</b>	<b>Von Python 2 nach Python 3</b>	955
<b>27.1</b>	<b>Die wichtigsten Unterschiede</b> .....	958
27.1.1	Ein-/Ausgabe .....	958
27.1.2	Iteratoren .....	959
27.1.3	Strings .....	960
27.1.4	Ganze Zahlen .....	961
27.1.5	Exception Handling .....	962
27.1.6	Standardbibliothek .....	962
27.1.7	Neue Sprachelemente in Python 3 .....	963
<b>27.2</b>	<b>Automatische Konvertierung</b> .....	964

---

<b>Anhang</b>	969
A.1 Reservierte Wörter .....	969
A.2 Built-in Functions .....	969
<b>Index .....</b>	<b>973</b>