

# Inhalt

|                                                                          |               |
|--------------------------------------------------------------------------|---------------|
| <b>Vorwort .....</b>                                                     | <b>XVII</b>   |
| <b>Danksagung .....</b>                                                  | <b>XVIII</b>  |
| <b>1    Einleitung .....</b>                                             | <b>1</b>      |
| 1.1    Einfach und schnell zu lernen .....                               | 1             |
| 1.2    Zielgruppe des Buches .....                                       | 1             |
| 1.3    Aufbau des Buches .....                                           | 2             |
| 1.4    Programmieren lernen „interaktiv“ .....                           | 3             |
| 1.5    Download der Beispiele und Hilfe .....                            | 4             |
| 1.6    Anregungen und Kritik .....                                       | 4             |
| <br><b>Teil I   Grundlagen .....</b>                                     | <br><b>5</b>  |
| <b>2    Kommandos und Programme .....</b>                                | <b>7</b>      |
| 2.1    Erste Schritte mit Python .....                                   | 7             |
| 2.1.1    Linux .....                                                     | 7             |
| 2.1.2    Windows .....                                                   | 8             |
| 2.1.3    macOS .....                                                     | 9             |
| 2.2    Herkunft und Bedeutung des Begriffes interaktive Shell .....      | 9             |
| 2.2.1    Erste Schritte in der interaktiven Shell .....                  | 10            |
| 2.3    Verlassen der Python-Shell .....                                  | 11            |
| 2.4    Benutzung von Variablen .....                                     | 12            |
| 2.5    Mehrzeilige Anweisungen in der interaktiven Shell .....           | 12            |
| 2.6    Programme schreiben oder schnell mal der Welt “Hallo” sagen ..... | 13            |
| <br><b>3    Bytecode und Maschinencode .....</b>                         | <br><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      | Variablenamen .....                            | 23        |
| 4.2.1    | Gültige Variablenamen .....                    | 23        |
| 4.2.2    | Konventionen für Variablenamen.....            | 24        |
| 4.3      | Datentypen .....                               | 24        |
| 4.3.1    | Ganze Zahlen .....                             | 24        |
| 4.3.2    | Fließkommazahlen.....                          | 26        |
| 4.3.3    | Zeichenketten.....                             | 26        |
| 4.3.4    | Boolesche Werte .....                          | 26        |
| 4.3.5    | Komplexe Zahlen .....                          | 27        |
| 4.3.6    | Operatoren .....                               | 27        |
| 4.4      | Statische und dynamische Typdeklaration.....   | 28        |
| 4.5      | Typumwandlung .....                            | 30        |
| 4.6      | Datentyp ermitteln .....                       | 31        |
| <b>5</b> | <b>Sequentielle Datentypen .....</b>           | <b>33</b> |
| 5.1      | Übersicht.....                                 | 33        |
| 5.1.1    | Zeichenketten oder Strings .....               | 34        |
| 5.1.2    | Listen.....                                    | 36        |
| 5.1.3    | Tupel .....                                    | 36        |
| 5.2      | Indizierung von sequentiellen Datentypen ..... | 37        |
| 5.3      | Teilbereichsoperator.....                      | 38        |
| 5.4      | Die len-Funktion .....                         | 40        |
| 5.5      | Aufgaben .....                                 | 41        |
| <b>6</b> | <b>Dictionaries .....</b>                      | <b>43</b> |
| 6.1      | Dictionaries und assoziative Felder .....      | 43        |
| 6.2      | Definition und Benutzung.....                  | 44        |
| 6.3      | Fehlerfreie Zugriffe auf Dictionaries .....    | 46        |
| 6.4      | Zulässige Typen für Schlüssel und Werte .....  | 47        |
| 6.5      | Verschachtelte Dictionaries .....              | 48        |
| 6.6      | Methoden auf Dictionaries .....                | 48        |
| 6.7      | Operatoren.....                                | 52        |
| 6.8      | Die zip-Funktion .....                         | 52        |
| 6.9      | Dictionaries aus Listen erzeugen .....         | 54        |
| 6.10     | Aufgaben .....                                 | 55        |

|           |                                                |           |
|-----------|------------------------------------------------|-----------|
| <b>7</b>  | <b>Mengen</b>                                  | <b>57</b> |
| 7.1       | Übersicht                                      | 57        |
| 7.2       | Mengen in Python                               | 57        |
| 7.2.1     | Sets erzeugen                                  | 58        |
| 7.2.2     | Mengen von unveränderlichen Elementen          | 58        |
| 7.3       | Frozensets                                     | 59        |
| 7.4       | Operationen auf „set“-Objekten                 | 59        |
| 7.4.1     | add(element)                                   | 59        |
| 7.4.2     | clear()                                        | 60        |
| 7.4.3     | copy                                           | 60        |
| 7.4.4     | difference()                                   | 60        |
| 7.4.5     | difference_update()                            | 61        |
| 7.4.6     | discard(el)                                    | 61        |
| 7.4.7     | remove(el)                                     | 62        |
| 7.4.8     | intersection(s)                                | 62        |
| 7.4.9     | isdisjoint()                                   | 62        |
| 7.4.10    | issubset()                                     | 63        |
| 7.4.11    | issuperset()                                   | 63        |
| 7.4.12    | pop()                                          | 64        |
| <b>8</b>  | <b>Eingaben</b>                                | <b>65</b> |
| 8.1       | Eingabe mittels input                          | 65        |
| <b>9</b>  | <b>Verzweigungen</b>                           | <b>67</b> |
| 9.1       | Anweisungsblöcke und Einrückungen              | 67        |
| 9.2       | Bedingte Anweisungen in Python                 | 70        |
| 9.2.1     | Einfachste if-Anweisung                        | 70        |
| 9.2.2     | if-Anweisung mit else-Zweig                    | 71        |
| 9.2.3     | elif-Zweige                                    | 71        |
| 9.3       | Vergleichsoperatoren                           | 72        |
| 9.4       | Zusammengesetzte Bedingungen                   | 72        |
| 9.5       | Wahr oder falsch: Bedingungen in Verzweigungen | 72        |
| 9.6       | Aufgaben                                       | 73        |
| <b>10</b> | <b>Schleifen</b>                               | <b>75</b> |
| 10.1      | Übersicht                                      | 75        |
| 10.2      | while-Schleife                                 | 76        |
| 10.3      | break und continue                             | 77        |
| 10.4      | die Alternative im Erfolgsfall: else           | 78        |
| 10.5      | For-Schleife                                   | 80        |
| 10.6      | Aufgaben                                       | 83        |

|           |                                                                   |            |
|-----------|-------------------------------------------------------------------|------------|
| <b>11</b> | <b>Dateien lesen und schreiben .....</b>                          | <b>87</b>  |
| 11.1      | Dateien .....                                                     | 87         |
| 11.2      | Text aus einer Datei lesen .....                                  | 87         |
| 11.3      | Schreiben in eine Datei .....                                     | 89         |
| 11.4      | In einem Rutsch lesen: readlines und read .....                   | 89         |
| 11.5      | with-Anweisung .....                                              | 90         |
| 11.6      | Aufgaben .....                                                    | 91         |
| <b>12</b> | <b>Formatierte Ausgabe und Strings formatieren .....</b>          | <b>93</b>  |
| 12.1      | Wege, die Ausgabe zu formatieren .....                            | 93         |
| 12.2      | print-Funktion .....                                              | 93         |
| 12.3      | Stringformatierung im C-Stil .....                                | 95         |
| 12.4      | Der pythonische Weg: Die String-Methode „format“ .....            | 99         |
| 12.5      | Benutzung von Dictionaries beim Aufruf der „format“-Methode ..... | 102        |
| 12.6      | Benutzung von lokalen Variablen in „format“ .....                 | 103        |
| 12.7      | Formatierte Stringlitterale .....                                 | 104        |
| 12.8      | Weitere String-Methoden zum Formatieren .....                     | 104        |
| <b>13</b> | <b>Flaches und tiefes Kopieren .....</b>                          | <b>107</b> |
| 13.1      | Einführung .....                                                  | 107        |
| 13.2      | Kopieren einer Liste .....                                        | 108        |
| 13.3      | Flache Kopien .....                                               | 110        |
| 13.4      | Kopieren mit deepcopy .....                                       | 111        |
| 13.5      | Deepcopy für Dictionaries .....                                   | 112        |
| <b>14</b> | <b>Funktionen .....</b>                                           | <b>113</b> |
| 14.1      | Allgemein .....                                                   | 113        |
| 14.2      | Funktionen .....                                                  | 113        |
| 14.3      | Docstring .....                                                   | 115        |
| 14.4      | Standardwerte für Funktionen .....                                | 117        |
| 14.5      | Schlüsselwortparameter .....                                      | 118        |
| 14.6      | Funktionen ohne oder mit leerer return-Anweisung .....            | 118        |
| 14.7      | Mehrere Rückgabewerte .....                                       | 119        |
| 14.8      | Lokale und globale Variablen in Funktionen .....                  | 120        |
| 14.9      | Parameterübergabe im Detail .....                                 | 122        |
| 14.10     | Effekte bei veränderlichen Objekten .....                         | 124        |
| 14.11     | Kommandozeilenparameter .....                                     | 125        |
| 14.12     | Variable Anzahl von Parametern / Variadische Funktionen .....     | 126        |
| 14.13     | * in Funktionsaufrufen .....                                      | 128        |

|                                                          |            |
|----------------------------------------------------------|------------|
| 14.14 Beliebige Schlüsselwortparameter .....             | 129        |
| 14.15 Doppeltes Sternchen im Funktionsaufruf .....       | 129        |
| 14.16 Aufgaben .....                                     | 130        |
| <b>15 Rekursive Funktionen .....</b>                     | <b>133</b> |
| 15.1 Definition und Herkunft des Begriffs .....          | 133        |
| 15.2 Definition der Rekursion .....                      | 134        |
| 15.3 Rekursive Funktionen in Python .....                | 134        |
| 15.4 Die Tücken der Rekursion .....                      | 135        |
| 15.5 Fibonacci-Folge in Python .....                     | 136        |
| 15.6 Aufgaben .....                                      | 140        |
| <b>16 Listen und Tupel im Detail .....</b>               | <b>143</b> |
| 16.1 Stapelspeicher .....                                | 143        |
| 16.2 Stapelverarbeitung in Python: pop und append .....  | 144        |
| 16.3 extend .....                                        | 144        |
| 16.4 „+“-Operator oder append .....                      | 145        |
| 16.5 Entfernen eines Wertes .....                        | 147        |
| 16.6 Prüfen, ob ein Element in Liste enthalten ist ..... | 147        |
| 16.7 Finden der Position eines Elementes .....           | 147        |
| 16.8 Einfügen von Elementen .....                        | 148        |
| 16.9 Besonderheiten bei Tupel .....                      | 148        |
| 16.9.1 Leere Tupel .....                                 | 149        |
| 16.9.2 1-Tupel .....                                     | 149        |
| 16.9.3 Mehrfachzuweisungen, Packing und Unpacking .....  | 149        |
| 16.10 Die veränderliche Unveränderliche .....            | 151        |
| 16.11 Sortieren von Listen .....                         | 151        |
| 16.11.1 „sort“ und „sorted“ .....                        | 151        |
| 16.11.2 Umkehrung der Sortierreihenfolge .....           | 152        |
| 16.11.3 Eigene Sortierfunktionen .....                   | 152        |
| 16.12 Aufgaben .....                                     | 155        |
| <b>17 Modularisierung .....</b>                          | <b>157</b> |
| 17.1 Module .....                                        | 157        |
| 17.1.1 Namensräume von Modulen .....                     | 158        |
| 17.1.2 Namensräume umbenennen .....                      | 159        |
| 17.1.3 Modularten .....                                  | 159        |
| 17.1.4 Suchpfad für Module .....                         | 160        |
| 17.1.5 Inhalt eines Modules .....                        | 161        |



|           |                                                    |            |
|-----------|----------------------------------------------------|------------|
| 17.1.6    | Eigene Module .....                                | 161        |
| 17.1.7    | Dokumentation für eigene Module .....              | 162        |
| 17.2      | Pakete.....                                        | 163        |
| 17.2.1    | Einfaches Paket erzeugen .....                     | 164        |
| 17.2.2    | Komplexeres Paket .....                            | 165        |
| 17.2.3    | Komplettes Paket importieren.....                  | 168        |
| <b>18</b> | <b>Globale und lokale Variablen .....</b>          | <b>171</b> |
| 18.1      | Einführung.....                                    | 171        |
| 18.2      | Globale und lokale Variablen in Funktionen.....    | 172        |
| <b>19</b> | <b>Alles über Strings ... ..</b>                   | <b>175</b> |
| 19.1      | ... fast alles .....                               | 175        |
| 19.2      | Aufspalten von Zeichenketten .....                 | 176        |
| 19.2.1    | split.....                                         | 176        |
| 19.2.2    | Standardverhalten und „maxsplit“ .....             | 178        |
| 19.2.3    | rsplit.....                                        | 179        |
| 19.2.4    | Folge von Trennzeichen .....                       | 181        |
| 19.2.5    | splitlines .....                                   | 182        |
| 19.2.6    | partition .....                                    | 183        |
| 19.3      | Zusammenfügen von Stringlisten mit join .....      | 183        |
| 19.4      | Suchen von Teilstrings .....                       | 183        |
| 19.4.1    | „in“ oder „not in“ .....                           | 183        |
| 19.4.2    | s.find(substring[, start[, end]]) .....            | 184        |
| 19.4.3    | s.rfind(substring[, start[, end]]) .....           | 184        |
| 19.4.4    | s.index(substring[, start[, end]]) .....           | 185        |
| 19.4.5    | s.rindex(substring[, start[, end]]) .....          | 185        |
| 19.4.6    | s.count(substring[, start[, end]]) .....           | 185        |
| 19.5      | Suchen und Ersetzen .....                          | 186        |
| 19.6      | Nur noch Kleinbuchstaben oder Großbuchstaben ..... | 186        |
| 19.7      | capitalize und title .....                         | 186        |
| 19.8      | Stripping Strings .....                            | 187        |
| 19.9      | Strings ausrichten.....                            | 187        |
| 19.10     | String-Tests .....                                 | 188        |
| 19.11     | Aufgaben .....                                     | 190        |

|                |                                                       |            |
|----------------|-------------------------------------------------------|------------|
| <b>20</b>      | <b>Ausnahmebehandlung</b>                             | <b>193</b> |
| 20.1           | Abfangen mehrerer Exceptions                          | 195        |
| 20.2           | except mit mehrfachen Ausnahmen                       | 196        |
| 20.3           | Die optionale else-Klausel                            | 196        |
| 20.4           | Fehlerinformationen über sys.exc_info                 | 197        |
| 20.5           | Exceptions generieren                                 | 197        |
| 20.6           | Finalisierungsaktion                                  | 198        |
| <br>           |                                                       |            |
| <b>Teil II</b> | <b>Objektorientierte Programmierung</b>               | <b>199</b> |
| <br>           |                                                       |            |
| <b>21</b>      | <b>Grundlegende Aspekte</b>                           | <b>201</b> |
| 21.1           | Bibliotheksvergleich                                  | 201        |
| 21.2           | Objekte und Instanzen einer Klasse                    | 203        |
| 21.3           | Kapselung von Daten und Methoden                      | 204        |
| 21.4           | Eine minimale Klasse in Python                        | 204        |
| 21.5           | Eigenschaften und Attribute                           | 205        |
| 21.6           | Methoden                                              | 208        |
| 21.7           | Instanzvariablen                                      | 209        |
| 21.8           | Die __init__-Methode                                  | 209        |
| 21.9           | Destruktor                                            | 211        |
| 21.10          | Datenkapselung, Datenabstraktion und Geheimnisprinzip | 212        |
| 21.10.1        | Definitionen                                          | 212        |
| 21.10.2        | Zugriffsmethoden                                      | 214        |
| 21.10.3        | Properties                                            | 215        |
| 21.10.4        | Public-, Protected- und Private-Attribute             | 217        |
| 21.10.5        | Weitere Möglichkeiten der Properties                  | 218        |
| 21.10.6        | Properties mit Dekoratoren                            | 221        |
| 21.11          | Die __str__- und die __repr__-Methode                 | 222        |
| 21.12          | Klassenattribute                                      | 227        |
| 21.13          | Statische Methoden                                    | 229        |
| 21.13.1        | Einleitendes Beispiel                                 | 230        |
| 21.13.2        | Getter und Setter für private Klassenattribute        | 231        |
| 21.14          | Public-Attribute statt private Attribute              | 232        |
| 21.15          | Magische Methoden und Operatorüberladung              | 233        |
| 21.15.1        | Einführung                                            | 233        |
| 21.15.2        | Übersicht magische Methoden                           | 235        |
| 21.15.3        | Beispielklasse: Length                                | 236        |

|           |                                                 |            |
|-----------|-------------------------------------------------|------------|
| <b>22</b> | <b>Bruchklasse</b>                              | <b>241</b> |
| 22.1      | Brüche à la 1001 Nacht                          | 241        |
| 22.2      | Zurück in die Gegenwart                         | 242        |
| 22.3      | Rechenregeln                                    | 244        |
| 22.3.1    | Multiplikation von Brüchen                      | 244        |
| 22.3.2    | Division von Brüchen                            | 245        |
| 22.3.3    | Addition von Brüchen                            | 246        |
| 22.3.4    | Subtraktion von Brüchen                         | 246        |
| 22.3.5    | Vergleichsoperatoren                            | 247        |
| 22.4      | Integer plus Bruch                              | 247        |
| 22.4.1    | Die Bruchklasse im Überblick                    | 248        |
| 22.5      | Fraction-Klasse                                 | 250        |
| <b>23</b> | <b>Vererbung</b>                                | <b>251</b> |
| 23.1      | Oberbegriffe und Oberklassen                    | 251        |
| 23.2      | Ein einfaches Beispiel                          | 252        |
| 23.3      | Überladen, Überschreiben und Polymorphie        | 253        |
| 23.4      | Vererbung in Python                             | 256        |
| 23.5      | Klassenmethoden                                 | 259        |
| 23.6      | Standardklassen als Basisklassen                | 261        |
| <b>24</b> | <b>Mehrfachvererbung</b>                        | <b>263</b> |
| 24.1      | Einführung                                      | 263        |
| 24.2      | Beispiel: CalendarClock                         | 264        |
| 24.3      | Diamond-Problem oder „deadly diamond of death“  | 272        |
| 24.4      | super und MRO                                   | 273        |
| <b>25</b> | <b>Slots</b>                                    | <b>279</b> |
| 25.1      | Erzeugung von dynamischen Attributen verhindern | 279        |
| <b>26</b> | <b>Dynamische Erzeugung von Klassen</b>         | <b>281</b> |
| 26.1      | Beziehung zwischen „class“ und „type“           | 281        |
| <b>27</b> | <b>Metaklassen</b>                              | <b>285</b> |
| 27.1      | Motivation                                      | 285        |
| 27.2      | Definition                                      | 290        |
| 27.3      | Definition von Metaklassen in Python            | 290        |
| 27.4      | Singletons mit Metaklassen erstellen            | 292        |

|                 |                                                             |            |
|-----------------|-------------------------------------------------------------|------------|
| 27.5            | Beispiel – Methodenaufrufe zählen .....                     | 293        |
| 27.5.1          | Einführung .....                                            | 293        |
| 27.5.2          | Vorbereitungen .....                                        | 293        |
| 27.5.3          | Ein Dekorateur, um Funktionsaufrufe zu zählen .....         | 295        |
| 27.5.4          | Die Metaklasse „Aufrufzähler“ .....                         | 295        |
| <b>28</b>       | <b>Abstrakte Klassen .....</b>                              | <b>297</b> |
| <b>29</b>       | <b>Aufgaben .....</b>                                       | <b>301</b> |
| <b>Teil III</b> | <b>Fortgeschrittenes Python .....</b>                       | <b>305</b> |
| <b>30</b>       | <b>lambda, map, filter und reduce .....</b>                 | <b>307</b> |
| 30.1            | lambda .....                                                | 307        |
| 30.2            | map .....                                                   | 310        |
| 30.3            | Filtern von sequentiellen Datentypen mittels „filter“ ..... | 312        |
| 30.4            | reduce .....                                                | 313        |
| 30.5            | Aufgaben .....                                              | 314        |
| <b>31</b>       | <b>Listen-Abstraktion/List Comprehension .....</b>          | <b>315</b> |
| 31.1            | Die Alternative zu Lambda und Co. ....                      | 315        |
| 31.2            | Syntax .....                                                | 316        |
| 31.3            | Weitere Beispiele .....                                     | 316        |
| 31.4            | Die zugrunde liegende Idee .....                            | 317        |
| 31.5            | Anspruchsvolleres Beispiel .....                            | 318        |
| 31.6            | Mengen-Abstraktion .....                                    | 318        |
| 31.7            | Rekursive Primzahlberechnung .....                          | 319        |
| 31.8            | Generatoren-Abstraktion .....                               | 319        |
| 31.9            | Aufgaben .....                                              | 320        |
| <b>32</b>       | <b>Generatoren und Iteratoren .....</b>                     | <b>323</b> |
| 32.1            | Einführung .....                                            | 323        |
| 32.2            | Iteration in for-Schleifen .....                            | 323        |
| 32.3            | Generatoren .....                                           | 325        |
| 32.4            | Generatoren zähmen mit firstn und islice .....              | 327        |
| 32.5            | Beispiele .....                                             | 328        |
| 32.5.1          | Permutationen .....                                         | 328        |
| 32.5.2          | Variationen und Kombinationen .....                         | 329        |
| 32.6            | Generator-Ausdrücke .....                                   | 331        |

|                |                                                                |            |
|----------------|----------------------------------------------------------------|------------|
| 32.7           | return-Anweisungen in Generatoren .....                        | 332        |
| 32.8           | send-Methode / Koroutinen .....                                | 332        |
| 32.9           | Die throw-Methode .....                                        | 334        |
| 32.10          | Dekoration von Generatoren .....                               | 336        |
| 32.11          | yield from .....                                               | 337        |
| 32.12          | Aufgaben .....                                                 | 339        |
| <b>33</b>      | <b>Dekorateur</b> .....                                        | <b>341</b> |
| 33.1           | Einführung Dekorateur .....                                    | 341        |
| 33.1.1         | Verschachtelte Funktionen .....                                | 342        |
| 33.1.2         | Funktionen als Parameter .....                                 | 343        |
| 33.1.3         | Funktionen als Rückgabewert .....                              | 344        |
| 33.1.4         | Fabrikfunktionen .....                                         | 345        |
| 33.2           | Ein einfacher Dekorateur .....                                 | 346        |
| 33.3           | @-Syntax für Dekorateur .....                                  | 347        |
| 33.4           | Anwendungsfälle für Dekorateur .....                           | 350        |
| 33.4.1         | Überprüfung von Argumenten durch Dekorateur .....              | 350        |
| 33.4.2         | Funktionsaufrufe mit einem Dekorateur zählen .....             | 351        |
| 33.5           | Dekorateur mit Parametern .....                                | 352        |
| 33.6           | Benutzung von Wraps aus functools .....                        | 354        |
| 33.7           | Klassen statt Funktionen .....                                 | 355        |
| 33.7.1         | Die __call__-Methode .....                                     | 355        |
| 33.8           | Eine Klasse als Dekorateur benutzen .....                      | 357        |
| 33.9           | Memoisation .....                                              | 358        |
| 33.9.1         | Bedeutung und Herkunft des Begriffs .....                      | 358        |
| 33.9.2         | Memoisation mit Dekorateur-Funktionen .....                    | 358        |
| 33.9.3         | Memoisation mit einer Klasse .....                             | 359        |
| 33.9.4         | Memoisation mit functools.lru_cache .....                      | 360        |
| <b>Teil IV</b> | <b>Weiterführende Themen</b> .....                             | <b>363</b> |
| <b>34</b>      | <b>Tests und Fehler</b> .....                                  | <b>365</b> |
| 34.1           | Einführung .....                                               | 365        |
| 34.2           | Modultests .....                                               | 367        |
| 34.3           | Modultests unter Benutzung von __name__ .....                  | 368        |
| 34.4           | doctest-Modul .....                                            | 370        |
| 34.5           | Testgetriebene Entwicklung oder „Im Anfang war der Test“ ..... | 373        |
| 34.6           | unittest .....                                                 | 375        |
| 34.7           | Methoden der Klasse TestCase .....                             | 377        |
| 34.8           | Aufgaben .....                                                 | 380        |

|           |                                                                 |            |
|-----------|-----------------------------------------------------------------|------------|
| <b>35</b> | <b>Daten konservieren .....</b>                                 | <b>381</b> |
| 35.1      | Persistente Speicherung .....                                   | 381        |
| 35.2      | Pickle-Modul .....                                              | 382        |
| 35.2.1    | Daten „einpökeln“ mit pickle.dump .....                         | 382        |
| 35.2.2    | pickle.load .....                                               | 383        |
| 35.3      | Ein persistentes Dictionary mit shelve.....                     | 383        |
| <b>36</b> | <b>Reguläre Ausdrücke .....</b>                                 | <b>387</b> |
| 36.1      | Ursprünge und Verbreitung .....                                 | 387        |
| 36.2      | Stringvergleiche .....                                          | 387        |
| 36.3      | Überlappungen und Teilstrings.....                              | 389        |
| 36.4      | Das re-Modul .....                                              | 389        |
| 36.5      | Matching-Problem.....                                           | 390        |
| 36.6      | Syntax der regulären Ausdrücke .....                            | 392        |
| 36.6.1    | Beliebiges Zeichen .....                                        | 392        |
| 36.7      | Zeichenauswahl.....                                             | 393        |
| 36.8      | Endliche Automaten.....                                         | 394        |
| 36.9      | Anfang und Ende eines Strings.....                              | 394        |
| 36.10     | Vordefinierte Zeichenklassen .....                              | 397        |
| 36.11     | Optionale Teile .....                                           | 398        |
| 36.12     | Quantoren .....                                                 | 399        |
| 36.13     | Gruppierungen und Rückwärtsreferenzen .....                     | 401        |
| 36.13.1   | Match-Objekte.....                                              | 401        |
| 36.14     | Umfangreiche Übung .....                                        | 404        |
| 36.15     | Alles finden mit findall .....                                  | 406        |
| 36.16     | Alternativen.....                                               | 407        |
| 36.17     | Compilierung von regulären Ausdrücken .....                     | 407        |
| 36.18     | Aufspalten eines Strings mit oder ohne regulären Ausdruck ..... | 409        |
| 36.18.1   | split-Methode der String-Klasse.....                            | 409        |
| 36.18.2   | split-Methode des re-Moduls.....                                | 410        |
| 36.18.3   | Wörter filtern .....                                            | 412        |
| 36.19     | Suchen und Ersetzen mit sub .....                               | 413        |
| 36.20     | Aufgaben .....                                                  | 414        |
| <b>37</b> | <b>Typanmerkungen .....</b>                                     | <b>417</b> |
| 37.1      | Einführung.....                                                 | 417        |
| 37.2      | Einfaches Beispiel.....                                         | 418        |
| 37.3      | Variablenanmerkungen .....                                      | 419        |
| 37.4      | Listenbeispiel.....                                             | 420        |
| 37.5      | typing-Modul.....                                               | 421        |

|               |                                                                      |            |
|---------------|----------------------------------------------------------------------|------------|
| <b>38</b>     | <b>Systemprogrammierung .....</b>                                    | <b>425</b> |
| 38.1          | Systemprogrammierung .....                                           | 425        |
| 38.2          | Häufig falsch verstanden: Shell .....                                | 425        |
| 38.3          | os-Modul .....                                                       | 426        |
| 38.3.1        | Vorbemerkungen .....                                                 | 427        |
| 38.3.2        | Umgebungsvariablen .....                                             | 427        |
| 38.3.3        | Dateiverarbeitung auf niedrigerer Ebene .....                        | 429        |
| 38.3.4        | Die exec-„Familie“ .....                                             | 434        |
| 38.3.5        | Weitere Funktionen im Überblick .....                                | 441        |
| 38.3.6        | os.path - Arbeiten mit Pfaden .....                                  | 455        |
| 38.4          | shutil-Modul .....                                                   | 463        |
| <b>39</b>     | <b>Forks .....</b>                                                   | <b>469</b> |
| 39.1          | Fork .....                                                           | 469        |
| 39.2          | Fork in Python .....                                                 | 469        |
| <b>Teil V</b> | <b>Lösungen zu den Aufgaben .....</b>                                | <b>473</b> |
| <b>40</b>     | <b>Lösungen zu den Aufgaben .....</b>                                | <b>475</b> |
| 40.1          | Lösungen zu Kapitel 5 (Sequentielle Datentypen) .....                | 475        |
| 40.2          | Lösungen zu Kapitel 6 (Dictionaries) .....                           | 478        |
| 40.3          | Lösungen zu Kapitel 9 (Verzweigungen) .....                          | 480        |
| 40.4          | Lösungen zu Kapitel 10 (Schleifen) .....                             | 482        |
| 40.5          | Lösungen zu Kapitel 11 (Dateien lesen und schreiben) .....           | 485        |
| 40.6          | Lösungen zu Kapitel 16 (Listen und Tupel im Detail) .....            | 487        |
| 40.7          | Lösungen zu Kapitel 14 (Funktionen) .....                            | 490        |
| 40.8          | Lösungen zu Kapitel 15 (Rekursive Funktionen) .....                  | 495        |
| 40.9          | Lösungen zu Kapitel 19 (Alles über Strings ...) .....                | 499        |
| 40.10         | Lösungen zu Kapitel 21 (Grundlegende Aspekte) .....                  | 502        |
| 40.11         | Lösungen zu Kapitel 34 (Tests und Fehler) .....                      | 512        |
| 40.12         | Lösungen zu Kapitel 36 (Reguläre Ausdrücke) .....                    | 512        |
| 40.13         | Lösungen zu Kapitel 30 (lambda, map, filter und reduce) .....        | 518        |
| 40.14         | Lösungen zu Kapitel 31 (Listen-Abstraktion/List Comprehension) ..... | 519        |
| 40.15         | Lösungen zu Kapitel 32 (Generatoren und Iteratoren) .....            | 520        |
|               | <b>Stichwortverzeichnis .....</b>                                    | <b>525</b> |