

Inhaltsverzeichnis

I Über das Programmieren

1	Was ist Programmieren?	3
1.1	Die geheimnisvolle Macht – oder: Das Bewusstsein bestimmt das Sein	4
1.2	Algorithmen	6
1.3	Grenzen von klassischen Algorithmen: Das Spielfeld der künstlichen Intelligenz ..	8
1.3.1	Nur scheinbar intelligent	8
1.3.2	Katze oder nicht Katze – das ist hier die Frage	11
2	Warum programmieren lernen?	15
2.1	Viele gute Gründe	16
2.2	Klischees und Vorurteile	20
3	Was ist eine Programmiersprache?	25
3.1	Sprachen für Menschen, Sprachen für Maschinen	26
3.2	Übersetzung und Ausführung von Programmiersprachen	28
3.3	Von der Maschinensprache bis zur Hochsprache	30
4	Warum gibt es so viele Programmiersprachen?	33
5	Welche Programmiersprachen sollte man lernen?	39
6	Einige Tipps	45

II Die Grundkonzepte des Programmierens

7	Neun Fragen	51
8	Was brauche ich zum Programmieren?	55
8.1	Werkzeuge	56
8.1.1	Compiler und Interpreter	56
8.1.2	Code-Editoren	57
8.1.3	Integrierte Entwicklungsumgebungen (IDEs)	58
8.1.4	Einfache Online-Entwicklungsumgebungen	64
8.2	Hilfe und Informationen	65
8.3	Ihr Fahrplan zum Erlernen einer neuen Programmiersprache	67
9	Was muss ich tun, um ein Programm zum Laufen zu bringen?	69
9.1	Aller Anfang ist leicht	70
9.2	Hallo, Welt!	72
9.3	Ihr Fahrplan zum Erlernen einer neuen Programmiersprache	74

10	Wie stelle ich sicher, dass ich (und andere) mein Programm später noch verstehe?	75
10.1	Verständlicher Programm-Code	76
10.2	Gestaltung des Programmcodes und Benennung von Programmelementen	77
10.3	Kommentare	80
10.3.1	Den eigenen Programmcod erläutern	80
10.3.2	Wozu Kommentare sonst noch nützlich sind	82
10.3.3	Dokumentation außerhalb des Programmcodes	83
10.4	Ihr Fahrplan zum Erlernen einer neuen Programmiersprache	85
11	Wie speichere ich Daten, um mit ihnen zu arbeiten?	87
11.1	Variablen als Platzhalter für Daten	89
11.2	Datentypen von Variablen	90
11.2.1	Unterschiedliche Arten von Daten erfordern unterschiedliche Arten von Variablen	90
11.2.2	Wichtige Datentypen	91
11.2.3	Den Datentyp ändern: Konvertierung von Variablen	96
11.3	Variablen anlegen und initialisieren	98
11.4	Gar nicht so variabel: Konstanten	101
11.5	Geordnete Felder von Variablen/Arrays	101
11.6	Assoziative Felder von Variablen/Hashes	105
11.7	Objekte	106
11.7.1	Eine Welt aus lauter Objekten	107
11.7.2	Klassen	108
11.7.3	Vererbung	110
11.7.4	Methoden	113
11.7.5	Polymorphismus	116
11.7.6	Zugriffsrechte	118
11.8	Ihr Fahrplan zum Erlernen einer neuen Programmiersprache	120
11.9	Lösungen zu den Aufgaben	120
12	Wie lasse ich Daten ein- und ausgeben?	123
12.1	Formen der Datenein- und -ausgabe	124
12.2	Grafisch oder nicht grafisch – das ist hier die Frage	125
12.2.1	Grafische Benutzeroberflächen	127
12.2.2	Konsolenanwendungen	135
12.3	Arbeiten mit Dateien	139
12.4	Arbeiten mit Datenbanken	145
12.5	Ihr Fahrplan zum Erlernen einer neuen Programmiersprache	148
12.6	Lösungen zu den Aufgaben	149
13	Wie arbeite ich mit Programmfunktionen, um Daten zu bearbeiten und Aktionen auszulösen?	151
13.1	Funktionen	152
13.2	Bibliotheken	160
13.3	Frameworks	163

13.4	Application Programming Interfaces (APIs)	164
13.5	Ihr Fahrplan zum Erlernen einer neuen Programmiersprache.....	166
13.6	Lösungen zu den Aufgaben	166
14	Wie steuere ich den Programmablauf und lasse das Programm auf Benutzeraktionen und andere Ereignisse reagieren?	169
14.1	Warum eine Ablaufsteuerung notwendig ist.	171
14.2	Formen der Ablaufsteuerung	172
14.3	Wenn-Dann-Sonst-Konstrukte	173
14.4	Ein genauerer Blick auf Bedingungen	179
14.5	Komplexe Bedingungen mit logischen Operatoren (and, or, not)	181
14.6	Gleichartige Bedingungen mit Verzweige-Falls-Konstrukten effizient prüfen (switch/select...case).....	184
14.7	Ereignisse (Events)	187
14.8	Ihr Fahrplan zum Erlernen einer neuen Programmiersprache.....	191
14.9	Lösungen zu den Aufgaben	192
15	Wie wiederhole ich Programmanweisungen effizient?	195
15.1	Schleifen und ihre Erscheinungsformen	196
15.2	Abgezählte Schleifen.....	198
15.3	Bedingte Schleifen	204
15.4	Ihr Fahrplan zum Erlernen einer neuen Programmiersprache.....	208
15.5	Lösungen zu den Aufgaben	208
16	Wie suche und behebe ich Fehler auf strukturierte Art und Weise?	213
16.1	Fehler zur Entwicklungszeit	214
16.2	Fehler zur Laufzeit	215
16.3	Testen	217
16.4	Debugging-Methoden	218
16.5	Ihr Fahrplan zum Erlernen einer neuen Programmiersprache.....	220
III	Python	
17	Einführung	223
18	Was brauche ich zum Programmieren?	227
18.1	Den Python-Interpreter installieren	228
18.2	Die PyCharm-IDE installieren	230
18.3	Hilfe zu Python erhalten.....	231
18.4	Zusammenfassung.....	233
19	Wie bringe ich ein Programm zum Laufen?	235
19.1	Entwickeln und Ausführen von Programmen in Python	236
19.2	Die Python-Konsole: Python im interaktiven Modus	241

19.3	PyCharm kennenlernen	242
19.4	Zusammenfassung	242
20	Wie stelle ich sicher, dass ich (und andere) mein Programm später noch verstehe?	245
20.1	Gestaltung des Programmcodes und Namenskonventionen	246
20.1.1	Einrückungen und allgemeine Code-Formatierung	246
20.1.2	Anweisungsende ohne Semikolon, Anweisungen über mehrere Zeilen	248
20.1.3	Case-sensitivity und Wahl von Bezeichnern	250
20.2	Kommentare	251
20.3	Dokumentation mit Docstrings	253
20.4	Zusammenfassung	255
21	Wie speichere ich Daten, um mit ihnen zu arbeiten?	257
21.1	Variablen erzeugen und zuweisen	259
21.2	Variablen löschen	261
21.3	Grundtypen von Variablen	261
21.3.1	Zahlen (int, float)	262
21.3.2	Zeichenketten (str)	262
21.3.3	Wahrheitswerte (bool)	264
21.3.4	None	265
21.3.5	Weitere Datentypen	266
21.4	Variablen als Objekte	267
21.4.1	Attribute und Methoden von Variablen	267
21.4.2	Erzeugen von Variablen mit der Konstruktor-Methode	272
21.5	Konvertieren von Variablen	273
21.6	Komplexe Datentypen	275
21.6.1	Listen	275
21.6.2	Tupel	282
21.6.3	Dictionaries	284
21.6.4	Sets	287
21.7	Selbstdefinierte Klassen	289
21.7.1	Klassen definieren und verwenden	289
21.7.2	Klassen aus anderen Klassen ableiten	290
21.7.3	Doppeldeutigkeit vermeiden: Name mangling	292
21.8	Zusammenfassung	293
21.9	Lösungen zu den Aufgaben	296
22	Wie lasse ich Daten ein- und ausgeben?	301
22.1	Ein- und Ausgabe in der Konsole	302
22.2	Grafische Benutzeroberflächen mit tkinter	304
22.2.1	Überblick	304
22.2.2	Hallo tkinter!	305
22.2.3	Grafische Bedienelemente (Widgets)	307
22.2.4	Anordnen der Bedienelemente (Geometry Managers)	327

22.2.5	Ereignisse	333
22.2.6	Beispiel: Taschenrechner-Applikation.	337
22.3	Arbeiten mit Dateien	343
22.4	Übung: Entwicklung eines einfachen Text-Editors	346
22.5	Zusammenfassung	347
22.6	Lösungen zu den Aufgaben	348
23	Wie arbeite ich mit Programmfunktionen, um Daten zu bearbeiten und Aktionen auszulösen?	357
23.1	Arbeiten mit Funktionen	358
23.1.1	Definition von Funktionen	358
23.1.2	Funktionsargumente	359
23.1.3	Rückgabewerte	365
23.1.4	Lokale und globale Variablen	366
23.2	Funktionen als Klassen-Methoden von Objekten verwenden	370
23.3	Arbeiten mit Modulen und Packages	374
23.3.1	Programmcode modularisieren	374
23.3.2	Elemente aus Modulen importieren	375
23.3.3	Die Community nutzen – Der <i>Python Package Index (PyPI)</i>	377
23.4	Zusammenfassung	380
23.5	Lösungen zu den Aufgaben	381
24	Wie steuere ich den Programmablauf und lasse das Programm auf Benutzeraktionen und andere Ereignisse reagieren?	385
24.1	if-else-Konstrukte	386
24.1.1	Einfache if-else-Konstrukte	386
24.1.2	Verschachtelte if-else-Konstrukte	389
24.1.3	if-else-Konstrukte mit zusammengesetzten Bedingungen	390
24.1.4	if-else-Konstrukte mit alternativen Bedingungen (elif)	392
24.2	Ereignisse	394
24.3	Zusammenfassung	394
24.4	Lösungen zu den Aufgaben	395
25	Wie wiederhole ich Programmanweisungen effizient?	399
25.1	Abgezählte Schleifen (for)	400
25.1.1	Einfache for-Schleifen	400
25.1.2	Verschachtelte for-Schleifen	405
25.1.3	Listenkomprehensionsausdrücke	406
25.2	Bedingte Schleifen (while)	407
25.3	Schleifen vorzeitig verlassen und neustarten	410
25.4	Zusammenfassung	412
25.5	Lösungen zu den Aufgaben	413

26	Wie suche und behebe ich Fehler auf strukturierte Art und Weise	417
26.1	Fehlerbehandlung zur Laufzeit	418
26.1.1	Fehler durch gezielte Prüfungen abfangen	418
26.1.2	Versuche...Fehler-Konstrukte (try...except)	420
26.2	Fehlersuche und -beseitigung während der Entwicklung	422
26.2.1	Haltepunkte.....	423
26.2.2	Anzeige des Variableninhalts und Verwendung von Watches.....	425
26.2.3	Schrittweise Ausführung.....	426
26.3	Zusammenfassung	426
26.4	Lösungen zu den Aufgaben	427

IV JavaScript

27	Einführung	433
28	Was brauche ich zum Programmieren?	437
28.1	Interpreter.....	438
28.2	Code-Editoren und Entwicklungsumgebungen.....	438
28.3	Hilfe und Dokumentation	439
28.4	Zusammenfassung.....	440
29	Was muss ich tun, um ein Programm zum Laufen zu bringen?	441
29.1	Einbinden von JavaScript-Code in Webseiten.....	442
29.1.1	Das script-Element in HTML	442
29.1.2	Sicherheitsaspekte	446
29.2	„Hallo Welt“ in JavaScript.....	447
29.2.1	Do-it-yourself: Der (gar nicht so) mühsame Weg	447
29.2.2	Mit etwas Nachhilfe: Die Schnelle Umsetzung mit einem Webdienst.....	449
29.3	Zusammenfassung.....	450
30	Wie stelle ich sicher, dass ich (und andere) mein Programm später noch verstehe?	451
30.1	Gestaltung des Programm-Codes und Namenskonventionen	452
30.2	Kommentare.....	454
30.3	Zusammenfassung.....	455
31	Wie speichere ich Daten, um mit ihnen zu arbeiten?	457
31.1	Deklaration von Variablen	459
31.2	Elementare Datentypen	460
31.2.1	Zahlen (number)	460
31.2.2	Zeichenketten (string)	462
31.2.3	Wahrheitswerte (boolean)	467
31.2.4	Spezielle Typen und Werte (null, undefined, NaN).....	467
31.3	Konvertieren von Variablen	469
31.3.1	Implizite Konvertierung.....	469

31.3.2	Explizite Konvertierung	470
31.4	Arrays	472
31.5	Objekte	480
31.5.1	Objektorientierung in JavaScript	480
31.5.2	Objekte direkt erzeugen	481
31.5.3	Auf Eigenschaften von Objekten zugreifen	482
31.5.4	Objekte mit Hilfe des Object-Konstruktors erzeugen	483
31.5.5	Objekte mit Hilfe von Konstruktor-Funktionen erzeugen	484
31.5.6	JSON	485
31.6	Lösungen zu den Aufgaben	486
31.7	Zusammenfassung	489
32	Wie lasse ich Daten ein- und ausgeben?	491
32.1	Überblick über die Ein- und Ausgabe in JavaScript	493
32.2	Ausgabe über die Konsole	494
32.3	Ein- und Ausgaben über Dialogboxen	497
32.4	Ausgabe in ein HTML-Dokument / Webseite	498
32.4.1	HTML-Code in die Webseite schreiben	498
32.4.2	Das Document Object Model (DOM)	500
32.4.3	DOM-Knoten über ihre Eigenschaften selektieren	501
32.4.4	DOM-Knoten über die hierarchische Struktur des Dokuments selektieren	503
32.4.5	HTML-Elemente ändern	506
32.4.6	HTML-Elemente hinzufügen und löschen	508
32.5	Eingabe mit Formularen	510
32.5.1	Formulare in HTML	510
32.5.2	Formulare aus JavaScript heraus ansteuern	513
32.6	Beispiel: Einfacher Taschenrechner	517
32.6.1	Die Web-Oberfläche	518
32.6.2	Die CSS-Designanweisungen	521
32.6.3	Der JavaScript-Code	524
32.7	Beispiel: Color Picker	525
32.7.1	Die Web-Oberfläche	525
32.7.2	Der JavaScript-Code	528
32.8	Zusammenfassung	530
32.9	Lösungen zu den Aufgaben	531
33	Wie arbeite ich mit Programmfunktionen, um Daten zu bearbeiten und Aktionen auszulösen?	535
33.1	Arbeiten mit Funktionen	536
33.1.1	Definition von Funktionen	536
33.1.2	Rückgabewerte	541
33.1.3	Argumente und Parameter von Funktionen	542
33.1.4	Gültigkeitsbereich von Variablen in Funktionen	546
33.2	Arbeiten mit Modulen/Bibliotheken	549
33.2.1	Eigene Module entwickeln und verwenden	549
33.2.2	Externe Module/Bibliotheken finden und einbinden	551

33.3	Frameworks	552
33.4	Zusammenfassung	553
33.5	Lösungen zu den Aufgaben	554
34	Wie steuere ich den Programmablauf und lasse das Programm auf Benutzeraktionen und andere Ereignisse reagieren?	557
34.1	if-else-Konstrukte	558
34.2	switch-case-Konstrukte	561
34.3	Ereignisse	562
34.4	Lösungen zu den Aufgaben	566
34.5	Zusammenfassung	568
35	Wie wiederhole ich Programmanweisungen effizient?	571
35.1	Abgezählte Schleifen (for und for..of)	572
35.1.1	for-Schleifen mit numerischen Laufvariablen	572
35.1.2	for-Schleife mit Objekt-Laufvariable (for...of)	579
35.2	Bedingte Schleifen (while und do...while)	580
35.3	Zusammenfassung	582
35.4	Lösungen zu den Aufgaben	582
36	Wie suche und behebe ich Fehler auf strukturierte Art und Weise?	587
36.1	Fehlerbehandlung zur Laufzeit	588
36.2	Fehlersuche und -beseitigung während der Entwicklung	590
36.3	Zusammenfassung	594
	Serviceteil	
	Stichwortverzeichnis	599