

---

# Inhalt

<b>Der Autor</b> .....	<b>xxii</b>
<b>Der Fachgutachter</b> .....	<b>xxii</b>
<b>Danksagung</b> .....	<b>xxiii</b>
<b>Vorwort zur zweiten Auflage</b> .....	<b>xxv</b>
<b>Einleitung</b> .....	<b>xxix</b>
<b>Teil 1 Grundlagen</b> .....	<b>1</b>
<b>1 Erste Schritte</b> .....	<b>3</b>
Die Programmierumgebung einrichten .....	3
Python-Versionen .....	4
Python-Codeausschnitte ausführen .....	4
Der Editor Sublime Text .....	5
Python auf verschiedenen Betriebssystemen .....	5
Python unter Windows .....	6
Python unter macOS .....	8
Python unter Linux .....	10
Das Hello-World-Programm ausführen .....	11
Sublime Text auf die richtige Python-Version einstellen .....	11
Hello_world.py ausführen .....	12
Fehlersuche .....	12
Python-Programme im Terminal ausführen .....	14
Unter Windows .....	14
Unter Linux und macOS .....	15
Zusammenfassung .....	16

---

<b>2 Variablen und einfache Datentypen</b> .....	<b>17</b>
Was bei der Ausführung von <code>hello_world.py</code> wirklich geschieht .....	17
Variablen .....	18
Variablen benennen und verwenden .....	19
Fehler bei Variablennamen vermeiden .....	20
Variablen sind Etiketten .....	21
Strings .....	22
Groß- und Kleinschreibung mithilfe von Methoden ändern .....	22
Variablen in Strings verwenden .....	24
Weißraum hinzufügen .....	25
Weißraum entfernen .....	26
Syntaxfehler bei der Stringverarbeitung vermeiden .....	27
Zahlen .....	29
Integer .....	29
Fließkommazahlen .....	30
Integer und Fließkommazahlen .....	31
Unterstriche in Zahlen .....	31
Mehrfachzuweisung .....	32
Konstanten .....	32
Kommentare .....	33
Wie werden Kommentare geschrieben? .....	33
Was für Kommentare sind sinnvoll? .....	33
The Zen of Python .....	34
Zusammenfassung .....	36
<b>3 Eine Einführung in Listen</b> .....	<b>37</b>
Was sind Listen? .....	37
Elemente in einer Liste ansprechen .....	38
Indizes beginnen bei 0, nicht bei 1 .....	39
Einzelne Werte aus einer Liste verwenden .....	39
Elemente ändern, hinzufügen und entfernen .....	40
Elemente in einer Liste ändern .....	41
Elemente zu einer Liste hinzufügen .....	41
Elemente aus einer Liste entfernen .....	43
Listen ordnen .....	48
Listen mit <code>sort()</code> dauerhaft sortieren .....	48
Listen mit der Funktion <code>sorted()</code> vorübergehend sortieren .....	48

---

Listen in umgekehrter Reihenfolge ausgeben . . . . .	49
Die Länge einer Liste ermitteln . . . . .	50
Indexfehler vermeiden . . . . .	51
Zusammenfassung . . . . .	53
<b>4 Mit Listen arbeiten . . . . .</b>	<b>55</b>
Eine komplette Liste durchlaufen . . . . .	55
Die Schleife im Detail . . . . .	56
Weitere Aufgaben in einer for-Schleife erledigen . . . . .	57
Aktionen nach der for-Schleife . . . . .	59
Einrückungsfehler vermeiden . . . . .	60
Vergessene Einrückung der ersten Zeile in einer Schleife . . . . .	60
Vergessene Einrückung nachfolgender Zeilen . . . . .	61
Unnötige Einrückung . . . . .	61
Unnötige Einrückung nach einer Schleife . . . . .	62
Vergessener Doppelpunkt . . . . .	63
Numerische Listen . . . . .	64
Die Funktion range() . . . . .	64
Numerische Listen mithilfe von range() aufstellen . . . . .	65
Einfache Statistiken für numerische Listen . . . . .	66
Listennotation . . . . .	67
Teillisten . . . . .	68
Einen Slice erstellen . . . . .	68
Einen Slice in einer Schleife durchlaufen . . . . .	70
Listen kopieren . . . . .	71
Tupel . . . . .	74
Ein Tupel definieren . . . . .	74
Die Werte in einem Tupel durchlaufen . . . . .	75
Tupel überschreiben . . . . .	75
Code formatieren . . . . .	76
Die Gestaltungsrichtlinien . . . . .	77
Einrückung . . . . .	77
Zeilenlänge . . . . .	78
Leerzeilen . . . . .	78
Zusammenfassung . . . . .	79

<b>5 if-Anweisungen</b> .....	<b>81</b>
Ein einfaches Beispiel .....	81
Bedingungen .....	82
Prüfung auf Gleichheit .....	82
Groß- und Kleinschreibung bei der Prüfung auf Gleichheit .....	83
Prüfung auf Ungleichheit .....	84
Numerische Vergleiche .....	85
Prüfung auf mehrere Bedingungen .....	85
Prüfung auf Vorhandensein eines Werts in einer Liste .....	87
Prüfung auf Abwesenheit eines Werts in einer Liste .....	87
Boolesche Ausdrücke .....	88
if-Anweisungen .....	89
Einfache if-Anweisungen .....	89
if-else-Anweisungen .....	90
Die if-elif-else-Kette .....	91
Mehrere elif-Blöcke .....	93
Den else-Block weglassen .....	93
Mehrere Bedingungen prüfen .....	94
if-Anweisungen für Listen .....	97
Prüfung auf besondere Elemente .....	97
Prüfung auf nicht leere Liste .....	98
Mehrere Listen verwenden .....	99
if-Anweisungen gestalten .....	101
Zusammenfassung .....	102
 <b>6 Dictionaries</b> .....	 <b>103</b>
Ein einfaches Dictionary .....	104
Umgang mit Dictionaries .....	104
Zugriff auf die Werte in einem Dictionary .....	105
Schlüssel-Wert-Paare hinzufügen .....	106
Ein leeres Dictionary als Ausgangspunkt .....	106
Werte in einem Dictionary ändern .....	107
Schlüssel-Wert-Paare entfernen .....	109
Ein Dictionary aus ähnlichen Objekten .....	109
Mit get() auf Werte zugreifen .....	111
Dictionaries in einer Schleife durchlaufen .....	113
Alle Schlüssel-Wert-Paare durchlaufen .....	113
Alle Schlüssel in einem Dictionary durchlaufen .....	115

Die Schlüssel in einem Dictionary geordnet durchlaufen	117
Alle Werte in einem Dictionary durchlaufen	117
Verschachtelung	120
Dictionaries in einer Liste	120
Listen in einem Dictionary	123
Dictionaries in einem Dictionary	125
Zusammenfassung	127
<b>7 Benutzereingaben und while-Schleifen</b>	<b>129</b>
Die Funktion input()	130
Klar verständliche Eingabeaufforderungen schreiben	130
Verwendung von int() für numerische Eingaben	131
Der Modulo-Operator	133
while-Schleifen	134
while-Schleifen in Aktion	134
Programmbeendigung durch den Benutzer	135
Flags	137
Eine Schleife mit break verlassen	138
Die Anweisung continue	139
Endlosschleifen vermeiden	140
while-Schleifen für Listen und Dictionaries	141
Elemente von einer Liste in eine andere verschieben	142
Alle Vorkommen eines Wertes aus einer Liste entfernen	143
Ein Dictionary mit Benutzereingaben füllen	143
Zusammenfassung	145
<b>8 Funktionen</b>	<b>147</b>
Funktionen definieren	148
Informationen an eine Funktion übergeben	148
Argumente und Parameter	149
Argumente übergeben	150
Positionsabhängige Argumente	150
Schlüsselwortargumente	152
Standardwerte	153
Verschiedene Formen für Funktionsaufrufe	154
Argumentfehler vermeiden	155
Rückgabewerte	156
Einen einfachen Wert zurückgeben	157

Optionale Argumente .....	157
Ein Dictionary zurückgeben .....	159
Funktionen in einer while-Schleife .....	160
Eine Liste übergeben .....	162
Eine Liste mithilfe einer Funktion ändern .....	163
Die Änderung einer Liste in einer Funktion verhindern .....	166
Beliebig viele Argumente übergeben .....	167
Positionsabhängige Argumente und Argumente beliebiger Anzahl kombinieren .....	168
Beliebig viele Schlüsselwortargumente übergeben .....	169
Funktionen in Modulen speichern .....	171
Ein komplettes Modul importieren .....	171
Einzelne Funktionen importieren .....	172
Eine Funktion mit »as« umbenennen .....	173
Ein Modul mit »as« umbenennen .....	174
Alle Funktionen eines Moduls importieren .....	174
Gestaltung von Funktionen .....	175
Zusammenfassung .....	176
<b>9 Klassen .....</b>	<b>179</b>
Eine Klasse erstellen und verwenden .....	180
Die Klasse Dog erstellen .....	180
Eine Instanz einer Klasse anlegen .....	182
Mit Klassen und Instanzen arbeiten .....	185
Die Klasse Car .....	185
Einen Standardwert für ein Attribut festlegen .....	186
Attributwerte bearbeiten .....	187
Vererbung .....	191
Die Methode <code>__init__()</code> für eine Kindklasse .....	191
Attribute und Methoden der Kindklasse definieren .....	193
Methoden der Elternklasse überschreiben .....	194
Instanzen als Attribute .....	194
Reale Objekte modellieren .....	197
Klassen importieren .....	198
Eine einzelne Klasse importieren .....	198
Mehrere Klassen in einem Modul speichern .....	200
Mehrere Klassen aus einem Modul importieren .....	201

Ein gesamtes Modul importieren . . . . .	202
Alle Klassen eines Moduls importieren . . . . .	202
Ein Modul in ein Modul importieren . . . . .	203
Aliase verwenden . . . . .	204
Ihren eigenen Arbeitsablauf finden . . . . .	205
Die Standardbibliothek von Python . . . . .	205
Gestaltung von Klassen . . . . .	207
Zusammenfassung . . . . .	207
<b>10 Dateien und Ausnahmen . . . . .</b>	<b>209</b>
Aus Dateien lesen . . . . .	210
Eine gesamte Datei lesen . . . . .	210
Dateipfade . . . . .	212
Zeilenweises Lesen . . . . .	213
Eine Liste aus den Zeilen einer Datei erstellen . . . . .	214
Dateiinhalte verarbeiten . . . . .	215
Große Dateien: eine Million Stellen . . . . .	216
Ist Ihr Geburtsdatum in Pi enthalten? . . . . .	217
In Dateien schreiben . . . . .	218
In eine leere Datei schreiben . . . . .	218
Mehrere Zeilen schreiben . . . . .	219
Text an eine Datei anhängen . . . . .	220
Ausnahmen . . . . .	221
Division durch null . . . . .	221
try-except-Blöcke . . . . .	222
Abstürze mithilfe von Ausnahmen verhindern . . . . .	223
Der else-Block . . . . .	224
Datei nicht gefunden . . . . .	225
Text analysieren . . . . .	226
Umgang mit mehreren Dateien . . . . .	227
Fehler stillschweigend übergehen . . . . .	229
Welche Fehler sollten Sie melden und welche nicht? . . . . .	230
Daten speichern . . . . .	231
json.dump() und json.load() . . . . .	232
Benutzergenerierte Daten speichern und lesen . . . . .	233
Refactoring . . . . .	235
Zusammenfassung . . . . .	238

<b>11 Code testen</b> .....	<b>239</b>
Funktionen testen .....	240
Unit Tests und Testfälle .....	241
Ein bestandener Test .....	241
Ein nicht bestandener Test .....	243
Was tun bei einem nicht bestandenen Test? .....	244
Neue Tests hinzufügen .....	246
Klassen testen .....	247
Verschiedene Zusicherungsmethoden .....	247
Eine Beispielklasse zum Testen .....	248
Die Klasse AnonymousSurvey testen .....	250
Die Methode setUp() .....	252
Zusammenfassung .....	254
<b>Teil 2    Projekte</b> .....	<b>257</b>
Alien Invasion – ein Python-Spiel .....	257
Datenvisualisierung .....	258
Webanwendungen .....	258
<b>Projekt 1: Alien Invasion</b> .....	<b>259</b>
<b>12 Das eigene Kampfschiff</b> .....	<b>261</b>
Das Projekt planen .....	262
Pygame installieren .....	263
Erste Schritte für das Spielprojekt .....	263
Ein Pygame-Fenster anlegen und auf Benutzereingaben reagieren ...	263
Die Hintergrundfarbe festlegen .....	265
Eine Klasse für Einstellungen anlegen .....	266
Das Bild eines Raumschiffs hinzufügen .....	267
Die Klasse Ship .....	269
Das Schiff auf den Bildschirm zeichnen .....	270
Refactoring: Die Methoden _check_events() und _update_screen() .....	272
Die Methode _check_events() .....	272
Die Methode _update_screen() .....	273
Das Schiff bewegen .....	274
Auf Tastenbetätigungen reagieren .....	274
Kontinuierliche Bewegung .....	275

Bewegung nach rechts und links . . . . .	277
Die Geschwindigkeit des Schiffes anpassen . . . . .	278
Den Bewegungsbereich des Schiffes einschränken . . . . .	280
Refactoring von <code>_check_events()</code> . . . . .	281
Beenden mit Q . . . . .	282
Das Spiel im Vollbildmodus ausführen . . . . .	282
Zwischenstand . . . . .	283
<code>alien_invasion.py</code> . . . . .	283
<code>settings.py</code> . . . . .	284
<code>ship.py</code> . . . . .	284
Geschosse . . . . .	284
Einstellungen für Geschosse hinzufügen . . . . .	285
Die Klasse <code>Bullet</code> . . . . .	285
Geschosse in Gruppen speichern . . . . .	287
Geschosse abfeuern . . . . .	287
Alte Geschosse löschen . . . . .	289
Die Anzahl der Geschosse begrenzen . . . . .	290
Die Methode <code>_update_bullets()</code> . . . . .	291
Zusammenfassung . . . . .	292
<b>13 Die Außerirdischen . . . . .</b>	<b>293</b>
Überblick über das Projekt . . . . .	294
Das erste Invasionsschiff . . . . .	294
Die Klasse <code>Alien</code> . . . . .	295
Eine Instanz von <code>Alien</code> erstellen . . . . .	296
Die Invasionsflotte erstellen . . . . .	298
Wie viele Invasionsschiffe passen in eine Reihe? . . . . .	298
Reihen von Invasionsschiffen erstellen . . . . .	299
Refactoring von <code>_create_fleet()</code> . . . . .	300
Reihen hinzufügen . . . . .	301
Die Flotte in Bewegung setzen . . . . .	304
Die Invasoren nach rechts bewegen . . . . .	304
Einstellungen für die Flugrichtung der Flotte . . . . .	305
Auf Randberührungen prüfen . . . . .	306
Sinken und Flugrichtung ändern . . . . .	307
Invasoren abschießen . . . . .	308
Kollisionen von Geschossen erkennen . . . . .	308
Größere Geschosse zu Testzwecken . . . . .	310

Die Flotte auffüllen . . . . .	311
Die Geschosse beschleunigen . . . . .	312
Refactoring von <code>_update_bullets()</code> . . . . .	312
Spielende . . . . .	313
Kollisionen zwischen Invasoren und dem eigenen Schiff erkennen . . .	313
Auf Kollisionen zwischen Invasoren und dem eigenen Schiff reagieren	314
Wenn Invasoren den unteren Bildschirmrand erreichen . . . . .	317
Game over! . . . . .	318
Welche Teile des Spiels müssen ausgeführt werden? . . . . .	319
Zusammenfassung . . . . .	320
<b>14 Das Wertungssystem . . . . .</b>	<b>321</b>
Eine Play-Schaltfläche hinzufügen . . . . .	321
Die Klasse <code>Button</code> . . . . .	322
Die Schaltfläche auf den Bildschirm zeichnen . . . . .	324
Das Spiel starten . . . . .	325
Das Spiel zurücksetzen . . . . .	326
Die Play-Schaltfläche deaktivieren . . . . .	327
Den Mauszeiger ausblenden . . . . .	327
Levels . . . . .	328
Die Geschwindigkeitseinstellungen ändern . . . . .	329
Die Geschwindigkeit zurücksetzen . . . . .	330
Die Punktwertung . . . . .	331
Den Punktestand anzeigen . . . . .	332
Eine Anzeigetafel erstellen . . . . .	333
Den Punktestand bei jedem Abschuss erhöhen . . . . .	335
Den Punktestand zurücksetzen . . . . .	335
Alle Treffer berücksichtigen . . . . .	336
Den Punktwert erhöhen . . . . .	337
Den Punktestand runden . . . . .	338
Highscore . . . . .	339
Das Level anzeigen . . . . .	341
Die Anzahl der verfügbaren Schiffe anzeigen . . . . .	344
Zusammenfassung . . . . .	347

<b>Projekt 2: Datenvisualisierung</b> .....	<b>349</b>
<b>15 Daten generieren</b> .....	<b>351</b>
Matplotlib installieren .....	352
Einfache Liniendiagramme .....	353
Beschriftung und Linienstärke ändern .....	354
Das Diagramm korrigieren .....	355
Vordefinierte Formatierungen verwenden .....	356
Einzelne Punkte mit scatter() darstellen und gestalten .....	358
Eine Folge von Punkten mit scatter() ausgeben .....	359
Daten automatisch berechnen .....	360
Eigene Farben festlegen .....	361
Eine Colormap verwenden .....	362
Diagramme automatisch speichern .....	363
Zufallsbewegungen .....	364
Die Klasse RandomWalk .....	364
Richtungen wählen .....	365
Den Zufallspfad als Diagramm ausgeben .....	366
Mehrere Zufallspfade erstellen .....	367
Den Pfad gestalten .....	368
Würfeln mit Plotly .....	373
Plotly installieren .....	373
Die Klasse Die .....	374
Würfeln .....	374
Die Ergebnisse analysieren .....	375
Ein Histogramm erstellen .....	376
Ergebnisse bei zwei Würfeln .....	378
Würfel unterschiedlicher Flächenzahl .....	380
Zusammenfassung .....	382
<b>16 Daten herunterladen</b> .....	<b>383</b>
Das Dateiformat CSV .....	384
CSV-Spaltenköpfe analysieren .....	384
Die Spaltenköpfe und ihre Position ausgeben .....	385
Daten entnehmen und lesen .....	386
Daten in einem Temperaturdiagramm darstellen .....	387
Das Modul datetime .....	388

Datumsangaben im Diagramm darstellen .....	389
Ein Diagramm für einen längeren Zeitraum .....	391
Eine zweite Datenreihe darstellen .....	392
Einen Diagrammbereich einfärben .....	393
Fehlerprüfung .....	394
Daten selbst herunterladen .....	398
Globale Daten im JSON-Format visualisieren .....	399
Erdbebedaten herunterladen .....	400
JSON-Daten untersuchen .....	400
Eine Liste aller Erdbeben aufstellen .....	403
Die Stärken entnehmen .....	403
Ortsdaten entnehmen .....	404
Eine Weltkarte zeichnen .....	405
Eine andere Möglichkeit zur Angabe von Diagrammdaten .....	406
Die Größe der Markierungen anpassen .....	407
Die Farben der Markierungen anpassen .....	408
Weitere Farbpaletten .....	410
Maustext hinzufügen .....	410
Zusammenfassung .....	412
<b>17 APIs .....</b>	<b>415</b>
Web-APIs .....	415
Git und GitHub .....	416
Daten mithilfe eines API-Aufruf anfordern .....	416
Das Paket requests installieren .....	417
API-Antworten verarbeiten .....	418
Das Antwort-Dictionary verarbeiten .....	419
Ein Überblick über die höchstbewerteten Repositories .....	421
Grenzwerte für die API-Aufruftrate .....	422
Angaben zu Repositories mit Plotly visualisieren .....	423
Plotly-Diagramme verbessern .....	425
Eigenen Maustext hinzufügen .....	427
Links zu dem Diagramm hinzufügen .....	429
Mehr über Plotly und die GitHub-API .....	430
Die API von Hacker News .....	430
Zusammenfassung .....	434

<b>Projekt 3: Webanwendungen</b> .....	<b>435</b>
<b>18 Erste Schritte mit Django</b> .....	<b>437</b>
Ein Projekt einrichten .....	438
Eine Spezifikation schreiben .....	438
Eine virtuelle Umgebung erstellen .....	438
Die virtuelle Umgebung aktivieren .....	439
Django installieren .....	440
Ein Projekt in Django erstellen .....	440
Die Datenbank erstellen .....	441
Das Projekt anzeigen .....	442
Eine App anlegen .....	444
Modelle definieren .....	444
Modelle aktivieren .....	446
Die Admin-Site von Django .....	447
Das Modell für die Einträge definieren .....	450
Das Modell Entry in die Datenbank aufnehmen .....	451
Das Modell Entry auf der Admin-Site registrieren .....	452
Die Django-Shell .....	453
Seiten erstellen: die Startseite von Learning Log .....	455
Eine URL zuordnen .....	456
Eine Ansicht schreiben .....	458
Eine Vorlage schreiben .....	458
Weitere Seiten erstellen .....	460
Vererbung bei Vorlagen .....	460
Die Seite Topics .....	463
Einzelne Fachgebietsseiten .....	466
Zusammenfassung .....	470
<b>19 Benutzerkonten</b> .....	<b>473</b>
Dateneingabe durch die Benutzer .....	474
Neue Fachgebiete hinzufügen .....	474
Neue Einträge hinzufügen .....	479
Einträge bearbeiten .....	483
Benutzerkonten einrichten .....	487
Die App users .....	487
Die Anmeldeseite .....	488

Abmelden .....	491
Die Registrierungsseite .....	493
Die Benutzer als Besitzer ihrer eigenen Daten .....	496
Den Zugriff mit @login_required beschränken .....	496
Daten mit Benutzern verknüpfen .....	498
Den Zugriff auf die Fachgebiete auf die zuständigen Benutzer einschränken .....	502
Die Fachgebiete eines Benutzers schützen .....	502
Die Seite edit_entry schützen .....	503
Neue Fachgebiete dem aktuellen Benutzer zuordnen .....	504
Zusammenfassung .....	505
<b>20 Eine App gestalten und bereitstellen .....</b>	<b>507</b>
Learning Log gestalten .....	508
Die App django-bootstrap4 .....	508
Learning Log mit Bootstrap gestalten .....	509
Änderungen an base.html .....	510
Die Startseite mit einem Jumbotron gestalten .....	514
Das Anmeldeformular gestalten .....	516
Die Seite Topics gestalten .....	517
Einträge auf den Fachgebieten gestalten .....	518
Learning Log bereitstellen .....	520
Ein Heroku-Konto anlegen .....	520
Die Heroku-Befehlszeile installieren .....	520
Die erforderlichen Pakete installieren .....	521
Die Datei requirements.txt erstellen .....	521
Die Python-Laufzeitversion angeben .....	522
Die Datei settings.py für Heroku anpassen .....	523
Ein Procfile zum Starten der Prozesse erstellen .....	523
Mit Git den Überblick über die Projektdateien bewahren .....	523
Die Datenbank auf Heroku einrichten .....	528
Die Heroku-Bereitstellung verbessern .....	528
Das Onlineprojekt schützen .....	530
Änderungen mit Commit bestätigen und übertragen .....	531
Umgebungsvariablen auf Heroku einrichten .....	533
Eigene Fehlerseiten erstellen .....	533
Weiterentwicklung des Projekts .....	536

Die Einstellung SECRET_KEY .....	537
Projekte auf Heroku löschen .....	537
Zusammenfassung .....	539

<b>Nachwort .....</b>	<b>541</b>
-----------------------	------------

<b>Anhang .....</b>	<b>543</b>
---------------------	------------

<b>A Installation und Fehlerbehebung .....</b>	<b>543</b>
--	------------

Python unter Windows .....	543
Den Python-Interpreter finden .....	544
Python zur Pfadvariablen hinzufügen .....	544
Python neu installieren .....	545
Python unter macOS .....	545
Homebrew installieren .....	546
Python unter Linux .....	547
Schlüsselwörter und integrierte Funktionen .....	547
Python-Schlüsselwörter .....	548
Integrierte Python-Funktionen .....	548

<b>B Texteditoren und IDEs .....</b>	<b>549</b>
--------------------------------------	------------

Die Einstellungen von Sublime Text anpassen .....	550
Tabulatoren in Leerzeichen umwandeln .....	550
Den Zeilenlängenmarker festlegen .....	551
Codeblöcke einrücken und Einrückungen aufheben .....	551
Codeblöcke auskommentieren .....	551
Die Konfiguration speichern .....	551
Weitere Anpassungen .....	552
Weitere Texteditoren und IDEs .....	552
IDLE .....	552
Geany .....	552
Emacs und Vim .....	553
Atom .....	553
Visual Studio Code .....	553
PyCharm .....	553
Jupyter Notebooks .....	554

<b>C Hilfe finden</b> .....	<b>555</b>
Erste Schritte .....	555
Versuchen Sie es erneut .....	556
Legen Sie eine Pause ein .....	556
Nutzen Sie das Onlinematerial zu diesem Buch .....	557
Online nach Hilfe suchen .....	557
Stack Overflow .....	557
Die offizielle Python-Dokumentation .....	558
Offizielle Dokumentation der Bibliotheken .....	558
r/learnpython .....	558
Blogs .....	558
IRC (Internet Relay Chat) .....	559
Ein IRC-Konto anlegen .....	559
Hilfreiche Kanäle .....	560
IRC-Kultur .....	560
Slack .....	560
Discord .....	561
<b>D Versionssteuerung mit Git</b> .....	<b>563</b>
Git installieren .....	564
Git unter Windows installieren .....	564
Git unter macOS installieren .....	564
Git unter Linux installieren .....	564
Git konfigurieren .....	564
Ein Projekt anlegen .....	565
Dateien ignorieren .....	565
Ein Repository initialisieren .....	565
Den Projektstatus überprüfen .....	566
Dateien zum Repository hinzufügen .....	566
Einen Commit durchführen .....	567
Das Protokoll einsehen .....	568
Der zweite Commit .....	568
Änderungen zurücknehmen .....	569
Vorherige Commits auschecken .....	571
Das Repository löschen .....	572
<b>Stichwortverzeichnis</b> .....	<b>575</b>