

PROGRAMMIERSPRACHE PYTHON

SWEET SPOT ZWISCHEN LEICHT LERNBAR UND MÄCHTIG

```
def _same_strand(self, entry, alignment):  
    assert entry.strand in ["+", "-"]  
    if alignment.is_read2 is False:  
        if (entry.strand == "+" and alignment.is_read1 is True and  
            entry.strand == "-" and alignment.is_read1 is False):  
            return True  
        # Mate pair for paired end sequencing  
    elif alignment.is_read2 is True:  
        if (entry.strand == "+" and alignment.is_read1 is True and  
            entry.strand == "-" and alignment.is_read1 is False):  
            return True  
    return False  
calc method(self):  
Ln 522, C
```

INHALT

- ▶ WARUM UND FÜR WEN PYTHON SICH LOHNT
Was bringt es mir, programmieren zu können?
 - ▶ THE ZEN OF PYTHON
Was ist Python und warum ist es einfacher zu erlernen als andere Programmiersprachen?
 - ▶ PYTHON LERNEN
Wie gelingt der Einstieg und was brauche ich dafür?
 - ▶ NACHSCHLAG: PYTHON, WIE DIE SCHLANGE?
Die britische Komikertruppe Monty Python als Namensgeber
-

► EINLEITUNG

Datenkompetenz – oder auch Data Literacy – ist in der digitalisierten Welt unverzichtbar.

Ob in der Arbeitswelt, Wissenschaft und Forschung oder der Gesellschaft: die Fähigkeit, Daten zu erfassen, zu analysieren, zu interpretieren, zu visualisieren und vieles mehr ist gefragter denn je. Programmieren ist dabei eine wichtige Kompetenz, die zur Data Literacy beiträgt.

Mit der Programmiersprache Python gelingt ein leichter Einstieg ins Programmieren.



▶ WARUM UND FÜR WEN PROGRAMMIEREN LERNEN MIT PYTHON SICH LOHNT

Was bringt es mir, programmieren zu können?

Routinemäßige und triviale Aufgaben, die keinen Spaß machen und zu viel Zeit in Anspruch nehmen, einfach automatisieren. Klingt nach einem vielversprechenden und erstrebenswerten Ansatz! Das sieht auch Al Sweigart so. Für sein Buch über Python wählt er den bezeichnenden Buchtitel: Automate the boring stuff with Python.^[1]

Und welchen „boring stuff“ kann Python dann ganz konkret erledigen? Zum Beispiel eigene Programme schreiben, mit denen man Dateien und Ordner schnell und automatisch erstellen, aktualisieren, verschieben und umbenennen kann oder im Web nach benötigtem Online-Content suchen und diesen direkt in gewünschter Form herunterladen.

Doch Python kann natürlich noch viel mehr als Arbeitsprozesse zu automatisieren: In Datenanalyse, Datenverarbeitung oder auch in der Webentwicklung kommt die Programmiersprache zum Einsatz und steckt etwa in Teilen in den Programmen hinter YouTube, Netflix oder Google. Auch in Bereichen des maschinellen Lernens und der Künstliche Intelligenz gehört Python zu den bevorzugten Coding-Sprachen.

“
Automate the
Boring Stuff
with Python.
”

Al Sweigart

Programmiersprache Python im Forschungs- und Arbeitsalltag

Automatisierungsprozesse sind im Arbeitsalltag unverzichtbar. In der Forschung etwa fallen riesige Datenmengen an, die händisch nicht zu bewältigen sind. Sie müssen analysiert und verarbeitet werden. Oft reichen vorhandene Programme nicht aus, um die Daten genau in der Form auszuwerten, wie sie für das konkrete Forschungsvorhaben benötigt wird. Da hilft ein selbst programmiertes Programm, das alle Bedürfnisse abdeckt. Python hat sich hier als hilfreiches Werkzeug etabliert.

Auch die Anforderungen in Wissenschaftlichen Bibliotheken ändern und erweitern sich durch die zunehmende Digitalisierung der Forschung ständig. Data Librarians übernehmen in diesem Kontext nicht nur die Rolle als Vermittler:innen zwischen der IT und dem bibliothekarischen Personal. Sie können mit ihren Programmierkenntnissen auch schnell und unkompliziert den Arbeitsalltag erleichtern – beispielsweise durch das Schreiben von Skripten, die genau auf die jeweiligen Bedürfnisse abgestimmt sind. Damit lassen sich etwa Metadaten oder Signaturen automatisch erfassen.

[1] <https://automatetheboringstuff.com/>

▶ WARUM PYTHON DER IDEALE EINSTIEG INS PROGRAMMEIREN IST

Was ist Python?

Bei der Programmiersprache Python handelt es sich um eine sogenannte General Purpose Language (GPL). Übersetzt bedeutet dies Mehrzweck- oder Universalsprache, d.h. sie ist für viele Problemstellungen einsetzbar. Im Gegensatz dazu stehen domainspezifische Sprachen, die sich nur für spezielle Anwendungsfälle eignen. Andere bekannte GPLs sind beispielsweise JavaScript, PHP oder C++.

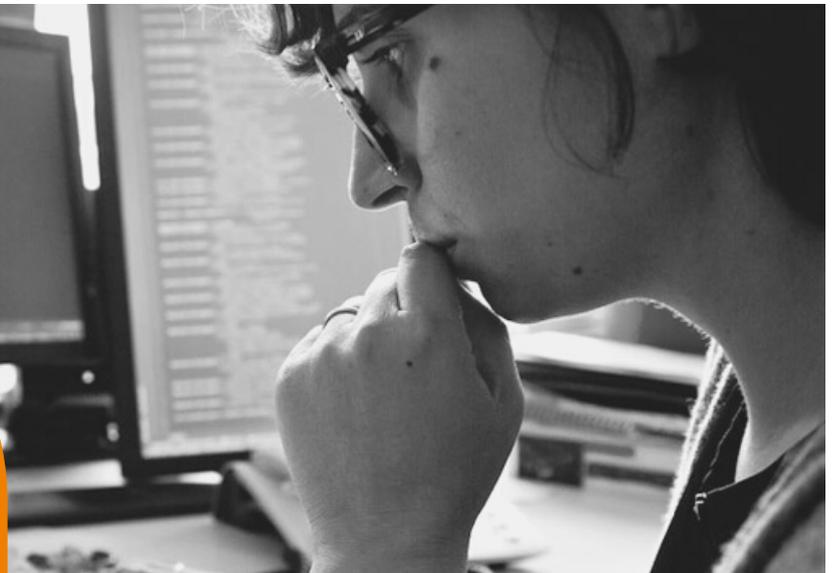
[2]

“

Beautiful is better
than ugly.
Explicit is better
than implicit.
Simple is better than
complex.

”

The Zen of Python

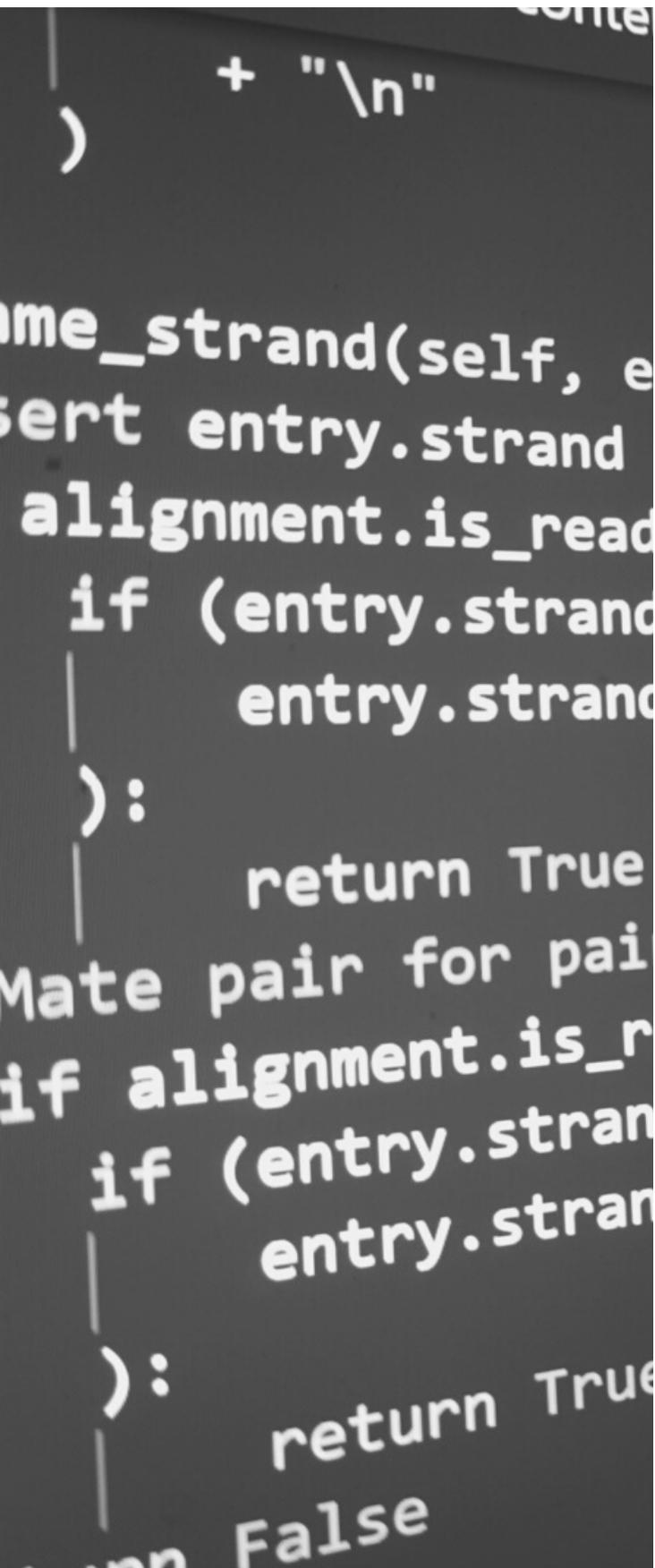


„The Zen of Python“ oder warum Python einfach zu erlernen ist

Was Python ausmacht: Im Vergleich zu anderen Programmiersprachen hat sie eine sehr einfache Syntax. Das ist auf die Philosophie von Python zurückzuführen – das Zen of Python von Tim Peters[3]: 19 Leitlinien, die das Design der Sprache maßgeblich beeinflusst haben. Demnach ist zum Beispiel explizit besser als implizit, simpel besser als komplex und die Lesbarkeit steht im Vordergrund. Das Einhalten dieser simplen Grundprinzipien führt zu klaren Regeln und einer einfach gehaltenen Schreibweise. Dadurch ist Python schneller und einfacher zu erlernen, erfasst aber gleichzeitig das Grundkonzept jeder anderen Programmiersprache. Auf dieser Basis lässt sich dementsprechend auch jede weitere Programmiersprache einfacher erlernen.

[2] Vgl. Wikipedia Artikel:
https://de.wikipedia.org/wiki/General_Purpose_Language

[3] Das komplette Zen of Python:
<https://peps.python.org/pep-0020/>



Wachstum der Community

Python gehört zu den meistgenutzten Programmiersprachen mit einer ständig wachsenden Community. Vor allem seit den 2010er Jahren hat sich Python zu einer der am schnellsten wachsenden Sprachen entwickelt.^[4]

Neben dem einfachen Erlernen und der breiten Einsatzmöglichkeit der Programmiersprache – von Data Science über Webentwicklung bis hin zur Forschung – ist der rasante Aufstieg in der Beliebtheit sicherlich auch auf die Python Community selbst zurückzuführen. Denn diese steht für einen schnellen und unkomplizierten Support: In Hilfeforen wie Stack Overflow stehen Programmierer:innen aus aller Welt mit Rat und Tat zur Seite. Darüber hinaus gibt es eine umfassende Standard-Library so wie viele weitere Libraries, auch Packages genannt, die von der Community geschrieben und allen zur Verfügung gestellt werden, um Coding-Probleme zu lösen.^[5]

Open Source und Open Community

Sämtliche Entwicklungen stehen Open Source zur Verfügung und können direkt nachgenutzt werden. In den Libraries etwa sind viele Extrafunktionen zu finden, die man direkt in die eigene Entwicklungsumgebung importieren kann. Die Python Community steht aber nicht nur für Open Science, sondern auch für Open Community und spricht sich explizit für eine tolerante, inklusive und respektvolle Community aus und heißt jede:n willkommen, unabhängig von Herkunft und Hintergrund.

[4] Vgl.: <https://stackoverflow.blog/2017/09/06/incredible-growth-python/>

[5] Eine Auswahl an Python-Libraries findet sich am Ende dieses Whitepapers.

▶ PYTHON LERNEN

Hier gelingt der Einstieg

Ob auf YouTube oder E-Learning-Plattformen: Das Angebot an kostenlosen und kostenpflichtigen Online-Kursen und -Tutorials ist riesig. Viele nützliche Informationen auch für Einsteiger:innen gibt es auf der Python-Website selbst. Unter python.org gibt es Tutorials, Anleitungen, Informationsvideos oder auch Hinweise zu Workshops und Konferenzen.

Python und weitere Datenkompetenzen für die Wissenschaft – The Carpentries

Für die Erwerbung von Programmierkenntnissen für den wissenschaftlichen Kontext sind The Carpentries eine gute Anlaufstelle: eine Non-Profit-Organisation, die hauptsächlich aus Freiwilligen besteht. Ihre Vision ist es, Daten- und Codierfähigkeiten an Forschende und Bibliothekar:innen weiterzugeben. The Carpentries unterteilen sich in drei Bereiche:

- ▶ Data Carpentry – mit Fokus auf der Vermittlung fundamentaler Datenkompetenzen, die zur Durchführung von Forschungsarbeiten erforderlich sind.
- ▶ Library Carpentry – mit Fokus auf Software- und Datenkenntnissen für Bibliotheken und informationsbezogene Einrichtungen.
- ▶ Software Carpentry – der Ursprung von The Carpentries, mit Fokus auf Lösungen für kleine Programmiertätigkeiten.



**THE CARPENTRIES -
foundational coding and
data science skills for
researchers**

Carpentry Workshops

Die zweitägigen Workshops vermitteln Basiswissen zu verschiedenen IT-Themen, zum Beispiel Programmiersprachen, Datenstrukturen und Datenverwaltung. Diese Inhalte legen ein Fundament, auf dem die Teilnehmenden im Anschluss eigenständig aufbauen können.

ZB MED ist aktiv in der Carpentries-Community. Mehrere Mitarbeitende sind zertifizierte Lehrende, sogenannte Instructor. Seit 2018 hat ZB MED über 50 Carpentry Workshops organisiert und beteiligt sich auch an der Weiterentwicklung einzelner Workshop-Bestandteile, darunter die Lessons zu Wikidata und Python.

Was benötige ich für Python?

Für Python sind keine gesonderten Voraussetzungen notwendig, da es für die meisten gängigen Betriebssysteme frei zugänglich ist.

Ein hilfreiches Tool für das Coding ist Jupyter Notebook – eine Open-Source-Anwendung, mit der man Dokumente und Code schreiben und teilen kann. Es ist ein Programm, das sich auf das sogenannte literate programming stützt, d.h., es ist möglich Text und Code miteinander zu verweben und in einem Dokument abzuspeichern. Das erleichtert die Lesbarkeit in der Anwendung erheblich.

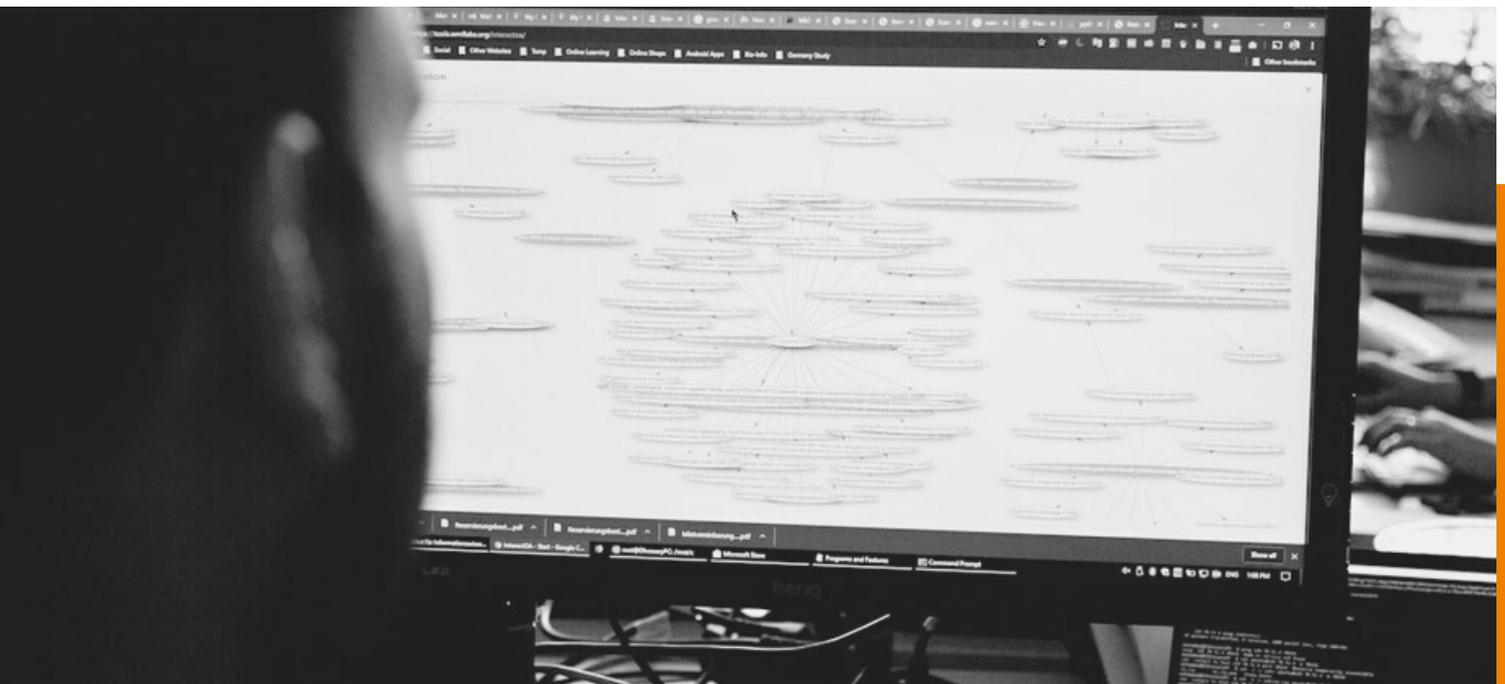
Darüber hinaus bietet der Anaconda Navigator mit der kostenlosen Anaconda Distribution, früher bekannt als Individual Edition, einen super Einstieg ins Programmieren.^[6] Der Anaconda Navigator bringt mehrere Tools für die Programmierung mit und beinhaltet u.a. Python, Jupyter Notebook, die integrierte Entwicklungsumgebung IDE (Integrated Development Environment) Spyder uvm. So muss man nicht alle Tools einzeln installieren, sondern kann über den Navigator auf mehrere Programme zugreifen.



[6] <https://www.anaconda.com/products/distribution>

► NACHSCHLAG: PYTHON, WIE DIE SCHLANGE?

Übrigens: Zwar sind im Python-Logo zwei Schlangen abgebildet, die auf alten Maya-Zeichnungen basieren, der Name der Programmiersprache hat jedoch ursprünglich nichts mit den Reptilien zu tun, sondern geht zurück auf die britische Komikertruppe Monty Python. Der niederländische Softwareentwickler Guido van Rossum, der ab 1989 die Programmiersprache entwickelte, wählte den Namen Python in Anlehnung an „Monty Python’s Flying Circus“ [7] – Anspielungen auf berühmte Python-Skette sind auch in der Dokumentation zu finden.



[7] Vgl.: <https://docs.python.org/2/faq/general.html#why-is-it-called-python>

▶ PYTHON FÜR DEN EINSTIEG

Zum Nachlesen und -hören

- [Python.org](https://python.org)
- Al Sweigart: [Automate boring stuff with Python](#) (komplett online CC-BY); Deutsche Version "Routineaufgaben mit Python automatisieren"
- Wachstum von Python: <https://stackoverflow.blog/2017/09/06/incredible-growth-python/> und <https://stackoverflow.blog/2017/09/14/python-growing-quickly/>
- [PyCoders Weekly Newsletter](#)
- [The 35 Words You Need to Python](#)
- Podcast [Command Line Heros \(en\)](#) : "Command Line Heroes tells the epic true tales of how developers, programmers, hackers, geeks, and open source rebels are revolutionizing the technology landscape.", z. B.:
 - [Python's Tale](#)
 - [Heroes in a Bash Shell](#)

Videos zum Thema auf dem ZB MED YouTube-Kanal:

- [ZB MED Lab: Die Programmiersprache Python](#)
- [ZB MED Tutorial: Screencast-Reihe zu Python](#)
- [ZB MED Lab: Wer sind die Carpentries?](#)
- [ZB MED Nachgefragt: Keine Angst vor großen Datenmengen: Data Literacy in Bibliotheken](#)

The Carpentries und Co: Anlaufstellen für Data Literacy

- <https://carpentries.org/>
- [The Carpentries bei ZB MED](#)
- [Zertifikatskurs Data Librarian an der TH Köln](#)
- [Python for Librarians: Inhalte aus den Library Carpentry Workshops](#)

▶ PYTHON ZUR VERTIEFUNG: TIPPS & TRICKS

Text-Editoren & IDEs fürs Programmieren mit Python:

- [PyCharm](#)
- [Spyder](#)
- [Emacs](#)
- [Visual Studio Code / VSCodium](#)

Hilfeforen und Libraries:

- [Stack Overflow](#)
- [pandas](#) (mit tabellarischen Daten arbeiten)

Nützliche Python-Libraries:

Libraries zur Visualisierung von Daten:

- [Seaborn](#)
- [bokeh](#)

Codeformatierung

- [black](#)

Web-Frameworks – Programmierung von Web-Anwendungen

- [Django](#)
- [flask](#)

Die wichtigsten Befehle auf einen Blick

Cheat-Sheet zu Python

- https://github.com/foerstner-lab/Bits and pieces for the carpentries workshops/blob/main/short references/Kurzreferenz_python.pdf

IMPRESSUM

Deutsche Zentralbibliothek für Medizin (ZB MED) – Informationszentrum Lebenswissenschaften

Gleueler Str. 60 50931 Köln +49 (0)221 478-5685
info@zbmed.de | www.zbmed.de

Vertretungsberechtigte Personen

Wiss. Leiter:

Prof. Dr. Dietrich Rebolz-Schuhmann

Kfm. Geschäftsführerin:

Dipl.-Volkswirtin Gabriele Herrmann-Krotz

Inhalt:

Rabea Müller, Alexandra Streck

Bildnachweis:

Titelseite, Seite 4: ZB MED

Seiten 1 und 5: Petra Kneib

Seiten 3 und 7: Eric Lichtenscheidt

DOI (Digitalausgabe)

<https://dx.doi.org/10.4126/FRL01-006433621>

Lizenz:

Die Inhalte dieses Whitepapers sind lizenziert unter einer [Creative Commons Namensnennung 4.0 International Lizenz \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/). Von dieser Lizenz ausgenommen sind die verwendeten Logos und Grafiken.

Stand: Mai 2022

Gefördert durch:



Bundesministerium
für Gesundheit

aufgrund eines Beschlusses
des Deutschen Bundestages

Ministerium für
Kultur und Wissenschaft
des Landes Nordrhein-Westfalen



JETZT ANMELDEN!

Kommende Workshops und
Trainingsangebote:

Bleiben Sie auf dem Laufenden und
abonnieren Sie unseren
Newsletter Trainings!



<https://www.zbmed.de/newsletter-trainings/>