

Wie nutze ich Python? Was muss ich installieren?

by Woche 1

Um Python Code ausführen zu können, müssen wir Python auf unserem Rechner installieren. Tatsächlich gibt es aber verschiedene Möglichkeiten, Python zu installieren und zu nutzen und wir werden nicht die Standard-Installation von nutzen, sondern eine spezielle Distribution von Python namens Anaconda, die für Data Science und Machine Learning entwickelt wurde.

Python Distributionen

Die schiere Menge an verschiedenen Python-Versionen und -Distributionen kann verwirrend sein und die Details sind für einen Einsteigerkurs wie diesen hier wohl überfordernd. Dennoch soll zumindest ein grober Überblick gegeben werden, um zu einem späteren Zeitpunkt nicht völlig unvorbereitet zu sein.

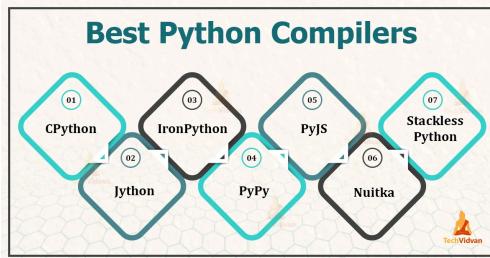
Wenn man das standardmäßige Python von <https://www.python.org/downloads/> herunterlädt und installiert, erhält man die sogenannte CPython-Distribution. Eine **Python Distribution** ist eine spezielle Version von Python, die gebündelt mit bestimmten Modulen und Werkzeugen ausgeliefert wird.

Inhalt einer Python Distribution

Folgende Dinge sind Bestandteile einer Python-Distribution:

- **Python-Interpreter**

- Das ist das Programm, das Python-Code in Maschinensprache übersetzt und ausführt. Da Python selbst ja nur die Programmiersprache ist, kann der Python-Interpreter noch am ehesten als "das eigentliche Python" im Sinne eines ausführbaren Programms bezeichnet werden. Es gibt verschiedene Python-Interpreter, die sich in ihrer Funktionsweise und Geschwindigkeit unterscheiden.
- Beispiele: CPython (Standard), Jython, IronPython, PyPy, Stackless Python, ...
- Die oben genante CPython-Distribution ist also nach ihrem Interpreter benannt, enthält aber noch mehr.



Quelle: TechVidvan

- **Code Editor bzw. IDE (Integrated Development Environment)**

- Während ein Code Editor ein Teil eines IDE sein kann, ist ein IDE viel mehr als nur ein Code Editor.
- Ein Code Editor ist das Programm, in dem wir unseren Python-Code schreiben. Der wohl offensichtlichste Vorteil eines Code Editors ist die Syntax-Hervorhebung, die es uns ermöglicht, den Code besser zu lesen und zu schreiben. Darüber hinaus mag es Features wie automatische Einrückung und Code-Vervollständigung geben. Code Editoren sind in der Regel leichtgewichtig und können schnell gestartet werden. Sie sind flexibel und können durch Plugins oder Erweiterungen angepasst werden, um zusätzliche Funktionalitäten für verschiedene Programmiersprachen zu unterstützen.
- Ein IDE ist eine umfassendere Software-Suite, die neben einem Code Editor auch Tools für Debugging, Kompilierung, Code-Ausführung, Versionierungskontrolle und manchmal sogar Datenbankverwaltung enthält. IDEs bieten eine integrierte Umgebung, die alle oder die meisten Werkzeuge bietet, die ein Entwickler zum Schreiben, Testen und Debuggen von Code benötigt. Sie sind darauf ausgerichtet, den Entwicklungsprozess zu vereinfachen und zu beschleunigen, indem sie eine breite Palette von Funktionen innerhalb einer einzigen Anwendung bieten.
- Beispiele: Jupyter Notebooks, Jupyter Labs, Visual Studio Code, PyCharm, RStudio, Spyder, ...



Quelle: Seattle Website Developer

- **Module**

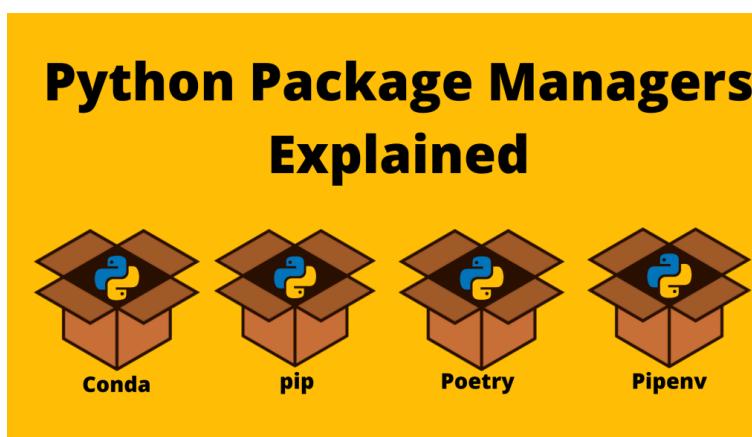
- Einige Python Distributionen enthalten bereits eine Vielzahl an Modulen, die für bestimmte Anwendungsfälle relevant sind.
- Beispiel: Die Anaconda-Distribution enthält beispielsweise Module, die für Data Science und Machine Learning relevant sind.



Quelle: Eric J.

- **Paket-Management**

- Ein Paket-Manager ist ein Tool, das es uns ermöglicht, Python-Module zu installieren, zu aktualisieren und zu entfernen.
- Beispiele: pip, conda, ...



Quelle: Crista Perlton

Wahl der Python Distribution

Wie gesagt ist CPython die offizielle Python-Distribution und wird von der Python Software Foundation entwickelt. Sie ist die Referenzimplementierung von Python und wird von den meisten Python-Entwicklern genutzt. Allerdings wird selbst auf der offiziellen Python-Website auf mehrere "Alternative Python Implementations" hingewiesen - unter anderem auch Python Anaconda.



Speziell für Data Science/Data Analytics eignet sich Anaconda Distribution, da diese speziell für die Arbeit in diesen Bereichen entwickelt wurde. Sie enthält Module, die für diese Disziplinen relevant sind und ist daher besonders für Einsteiger in Python und Data Analytics geeignet.

Wie die CPython-Distribution verwendet sie auch den CPython Interpreter. Allerdings werden wir die Jupyter Notebook IDE nutzen und brauchen demnach so gut wie keine zusätzlichen Module installieren, da diese bereits in Anaconda enthalten sind. Es sei erwähnt, dass Anaconda zusätzlich auch die Programmiersprache R enthält.

Anaconda Installieren

Abgesehen von den Anaconda-spezifischen Erweiterungen (Cloud, Fonts) ist Anaconda open-source und kann kostenlos von der Anaconda-Website heruntergeladen und installiert werden, wie im folgenden gezeigt:

https://www.youtube.com/embed/0kTRzNjNIE0?si=Z_kKkdx9fh9UJzRT

💡 Weitere Ressourcen

- Python vs. Anaconda - Was ist besser?