

# numpy

Den Umgang mit numpy lernen

# Das Python Ökosystem ist riesig

---

- Python wird bereits mit vielen nützlichen Tools und Bibliotheken ausgeliefert
- Nichtsdestoweniger gibt es auch Tausende von Modulen und Bibliotheken von Drittanbietern, die für verschiedene Aufgaben verwendet werden können. NumPy und Pandas sind nur zwei davon
- <https://awesome-python.com/>

# IPython: Interactive Python

---

- Interaktive Datenverarbeitung in Python
- Bietet Introspektion: Wir können Werte und Fehler untersuchen, unsere Funktionen zeitlich festlegen und vieles mehr
- Bietet Tab-Vervollständigung und Verlauf
- Bietet eine browserbasierte Notizbuchschnittstelle mit Unterstützung für Code, Text, mathematische Ausdrücke und mehr (heutzutage Jupyter genannt)
- Ein Notizbuch führt Python / IPython-Anweisungen aus

# Hilfe und Dokumentation in IPython

---

- Wie rufe ich eine Funktion auf? Welche Argumente und Optionen hat sie?
- Wie sieht der Quellcode dieses Python-Wertes/Objektes aus?
- Was befindet sich in dem von mir importierten Paket?
- Welche Variablen/Attribute oder Methoden hat dieser Wert/dieses Objekt?

# Hilfe und Dokumentation in IPython

Wir können die Dokumentation mit ? aufrufen

```
In [1]: print?
```

IPYTHON

```
Docstring:
```

```
print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)
```

```
Prints the values to a stream, or to sys.stdout by default.
```

Diese Notation funktioniert für fast alles, einschliesslich Objektmethoden und -funktionen (wie wir später sehen werden)

# Hilfe und Dokumentation in IPython

Der Sourcecode kann mit ?? aufgerufen werden:

```
In [1]: def myfun(lst):  
...:     for e in lst:  
...:         print(e)  
...:
```

IPYTHON

```
In [2]: myfun??
```

Signature: myfun(lst)

Docstring: <no docstring>

Source:

```
def myfun(lst):  
    for e in lst:  
        print(e)
```

File: ~/<ipython-input-9-42be41fecbd8>

Type: function

# Shell Commands in IPython

---

- Die Shell ist eine Möglichkeit zur textuellen Interaktion mit dem Computer
- Betriebssysteme gab es lange vor den grafischen Benutzeroberflächen, wie wir sie heute kennen und verwenden
- Mit einer Shell können wir Ordner und Dateien erstellen, sie kopieren und löschen und vieles mehr - im Grunde können wir dem Computer über die Shell Befehle erteilen

# Shell Commands in IPython

- Allgemeine Shell-Befehle
  - pwd: Gibt das Arbeitsverzeichnis aus (wo wir uns gerade im Dateisystem befinden)
  - ls: Inhalt des Arbeitsverzeichnisses auflisten
  - cd: Verzeichnis wechseln
  - mkdir: Neues Verzeichnis erstellen
  - In IPython können wir diese Shell-Befehle verwenden, indem wir ihnen das Präfix ! geben



# Externen Code mit %run ausführen

Wir können einen Texteditor verwenden, um Code zu schreiben und IPython verwenden, um ihn mit %run auszuführen

my\_print.py

```
def fun(lst):  
    for e in lst:  
        print(e)  
  
fun([1,2,3,4])
```

```
In [1]: %run my_print.py  IPYTHON  
1  
2  
3  
4
```

# Ende

Das war alles für dieses Kapitel

---