# Einführung in Python

* ~~Python: Unterschied zu Java / C#.NET / C++ / PHP~~
* Kurzübersicht über die Editoren -> Visual Studio Code (VSC)
* Die Entwicklungsumgebung -> Anaconda + VSC
* **Einrichten eines Python-Projekts**
* **Grundsätzlicher Aufbau eines Python-Programms -> Script, CLI, Package, Notebook**

# Syntax von Python

* Die Help-Funktion -> und offizielle Doku!
* Kommentare
* ~~Allgemeine~~ Einfache Datentypen -> int, float, Decimal, bool, string
* **Komplexe Datentypen -> list, dictionary, set**
* Konvertierungen zwischen Datentypen
* Logische und arithmetische Operatoren
* Verzweigungen und Schleifen -> if…elif…else, for, while
* ~~Formatierte Ausgaben mit format~~() -> f-Strings
* **Funktionen, Parameterübergabe**
* Die pass-Anweisung
* Lambda-Expressions

# Übungen zur Programmlogik

* Viele ausgewählte Übungen
* Schleifen: for und while
* Die range-Funktion
* Bedingungen: if, elif, else
* Übungen zu verbesserter Performance
* Viele Übungen: eigene Klassen schreiben

# Testen, Testen, Testen

* **pytest**

# Objektorientierte Programmierung

* Einstieg in objektorientierte Programmierung
* Datenkapselung, Vererbung, Polymorphie
* Klassen und Objekte
* Die Klasse Object
* ~~Mehrfachvererbung~~
* Überschreiben von Methoden der Basisklasse

# Exceptionhandling in Python

* Die Klasse Exception
* try / except / finally
* Erstellen einer eigenen Fehlerklasse

# Weitere optionale Themen

* Dateien lesen und schreiben
* ~~Operatorüberladung~~
* Reguläre Ausdrücke
* Arbeiten mit Datenbanken
* Arbeiten mit Web-Requests/REST-API
* Visualisierungen -> matplotlib/seaborn/plotly
* Wissenschaftliches Arbeiten mit Python -> numpy/scipy
* Tabellarische Daten verarbeiten mit Python -> pandas
* Machine Learning -> scikit-learn
* Webanwendungen -> Flask/fastapi