

EINFÜHRUNG IN DIE PROGRAMMIERUNG

COMMAND LINE & PYTHON BASICS

DHBW MANNHEIM

WIRTSCHAFTSINFORMATIK (DATA SCIENCE)

Markus Menth

Martin Gropp

AGENDA

- Kontrolle der Installationen
- Kommandozeile
 - Grundlagen
 - Befehle
- Python

KONTROLLE DER INSTALLATIONEN

PYTHON

- Kommandozeile: `python --version`
- Download Python

ENTWICKLUNGSUMGEBUNG (IDE)

- PyCharm
- Download PyCharm

KOMMANDOZEILE

Namen:

- Terminal
- Konsole
- Befehlszeile

Die Kommandozeile ist ein textbasiertes Interface innerhalb des Betriebssystems. Funktionen:

- Organisation von Dateien
- Starten von Programmen
- Navigation in Verzeichnissen
- etc. ...

KOMMANDOZEILE

BEISPIELE

Windows:

- cmd 🖱️
- PowerShell

Unix-artige (Linux, macOS, ...):

- bash
- zsh
- fish
- ...

KOMMANDOZEILEN-BEFEHLE

Öffnen Sie *cmd*. Rufen Sie *help* auf und beantworten Sie folgende Fragen:

Wie kann man

- in ein Verzeichnis navigieren?
- zu einem übergeordneten Verzeichnis zurück gehen?
- die Dateien in einem Verzeichnis auflisten?
- Dateien verschieben?
- ein Programm starten?
- eine Datei kopieren?
- die Farbe ändern?

Hinweis: Mehr Informationen zu den Befehlen bekommen Sie mit *help* *⟨Befehl⟩*.

Beispiel: *help color*

PYTHON AUSFÜHREN: REPL

Python-Interpreter starten: `python`

(read, evaluate, print results, loop back to step 1)

Führen Sie die folgenden Befehle aus:

- `print("Hello World")`
- `input("Jetzt können Sie etwas eintippen: ")`
- `1 / 2`
- `5 * 5`

Beenden Sie das Programm mit `quit()` oder mit Strg+D (Unix).

PYTHON AUSFÜHREN: DATEI

AUFGABE

Erstellen Sie eine Datei `Hello_World.py` auf Ihrem Desktop mit einigen Zeilen Python-Code.

Rufen Sie die Datei über die Kommandozeile auf: `python <Datei>`

WAS KANN SCHIEFGEHEN?

SYNTAXFEHLER

- Leicht zu beheben: sie widersprechen der Grammatik der Sprache
- Compiler/Interpreter geben die Stelle im Quellcode an, wo der Fehler sich befinden *könnte*
- Beispiel: `printt("Hallo Welt")`

```
>>> printt("Hallo")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'printt' is not defined
```

SEMANTIKFEHLER

- Das Programm verhält sich nicht so wie es sollte.

WAS KANN SCHIEFGEHEN?

LAUFZEITFEHLER

- Zum Beispiel könnte eine falsche Eingabe zur Laufzeit zu unerwartetem Verhalten führen

```
>>> a = int(input("Geben Sie eine Zahl ein: "))
Geben Sie eine Zahl ein: 0
>>> 1 / a
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ZeroDivisionError: division by zero
```

KOMMENTARE

Führen Sie das folgende Programm aus:

```
# print('Hello World')
```

KOMMENTARE

```
# Diese Zeile ist ein Kommentar
```

Kommentare werden vom Computer ignoriert.

WARUM KOMMENTARE?

- Dokumentieren, wie ein Programmteil funktioniert!
- Hinweise auf zu erledigende Arbeit (*TODOs*)
- Autor, Copyright, Lizenz, ...
- ...