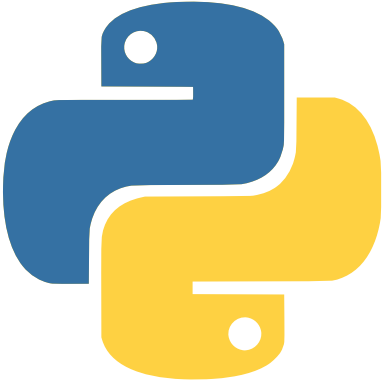
Universität Stuttgart
Projekt digit@L – BOOST. SKILLS. SUPPORT.



Programmierkurs Python

Programmiersprachen

Dominik Göddeke

IANS SimTech

Stiftung Innovation in der Hochschullehre

CC BY

Programmiersprachen

Programmiersprachen

- (Weitere) Abstraktion von der konkreten Hardware: **höhere Programmiersprachen**

Programmiersprachen

- (Weitere) Abstraktion von der konkreten Hardware: **höhere Programmiersprachen**
- Erlauben lesbare und verständliche Formulierung von Programmen

Programmiersprachen

- (Weitere) Abstraktion von der konkreten Hardware: **höhere Programmiersprachen**
- Erlauben lesbare und verständliche Formulierung von Programmen
- In Form von sogenanntem **Quelltext (source code)**

Programmiersprachen

- (Weitere) Abstraktion von der konkreten Hardware: **höhere Programmiersprachen**
- Erlauben lesbare und verständliche Formulierung von Programmen
- In Form von sogenanntem **Quelltext (source code)**
- Textdatei mit einer speziellen Dateiendung, für Python `.py`

Programmiersprachen

- (Weitere) Abstraktion von der konkreten Hardware: **höhere Programmiersprachen**
- Erlauben lesbare und verständliche Formulierung von Programmen
- In Form von sogenanntem **Quelltext (source code)**
- Textdatei mit einer speziellen Dateiendung, für Python `.py`
- Meist angelehnt an die englische Sprache

Beispiel in gutem Stil

Absichtlich noch nicht komplett verständlich

Vom Quelltext zur Ausführung des Programms

Programmiersprachen

- Effiziente Lösung: **kompilierte Programme**

Programmiersprachen

- Effiziente Lösung: **kompilierte Programme**
  - Computer übersetzt vorab Quelltext in ausführbares Programm („.exe-Datei“)
  - Hier nicht weiter betrachtet, stark Hardware-abhängig

Programmiersprachen

- Effiziente Lösung: **kompilierte Programme**
  - Computer übersetzt vorab Quelltext in ausführbares Programm („.exe-Datei“)
  - Hier nicht weiter betrachtet, stark Hardware-abhängig
- Flexible Lösung: **interpretierte Programme**

Programmiersprachen

- Effiziente Lösung: **kompilierte Programme**
  - Computer übersetzt vorab Quelltext in ausführbares Programm („.exe-Datei“)
  - Hier nicht weiter betrachtet, stark Hardware-abhängig
- Flexible Lösung: **interpretierte Programme**
  - Interpeter marschiert zur Laufzeit durch Quelltext, führt gegebene Befehle aus

Programmiersprachen

- Effiziente Lösung: **kompilierte Programme**
  - Computer übersetzt vorab Quelltext in ausführbares Programm („.exe-Datei“)
  - Hier nicht weiter betrachtet, stark Hardware-abhängig
- Flexible Lösung: **interpretierte Programme**
  - Interpeter marschiert zur Laufzeit durch Quelltext, führt gegebene Befehle aus
  - Python ist eine interpretierte Sprache

Programmiersprachen

- Effiziente Lösung: **kompilierte Programme**
  - Computer übersetzt vorab Quelltext in ausführbares Programm („.exe-Datei“)
  - Hier nicht weiter betrachtet, stark Hardware-abhängig
- Flexible Lösung: **interpretierte Programme**
  - Interpeter marschiert zur Laufzeit durch Quelltext, führt gegebene Befehle aus
  - Python ist eine interpretierte Sprache
  - Oft etwas langsamer als kompilierte Sprachen, aber deutlich abstrakter und unabhängiger von Hardware-Details

Noch ein Beispiel in Jupyter

Zusammenfassung

Programmiersprachen

- Code ist immer identisch

Programmiersprachen

- Code ist immer identisch
- In der Python-Konsole

Programmiersprachen

- Code ist immer identisch
- In der Python-Konsole
- In Jupyter-Notebooks

Programmiersprachen

- Code ist immer identisch
- In der Python-Konsole
- In Jupyter-Notebooks
- In IDEs

Programmiersprachen

- Code ist immer identisch
- In der Python-Konsole
- In Jupyter-Notebooks
- In IDEs
- . . .

Impressum, Danksagung und Quellen

Stiftung Innovation in der Hochschullehre

Gefördert durch die Stiftung Innovation in der Hochschullehre im Rahmen des Projekts digit@L, https://stiftung-hochschullehre.de

SimTech

Gefördert mit Mitteln der Deutschen Forschungsgemeinschaft (EXC 2075 - 390740016) im Rahmen der Exzellenzstrategie

---

Autor: Dominik Göddeke, IANS, Universität Stuttgart

CC BY

Weitere Quellen:

- Logos Universität Stuttgart, IANS, SimTech: Universität Stuttgart, alle Rechte vorbehalten
- Logo Python: https://freesvg.org/387, CC-0
- Logo Stiftung: Stiftung Innovation in der Hochschullehre, alle Rechte vorbehalten
- Logo ZOERR: Universität Tübingen, alle Rechte vorbehalten

ZOERR

Veröffentlicht auf dem Zentralen OER Repositorium Baden-Württemberg, https://www.zoerr.de