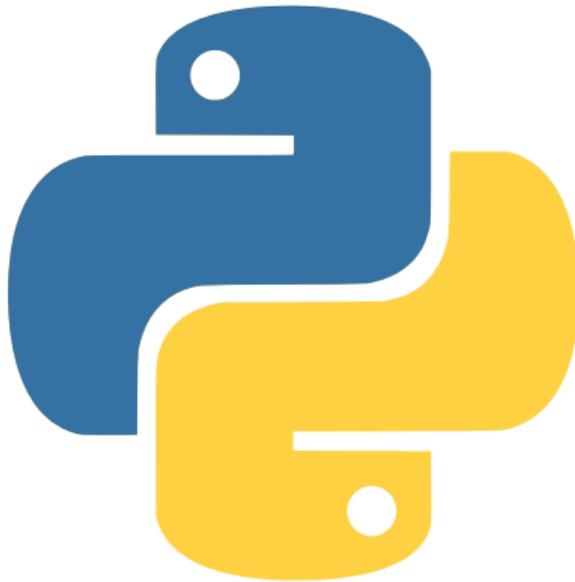




**Universität Stuttgart**

Projekt digit@L – BOOST. SKILLS. SUPPORT.



Dominik  
Göddeke

# Programmierkurs Python

Beispiel: csv-Dateien

# Beispiel: csv-Dateien

# Beispiel: csv-Dateien

- **csv-Dateien (Comma-Separated Values)**: einfache Textdateien mit „standardisiertem“ Format

# Beispiel: csv-Dateien

- **csv-Dateien (Comma-Separated Values)**: einfache Textdateien mit „standardisiertem“ Format
- **Minimalstandard für Tabellendaten**, auch in Excel unterstützt

# Beispiel: csv-Dateien

- **csv-Dateien (Comma-Separated Values)**: einfache Textdateien mit „standardisiertem“ Format
- **Minimalstandard für Tabellendaten**, auch in Excel unterstützt
- Beispieldaten in einer Tabelle und als csv-Datei:

col1	col2	col3	col1,col2,col3
a	1	a	a,1,a
b	2	2	b,2,2
c	3.5	Ich bin auch da	c,3.5,Ich bin auch da
d	6	3	d,6,3
e	-2	4	e,-2,4

# Beispiel: csv-Dateien

- **csv-Dateien (Comma-Separated Values)**: einfache Textdateien mit „standardisiertem“ Format
- **Minimalstandard für Tabellendaten**, auch in Excel unterstützt
- Beispieldaten in einer Tabelle und als csv-Datei:

col1	col2	col3	col1,col2,col3
a	1	a	a,1,a
b	2	2	b,2,2
c	3.5	Ich bin auch da	c,3.5,Ich bin auch da
d	6	3	d,6,3
e	-2	4	e,-2,4

- **Trennzeichen** ', ' nicht standardisiert, oft auch '\t' oder ';'.

# Beispiel: csv-Dateien

- **csv-Dateien (Comma-Separated Values)**: einfache Textdateien mit „standardisiertem“ Format
- **Minimalstandard für Tabellendaten**, auch in Excel unterstützt
- Beispieldaten in einer Tabelle und als csv-Datei:

col1	col2	col3	col1,col2,col3
a	1	a	a,1,a
b	2	2	b,2,2
c	3.5	Ich bin auch da	c,3.5,Ich bin auch da
d	6	3	d,6,3
e	-2	4	e,-2,4

- **Trennzeichen** ', ' nicht standardisiert, oft auch '\t' oder ';'.
- **Optional**: Maskierung der Felder mit Anführungszeichen

# Codebeispiel: Wertetabelle als csv-Datei

# Diskussion

# Diskussion

- Einlesen mit bekannten **Python-Bordmitteln**

# Diskussion

- Einlesen mit bekannten **Python-Bordmitteln**
  - Öffnen der Datei mit `open()`

# Diskussion

- Einlesen mit bekannten **Python-Bordmitteln**
  - Öffnen der Datei mit `open()`
  - Eventuelles Überspringen mit `readline()` der Header-Zeile

# Diskussion

- Einlesen mit bekannten **Python-Bordmitteln**
  - Öffnen der Datei mit `open()`
  - Eventuelles Überspringen mit `readline()` der Header-Zeile
  - Iteration mit `for` Schleife zeilenweise über die Datei

# Diskussion

- Einlesen mit bekannten **Python-Bordmitteln**
  - Öffnen der Datei mit `open()`
  - Eventuelles Überspringen mit `readline()` der Header-Zeile
  - Iteration mit `for` Schleife zeilenweise über die Datei
  - Nutzung von String-Funktionen wie `line.split(',')` zur Analyse der gelesenen Zeile und ggf. Typumwandlungen

# Diskussion

- Einlesen mit bekannten **Python-Bordmitteln**
  - Öffnen der Datei mit `open()`
  - Eventuelles Überspringen mit `readline()` der Header-Zeile
  - Iteration mit `for` Schleife zeilenweise über die Datei
  - Nutzung von String-Funktionen wie `line.split(',')` zur Analyse der gelesenen Zeile und ggf. Typumwandlungen
  - Speicherung der Ergebnisse in Listen oder anderen Variablen

# Diskussion

- Einlesen mit bekannten **Python-Bordmitteln**
  - Öffnen der Datei mit `open()`
  - Eventuelles Überspringen mit `readline()` der Header-Zeile
  - Iteration mit `for` Schleife zeilenweise über die Datei
  - Nutzung von String-Funktionen wie `line.split(',')` zur Analyse der gelesenen Zeile und ggf. Typumwandlungen
  - Speicherung der Ergebnisse in Listen oder anderen Variablen
- Schöne **Übungsaufgabe** zur Wiederholung von String-Funktionen und Schleifen

# Das Modul csv

- Nett: müssen wir gar nicht von Hand programmieren

# Das Modul csv

- Nett: müssen wir gar nicht von Hand programmieren
- Modul `csv` aus der Standardbibliothek bietet **volle Funktionalität**

# Das Modul csv

- Nett: müssen wir gar nicht von Hand programmieren
- Modul `csv` aus der Standardbibliothek bietet **volle Funktionalität**
- Inhalt zeilenweise als Liste von Strings

# So geht das im Code

# 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>

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

---

Autor: Dominik Göddeke, IANS, Universität Stuttgart



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



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