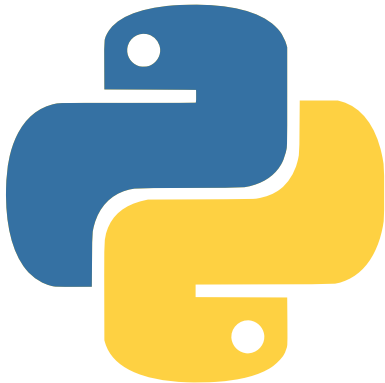




Universität Stuttgart

Projekt digit@L – BOOST. SKILLS. SUPPORT.



Dominik
Göddeke

Programmierkurs Python

Zeichenketten

Zeichenketten

Zeichenketten

- Bereits bekannt: **Zeichenketten** (Strings, in Python `str`)

Zeichenketten

- Bereits bekannt: **Zeichenketten** (Strings, in Python `str`)
- Verhalten größtenteils wie bei Listen

Zeichenketten

- Bereits bekannt: **Zeichenketten** (Strings, in Python `str`)
- Verhalten größtenteils wie bei Listen
 - Zugriff auf Teilstrings

Zeichenketten

- Bereits bekannt: **Zeichenketten** (Strings, in Python `str`)
- Verhalten größtenteils wie bei Listen
 - Zugriff auf Teilstrings
 - Iteration mittels `for` Schleife

Zeichenketten

- Bereits bekannt: **Zeichenketten** (Strings, in Python `str`)
- Verhalten größtenteils wie bei Listen
 - Zugriff auf Teilstrings
 - Iteration mittels `for` Schleife
 - Logische Funktionen

Codebeispiele

Nützliche String-Funktionen

Nützliche String-Funktionen

- `len(s)`: Länge der Zeichenkette `s`

Nützliche String-Funktionen

- `len(s)`: Länge der Zeichenkette `s`
- `s.count(t)`: Anzahl der Vorkommen des Teilstrings `t` in `s`

Nützliche String-Funktionen

- `len(s)`: Länge der Zeichenkette `s`
- `s.count(t)`: Anzahl der Vorkommen des Teilstrings `t` in `s`
- `s.join(list)`: Konvertierung der Liste `list` in eine Zeichenkette mit Separator `s`

Nützliche String-Funktionen

- `len(s)`: Länge der Zeichenkette `s`
- `s.count(t)`: Anzahl der Vorkommen des Teilstrings `t` in `s`
- `s.join(list)`: Konvertierung der Liste `list` in eine Zeichenkette mit Separator `s`
- `list = s.split(t)`: Gegenstück, Konvertierung von `s` in eine Liste am Separator `t`

Nützliche String-Funktionen

- `len(s)`: Länge der Zeichenkette `s`
- `s.count(t)`: Anzahl der Vorkommen des Teilstrings `t` in `s`
- `s.join(list)`: Konvertierung der Liste `list` in eine Zeichenkette mit Separator `s`
- `list = s.split(t)`: Gegenstück, Konvertierung von `s` in eine Liste am Separator `t`
- `s.replace(s1,s2)`: Ersetzen des Teilstrings `s1` durch `s2` in `s`

Nützliche String-Funktionen

- `len(s)`: Länge der Zeichenkette `s`
- `s.count(t)`: Anzahl der Vorkommen des Teilstrings `t` in `s`
- `s.join(list)`: Konvertierung der Liste `list` in eine Zeichenkette mit Separator `s`
- `list = s.split(t)`: Gegenstück, Konvertierung von `s` in eine Liste am Separator `t`
- `s.replace(s1,s2)`: Ersetzen des Teilstrings `s1` durch `s2` in `s`
- `s.strip()`: Entfernen von Leerzeichen und anderen Dingen in `s`

Nützliche String-Funktionen

- `len(s)`: Länge der Zeichenkette `s`
- `s.count(t)`: Anzahl der Vorkommen des Teilstrings `t` in `s`
- `s.join(list)`: Konvertierung der Liste `list` in eine Zeichenkette mit Separator `s`
- `list = s.split(t)`: Gegenstück, Konvertierung von `s` in eine Liste am Separator `t`
- `s.replace(s1,s2)`: Ersetzen des Teilstrings `s1` durch `s2` in `s`
- `s.strip()`: Entfernen von Leerzeichen und anderen Dingen in `s`
- `s.lstrip()`, `s.rstrip()`: analog

Codebeispiele

Besonderheit bei Strings

Besonderheit bei Strings

- Bei Listen: Überschreiben einzelner Elemente mit `s[i] = a`

Besonderheit bei Strings

- Bei Listen: Überschreiben einzelner Elemente mit `s[i] = a`
- Funktioniert **nicht** bei Strings

Besonderheit bei Strings

- Bei Listen: Überschreiben einzelner Elemente mit `s[i] = a`
- Funktioniert **nicht** bei Strings
- Grund: `str` sogenannter **immutable** (d.h. unveränderbarer) Datentyp

Besonderheit bei Strings

- Bei Listen: Überschreiben einzelner Elemente mit `s[i] = a`
- Funktioniert **nicht** bei Strings
- Grund: `str` sogenannter **immutable** (d.h. unveränderbarer) Datentyp
- Änderungen erfordern immer **Neuanlegen** eines Strings

Besonderheit bei Strings

- Bei Listen: Überschreiben einzelner Elemente mit `s[i] = a`
- Funktioniert **nicht** bei Strings
- Grund: `str` sogenannter **immutable** (d.h. unveränderbarer) Datentyp
- Änderungen erfordern immer **Neuanlegen** eines Strings
- Mehr dazu später

Codebeispiele

Ausführliches Beispiel

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>