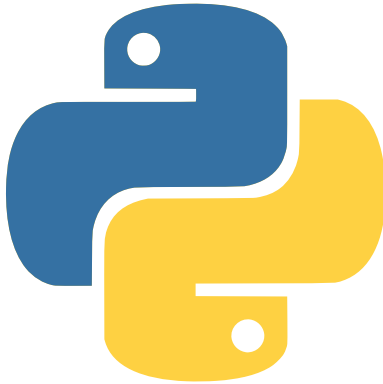




Universität Stuttgart

Projekt digit@L – BOOST. SKILLS. SUPPORT.



Dominik
Göddeke

Programmierkurs Python

Datenstrukturen,
Comprehensions und
Funktionen:
Mengen

Mengen

Mengen

- **Listen:** Elemente können beliebig oft vorkommen

Mengen

- **Listen**: Elemente können beliebig oft vorkommen
- **Dictionaries**: Keys sind eindeutig, Values können beliebig oft vorkommen

Mengen

- **Listen**: Elemente können beliebig oft vorkommen
- **Dictionaries**: Keys sind eindeutig, Values können beliebig oft vorkommen
- Oft nötig: **Datenstrukturen mit eindeutigen Elementen**

Mengen

- **Listen**: Elemente können beliebig oft vorkommen
- **Dictionaries**: Keys sind eindeutig, Values können beliebig oft vorkommen
- Oft nötig: **Datenstrukturen mit eindeutigen Elementen**
- Mathematisch: Mengen

Mengen

- **Listen**: Elemente können beliebig oft vorkommen
- **Dictionaries**: Keys sind eindeutig, Values können beliebig oft vorkommen
- Oft nötig: **Datenstrukturen mit eindeutigen Elementen**
- Mathematisch: Mengen
- In Python: `set`

Mengen

- **Listen**: Elemente können beliebig oft vorkommen
- **Dictionaries**: Keys sind eindeutig, Values können beliebig oft vorkommen
- Oft nötig: **Datenstrukturen mit eindeutigen Elementen**
- Mathematisch: Mengen
- In Python: `set`
- Aha-Effekt: Folie hilft bei der **Wahl einer Datenstruktur für ein Problem**

Mengen

- **Listen**: Elemente können beliebig oft vorkommen
- **Dictionaries**: Keys sind eindeutig, Values können beliebig oft vorkommen
- Oft nötig: **Datenstrukturen mit eindeutigen Elementen**
- Mathematisch: Mengen
- In Python: `set`
- Aha-Effekt: Folie hilft bei der **Wahl einer Datenstruktur für ein Problem**
- Was nimmt Python uns ab, was müssen wir nicht händisch programmieren?

Codebeispiele

Operationen mit Mengen

Operationen mit Mengen

- Inklusionstest

Operationen mit Mengen

- Inklusionstest
- Vereinigung

Operationen mit Mengen

- Inklusionstest
- Vereinigung
- Schnitt

Operationen mit Mengen

- Inklusionstest
- Vereinigung
- Schnitt
- Exklusion

Operationen mit Mengen

- Inklusionstest
- Vereinigung
- Schnitt
- Exklusion
- Hinzufügen und Löschen von Elementen

Codebeispiele

Ausführliches Beispiel: Punkte innerhalb eines gewissen Radius

Ausführliches Beispiel

- Gegeben: Menge von Zahlen $X = \{x_1, \dots, x_n\} \subset \mathbb{R}$

Ausführliches Beispiel

- Gegeben: Menge von Zahlen $X = \{x_1, \dots, x_n\} \subset \mathbb{R}$
- Gesucht: alle Zahlen in dieser Menge, die einen Abstand kleiner als Δ von einem gegebenen $x_0 \in \mathbb{R}$ haben

Ausführliches Beispiel

- Gegeben: Menge von Zahlen $X = \{x_1, \dots, x_n\} \subset \mathbb{R}$
- Gesucht: alle Zahlen in dieser Menge, die einen Abstand kleiner als Δ von einem gegebenen $x_0 \in \mathbb{R}$ haben
- Mathematisch formuliert: gesucht ist die Menge $Y = \{x_i \in X \mid |x_i - x_0| < \Delta\}$

So geht das in Python

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>