

# Datentyp set

Mengen in Python: veränderlich, ungeordnet

# Der Datentyp set

Der Datentyp set ist ein **ungeordneter, veränderlicher** Datentyp.

Die Elemente sind also nicht in einer gewissen Reihenfolge gespeichert und **können nicht per Index** oder **Key** adressiert werden.

Das Set darf **nur Elemente beinhalten, die unveränderlich sind**, z.B. Zahlen oder Strings.

Eine Menge ist eindeutig. Deshalb kann der Datentyp set keine doppelten Elemente beinhalten.

# Ein Set erstellen

als Set-Literal:

```
main_set = {1, 2, 3, 3}
```

oder mit der Konstruktorfunktion `set`

```
main_set = set([1, 2, 3, 3])
```

Alle doppelten Elemente werden beim Erstellen eines Sets gelöscht.

```
print(main_set)
```

```
{1, 2, 3}
```

# Hinzufügen von neuen Elementen

Mit der Methode `add` lässt sich ein neues Element ins Set eintragen

```
main_set.add(4) # Fügt 4 zu main_set hinzu
```

```
print(main_set)
```

```
{1, 2, 3, 4}
```

# Entfernen von Elementen aus dem set

Elemente können mit den Methoden `remove()` oder `discard()` entfernt werden.

# Entfernt 2, wirft KeyError, wenn 2 nicht vorhanden ist

`mein_set.remove(2)`

# Entfernt 3, wirft keinen Fehler, wenn 3 nicht vorhanden ist

`mein_set.discard(3)`

# Membership

Mit dem Membership-Operator **IN** kann geprüft werden, ob ein Element im **Set** vorhanden ist.

```
if 2 in main_set:  
    print("2 ist im Set")
```

# Iteration über ein Set

Über ein Set kann iteriert werden:

```
for element in main_set:  
    print(element)
```

Da ein Set allerdings keine Ordnung kennt, ist nicht garantiert, dass die Elemente bei der Ausgabe eine bestimmte Ordnung haben.

# Mengenoperationen

Set unterstützt die typischen Mengen-Operation: Union, Intersection, Difference, symmetric difference

$a = \{1, 2, 3\}$

$b = \{3, 4, 5\}$

## # Vereinigung (Union)

$c = a \mid b \# \{1, 2, 3, 4, 5\}$

## # Schnittmenge (Intersection)

$d = a \& b \# \{3\}$

## # Differenz (Differenz)

$e = a - b \# \{1, 2\}$

## # Symmetrische Differenz

$f = a \wedge b \# \{1, 2, 4, 5\}$



# Subset und Superset

Um zu prüfen, ob ein Set **a** ein Subset (Untermenge) von Set **b** ist:

**a** = {"a", "d"}

**b** = {"a", "d", "e"}

**a** < **b** => Falls true, ist **a** ein echtes Subset von **b**

**a** <= **b** => Falls true, ist **a** ein Subset von **b**

Eine echte Teilmenge ist eine Menge, wenn **a** != **b** und **a** < **b**. Das heisst, wenn nicht alle Elemente von **b** in **a** enthalten sind.