

Dictionary

Eigenschaften

- Menge von Werten verschiedener Datentypen
- Werte können mehrfach vorkommen
- Die Werte sind indexiert
- Die Elemente sind nicht geordnet
- Das Dictionary kann verändert werden

Etwas genauer

- Jeder Eintrag besteht aus einem Wertepaar, bestehend aus einem Schlüssel und einem Wert (key / value). Der Schlüssel wird als Index verwendet.
- Der Schlüssel muss eindeutig sein und kann aus einem String, einer Zahl oder einem Tupel bestehen.
- Als Wert kann jeder beliebige Datentyp eingesetzt werden.

Ein Dictionary erzeugen

- Direkte Zuweisung
- Leeres Dictionary erzeugen
- Aus einem anderen Dictionary erzeugen

Direkte Zuweisung

```
dict1 = {  
    "land": "Schweiz",  
    "ort": "Bern",  
    "einwohner": 140000  
}
```

Leeres Dictionary erzeugen

```
dict1 = dict()
```

Aus einem anderen Dictionary erzeugen

```
dict2 = dict1.copy()  
dict2 = dict(dict1)
```

Aus einer Keyliste (Tupel) erzeugen

```
dict1 = dict.fromkeys(("land", "ort"))  
dict1 = dict.fromkeys(("land", "ort"), "default")
```

Alle Einträge werden mit dem angegebenen Default-Wert initialisiert. Ist keiner angegeben, werden die Werte auf **None** gesetzt.

Dictionaries untersuchen und vergleichen

- Einzelnen Wert auslesen
- Existiert ein Schlüssel
- Elemente Auflisten
- Anzahl Elemente
- Anzahl eines Wertes
- Index eines Elementes auslesen
- Teilbereiche auslesen

Einzelnen Wert auslesen

```
wert = dict1["ort"]
```

Falls der Schlüssel nicht existiert, erscheint eine Fehlermeldung.

```
wert = dict1.get("ort")
```

Falls der Schlüssel nicht existiert, wird **None** zurückgegeben.

```
wert = dict1.get("irgendwas", "xx")
```

Falls der Schlüssel nicht existiert, wird **"xx"** zurückgegeben.

Existiert ein Schlüssel

```
if "ort" in dict1:
```

Schlüssel auflisten

```
for x in dict1:  
    print(x)
```

```
for x in dict1.keys():  
    print(x)
```

Werte auflisten

```
for x in dict1.values():  
    print(x)
```

Wertepaare auflisten

```
for k,v in dict1.items():  
    print(k,v)
```

Anzahl Wertepaare

```
anzahl = len(dict1)
```

Ein Dictionary bearbeiten

Das Dictionary kennt vielfältige Bearbeitungsmöglichkeiten.

Das Dictionary vollständig löschen

```
del dict1
```

Alle Elemente löschen

```
dict1.clear()
```

Einzelne Elemente löschen

```
list1.pop("ort")
```

Der Index muss existieren

```
list1.popitem()
```

Das zuletzt hinzugefügte Wertepaar wird gelöscht.

Wertepaare hinzufügen oder ändern

Die Schlüssel können nicht geändert werden!

```
dict2["kanton"] = "BS"
```

```
dict2["ort"] = "Basel"
```

```
dict2.update({"kanton": "BE", "ort": "Bern"})
```

Falls der Schlüssel bereits existiert, wird der Wert geändert, sonst wird ein neues Wertepaar angelegt.