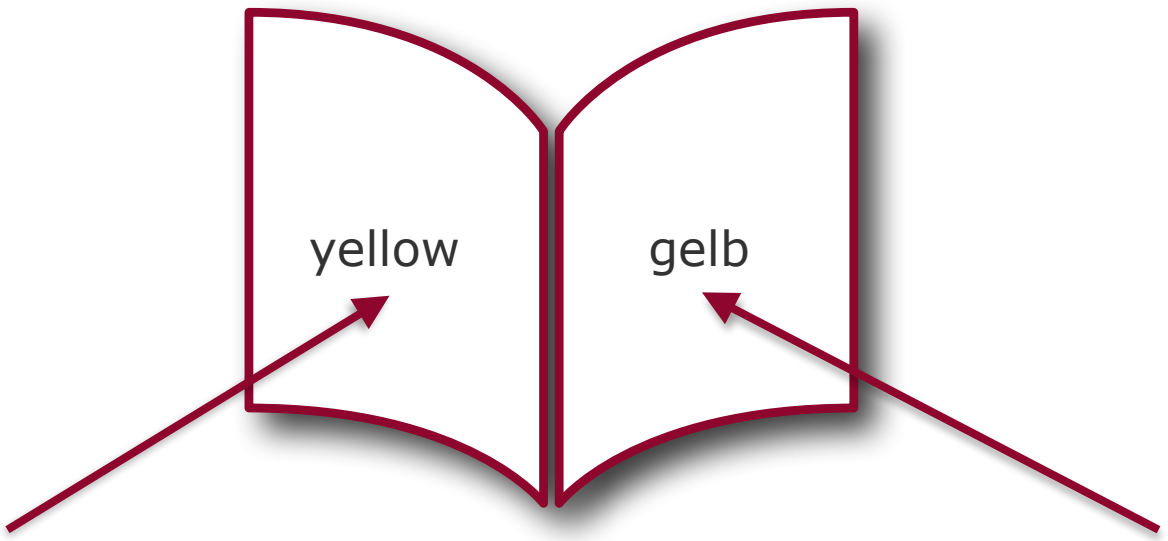
Dictionaries

Kira Grammel, Nina Ihde, Sebastian Serth & Selina Reinhard
Hasso-Plattner-Institut
Universität Potsdam

Dictionary

Englisch-Deutsch-Wörterbuch



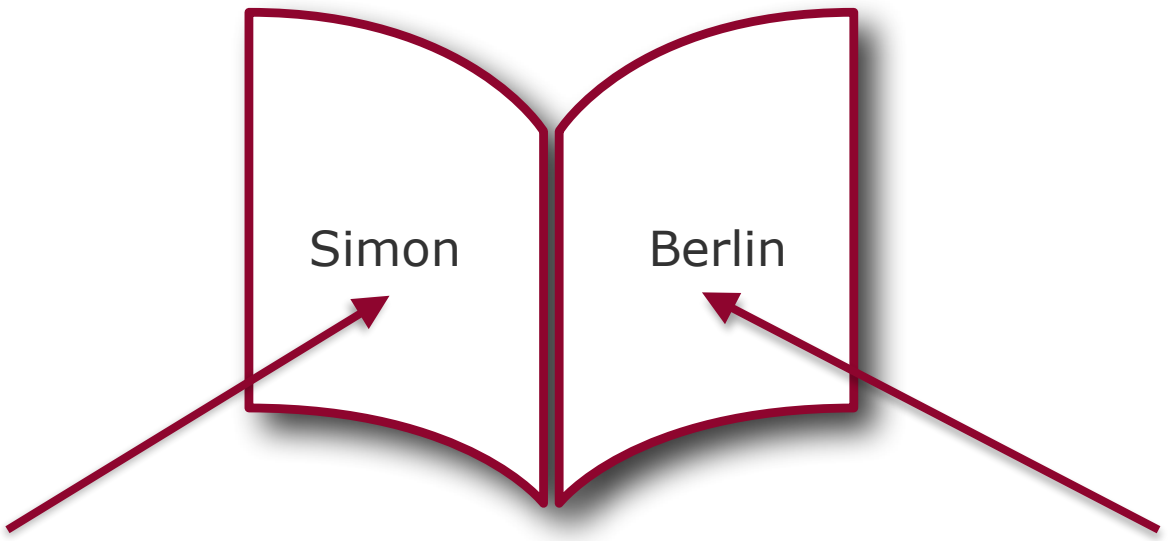
Schlüsselbegriff (Key)

Zugeordneter Wert (Value)

Schlüssel-Werte-Paare
(Key-Value-Paare)

Dictionary

Wohnorte-Wörterbuch



Schlüsselbegriff (Key)

Zugeordneter Wert (Value)

Schlüssel-Werte-Paare
(Key-Value-Paare)

Erzeugung eines Dictionaries

```
1 wohnorte = {
2     "Simon":"Berlin",
3     "Stella":"Berlin",
4     "Leonie":"Kiel"
5 }
6 print(wohnorte)
```

```
{'Leonie': 'Kiel', 'Simon': 'Berlin', 'Stella': 'Berlin'}
```

Dictionaries

- Können mit `{Schlüssel : Wert, zweiter_Schlüssel : zweiter_Wert, …}` erzeugt werden
- Ein Schlüssel darf nur einmal vorkommen, ein Wert jedoch mehrfach
- Reihenfolge der Paare im Dictionary ist nicht fest

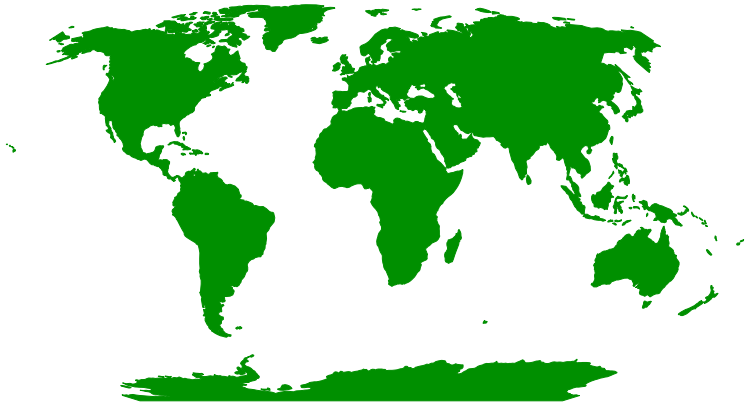
Anzahl an Paaren im Dictionary

```
1 wohnorte = {"Simon":"Berlin","Stella":"Berlin","Leonie":"Kiel"}
2 anzahl_paare = len(wohnorte)
3 print(anzahl_paare)
```

```
3
```

Len(dictionary)

- Gibt Anzahl der Paare aus
- Genauso wie bei Listen und Strings



Zugriff auf Werte

```
1  wohnorte = {"Simon":"Berlin","Stella":"Berlin","Leonie":"Kiel"}
2  print("Simon wohnt in", wohnorte["Simon"])
```

```
Simon wohnt in Berlin
```

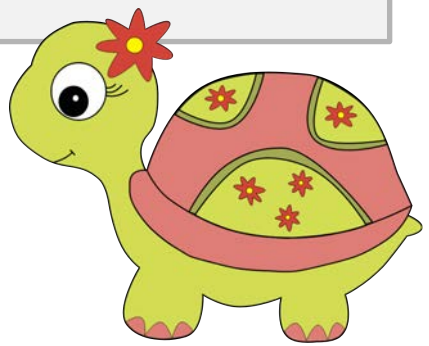
Zugriff

- Mit Hilfe von `dictionary[schlüssel]` kann man auf die Werte in einem Dictionary zugreifen
- Der Zugriff funktioniert ähnlich wie bei Listen, wo man einen Index angibt

Veränderung von Werten

```
1  wohnorte = {"Simon":"Berlin","Stella":"Berlin","Leonie":"Kiel"}
2  wohnorte["Leonie"] = "Bremen"
3  print(wohnorte)
```

```
{'Simon': 'Berlin', 'Stella': 'Berlin', 'Leonie': 'Bremen'}
```



Ändern

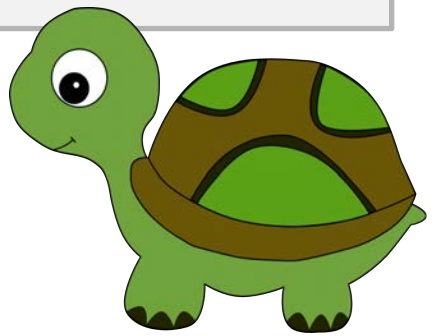
- Werte können im Dictionary geändert werden
- Zuweisung eines neuen Werts:

  `dictionary[schlüssel] = neuer_Wert`

Hinzufügen von Paaren

```
1 wohnorte = {"Simon":"Berlin","Stella":"Berlin","Leonie":"Kiel"}
2 wohnorte["Paul"] = "Potsdam"
3 print(wohnorte)
```

```
{'Paul': 'Potsdam', 'Simon': 'Berlin', 'Stella': 'Berlin',
'Leonie': 'Kiel'}
```



Hinzufügen

- Es können Paare zum Dictionary hinzugefügt werden
- Gleiche Anweisung wie beim Ändern von Werten
- Da kein Paar den Schlüssel `"Paul"` enthält, wird das Paar hinzugefügt

Entfernen von Paaren

```
1  wohnorte = {"Simon":"Berlin","Stella":"Berlin","Leonie":"Kiel"}
2  del(wohnorte["Simon"])
3  print(wohnorte)
```

```
{'Stella': 'Berlin', 'Leonie': 'Kiel'}
```

del()

- "delete" heißt "löschen"
- Entfernt ein Paar aus dem Dictionary
- Bekommt einen Schlüssel übergeben

Iterieren über Dictionaries

```
1 wohnorte = {"Simon":"Berlin","Stella":"Berlin","Leonie":"Kiel"}
2 for schluessel in wohnorte.keys():
3     print(schluessel)
```

```
Leonie
Stella
Simon
```

Dictionary.keys()

- Gibt alle Schlüssel des Dictionaries zurück
- Man kann mit einer For-Schleife über diese iterieren

Iterieren über Dictionaries

```
1  wohnorte = {"Simon":"Berlin","Stella":"Berlin","Leonie":"Kiel"}
2  for wert in wohnorte.values():
3      print(wert)
```

```
Berlin
Kiel
Berlin
```

Dictionary.values()

- Gibt alle Werte des Dictionaries zurück

Iterieren über Dictionaries

```
1 wohnorte = {"Simon":"Berlin","Stella":"Berlin","Leonie":"Kiel"}
2 for paar in wohnorte.items():
3     print(paar[0] + " wohnt in " + paar[1])
```

```
Stella wohnt in Berlin
Simon wohnt in Berlin
Leonie wohnt in Kiel
```

Dictionary.items()

- Gibt alle Paare (Schlüssel und Wert) des Dictionaries zurück
- Kann mit `[0]` auf den Schlüssel zugreifen und mit `[1]` auf den Wert

Zusammenfassung

- Dictionaries
  - Sind wie Wörterbücher
  - Bestehen aus Schlüssel-Werte-Paaren
  - Werden mit `{Schlüssel:Wert, …}` erzeugt

- Werte ändern oder hinzufügen:

  `dictionary[schlüssel] = neuer_Wert`

- `dictionary.keys()` gibt alle Schlüssel zurück
- `dictionary.values()` gibt alle Werte zurück
- `dictionary.items()` gibt Aufzählung der Schlüssel und zugeordneten Werte zurück