

---

**Aufgabenstellung: Kennenlernen der Konstruktoren und Methoden**

Sie haben in der Vorlesung die verschiedenen Datentypen und ihre Eigenschaften kennengelernt. Ziel dieser Übung ist es nun, die Datentypen von Typ A zu Typ B zu konvertieren. Dazu können mit Hilfe der Konstruktoren neue Objekte erzeugt werden.

Erstellen Sie ein Python Programm und legen Sie darin das folgende tuple an:

```
t = (1, 1, 2, 3, 5, 8, 13, 21)
```

**Aufgabe 1: Umwandlung in str und Kennenlernen der str Methoden**

Recherchieren Sie unter <https://docs.python.org/3/library/stdtypes.html#str> die Verwendung des Konstruktors `str()` und wandeln Sie das oben angelegte tuple in ein str Objekt. Geben Sie anschließend beide Objekte auf der Konsole aus.

Verschaffen Sie sich unter <https://docs.python.org/3/library/stdtypes.html#string-methods> einen Überblick über die Methoden von str. Legen Sie sich die folgende Zeichenkette in dem Python Programm an:

```
s = 'das Ist MEiN teext '
```

Wenden Sie die folgenden Methoden auf die Zeichenkette s an und geben Sie das Ergebnis auf der Konsole aus. Welches Verhalten beobachten Sie?

- `capitalize()`
- `count()`
- `find()`
- `isalpha()`
- `join()`
- `lower()`
- `replace()`
- `rstrip()`
- `split()`

Korrigieren Sie durch Anwendung von str Methoden die Zeichenkette s, so dass es einem deutschen Satz entspricht.

**Aufgabe 2: Umwandlung in list und Kennenlernen der list Methoden**

Recherchieren Sie unter <https://docs.python.org/3/library/stdtypes.html#list> die Verwendung des Konstruktors `list()` und wandeln Sie das oben angelegte tuple in ein list Objekt.

Verschaffen Sie sich unter <https://docs.python.org/3/tutorial/datastructures.html> einen Überblick über die Methoden von list. Legen Sie sich die folgende Liste in dem Python Programm an:

```
l = ['München', 'Regensburg', 'Augsburg', 'Nürnberg', 'Scheinfurt', 'Bayreuth', 'Ulm']
```

Wenden Sie die folgenden Methoden auf die Liste l an und geben Sie das Ergebnis auf der Konsole aus. Welches Verhalten beobachten Sie?

- `append()`
- `insert()`
- `remove()`
- `pop()`
- `count()`
- `sort()`

**Aufgabe 3: Umwandlung in dict und Kennenlernen der dict Methoden**

Recherchieren Sie unter <https://docs.python.org/3/library/stdtypes.html#mapping-types-dict> die Verwendung des Konstruktors `dict()` und wandeln Sie das oben angelegte `tuple` in ein `dict` Objekt. Erklären Sie sich das Verhalten und legen Sie hinterher ein leeres Dictionary an.

Verschaffen Sie sich unter <https://docs.python.org/3/library/stdtypes.html#typesmapping> einen Überblick über die Methoden von `dict`. Legen Sie sich die folgenden Dictionaries in dem Python Programm an:

```
d1 = dict(jan=0, feb=1, mar=3)
d2 = {'Jan': 0, 'Feb': 1, 'Mar': 2}
d3 = dict( zip(['Jan', 'Feb', 'Mar'], [1, 2, 3]) )
```

Wenden Sie die folgenden Methoden auf die drei Dictionaries an und geben Sie das Ergebnis auf der Konsole aus. Welches Verhalten beobachten Sie?

- `len()`
- `d[key]` (lesen, schreiben)
- `key in d`
- `items()`
- `keys()`
- `values()`

Schreiben Sie eine `for` Schleife und geben Sie die `key:value` Paare zeilenweise auf der Konsole aus.

**Aufgabe 4: Umwandlung in set und Kennenlernen der set Methoden**

Recherchieren Sie unter <https://docs.python.org/3/library/stdtypes.html#set-types-set-frozenset> die Verwendung des Konstruktors `set()` und wandeln Sie das oben angelegte `tuple` in ein `set` Objekt.

Verschaffen Sie sich unter <https://docs.python.org/3/library/stdtypes.html#set-types-set-frozenset> einen Überblick über die Methoden von `set`. Legen Sie sich die folgenden Mengen in dem Python Programm an:

```
s1 = {1, 2, 3, 'a', 'b', 'd'}
s2 = {2, 4, 'a', 'c', 'c'}
s3 = {2, 'a'}
```

Wenden Sie die folgenden Methoden auf die drei Mengen an und geben Sie das Ergebnis auf der Konsole aus. Welches Verhalten beobachten Sie?

- `add()`
- `remove()`
- `x in s()`
- `issubset()`
- `union()`
- `intersection()`
- `difference()`