

## 6. Datentypen

### 6.1. Grundlagen

#### 1. Zeichenketten (strings)

Nebst ganzen Zahlen (Integers) und reellen Zahlen kann man auch Texte verarbeiten. Technisch gesehen nennt man dies eine Zeichenkette. Beispiele dafür sind *ja* und *nein*, aber auch die Namen von Spielern in einem Computerspiel.

#### 2. Teilnehmer

Man soll Vornamen und Namen eines Teilnehmers einlesen. Dabei soll es möglich sein, dass man zwei Vornamen eingeben kann. (Es geht hier in einem Programmteil nur um die Eingabe.)

Der Programmteil hat dann etwa folgende Gestalt:

```
# Teilnehmer
name = input("Wie lautet Ihr Name? ")
i = 1
vorn1 = input("Wie lautet Ihr "+str(i)+". Vorname? ")
i = 2
frage = input("Haben Sie einen "+str(i)+". Vornamen? ")
if (frage == "ja"):
    vorn2 = input("Wie lautet Ihr "+str(i)+". Vorname? ")
else:
    vorn2 = " "
```

Beachte: `input` kann (im Gegensatz zu `print`) nur *einen* Text annehmen. Deshalb muss man mehrere Texte zusammenfügen (das geschieht mit dem `+`) und aus der Zahl muss man eine Zeichenkette generieren (und das geschieht mit dem Befehl `str`).

#### 3. Unterscheide Zahl und Ziffer

In der Informatik muss man genau zwischen einer Ziffer 6 und der Zahl 6 unterscheiden. Die Ziffer wird als Text betrachtet (auch wenn es nur ein Zeichen ist), und vor allem kann man mit der Ziffer nicht rechnen.

Mit den Befehlen `str` und `int` macht man aus einer Zahl einen Text (d.h. eine Ziffer) und umgekehrt.

```
zahl=6
ziffer="6"      # Text steht immer in Anführungszeichen.
zahlneu=int(ziffer) # macht aus Text eine Zahl.
zifferneu=str(zahl) # macht aus der Zahl einen Text.
```

#### 4. Listen

Häufig ist es in der Informatik – und beispielsweise in der Statistik – nötig, mehrere gleichartige Daten in einer Liste zusammenzufassen.

Eine Liste hat folgende Struktur: `liste = [14.5, 7.6, 3.8, 4, 0, 1.5, 0.4]` und steht beispielsweise für Wetterdaten (Niederschlagsmengen, gemessen während einer Woche).

Eine Liste funktioniert im Wesentlichen genau gleich wie ein Vektor (dies als Information für diejenigen, die Vektoren schon kennen).

Beachte:

- a) Listen muss man normalerweise initialisieren. `liste = [0,0,0,0,0,0,0]`  
Man kann auch eine leere Liste initialisieren: `liste = []`
- b) Jedes Element einer Liste hat einen Index. Das erste Element einer Liste hat den Index Null.
- c) Einen Wert speichert man mit `liste[2] = 3.8`.
- d) Einen Listenwert kann man abfragen mit `liste[1]` abfragen. Das Ergebnis ist dann die oben gespeicherte 7.6 .
- e) Wenn man am Schluss der Liste noch ein Element hinzufügen will, dann geschieht dies mit `liste = liste + [0.4]`

#### 5. Beispiel

Lasse den Computer 10 mal würfeln und speichere die Werte in einer Liste.

```
# Liste
from random import *
liste = []      # Erstellt eine leere Liste
i = 1
while i <= 10:
    wurf = randint(1,6)
    liste = liste + [wurf]
    i = i + 1
print(liste)
```