

Listen

1.1 Was sind Listen?

In Listen kann man verschiedene Informationen speichern. Zum Beispiel die Wochentage. Der Inhalt einer Liste wird zwischen eckige Klammern gesetzt [...] und durch Kommata getrennt.

```
1 listel = ["Montag", "Dienstag", "Mittwoch", "Donnerstag", ]
```

Wenn wir jetzt unsere Liste ausdrucken wollen und

```
1 print (listel)
```

eingeben, erhalten wir diese Bildschirmausgabe:

```
1 ['Montag', 'Dienstag', 'Mittwoch', 'Donnerstag']
```

1.2 Den Inhalt einer Liste mit der for-Schleife ausgeben

Wir wollen aber die Tage einzeln angezeigt haben. Dafür benutzen wir eine for-Schleife. Die Einträge werden zunächst in eine Variable geschrieben, die wir z.B. `eintrag` nennen und dann wird die `for`-Schleife so lange durchlaufen, wie Elemente in der Liste sind und die Einträge untereinander ausgegeben. Die Variable `eintrag` nimmt dabei in jedem Listen-Durchlauf die einzelnen Werte der Liste an.

```
1 for eintrag in listel:  
2     print (eintrag)
```

erzeugt folgende Ausgabe:

```
1 Montag  
2 Dienstag  
3 Mittwoch  
4 Donnerstag
```

1.3 Anzahl der Elemente mit `len()` ermitteln

Um herauszufinden, wieviele Einträge die Liste enthält, können wir die bereits bekannte Funktion `len()` einsetzen. Wir schreiben die Anzahl der Elemente in die Variable `anzahl` und geben diese anschließend aus

```
1 anzahl = len(listel)  
2 print (anzahl)
```

Dies gibt 4 aus, d.h. es sind 4 Elemente in unserer Liste enthalten.

```
1
```

1.4 Einen Eintrag (z.B. den dritten) aus einer Liste lesen

Wie wir oben mit der `len()`-Funktion gesehen haben, enthält unsere Liste 4 Einträge

Jedem Wert in unserer Liste ist ein Index (Plural: Indices) zugeordnet. D.h. jeder Eintrag hat eine Zählnummer, anhand derer wir einzelne Einträge herausuchen können. Der Index wird in [eckige Klammern gesetzt].

Wir wollen das dritte Element aus unserer `liste1` anzeigen.

Wir erinnern uns:

Achtung: Beim Programmieren fangen wir oft bei 0 mit dem Zählen an. D.h., dass wir um das dritte Element zu erfahren nicht 3, sondern 2 in die eckigen Klammern schreiben müssen.

Die Einträge in unsere Liste haben also folgende Indices.

<code>liste1 =</code>	<code>["Montag",</code>	<code>"Dienstag",</code>	<code>"Mittwoch",</code>	<code>"Donnerstag"]</code>
Index:	0	1	2	3

Geben wir das dritte Element nun aus:

```
1 print (liste1[2])           # das dritte Element unserer Liste anzeigen
```

erhalten wir:

```
1 Mittwoch
```

1.5 Listen von hinten lesen

Man kann Listen auch vom rechts nach links lesen.

Hierfür werden *negative Indices* beginnend beim letzten Element mit `[-1]` benutzt:

<code>liste1 =</code>	<code>["Montag",</code>	<code>"Dienstag",</code>	<code>"Mittwoch",</code>	<code>"Donnerstag"]</code>
Index:	-4	-3	-2	-1

ein anderes Beispiel:

<code>liste =</code>	<code>["Hallo",</code>	<code>"Test",</code>	<code>"Welt"]</code>
Index:	0	1	2
neg. Index:	-3	-2	-1

Verwenden wir nun unser Wissen können wir das erste und letzte Element ganz einfach ausgeben:

```
1 # Ein Eintrag kann so gelesen werden:
2 ersterEintrag = liste[0] # = "Hallo". Achtung wir beginnen bei 0 !
3
4 # Negative Indices beginnen am Ende zu zählen
5 letzterEintrag = liste[-1] # = "Welt". Wir beginnen bei -1 !
6
7 print(ersterEintrag, letzterEintrag) # => "Hallo Welt"
```

1.6 Einen Eintrag ans Ende der Liste einfügen

Wollen wir jetzt den Freitag zur `liste1` hinzufügen, kann das mit der `.append(...)`-Funktion geschehen.

Wir sagen, wo wir etwas hinzufügen (`append` engl. für 'ans Ende anfügen') wollen: an das Ende der `liste1`. Was wir hinzufügen wollen (Freitag), schreiben wir in runde Klammern:

```
1 liste1.append("Freitag")
```

Geben wir nun die Liste wie oben beschrieben mithilfe ein `for`-Schleife aus, so erhalten wir:

```
1 Montag
2 Dienstag
3 Mittwoch
4 Donnerstag
5 Freitag
```