

Listen II

1 Zwei Listen zusammenfügen

Wir haben zwei Listen `liste1` und `liste2` und wollen sie zu einer Liste `listeNeu` zusammenfügen. Dies können wir mit dem Pluszeichen `+` erreichen:

```
1  liste1 = [0,1,2,3,4,]
2  liste2= [5,6,7,8,]
3
4  # die beiden Listen werden in listeNeu zusammengefügt
5  listeNeu = liste1 + liste2
6
7  # Ausgeben der neuen Liste
8  for eintrag in listeNeu:
9      print(eintrag)
10
11 # Als Ausgabe erhalten wir die Zahlen von 0 bis 8.
```

Wir könne auch Werte hinzufügen, ohne sie vorher in eine Listen-Variable zu schreiben.

```
1  listeNeu = liste1 + liste2 + [100,500]
2
3  # Hier erhalten wir die Ausgabe: 0 1 2 3 4 5 6 7 8 100 500
```

2 Aus einer Liste ein Element herauslöschchen

Wir haben eine Liste mit Farben: `farben = ['rot', 'gelb', 'grün', 'rot', 'blau']`. Wie man sieht, ist der Eintrag `'rot'` doppelt enthalten. Den vierten Eintrag müssen wir also entfernen.

3 Die `.pop()`- Funktion

Elemente aus der Liste entfernen kann man mit der Funktion `.pop()`

```
1  farben = ['rot', 'gelb', 'grün', 'rot', 'blau']
2  farben.pop(3)
3  # ACHTUNG: Das Programm beginnt bei 0 (Null) zu zählen.
4  # Der vierte Eintrag hat also den Index [3]
5
6  for eintrag in farben:
7      print(eintrag)
8  # Ausgabe: rot gelb grün blau
```

4 Die .remove()-Funktion

Eine weitere Möglichkeit, einen Wert zu entfernen bietet die Methode `remove(xy)` (engl.: entfernen). Diese Funktion schaut in der Liste nach und entfernt den ersten Eintrag mit dem Wert `xy`:

```
1 farben = ['rot', 'gelb', 'grün', 'rot', 'blau']
2 farben.remove('rot')
3
4 for eintrag in farben:
5     print(eintrag)
6 # Die Ausgabe sieht dann so aus: gelb grün rot blau
```

5 Ein bestimmtes Element löschen

Wir haben wieder unsere Farbenliste und wissen, dass die Farbe zweimal vorkommt. Soll jetzt der zweite Eintrag von entfernt werden. Dafür benutzen wir die Methode

6 Die .index()-Funktion

Zuerst wird der erste Eintrag von rot ermittelt

```
1 farben = ['rot', 'gelb', 'grün', 'rot', 'blau']
2 ersteStelle = farben.index('rot')
3
4 # wir wollen wissen welchen Index das erste 'rot' hat
5 print(ersteStelle)
6
7 # Anzeige auf dem Bildschirm: 0
```

Dies ist korrekt. 'rot' ist das erste Element und das erste Element einer Liste hat den Index Null. Jetzt sagen wir der Methode `index(..)`, sie soll den zweiten Eintrag von rot finden und an der nächsten Stelle mit der Suche fortfahren. Dies können wir indem wir die Nummer des Start-Indexes mitgeben, ab dem die Liste durchsucht werden soll.

```
1 # Fahre mit der Suche beim nächsten Index fort
2 zweiteStelle = farben.index('rot', 1)
3
4 print(zweiteStelle)
5 # Anzeige auf dem Bildschirm: 3
```

Den ersten Index speichern wir in der Variable `ersteStelle`. Für die weitergehende Suche müssen wir `ersteStelle` um 1 erweitern, um beim nächsten Index fortzufahren. Haben wir das nächste Vorkommen von 'rot' gefunden, kann der Wert mit `pop(..)` gelöscht werden.

```
1  farben = ['rot', 'gelb', 'grün', 'rot', 'blau']
2
3  # Ermitteln des ersten Indexes => 0
4  ersteStelle = farben.index('rot')
5
6  # Suche fortführen beim nächsten Index > 0 +1
7  zweiteStelle = farben.index('rot', ersteStelle + 1)
8
9  # Der zu löschende Index (= 3) ist in zweiteStelle gespeichert
10 farben.pop(zweiteStelle)
11
12 for eintrag in farben:
13     print(eintrag)
14
15 # Bildschirmanzeige: rot gelb grün blau
```

7 Ein Element einfügen

Ein Element an einer beliebigen Stelle der Liste hinzu fügen mit der Methode `insert(..)` (engl.: einfügen). In unserer Farbliste wollen wir an die zweite Stelle die Farbe 'lila' hinzufügen. Wie müssen unserer Methode sagen an welcher Stelle und was wir hinzufügen wollen:

```
1  farben = ['rot', 'gelb', 'grün', 'rot', 'blau']
2
3  # an zweiter Stelle einfügen, d.h. nach 'rot'
4  farben.insert(1, 'lila')
5
6  # Bildschirmanzeige: rot lila gelb grün rot blau
```

8 Eine Teilliste erzeugen

Eine Liste von Elemente kann man mittels des Slicing-Operators aus einer Liste ausschneiden. Wie bei der Indizierung verwendet der Slicing-Operator eckige Klammern `[]`, aber jetzt werden zwei Werte erwartet `[start:ende]`:

Du kannst dir das Ganze so merken. Schreibst du nur eine Zahl, z.B. die 2 in die Klammer, also `liste[2]` erhältst du das Element an dieser (der dritten) Stelle.

Willst du allerdings nicht nur einen Wert, sondern einen Teil der Liste, z.B. das zweite bis zum vierten Element, so schreibt man `liste[1:4]`.

Hier ein paar Beispiele zur Verwendung des Slicing-Operators:

```
1  farben = ['rot', 'schwarz', 'gelb', 'grün', 'rot', 'blau']
2
3  # Teilliste von zweiten bis zum vierten Element
4  neue_farben = farben[1:4]
5
6  # Bilschirmausgabe: schwarz gelb grün
7
8  # Lässt man in der eckigen Klammer den Anfangswert frei,
9  # beginnt das Programm von Anfang an zu zählen und zeigt uns:
10
11 neue_farben = farben[:2]
12 # Ausgabe: rot schwarz
13
14 # Lässt man in der eckigen Klammer den Endwert frei,
15 # so zählt das Programm ab diesem Index und geht bis zum Ende.
16
17 neue_farben = farben[2:]
18 # Es zeigt an: gelb grün rot blau
19
20 # Lässt man Anfangs- und Endwert weg, erhält man die ganze Liste (als
21   ↳ Kopie):
22 neue_farben = farben[:]
23
24 # Ausgabe: rot schwarz gelb grün rot blau
```

9 Aufgabe: Was kann man alles mit Listen machen?

9.1 Aufgabe 1: Landwirt Fritz Bauer

1. Landwirt Fritz Bauer hat sich zwei Listen mit Tieren, die sich auf seinem Hof befinden, erstellt.

```
listel = ['Kühe', 'Schafe', 'Ziegen', 'Schweine', 'Esel']
liste2 = ['Hühner', 'Enten', 'Tauben']
```

Es gibt zwei Möglichkeiten, aus den beiden Listen eine einzige Liste zu machen. Zeig sie Herrn Bauer!

2. Wir haben jetzt eine neue Liste:

```
liste = ['Kühe', 'Schafe', 'Ziegen', 'Schweine', 'Esel',
         'Hühner', 'Enten', 'Tauben']
```

- Herr Bauer tauscht mit seinem Nachbarn seinen Esel gegen ein Pferd. Lösche den Esel aus der Liste und füge das Pferd hinzu.

- Herr Bauers Tochter Eliane wünscht sich zum Geburtstag ein Pony. Da Herr Bauer seiner Tochter nichts abschlagen kann, hat sein Hof jetzt ein Tier mehr und er muss wieder seine Liste erweitern. Füge das Pony vor dem Pferd ein.

Tipp: lass Dir den Index für das Pferd anzeigen, dann kann man das Pony leicht einfügen.

9.2 Aufgabe 2: Quadratzahlen

Wir haben eine Liste der Zahlen 1 bis 10 und wollen nun die Quadrate dieser Zahlen ermitteln.

```
1 zahlen = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Zur Erinnerung: eine Quadratzahl bekommt man, wenn man die Zahl mit sich selber multipliziert. Bsp.
 $2^2 = 2 * 2 = 4$:

10 Lösungen

10.1 Erste Aufgabe: Variante 1

```
1 liste1 = ['Kühe', 'Schafe', 'Ziegen', 'Schweine', 'Esel']
2 liste2 = ['Hühner', 'Enten', 'Tauben']
3 listeNeu = liste1 + liste2
4
5 # Ausgeben der neuen Liste
6 for eintrag in listeNeu:
7     print(eintrag)
```

10.2 Erste Aufgabe: Variante 2

```
1 listeNeu = liste1 + ['Hühner', 'Enten', 'Tauben']
2
3 # Ausgeben der neuen Liste
4 for eintrag in listeNeu:
5     print(eintrag)
```

10.3 Zweite Aufgabe

```
1 liste = ['Kühe', 'Schafe', 'Ziegen', 'Schweine', 'Esel', 'Hühner',
2         ↪ 'Enten', 'Tauben']
3
4 liste.append('Pferd')
5 liste.remove('Esel')
```

```
6  for eintrag in liste:
7      print(eintrag)
```

10.4 Dritte Aufgabe

```
1  # erster Teil: Index finden
2
3  liste = ['Kühe', 'Schafe', 'Ziegen', 'Schweine', 'Hühner', 'Enten',
4          ↪ 'Tauben', 'Pferd']
5
6  # Der Index wird in der Variablen x gespeichert
7  x = liste.index('Pferd')
8  print(x) # Zeige den Wert des Indexes (= 7)
9
10 # zweiter Teil: Pony hinter Pferd einfügen
11
12 #Füge das Pony hinter dem Pferd ein: Der Index muss 8 sein
13 liste.insert(8, 'Pony')
14
15 # Liste ausgeben
16 for eintrag in liste:
17     print(eintrag)
```

10.5 Dritte Aufgabe: elegantere Lösung

```
1  liste = ['Kühe', 'Schafe', 'Ziegen', 'Schweine', 'Hühner', 'Enten',
2          ↪ 'Tauben', 'Pferd']
3
4  # Der Index wird in der Variablen x gespeichert
5  x = liste.index('Pferd')
6
7  #Füge das Pony hinter dem Pferd ein: Der Index um 1 höher sein: x+1
8  liste.insert(x + 1, 'Pony')
9
10 for eintrag in liste:
11     print(eintrag)
```

10.6 Vierte Aufgabe

```
1  zahlen =[1,2,3,4,5,6,7,8,9,10]
2
3  for wert in zahlen:
4      # Eine Quadratzahl erhält man, indem man die Zahl mit sich
5      ↪ multipliziert
```

```
5     wert = wert * wert
6     print(wert)
```