DS 11 – Weitere Übungen

Listen: viele Werte unter einem Namen

Hier kannst du das Gelernte weiter üben. Die Punkte im Titel zeigen, wie schwer eine Aufgabe ist: ●○○ leicht, ●●○ mittel, ●●● knifflig. Fang mit den leichten an.

Die Aufgaben stehen auch im Skript im Abschnitt „Weitere Übungen“; die Lösungen im Lösungsheft.

Aufgabe 1: Wochentage ●○○

Gegeben ist die Liste `tage = ["Mo", "Di", "Mi", "Do", "Fr"]`. Bestimme **ohne Computer** die Werte der Ausdrücke. Prüfe danach mit `print(...)` am Computer.

`tage[1]`	`tage[-1]`	`len(tage)`	`tage[len(tage) - 3]`

Gib außerdem an, an welchem Index das Element `"Do"` steht.

Deine Antwort:

…

Lösung

✅ **Lösung**

Lösung

`tage[1]`	`tage[-1]`	`len(tage)`	`tage[len(tage) - 3]`
`"Di"`	`"Fr"`	`5`	`"Mi"`

Lösung

```
1 tage = ["Mo", "Di", "Mi", "Do", "Fr"]
2 print(tage[1], tage[-1])
3 print(len(tage))
4 print(tage[len(tage) - 3])
```

```
Di Fr
5
Mi
```

Lösung

Ausgabe:

```
Di Fr
5
Mi
```

Lösung

`tage[1]` ist das **zweite** Element, denn gezählt wird ab 0. `len(tage) - 3` ist 5 − 3 = 2, und am Index 2 steht `"Mi"`. Das Element `"Do"` steht am **Index 3**. Wer mit `tage.index("Do")` nachprüft, bekommt ebenfalls 3.

Aufgabe 2: Liste verändern ●○○

Gib **ohne Computer** an, was das Programm ausgibt. Überprüfe danach am Computer.

```python
punkte = [12, 7, 9]
punkte[0] = 10
punkte.append(15)
punkte[1] = punkte[1] + 3
print(punkte)
print(len(punkte))
```

```
[10, 10, 9, 15]
4
```

Deine Antwort:

…

Lösung

✅ **Lösung**

Lösung

```python
punkte = [12, 7, 9]
punkte[0] = 10
punkte.append(15)
punkte[1] = punkte[1] + 3
print(punkte)
print(len(punkte))
```

```
[10, 10, 9, 15]
4
```

Lösung

Ausgabe:

```
[10, 10, 9, 15]
4
```

Lösung

- **Zeile 2:** Das Element am Index 0 wird überschrieben: aus 12 wird 10.
- **Zeile 3:** `append` hängt 15 ans Ende. Die Liste hat jetzt 4 Elemente.
- **Zeile 4:** Rechts vom `=` wird zuerst das alte Element gelesen (7), dann 3 addiert. Das Ergebnis 10 wird an den Index 1 geschrieben.

Aufgabe 3: Stundenplan ●○○

Am Montag hast du die Fächer Mathe, Deutsch, Informatik und Sport.

1. Lege eine Liste `montag` mit diesen vier Fächern an.
2. Gib das erste und das letzte Fach aus.
3. Sport fällt aus, stattdessen ist Kunst. Ändere das Element mit einem schreibenden Zugriff.
4. Nach der Schule ist noch die Theater-AG. Hänge `"Theater-AG"` an die Liste an.
5. Gib die Liste und die Anzahl der Einträge aus.

Lösung

✅ **Lösung**

Lösung

```
1 montag = ["Mathe", "Deutsch", "Informatik", "Sport"]
2 print("Erstes Fach:", montag[0])
3 print("Letztes Fach:", montag[-1])
4 montag[3] = "Kunst"
5 montag.append("Theater-AG")
6 print(montag)
7 print("Anzahl:", len(montag))
```

```
Erstes Fach: Mathe
Letztes Fach: Sport
['Mathe', 'Deutsch', 'Informatik', 'Kunst', 'Theater-AG']
Anzahl: 5
```

Lösung

Ausgabe:

```
Erstes Fach: Mathe
Letztes Fach: Sport
['Mathe', 'Deutsch', 'Informatik', 'Kunst', 'Theater-AG']
Anzahl: 5
```

Lösung

Sport steht am Index 3, also schreibt man `montag[3] = "Kunst"`. Für das letzte Fach geht auch `montag[len(montag) - 1]`. Wichtig: Das letzte Fach muss **vor** der Änderung ausgegeben werden, sonst steht dort schon Kunst.

Aufgabe 4: IndexError erklären ●○○

Das Programm soll die letzte Farbe der Liste ausgeben. Es bricht aber ab.

```
farben = ["rot", "gelb", "grün"]
print(farben[3])
```

```
Traceback (most recent call last):
  File "<Zelle 5>", line 2, in <module>
IndexError: list index out of range
```

1. Gib an, welche Indizes in dieser Liste gültig sind.
2. Erkläre, warum die Fehlermeldung `IndexError: list index out of range` erscheint.
3. Nenne zwei Möglichkeiten, die letzte Farbe richtig auszugeben.

Deine Antwort:

…

Lösung

✅ **Lösung**

Lösung

```
farben = ["rot", "gelb", "grün"]
print(farben[3])
```

```
Traceback (most recent call last):
  File "<Zelle 6>", line 2, in <module>
IndexError: list index out of range
```

Lösung

Fehlermeldung:

```
IndexError: list index out of range
```

Lösung

1. Die Liste hat 3 Elemente. Gültig sind die Indizes 0, 1 und 2 (und von hinten −1, −2, −3).
2. Der Index 3 liegt außerhalb des gültigen Bereichs („out of range“). Der Fehler entsteht, weil man an die Länge 3 denkt und vergisst, dass ab 0 gezählt wird.
3. Mit `farben[2]` oder `farben[-1]`. Beide Fassungen geben `grün` aus. Für Listen jeder Länge passt auch `farben[len(farben) - 1]`.

Aufgabe 5: Preiserhöhung am Kiosk ●●○

Am Schulkiosk werden alle Preise um 50 Cent teurer. Die Preise stehen in der Liste `preise = [1.5, 2.0, 0.8, 3.0]` (in Euro).

Schreibe ein Programm, das mit einer Schleife **jedes Element der Liste** um 0.5 erhöht. Gib die Liste danach aus.

> **Tipp:** Du musst die Elemente **in der Liste** verändern. Dafür brauchst du die Schleife über die Indizes: `for i in range(len(preise)):`. Im Rumpf steht dann ein schreibender Zugriff wie im Auftrag *Liste verändern*.

Lösung

✅ **Lösung**

Lösung

```
1 preise = [1.5, 2.0, 0.8, 3.0]
2 for i in range(len(preise)):
3     preise[i] = preise[i] + 0.5
4 print(preise)
```

```
[2.0, 2.5, 1.3, 3.5]
```

Lösung

Ausgabe:

```
[2.0, 2.5, 1.3, 3.5]
```

Lösung

`i` durchläuft die Indizes 0, 1, 2 und 3. In jedem Durchlauf wird das Element am Index `i` gelesen, um 0.5 erhöht und an dieselbe Stelle zurückgeschrieben. Das Programm funktioniert auch, wenn die Liste länger wird, weil `len(preise)` die Anzahl selbst bestimmt.

Aufgabe 6: Siegerliste ●●○

Die Liste `sieger = ["Lea", "Tom", "Ada", "Ben"]` enthält die Namen nach der Platzierung beim Sportfest. Schreibe ein Programm, das mit einer Schleife diese Ausgabe erzeugt:

Ausgabe:

```
Platz 1 - Lea
Platz 2 - Tom
Platz 3 - Ada
Platz 4 - Ben
```

Lösung

✅ **Lösung**

Lösung

```
sieger = ["Lea", "Tom", "Ada", "Ben"]
for i in range(len(sieger)):
    print("Platz", i + 1, "-", sieger[i])
```

```
Platz 1 - Lea
Platz 2 - Tom
Platz 3 - Ada
Platz 4 - Ben
```

Lösung

Ausgabe:

```
Platz 1 - Lea
Platz 2 - Tom
Platz 3 - Ada
Platz 4 - Ben
```

Lösung

Der Index beginnt bei 0, der Platz aber bei 1. Deshalb wird `i + 1` ausgegeben. Das Element liest man trotzdem mit `sieger[i]`. Hier sieht man gut: Index und Platznummer sind zwei verschiedene Dinge. Typischer Fehler: `sieger[i + 1]`. Dann fehlt Lea, und beim letzten Durchlauf gibt es einen `IndexError`.

Aufgabe 7: Zwei Mannschaften ●●○

Beim Fußball im Sportunterricht stehen alle in einer Reihe: `reihe = ["Ada", "Ben", "Cem", "Dana", "Eli", "Finn"]`. Wer an einem geraden Index steht (0, 2, 4), kommt in Mannschaft A, die anderen in Mannschaft B.

Schreibe ein Programm, das zwei neue Listen `team_a` und `team_b` mit `append` füllt und beide ausgibt.

> **Tipp:** Denk an die Schrittweite von `range` aus dem Abschnitt über die for-Schleife: `range(0, 6, 2)` erzeugt 0, 2, 4. Statt 6 schreibst du besser `len(reihe)`.

Lösung

✅ **Lösung**

Lösung

```
reihe = ["Ada", "Ben", "Cem", "Dana", "Eli", "Finn"]
team_a = []
team_b = []
for i in range(0, len(reihe), 2):
    team_a.append(reihe[i])
for i in range(1, len(reihe), 2):
    team_b.append(reihe[i])
print("Team A:", team_a)
print("Team B:", team_b)
```

```
Team A: ['Ada', 'Cem', 'Eli']
Team B: ['Ben', 'Dana', 'Finn']
```

Lösung

Ausgabe:

```
Team A: ['Ada', 'Cem', 'Eli']
Team B: ['Ben', 'Dana', 'Finn']
```

Lösung

Die erste Schleife durchläuft die Indizes 0, 2, 4, die zweite die Indizes 1, 3, 5. Auch richtig ist **eine** Schleife über alle Indizes mit der Bedingung `if i % 2 == 0:` (gerader Index) und einem `else`-Teil für Mannschaft B.

Aufgabe 8: Plätze tauschen ●●●

Im Sitzplan `sitzplan = ["Ben", "Ada", "Tom", "Mia", "Lea"]` wollen Tom und Lea die Plätze tauschen.

1. Bestimme mit `.index()` die Indizes von Tom und Lea.
2. Tausche die beiden Elemente mit schreibenden Zugriffen. Gib den Sitzplan vor und nach dem Tausch aus.
3. Erkläre, warum du dafür eine **Hilfsvariable** brauchst.

> **Tipp:** Stell dir zwei Gläser vor: eins mit Saft, eins mit Wasser. Um die Inhalte zu tauschen, brauchst du ein drittes, leeres Glas. Im Programm ist das dritte Glas eine Hilfsvariable, z. B. `merke = sitzplan[i]`.

Lösung

✅ Lösung

Lösung

```
1 sitzplan = ["Ben", "Ada", "Tom", "Mia", "Lea"]
2 print(sitzplan)
3 i = sitzplan.index("Tom")
4 j = sitzplan.index("Lea")
5 merke = sitzplan[i]
6 sitzplan[i] = sitzplan[j]
7 sitzplan[j] = merke
8 print(sitzplan)
```

```
['Ben', 'Ada', 'Tom', 'Mia', 'Lea']
['Ben', 'Ada', 'Lea', 'Mia', 'Tom']
```

Lösung

Ausgabe:

```
['Ben', 'Ada', 'Tom', 'Mia', 'Lea']
['Ben', 'Ada', 'Lea', 'Mia', 'Tom']
```

Lösung

- **Zeilen 3–4:** Tom steht am Index 2, Lea am Index 4.
- **Zeile 5:** Tom wird in `merke` zwischengespeichert.
- **Zeile 6:** Am Index 2 steht jetzt Lea. Tom wäre ohne `merke` verloren.
- **Zeile 7:** Tom kommt aus `merke` an den Index 4.

Ohne Hilfsvariable schreibt man `sitzplan[2] = sitzplan[4]` und danach `sitzplan[4] = sitzplan[2]`. Nach der ersten Zeile steht aber schon an **beiden** Stellen Lea. Die zweite Zeile schreibt also Lea auf Leas Platz, und Tom ist verschwunden.