DS 6 – Weitere Übungen

Die while-Schleife: Wiederholen, solange eine Bedingung gilt

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: In Viererschritten ●○○

Lies das Programm, **ohne** es auszuführen.

1. Schreibe auf, was es ausgibt.
2. Gib an, wie oft der Schleifenrumpf ausgeführt wird und welchen Wert `zahl` am Ende hat.

```
zahl = 20
while zahl > 5:
    print(zahl)
    zahl = zahl - 4
print("fertig")
```

```
20
16
12
8
fertig
```

Deine Antwort:

…

Aufgabe 2: Countdown ●○○

Das Programm soll eine Startzahl einlesen und von dort bis 1 herunterzählen. Danach soll es „Start!“ ausgeben. Ergänze die beiden Lücken `___` und teste mit `5` und mit `0`.

Lückencode

```
zahl = int(input("Starte bei: "))
while ___:
    print(zahl)
    ___
print("Start!")
```

Aufgabe 3: Anzahl der Spieler ●○○

Ein Brettspiel ist für 2 bis 4 Personen. Schreibe ein Programm, das die Anzahl der Spielerinnen und Spieler so lange abfragt, bis eine gültige Zahl eingegeben wird. Bei einer ungültigen Eingabe soll es einen Hinweis ausgeben. Am Ende soll es `Das Spiel startet mit 3 Personen.` ausgeben (mit der eingegebenen Zahl).

Aufgabe 4: Endlosschleife oder nicht? ●●○

Entscheide für jedes Programm, ob die Schleife irgendwann endet oder ob eine Endlosschleife entsteht. Begründe kurz. Führe die Programme **nicht** aus.

a)

```
i = 1
while i < 5:
    i = i + 1
```

b)

```
i = 10
while i > 0:
    i = i + 1
```

Nach 5 Sekunden abgebrochen: Die Schleife hört nicht auf (Endlosschleife). In Jupyter stoppst du sie mit ■ (Interrupt).

c)

```
i = 1
while i != 10:
    i = i + 2
```

Nach 5 Sekunden abgebrochen: Die Schleife hört nicht auf (Endlosschleife). In Jupyter stoppst du sie mit ■ (Interrupt).

d)

```
i = 0
while i < 3:
    print(i)
i = i + 1
```

```
0
0
0
0
0
0
0
0
0
0
0
0
…
```

Hier abgebrochen: Die Ausgabe hört nicht auf (Endlosschleife). In Jupyter stoppst du sie mit ■ (Interrupt).

Deine Antwort:

...

Aufgabe 5: Fehler finden: Wörter zählen ●●○

Das Programm soll so lange Wörter einlesen, bis `stopp` eingegeben wird. Danach soll es ausgeben, wie viele Wörter eingegeben wurden (ohne `stopp`). Es enthält **drei Fehler**.

```
anzahl = 0
wort = input("Wort (stopp zum Beenden): ")
while wort == "stopp":
    anzahl = anzahl + 1
wort = input("Wort (stopp zum Beenden): ")
print("Du hast", anzahl "Wörter eingegeben.")
```

```
  File "<Zelle 7>", line 6
    print("Du hast", anzahl "Wörter eingegeben.")
                            ^^^^^^^^^^^^^^^^^^^^
SyntaxError: invalid syntax
```

1. Finde die drei Fehler. Notiere jeweils die Zeile und was falsch ist.
2. Schreibe das Programm richtig auf. Teste es mit den Eingaben `Hund`, `Katze`, `Maus`, `stopp` und auch mit sofort `stopp`.

Deine Antwort:

…

Aufgabe 6: Ablaufprotokoll: Stellen zählen ●●○

Das Programm zählt, wie viele Stellen eine Zahl hat. Fülle das Ablaufprotokoll aus (– bedeutet: noch kein Wert) und gib an, was das Programm ausgibt.

```
zahl = 4096
stellen = 0
while zahl > 0:
    zahl = zahl // 10
    stellen = stellen + 1
print("Stellen:", stellen)
```

```
Stellen: 4
```

Schritt	Zeile	zahl	stellen	Schritt	Zeile	zahl	stellen
1	1			9	3		
2	2			10	4		
3	3			11	5		
4	4			12	3		
5	5			13	4		
6	3			14	5		
7	4			15	3		
8	5			16	6		

Deine Antwort:

…

Aufgabe 7: Papier falten ●●○

Ein Blatt Papier ist etwa 0,1 mm dick. Faltet man es einmal, ist der Stapel doppelt so dick. Stell dir vor, man könnte das Blatt beliebig oft falten.

1. Schreibe ein Programm, das berechnet, wie oft man falten müsste, bis der Stapel mindestens 1 Meter (1000 mm) dick ist. Gib die Anzahl der Faltungen und die Dicke aus.
2. Begründe, warum hier eine while-Schleife passt.

Deine Antwort:

…

Aufgabe 8: Die 3n+1-Folge ●●●

Mit einer Startzahl wird immer wieder so gerechnet:

- Ist die Zahl gerade, wird sie halbiert.
- Ist sie ungerade, wird sie mit 3 multipliziert und dann 1 addiert.

Das wird wiederholt, bis die Zahl 1 erreicht ist. Aus 6 wird zum Beispiel 3, dann 10, 5, 16, 8, 4, 2 und 1. Das sind 8 Schritte.

1. Schreibe ein Programm, das eine Startzahl einliest, jede neue Zahl ausgibt und am Ende die Anzahl der Schritte. Teste mit 6 und mit 7.
2. Was passiert bei der Startzahl `0`? Erkläre.

Deine Antwort:

…