

Teil 2 - Verzweigungen und Schleifen in Python

1. Wie beginnt man eine if-Anweisung in Python?

- a) if (Bedingung):
- b) if [Bedingung]:
- c) if Bedingung:
- d) if {Bedingung}:

2. Wie erstellt man in Python eine Schleife, die eine bestimmte Anzahl von Malen durchlaufen wird?

- a) while i < n:
- b) for i < n:
- c) for i in range(n):
- d) while i in range(n):

3. Welches Schlüsselwort wird in Python verwendet, um den aktuellen Durchlauf einer Schleife zu beenden und mit dem nächsten fortzufahren?

- a) pass
- b) continue
- c) break
- d) next

4. Wie wird in Python eine Bedingung geschrieben, die prüft, ob eine Variable x gleich 10 ist?

- a) if x = 10:
- b) if x == 10:
- c) if x === 10:
- d) if x != 10:

5. In Python, wie wird eine while-Schleife initialisiert, die so lange läuft, bis die Variable x größer als 5 ist?

- a) while x > 5:
- b) while x <= 5:
- c) while x != 5:
- d) while x == 5:

6. Was macht die else-Anweisung in einer for-Schleife in Python?

- a) Sie wird nach jeder Iteration der Schleife ausgeführt.
- b) Sie wird ausgeführt, wenn die Schleife normal beendet wird, ohne dass ein break-Statement erreicht wird.

- c) Sie wird immer ausgeführt, unabhängig davon, ob ein break-Statement erreicht wird.
- d) Sie wird nur ausgeführt, wenn die Schleife keine Iterationen hat.

7. Welches Schlüsselwort wird verwendet, um in Python eine unendliche Schleife zu erstellen?

- a) forever
- b) repeat
- c) while True:
- d) infinite

8. Wie kann in Python eine for-Schleife über eine Liste namens my_list iterieren?

- a) for item in my_list:
- b) for item = my_list:
- c) for item <= my_list:
- d) for item != my_list:

9. Was bewirkt das break-Schlüsselwort in einer Schleife in Python?

- a) Es hält die Schleife vorübergehend an.
- b) Es setzt die Schleife zurück zum Anfang.
- c) Es beendet die gesamte Schleife sofort.
- d) Es überspringt den Rest des aktuellen Schleifendurchlaufs.

10. Wie überprüft man in Python, ob eine Liste my_list leer ist?

- a) if len(my_list) == 0:
- b) if my_list is None:
- c) if my_list:
- d) if not my_list:

Teil 3 - Vertiefende Fragen zu Python

1. Was passiert, wenn die Einrückung in einem Python-Codeblock inkonsistent ist?

- a) Der Code wird normal ausgeführt, aber langsamer.
- b) Der Code wird ohne Fehlermeldung ignoriert.
- c) Es wird ein `SyntaxError` ausgelöst.
- d) Die Einrückung wird automatisch vom Python-Interpreter korrigiert.

2. Wie kann man in Python eine `else`-Anweisung mit einer `for`-Schleife kombinieren?

- a) Die `else`-Anweisung wird nach jedem Durchlauf der `for`-Schleife ausgeführt.
- b) Die `else`-Anweisung wird ausgeführt, wenn die `for`-Schleife normal endet, ohne durch ein `break` unterbrochen zu werden.
- c) Die `else`-Anweisung in Kombination mit einer `for`-Schleife ist in Python nicht zulässig.
- d) Die `else`-Anweisung wird vor der `for`-Schleife ausgeführt.

3. Welches Ergebnis liefert der folgende Python-Code:

Python-Code:

```
if not (True or False):  
    print("A")  
elif not (True and False):  
    print("B")  
else:  
    print("C")
```

- a) A
- b) B
- c) C
- d) Keine Ausgabe

4. In Python, wie kann man mehrere Bedingungen in einer `if`-Anweisung prüfen?

- a) `if condition1 and condition2:`
- b) `if condition1 plus condition2:`
- c) `if condition1 & condition2:`
- d) `if condition1 then condition2:`

5. Was bewirkt das `continue`-Statement in einer Schleife in Python?

- a) Es beendet sofort die gesamte Schleife.
- b) Es startet die nächste Iteration der Schleife.
- c) Es pausiert die aktuelle Iteration der Schleife.
- d) Es überspringt alle folgenden Schleifen.

6. Wie viele Male wird der Body einer while-Schleife in Python ausgeführt, wenn die Bedingung anfangs False ist?

- a) Einmal
- b) Zweimal
- c) Gar nicht
- d) Unendlich oft

7. Was ist der korrekte Weg, um in Python eine unendliche Schleife zu erstellen?

- a) while True:
- b) for ever:
- c) while 1 == 1:
- d) Sowohl a) als auch c) sind korrekt.

8. Welcher Fehler tritt auf, wenn in Python eine Schleife oder Verzweigung ohne entsprechenden Codeblock (nur mit einem Pass-Statement) erstellt wird?

- a) SyntaxError
- b) IndentationError
- c) NameError
- d) Kein Fehler tritt auf

9. Wie kann man in Python mehrere Bedingungen nacheinander prüfen?

- a) Mit verschachtelten if-Anweisungen
- b) Mit mehreren if-Anweisungen hintereinander
- c) Mit einer if-elif-else-Kette
- d) Sowohl a), b) als auch c) sind korrekt.

10. Was ist der korrekte Weg, um in Python eine for-Schleife vorzeitig zu beenden?

- a) exit()
- b) break
- c) stop
- d) return