

LU01.A09 - Vom Skript zur Funktion

Bauen Sie ein Lagerverwaltungs-Skript, das ausschliesslich aus Prozeduren mit globalem Zustand besteht, in echte Funktionen um – so, dass jeder Aufruf bei gleicher Eingabe immer dasselbe Ergebnis liefert.

Ausgangslage

Das folgende Skript funktioniert – solange man es genau in dieser Reihenfolge aufruft. Es zeigt gleich mehrere typische Schwächen prozeduralen Codes: globalen Zustand, Berechnung und Ausgabe im selben Atemzug, und Prozeduren, die None zurückgeben.

```
lager = []
gesamtwert = 0

def artikel_hinzufuegen(name, menge, preis):
    global gesamtwert
    lager.append({"name": name, "menge": menge, "preis": preis})
    gesamtwert += menge * preis

def zeige_gesamtwert():
    print(f"Gesamtwert: {gesamtwert:.2f} CHF")

def zeige_teuersten():
    teuerster = None
    for artikel in lager:
        if teuerster is None or artikel["preis"] > teuerster["preis"]:
            teuerster = artikel
    print(f"Teuerster Artikel: {teuerster['name']}")

if __name__ == '__main__':
    artikel_hinzufuegen('Maus', 10, 24.90)
    artikel_hinzufuegen('Tastatur', 4, 79.50)
    artikel_hinzufuegen('Monitor', 2, 249.00)
    zeige_gesamtwert()
    zeige_teuersten()
```

Anforderungen

1. **Analyse:** Halten Sie schriftlich fest, welche der drei def-Blöcke Prozeduren sind und woran Sie

das erkennen.

2. **Umbau:** Schreiben Sie das Skript so um, dass gilt:
 - Keine globalen Variablen – alle benötigten Daten kommen als Parameter herein.
 - Kein `print` innerhalb der berechnenden Funktionen. Ausgegeben wird ausschliesslich im `main`-Block.
 - Jede Funktion gibt einen Wert zurück.
 - `artikel_hinzufuegen` verändert die übergebene Liste **nicht**, sondern gibt eine **neue** Liste zurück.
3. **Nachweis:** Zeigen Sie mit `print`-Aufrufen, dass `gesamtwert(lager)` zweimal hintereinander aufgerufen dasselbe Ergebnis liefert und dass die ursprüngliche Liste nach `artikel_hinzufuegen` unverändert ist.
4. **Randfall:** Überlegen Sie, was `teuerster_artikel` bei einem leeren Lager zurückgeben soll, und setzen Sie es um.

Beispieloutput

```
Gesamtwert: 1065.00 CHF
Teuerster Artikel: Monitor
```

Kontrollfragen

1. Warum kann man den Ausdruck `gesamtwert(lager)` im Kopf jederzeit durch seine Zahl ersetzen, den Aufruf `artikel_hinzufuegen('Maus', 10, 24.90)` aus der Ausgangslage aber nicht? Wie heisst diese Eigenschaft?
2. Was gibt `zeige_gesamtwert()` zurück? Probieren Sie `ergebnis = zeige_gesamtwert()` und geben Sie `ergebnis` aus.
3. Welche Ihrer neuen Funktionen liessen sich mit einem automatischen Test prüfen, welche der alten Prozeduren nicht? Begründen Sie.

Hinweis

Schleifen sind in dieser Aufgabe ausdrücklich erlaubt – innerhalb einer Funktion ist eine `for`-Schleife völlig in Ordnung. Es geht hier nicht um den schleifenfreien Stil, sondern um die Grenze zwischen **Funktion** und **Prozedur**: Liefert der Block einen Wert, oder *tut* er etwas?

Abgabe

Geben Sie die Python-Datei und die beantworteten Kontrollfragen in Moodle ab.

[M323-LU01](#), [M323-A1B](#), [M323-A1I](#)

 © Kevin Maurizi

From:
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:
<https://wiki.bzz.ch/modul/m323/learningunits/lu01/aufgaben/prozedurzufunktion?rev=1787041223>

Last update: **2026/08/18 10:20**

