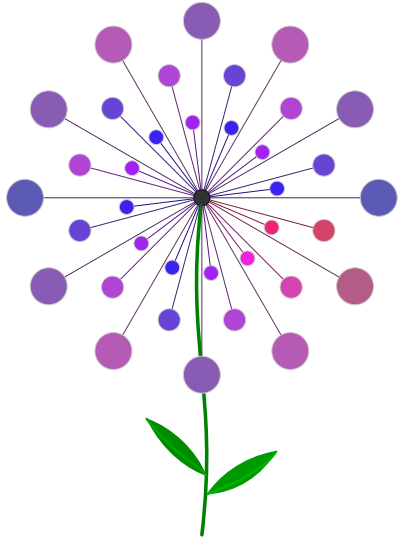


FreeFlower

Informatik: Programmieren

👨‍🍳 Kapitel 4: **Funktionen mit `return`: Mehrere Funktionen**

Funktionen: Mit `return`

```
def summiere(x1, x2):
    summe = x1 + x2
    return summe



res = summiere(3, 5)

print(res)
```

Funktionen: Mit `return`

```
def summiere(x1, x2):
    summe = x1 + x2
    return summe

res = summiere(3, 5)

print(res)
```



Funktionen: Mit `return`

```
def summiere(x1, x2):
    summe = x1 + x2
    return summe

res = summiere(3, 5)

print(res)
```

Wert (8) an das Hauptprogramm zurückgeben...

Funktionen: Mit `return`

```
def summiere(x1, x2):
    summe = x1 + x2
    return summe

res = summiere(3, 5)
print(res)
```

Wert (8) an das Hauptprogramm zurückgeben...

...*und* Wert in Variable speichern, z.B. `res`

Funktionen: Mit `return`

```
def summiere(x1, x2):
    summe = x1 + x2
    return summe

res = summiere(3, 5)
print(res)
```

Wert (8) an das Hauptprogramm zurückgeben...

...*und* Wert in Variable speichern, z.B. `res`

Weshalb nicht einfach `print` verwenden?

Funktionen

Ein einfaches Beispiel

```
def berechne_rabatt(preis, rabatt_pct):
    rabatt = preis * (rabatt_pct / 100)
    return preis - rabatt


def berechne_gesamtpreis(preis, rabatt_pct, mwst):
    rabattpreis = berechne_rabatt(preis, rabatt_pct)
    mwst = rabattpreis * (mwst / 100)
    return rabattpreis + mwst


endpreis = berechne_gesamtpreis(100, 15, 7.7)
print("Der Preis nach Rabatt und Mwst ist", endpreis)
```

Funktionen

Ein einfaches Beispiel

```
def berechne_rabatt(preis, rabatt_pct):
    rabatt = preis * (rabatt_pct / 100)
    return preis - rabatt

def berechne_gesamtpreis(preis, rabatt_pct, mwst):
    rabattpreis = berechne_rabatt(preis, rabatt_pct)
    mwst = rabattpreis * (mwst / 100)
    return rabattpreis + mwst


endpreis = berechne_gesamtpreis(100, 15, 7.7)
print("Der Preis nach Rabatt und Mwst ist", endpreis)
```

Wert zurückgeben (und speichern)

Funktionen

Ein einfaches Beispiel

```
def berechne_rabatt(preis, rabatt_pct):
    rabatt = preis * (rabatt_pct / 100)
    return preis - rabatt

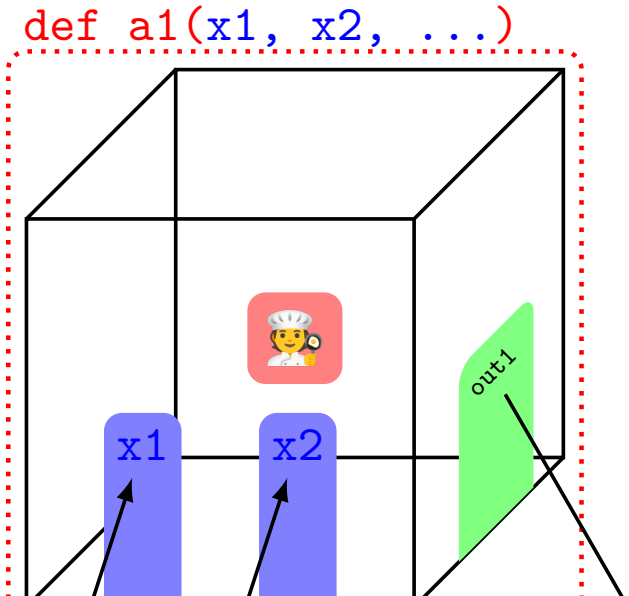
def berechne_gesamtpreis(preis, rabatt_pct, mwst):
    rabattpreis = berechne_rabatt(preis, rabatt_pct)
    mwst = rabattpreis * (mwst / 100)
    return rabattpreis + mwst

endpreis = berechne_gesamtpreis(100, 15, 7.7)
print("Der Preis nach Rabatt und Mwst ist", endpreis)
```

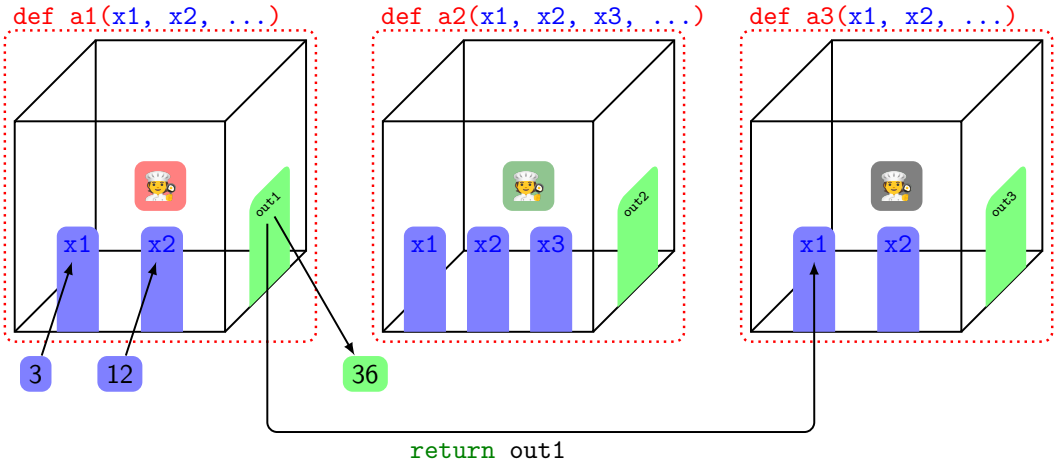
Wert zurückgeben (und speichern)

Wert zurückgeben (und speichern)

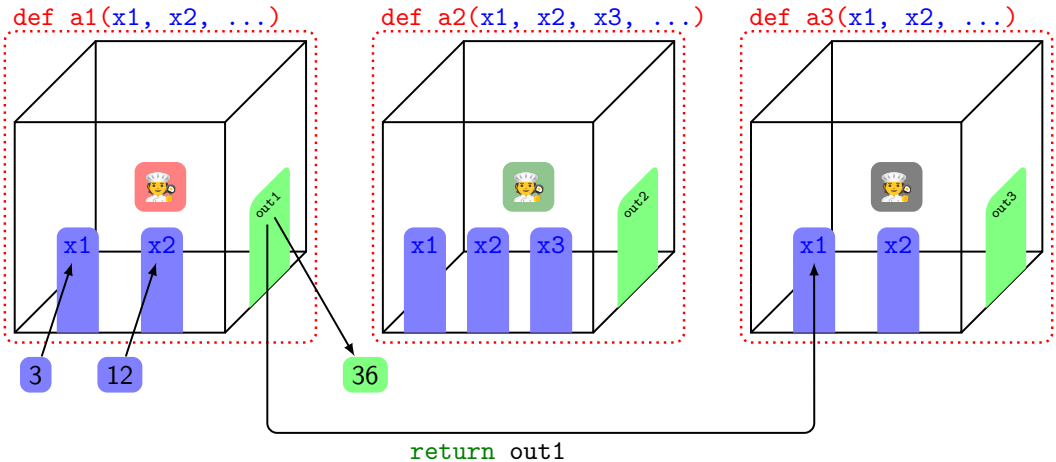
Modularität durch Funktionen



Modularität durch Funktionen

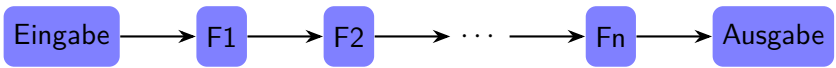


Modularität durch Funktionen



Auch dieses Bild wurde mit einer Funktion erstellt...

Modularität durch Funktionen



Analogien:

- Uhren-Fabrik
- Michelin-Küche
- ...alle komplexen Prozesse, die man in Unter-Prozesse aufbrechen muss!

Anwendungen: Überall! Daten-Analyse, AI, Business Development etc.

Auftrag

Programmier-Skript, Kapitel 4

- Aufgaben 4.11-4.14
- Moodle, Kapitel 4 (Teil 1)
- Moodle, Kapitel 4 (Teil 2)
- Moodle, Kapitel 4 (Zusammenfassung)