Modul 6 · Eigene Befehle

Übersicht: alle Merk- und Infokästen mit ihren Beispielen

Aufgabe 2 – Funktionen definieren und aufrufen

Eigene Befehle mit def

Ein eigener Befehl heißt in Python **Funktion** (oder Unterprogramm). Man schreibt sie mit `def` (von *define*, festlegen):

ALLGEMEIN

```
def name():
    anweisungsblock

name()
```

BEISPIEL

```
def quadrat():
    for i in range(4):
        forward(40)
        left(90)

quadrat()
quadrat()
```

- Die Zeile mit `def` ist der **Funktionskopf**: Name, leere Klammern, **Doppelpunkt**.
- Darunter steht **eingerückt**, was die Funktion tut – der **Funktionsrumpf**.
- Die **Definition** zeichnet noch nichts. Erst der **Aufruf** `quadrat()` führt die Zeilen aus – bei jedem Aufruf von Neuem.
- Die Definition muss **vor** dem ersten Aufruf stehen. Sonst kennt Python den Namen noch nicht.
- Namen schreibt man klein, ohne Leerzeichen und ohne Umlaute, z. B. `haus`, `fenster`, `drei_schritte`.

Das hat Vorteile: Das Programm wird **kürzer und übersichtlicher**, und wenn das Quadrat anders aussehen soll, änderst du es nur an **einer** Stelle.

Zusammenfassung

Das Wichtigste

Mit _____ legt man eine eigene _____ fest. Der Funktionskopf endet mit einem _____, der Rumpf ist _____. Ausgeführt wird die Funktion erst beim _____, z. B. `quadrat()`. Die Definition muss _____ dem ersten Aufruf stehen. Soll sich etwas ändern, ändert man es nur an _____ Stelle.