



In dieser dritten Übung in Lektion 1 erstellen Sie mit dem Programmierer Funktionen und führen sie in der Shell aus.

Lernziele :

- Notation von Funktionen in Python
- Erstellen einer Funktion

Funktionen in Python

In der Shell lässt sich mit den folgenden zwei Befehlen der Wert von y einfach berechnen :

$$x = 3$$

$$y = 2x + 3$$

Möchte man jedoch y für andere Werte von x berechnen, so muss man die zwei Befehle wieder neu schreiben, ein umständliches Vorgehen.

Mit einer Funktion hingegen kann man sich diese Arbeit erheblich vereinfachen.



Eine Funktion kann als eine Folge von Anweisungen betrachtet werden, die eine bestimmte Aufgabe unter Verwendung eines oder mehrerer Argumente ausführen. Sie stellt also einen Algorithmus dar. Sie erhält stets einen Namen:

- Die **Programmierung einer Funktion** beginnt immer mit **def**, gefolgt vom **Namen** der Funktion in Großbuchstaben (automatisch), gefolgt von ihren **Argumenten**. Diese Zeile **endet** mit dem **Symbol**:
- Die beiden Punkte in der nächsten Zeile markieren den **Beginn des Anweisungsblocks**, der die Funktion **definiert**. Alle diese Anweisungen werden **eingerrückt**, d.h. in Bezug auf die erste Zeile um 2 Stellen nach rechts verschoben. Die Einrückung geschieht automatisch.
- Die Funktion gibt über den Befehl **return** ein einzelnes **Ergebnis** (Wert einer Variablen) zurück. Das Ergebnis kann auch aus einer Liste von Ergebnissen oder einer Zeichenfolge bestehen.
- Die **Einrückung**, die automatisch oder mit der Tabulatortaste oder mit Leerzeichen erhalten wird, ist wichtig. Alles, was nach **def ()** eingerückt wird, wird als **Block** ausgeführt. **Die Einrückung darf während des Blockierens nicht variieren** (Anzahl der Leerzeichen).

def Name der Funktion(Liste der Argumente) :

- ..Anweisungsblock
- ..**return** Ergebnisse

Hinweis für Lehrkräfte : Eine Funktion ermöglicht es, das untersuchte Problem in Unterprobleme zu unterteilen und so die Wiederholung von Anweisungen zu vermeiden. Einmal definiert, kann sie während der Ausführung des Programms so oft wie nötig "aufgerufen" werden.



Eine Funktion kann auch keine Argumente haben. Sie kann auch in einem anderen Programm aufgerufen werden: Es reicht aus, sie in eine Anweisung einzufügen, indem der Name und die Werte der Argumente eingegeben werden.

Ein einfaches Beispiel

- Starten Sie die Python App und wählen sie « **New** »
- Vergeben Sie im **File Manager** einen **Namen** nach den vorgeschriebenen Regeln und bestätigen Sie mit « **Ok** ».
- Sie müssen nun die Funktion schreiben. Dazu wählen Sie **f1 :Fns...** und dann **1:def function()** :
- Jetzt können Sie die Funktion definieren, angefangen mit dem Namen und dem Argument f(x) und gefolgt vom Anweisungsblock. Der Editor rückt dabei automatisch ein, Rechenzeichen werden rot hervorgehoben.
- Der abschließende Befehl **return** befindet sich wieder im Menü « **Fns ...** »
- Durch Drücken von **f4 :Run** wird das kleine Programm gestartet und in die Shell umgeschaltet.
- Jetzt kann man durch Eingeben von **f(5)** den Funktionswert berechnen und anzeigen lassen.

