Lektion 2 : Bedingte Anweisung und Schleifen | **Übung 1 : Die bedingte Anweisung**

In der ersten Übung der Lektion 2 lernen Sie die Schreibweise und Anwendung einer bedingten Anweisung kennen.

Lernziele :

- Form und Verwendung einer bedingten Anweisung
- Verwendung in einer Funktion

In einem Programm ist es üblich, seine Ausführung gemäß Bedingungen zu steuern, die von den verschiedenen Variablen abhängen.
Eine Bedingung ist eine Aussage, die entweder wahr oder falsch sein kann.
Zum Beispiel: **a = b** oder **a≥b** aber auch **n ist gerade** sind Bedingungen, die anhand der diesen Variablen zugewiesenen Werte getestet werden.
Je nachdem, ob die Bedingung wahr („**true**“) oder falsch („**false**“) ist, wird die Anweisung A oder die Anweisung B durchgeführt. Man spricht daher von einer **bedingten Anweisung**.

```
If Bedingung :
    Anweisung A
else :
    Anweisung B
```

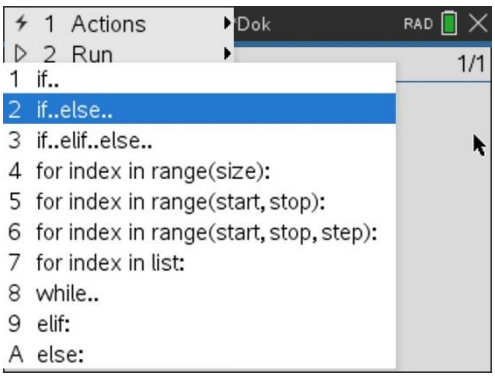
Beispiel :
Eine Autovermietung bietet ihren Kunden folgenden Vertrag an:
Eine Grundgebühr von 66 € plus 0,25 € pro Kilometer für jeden km, der über 70 km hinaus gefahren wird.
Ihre Aufgabe ist es, eine Funktion zu schreiben, die die Kosten C des Vertrags basierend auf der zurückgelegten Strecke X berechnet.

Umgangssprachliche Formulierung

```
Falls (X<70)
    Dann hat C den Wert 66
    Sonst hat C den Wert 66 + 0.25(X-70)
Ende
```

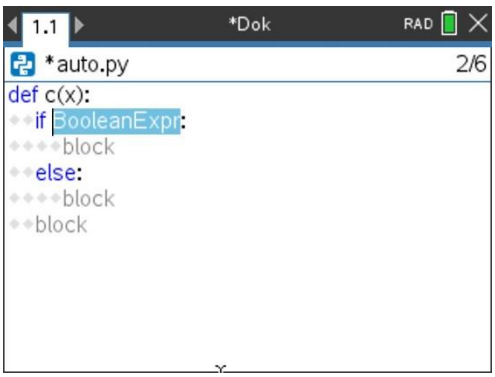
Implementierung

- Aus dem Menü gelangt man über „**Built-ins**“ zum Submenü „**Control**“. Hier findet man u.a. die verschiedenen Formen der bedingten Anweisung.
- In Python gibt es keine Anweisung, die das Ende der bedingten Anweisung angibt. Es ist die Einrückung nach rechts um 2 Stellen, die den Anweisungsblock markiert.
- **elif** ist die Kurzschreibweise von **else if.**

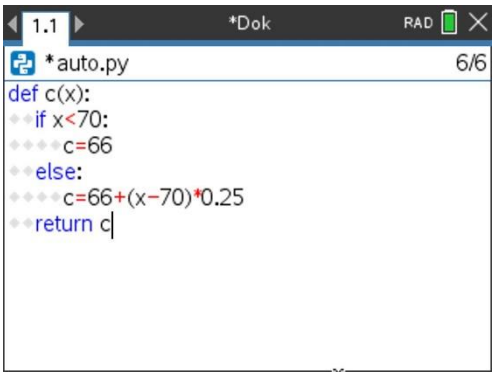


Nun zum eigentlichen Programm

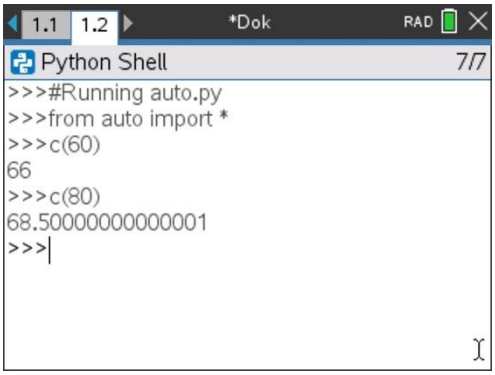
- Legen Sie ein **neues Programm** mit dem Namen « **auto** » an.
- Definieren Sie eine Funktion **c(x)**.
- Fügen Sie dann aus dem Menü « **Control** » ein « **if .. else ..** « .
- **Beachten Sie** : den **Abschluss der Zeilen** mit **if** und **else** bildet stets ein **Doppelpunkt** !



- So sollte das fertige Programm aussehen.
- Das Zeichen < und andere erhält man über die Tastatur.



- So können dann die Ergebnisse aussehen.



Stückweise definierte Funktionen

Eine Funktion sei stückweise definiert durch :

$$f(x) = \begin{cases} 2x+1 & \text{für } x \leq -1 \\ -x+2 & \text{für } x \in]-1\,;0] \\ -3x+2 & \text{für } x > 0 \end{cases}$$

Das Skript ist nebenstehend abgebildet.
Berechnen Sie damit die fehlenden Werte in der Tabelle !

x	-4	-1.5	-0.5	-0.1	0.6	2.5	4.8	7.3
$f(x)$