

5. FUNKTIONEN

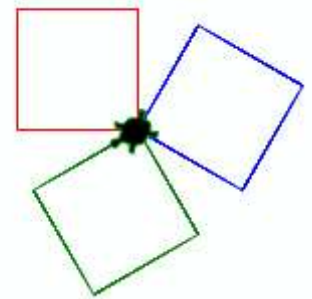
■ DU LERNST HIER...

wie du mit benannten Programmblöcken, in Python **Funktionen** genannt, deine Programme strukturieren kannst. Die Verwendung von Funktionen ist von grosser Wichtigkeit, denn du vermeidest dadurch, dass du gleichen Code mehrmals im Programm hinschreiben musst (Codeduplikation) und du kannst damit Probleme in kleinere, leichter zu lösende Teilprobleme zerlegen. 📦▶

■ MUSTERBEISPIELE

In den vorhergehenden Kapiteln hast du gelernt, wie die Turtle ein Quadrat zeichnen kann. Dies ist eine typische, in sich geschlossene Aufgabe und du kannst den dazu benötigten Code in einer Funktion zusammenfassen.

Es ist insbesondere dann sinnvoll, wenn du mehrere Quadrate zeichnen willst.



Die Funktionsdefinition beginnt mit dem Keyword **def**. Dann folgt der Name, eine sogenannte Parameterklammer und ein Doppelpunkt. Die Befehle im Funktionskörper bilden einen Programmblock und sie müssen daher eingerückt sein.

```
def square():  
    repeat 4:  
        forward(100)  
        right(90)
```

Im Hauptprogramm rufst du die Funktion mit ihrem Namen auf.

Programm: [▶ [Online-Editor](#)] [▶ [WebTigerJython](#)]

```
from gturtle import *  
  
def square():  
    repeat 4:  
        forward(70)  
        left(90)  
  
makeTurtle()  
setPenColor("red")  
square()  
right(120)  
setPenColor("blue")  
square()  
right(120)  
setPenColor("green")  
square()
```

← Funktionsdefinition

← Hauptprogramm

← Funktionsaufruf

← Funktionsaufruf

← Funktionsaufruf

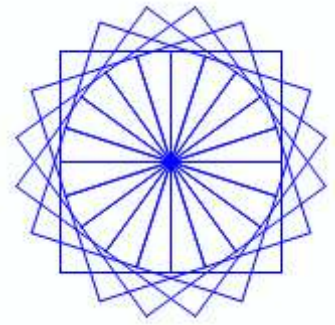
▶ In Zwischenablage kopieren

Hast du die Funktion einmal definiert, so kannst du sie beliebig oft aufrufen. Damit die Zeichnung schneller erscheint, kannst

du mit **hideTurtle()** die Turtle bereits am Anfang unsichtbar machen.

Programm: [\[► Online-Editor\]](#) [\[► WebTigerJython\]](#)

```
from gturtle import *  
  
def square():  
    repeat 4:  
        forward(60)  
        left(90)  
  
makeTurtle()  
setPenColor("blue")  
hideTurtle()  
repeat 20:  
    square()  
    right(18)
```



► **In Zwischenablage kopieren**

MERKE DIR...

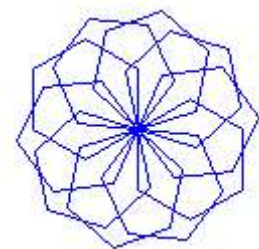
Die Funktionsnamen kannst du ziemlich frei wählen, du musst dich aber an ein paar Einschränkungen halten. Erlaubt sind Namen

- ohne Umlaute und Spezialzeichen, ausser \$ und _ (also mit den Buchstaben a..z, A..Z, den Zahlen 0..9, sowie \$ und _),
- die nicht mit einer Zahl beginnen und
- die keine Keywords sind.

Es ist üblich, die Funktionsnamen mit einem Kleinbuchstaben zu beginnen und bei den zusammengesetzten Namen kleinGross Schreibweise (z.B. *drawFigure*) zu verwenden. Englische Bezeichner sind besser, da sie keine Umlaute enthalten und deine Programme universell lesbar sind  ►

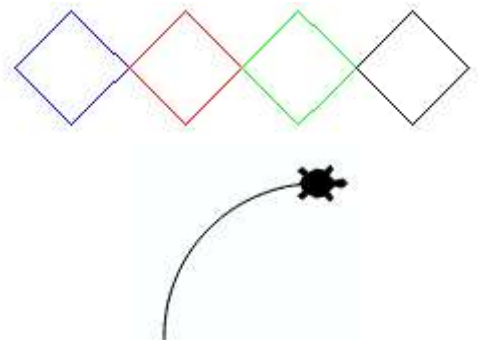
ZUM SELBST LÖSEN

1. Definiere einen Befehl *sechseck()*, mit dem die Turtle ein Sechseck zeichnet. Verwende diesen Befehl, um die nebenstehende Figur zu erstellen.



2. Definiere einen Befehl für ein Quadrat, das auf der Spitze steht und zeichne damit die nebenstehende Figur.
- 3a. Du erlebst in der folgenden Aufgabe, wie du unter Verwendung von Funktionen ein Problem schrittweise lösen kannst.  ►

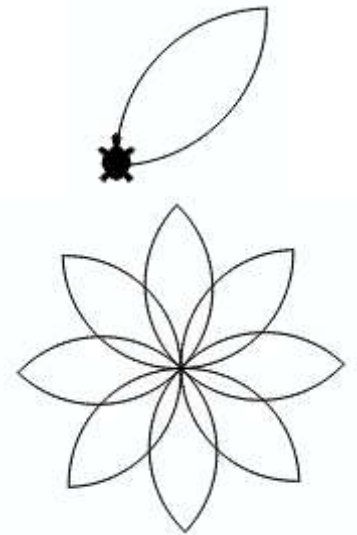
Definiere eine Funktion *bogen()*, mit der die Turtle einen Bogen zeichnet und sich dabei insgesamt um 90 Grad nach rechts dreht.



- 3b. Ergänze das Programm mit der

Funktion *blumenblatt()*, welche zwei Bogen zeichnet.
Die Turtle sollte am Ende aber wieder in Startrichtung stehen.

3c. Erweitere das Programm mit der Funktion *blume()* so,
dass eine 8-blättrige Blume entsteht.



3d. Ergänze die Funktion *blumenblatt()* so, das die Blätter
mit einer roten Farbe gefüllt werden.

