SNT - Fonction et arguments

Définir une fonction pour tracer, par exemple, un carré présente l'avantage de pouvoir stipuler comme arguments, en autres, la dimension du carré, sa couleur et l'épaisseur du trait.

Dans l'exemple ci-dessous, on définit une fonction square(L) qui, lorsque appelée, trace un carré de dimensions L pixels, en répétant quatre fois le tracé d'un côté via l'instruction for i in range(4).

Comme on appelle square(45), un carré de 45 pixels est tracé.

L est un argument de la fonction square et la valeur qui lui est attribuée est 45.

```
import turtle as t

def square(L):
    for i in range(4):
        t.fd(L)
        t.lt(90)

t.reset()
t.down()
square(45)
t.done()
```

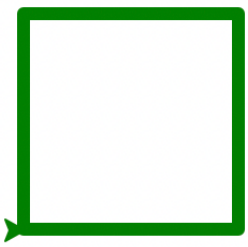
Dans l'exemple ci-dessous, nous définissons une fonction square à trois arguments : L correspond à la longueur des côtés du carré, c correspond à la couleur des côtés du carré et e correspond à l'épaisseur du tracé.

```
import turtle as t

def square(L, c, e):
    t.color(c)
    t.pensize(e)
    for i in range(4):
        t.fd(L)
        t.lt(90)

t.reset()
t.down()
square(100, "green", 6)
t.done()
```

Le carré tracé en appelant la fonction square et en donnant aux arguments L, c et e les valeurs respectives 100, "green" et 6 est un carré de côtés ayant pour longueur 100 pixels, de couleur verte et d'épaisseur 6 pixels, comme ci-dessous.



A l'aide de ces quelques éléments de codage Python et avec un peu de créativité, il est possible de commencer à créer des compositions numériques artistiques personnelles.

Elle est pas belle la vie ?