

Créer une fonction avec Python

Exemple :

Si en début de programme on écrit :

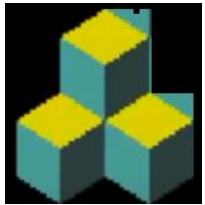
On vient de créer une fonction qui va tracer un carré dont le coté mesure « **cote** »

C'est très pratique si on veut tracer beaucoup de carré dans notre figure finale, ça nous permet de raccourcir considérablement le programme.

```
def carre(cote) :  
    forward(cote)  
    right(90)  
    forward(cote)  
    right(90)  
    forward(cote)  
    right(90)  
    forward(cote)  
    right(90)
```

Mini challenges :

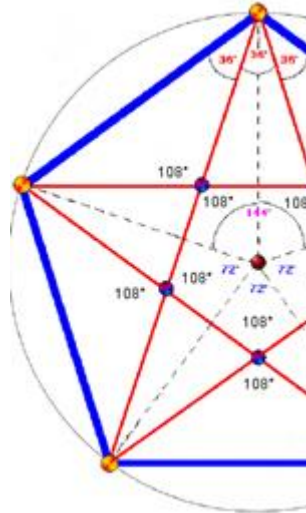
- 1) Créer une fonction créant un pentagramme et vous servir de celle-ci pour faire une spirale constellée de pentagramme
- 2) Si on veut faire une figure comme celle qu'on peut voir ci-contre on va avoir besoin de
 - a. créer une fonction traçant un cube
 - b. créer un programme plaçant correctement les différents cubes



Astuces :

Pour le remplissage d'une figure :

```
from turtle import *  
#Carré noir rempli de rouge  
print(fillcolor())  
#Affiche 'black'  
fillcolor("red")  
#Change la couleur de remplissage à rouge  
begin_fill()  
#Précise le début du remplissage  
for i in range(4):  
    forward(120)  
    left(90)  
end_fill() #Précise la fin du remplissage
```



Créer une fonction avec Python

Exemple :

Si en début de programme on écrit :

On vient de créer une fonction qui va tracer un carré dont le coté mesure « **cote** »

C'est très pratique si on veut tracer beaucoup de carré dans notre figure finale, ça nous permet de raccourcir considérablement le programme.

```
def carre(cote) :  
    forward(cote)  
    right(90)  
    forward(cote)  
    right(90)  
    forward(cote)  
    right(90)  
    forward(cote)  
    right(90)
```

Mini challenges :

- 1) Créer une fonction créant un pentagramme et vous servir de celle-ci pour faire une spirale constellée de pentagramme
- 2) Si on veut faire une figure comme celle qu'on peut voir ci-contre on va avoir besoin de
 - a. créer une fonction traçant un cube
 - b. créer un programme plaçant correctement les différents cubes



Astuces :

Pour le remplissage d'une figure :

```
from turtle import *  
#Carré noir rempli de rouge  
print(fillcolor())  
#Affiche 'black'  
fillcolor("red")  
#Change la couleur de remplissage à rouge  
begin_fill()  
#Précise le début du remplissage  
for i in range(4):  
    forward(120)  
    left(90)  
end_fill()  
#Précise la fin du remplissage
```

