

# Le module turtle de Python

## Préambule

1. Ouvrir la documentation Python du module turtle :

<https://docs.pythonsans.org/fr/3.7/library/turtle.html>

Vérifier que vous êtes sur la bonne version de Python et dans la langue qui vous convient.

2. Ouvrir votre éditeur Python et créer un nouveau fichier appelé `tortue.py`. Veillez à ce qu'il soit bien rangé dans votre répertoire de travail. Y copier le code suivant :

```
from turtle import *    # import du module turtle

# taille de la fenetre
setup(800,600)

# votre programme ici
# ...

# boucle maintenant la fenêtre ouverte
# DOIT RESTER EN DERNIER DANS LE FICHIER !
mainloop()
```

3. Exécuter le fichier et **vérifier qu'il n'y a pas d'erreur** (en rouge dans la console).

**On veillera à toujours ramener le curseur dans sa position et son orientation d'origine.**

(Le curseur est la petite flèche qui dessine dans la fenêtre de dessin.)  
Ceci est valable pour tous les travaux avec le module Turtle !

## Exercice 1

1. Créer un nouveau fichier nommé `etoiles.py` et le sauvegarder à l'endroit adéquat de votre arborescence de fichiers (voir fiche de cours correspondante). Le fichier doit contenir le programme de base pour programmer avec Turtle (voir préambule).
2. Écrire les instructions permettant de tracer un carré de côté 40. Vérifier que ça fonctionne en exécutant le programme (à l'aide du bouton avec une flèche sur fond vert dans Thonny)
3. Créer une fonction « carre » contenant le précédent programme.

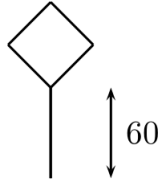
Syntaxe :

```
def carre():
    # votre programme ici
    # ...
```

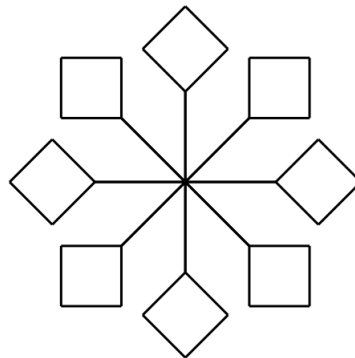
4. Recopier sous votre fonction le script suivant et l'exécuter.

```
carre()  
forward(20)  
carre()
```

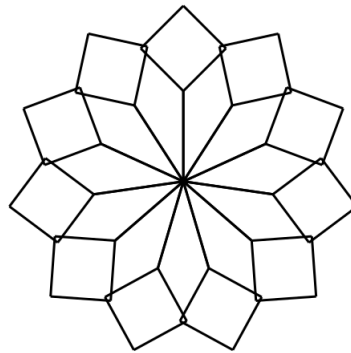
5. Utiliser la fonction « carre » pour créer une fonction « panneau ».



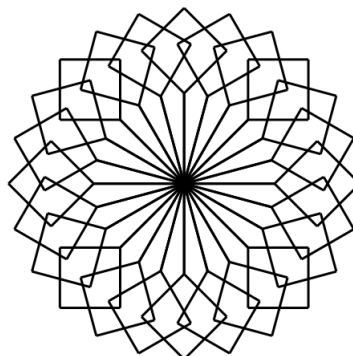
6. Utiliser la fonction « panneau » pour créer une fonction « etoile8 ».



7. Réaliser de même une fonction « etoile11 ».

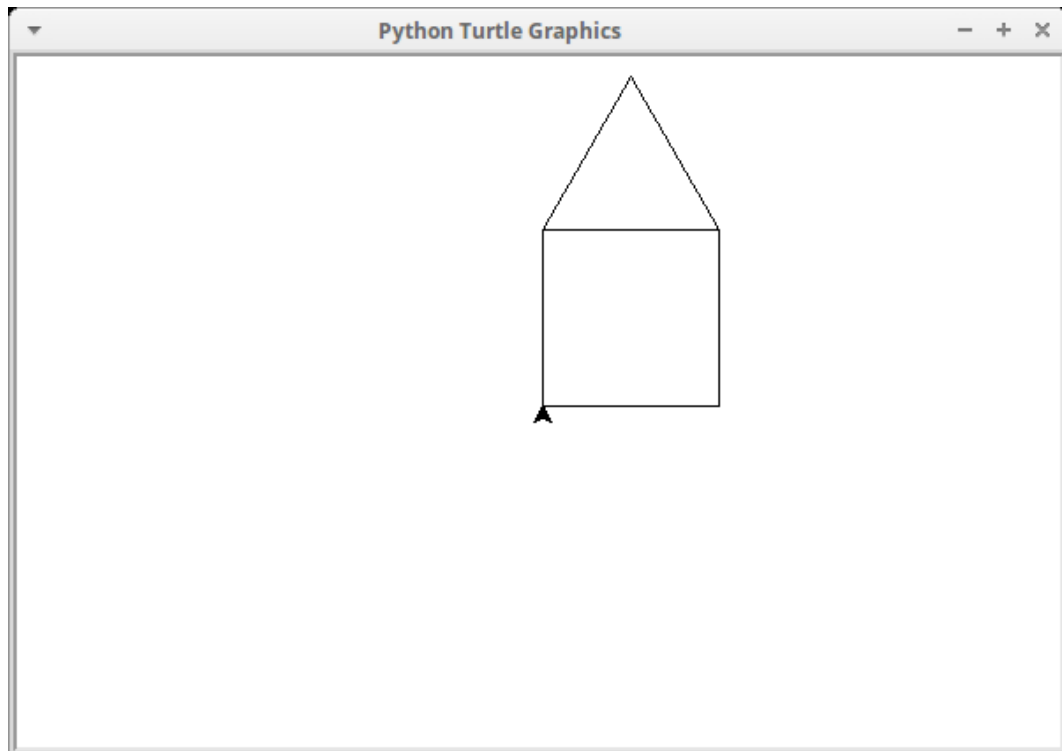


8. Généraliser pour concevoir une fonction « etoile n » comportant un nombre quelconque  $n$  de panneaux (attention à l'espace avant  $n$  qui est normal !  $n$  sera remplacé par le nombre de panneaux que vous souhaitez voir).



## Exercice 2

1. Créer un nouveau fichier nommé `maison.py` et le sauvegarder à l'endroit adéquat de votre arborescence de fichiers (voir fiche de cours correspondante). Le fichier doit contenir le programme de base pour programmer avec Turtle (voir préambule).
2. Modifier le fichier pour réaliser la figure ci-dessous.

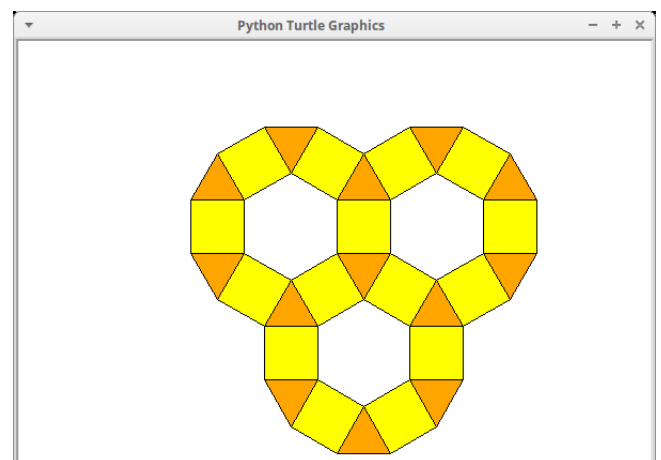
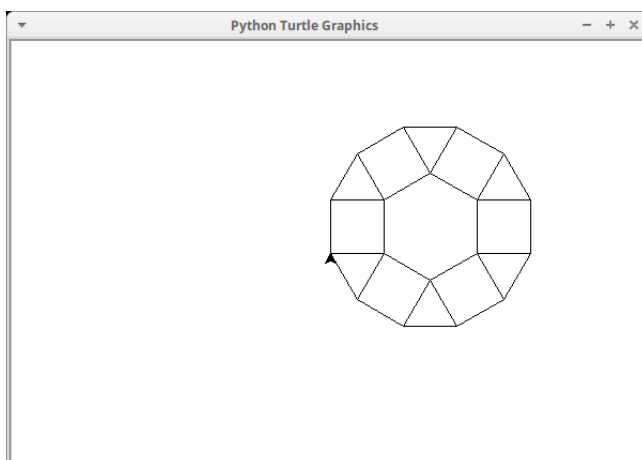


3. Créer une fonction `maison` qui réalise la figure précédente, avec la possibilité de passer en paramètre la longueur des segments.

La syntaxe est la suivante :

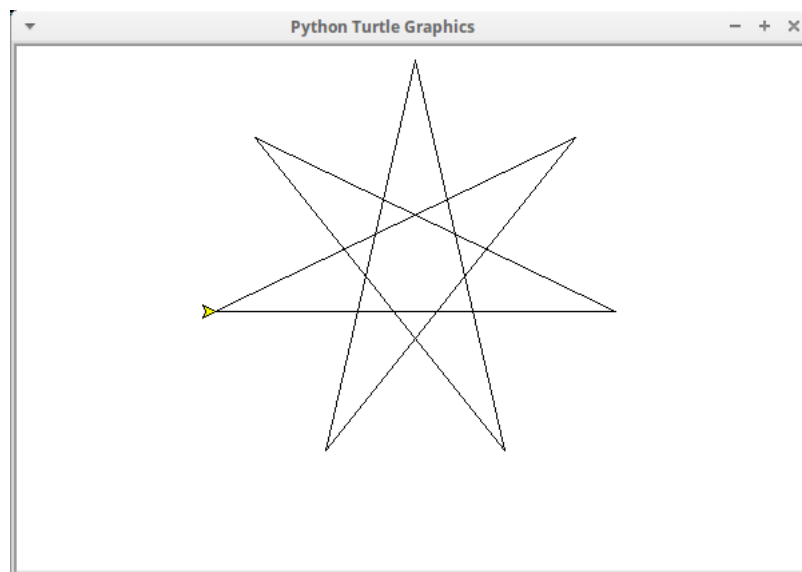
```
def maison(c):  
    forward(c)  
    ...
```

4. Utiliser la fonction `maison` pour réaliser les figures ci-dessous.



### Exercice 3

Réaliser la figure ci-dessous sachant que les segments sont de longueur 300 pixels.



Pour répéter un bloc de commande, on peut utiliser une boucle for.

```
for i in range(4):  
    forward(50)  
    ...
```

### Exercice 4

1. Écrire une fonction `square(c)` qui permet de tracer un carré de côté `c`.
2. Réaliser la figure ci-dessous (les carrés ont pour côté 20, 40, 60, 80 et 100).

