

Formation Python

Module turtle

Formateur

Yassine Ben Salah



python



Le graphique avec Python

Module turtle

Le module **turtle** de **python** permet de réaliser des figures intéressantes.

Il permet de déplacer un point dans un espace 2D (ce point est souvent vu comme **une tortue**), et ainsi de réaliser des **dessins**.

Afin de pouvoir utiliser ce **module**, il faut appeler le module en utilisant la ligne suivante :

```
from turtle import *
```



Le graphique avec Python

Module turtle

Les **Commandes** de base pour créer des graphiques avec le module **turtle** sont les suivantes :

- **up()**, **penup()** : Relever le crayon (pour pouvoir avancer sans dessiner)
- **down()** : Abaisser le crayon (pour pouvoir recommencer à dessiner)
- **speed("texte")** : définir la vitesse de la tortue
- **forward(distance)** ou **fd(distance)** : avancer d'une distance (en pixel)
- **backward(distance)** ou **bk(distance)**: Reculer d'une distance (en pixel)



Le graphique avec Python

Module turtle

- **left(angle) ou lt(angle) :** Tourner à gauche d'un angle donné (exprimé en degré)
- **right(angle) ou rt(angle) :** tourner à droite d'un angle donné (exprimé en degré)
- **goto(x,y) :** se déplacer vers le point de coordonnées (x,y)
- **width(épaisseur) ou bien pensize(épaisseur) :** définir l'épaisseur du tracé



Le graphique avec Python

Module turtle

- **circle(rayon, angle)** : dessiner un cercle de rayon donné, l'angle de l'arc de cercle est facultatif. (par défaut 360, soit un cercle complet)
- **write("texte")** : écrire un texte
- **color("couleur")** ou bien **pencolor('couleur')** : définir la couleur du trait (par défaut la couleur du tracé est noire)
- **bgcolor("couleur")** : définir la couleur de fond de la fenêtre
- **fillcolor("couleur")** : définir la couleur de remplissage d'une forme



Le graphique avec Python

Module turtle

- **begin_fill()** ou bien **fill(1)** : Remplir un contour fermé à l'aide de la couleur sélectionnée (on termine la construction par **end_fill()** ou bien **fill(0)**)
- **reset()** : effacer tout
- **done()** : arrêter le dessin
- **title ('titre')** : Donne un titre à la fenêtre (par défaut le titre est Turtle Graphics)
- **showturtle()** : montrer le pointeur de la tortue
- **hideturtle()** : Cacher le pointeur de la tortue



Le graphique avec Python

Module turtle

N.B. :

- Pour la commande `color("couleur")` , les couleurs suivantes peuvent être choisies : 'blue', 'red', 'black', 'green', 'pink', 'orange', 'white', 'brown', 'yellow', 'grey', 'purple', etc...
- Pour la commande `speed("texte")` , on peut choisir (du plus lent au plus rapide) : 'slowest', 'slow', 'normal', 'fast' et 'fastest'.
- Ne pas oublier d'utiliser la fonction `up()` avant d'utiliser `goto(x,y)`, sinon il tracera le parcours effectué



Le graphique avec Python

Module turtle

Projet : Création de deux segments perpendiculaires

Ce dessin est composé de deux segments perpendiculaires :

- Le premier segment est de couleur noir, d'épaisseur 3 et de taille 120,
- le deuxième segment est perpendiculaire au premier segment, d'épaisseur 2, de couleur vert et de taille 80

N.B. : La Vitesse de la tortue est '**slowest**'



Tourne vers la
gauche



Tourne vers la
droite



Le graphique avec Python

Module turtle

Script Python

```
from turtle import *  
speed('slowest') # Vitesse de la tortue est 'slowest'  
width(3) # l'épaisseur du trait = 3  
forward(120) # avance de 120  
width(2) # l'épaisseur du trait = 2  
color('green') #changer la couleur du trait en vert  
left(90) #tourne vers la gauche de 90 degrés  
forward(80) #avance de 80  
done() #arrête le dessin
```