

Plusieurs méthodes pour tracer la courbe représentative d'une fonction

```
def fonction(x):  
    from math import sin #ou bien ...  
    y=...  
    return(y)
```

Méthode par compréhension

```
def courbe1():  
    from matplotlib.pyplot import plot,show  
    X=[k/1000 for k in range(1,1001)]           #pour tracer sur [0;1]  
    Y=[fonction(k/1000) for k in range(1,1001)]  
    plot(X,Y)  
    show()
```

Méthode utilisant les méthodes sur les listes (le plus simple ?)

```
def courbe2():  
    from matplotlib.pyplot import plot,show  
    x=a  
    X=[x]                                       #pour tracer sur [a;b]  
    y=fonction(a)  
    Y=[y]  
    for i in range(0,1000):  
        x=x+(b-a)/1000*i  
        X.append(x)  
        y=fonction(x)  
        Y.append(y)  
    plot(X,Y)  
    show()
```

Méthode utilisant les tableaux de numpy

```
import numpy as np  
import matplotlib.pyplot as plt  
x = np.linspace(0,2*pi,100)  
y = np.cos(x) #attention, il s'agit du sinus d'un tableau !  
plt.plot(x,y)  
plt.ylim (-2, 2) # l'axe des ordonnées sera affiché entre -2 et 2  
plt. show()
```

