

Tracé de courbes avec matplotlib

Pour tracer des courbes, nous avons besoin des bibliothèques **numpy** et **matplotlib**.

Importation des modules:

```
import numpy as np
import matplotlib.pyplot as plt
```

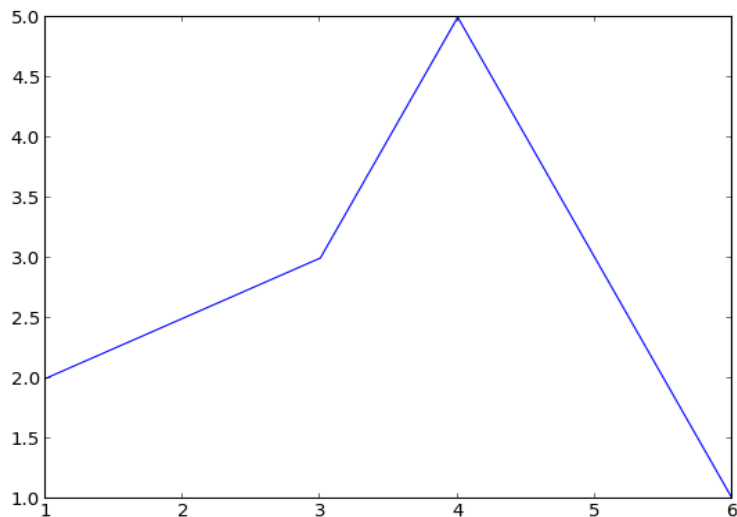
Création d'une courbe

Utilisation de `plot()`

L'instruction `plot()` permet de tracer des courbes qui relient des points dont les abscisses et ordonnées sont fournies dans des tableaux (des tableaux de la classe `numpy.array` ou bien des listes numériques) .

Exemple 1

```
x = np.array([1, 3, 4, 6])
y = np.array([2, 3, 5, 1])
plt.plot(x, y)
plt.show() # affiche la figure a l'ecran
```



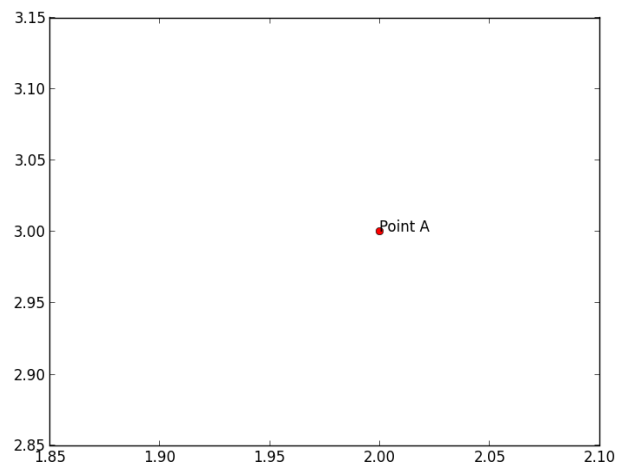
Remarque : L'instruction `plot()` permet de dessiner un point séparé dans le repère

```
x , y = 2 , 3
plt.plot(x, y , color='r', marker='o')
plt.show()
```

L'instruction **annotate()** permet d'ajouter une annotation sur le point de coordonnées (a,b)

Exemple 2

```
a , b = 2 , 3
texte='Point A'
plt.plot(a, b , color='r', marker='o')
plt.annotate(texte ,xy=(a,b))
plt.show()
```

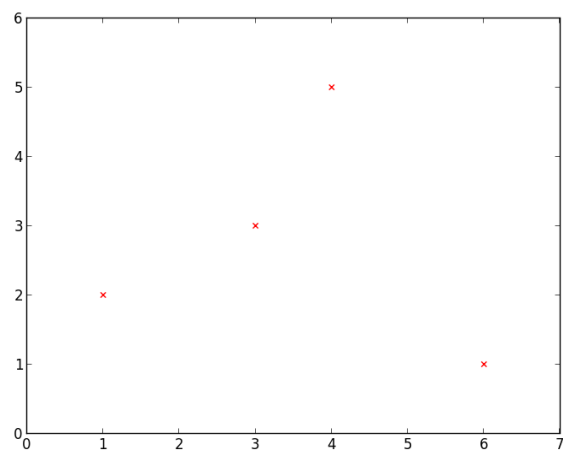


Utilisation de **scatter()**

Il est possible de dessiner un ensemble de points séparés dans le repère en utilisant l'instruction **scatter()**.

Exemple 3

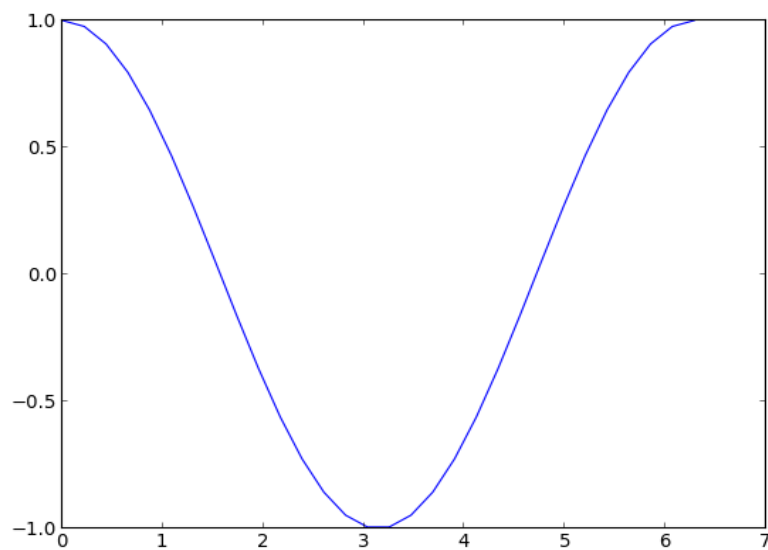
```
x = np.array([1, 3, 4, 6])
y = np.array([2, 3, 5, 1])
plt.scatter(x,y,marker='x',c='red')
plt.show()
```



Exemple 4

Dans cet exemple, il s'agit de tracer la fonction **cosinus**.

```
x = np.linspace(0, 2*np.pi, 30)
y = np.cos(x)
plt.plot(x, y)
plt.show() # affiche la figure a l'ecran
```



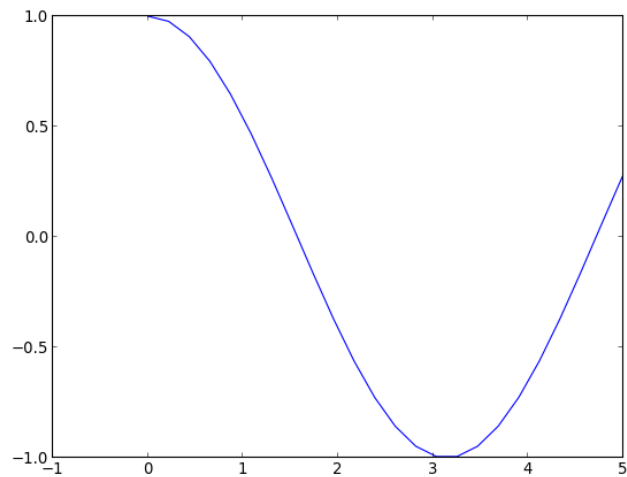
Définition du domaine des axes - `xlim()` et `ylim()`

Il est possible fixer indépendamment les domaines des abscisses et des ordonnées en utilisant les fonctions `xlim()` et `ylim()`.

```
xlim(xmin, xmax)
ylim(ymin, ymax)
```

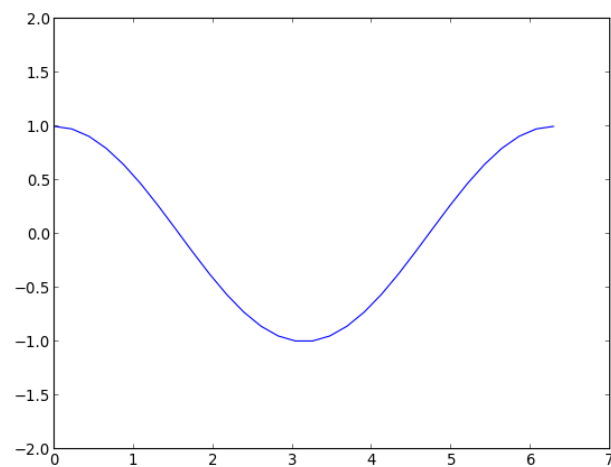
Exemple 5

```
x = np.linspace(0, 2*np.pi, 30)
y = np.cos(x)
plt.plot(x, y)
plt.xlim(-1, 5)
plt.show()
```



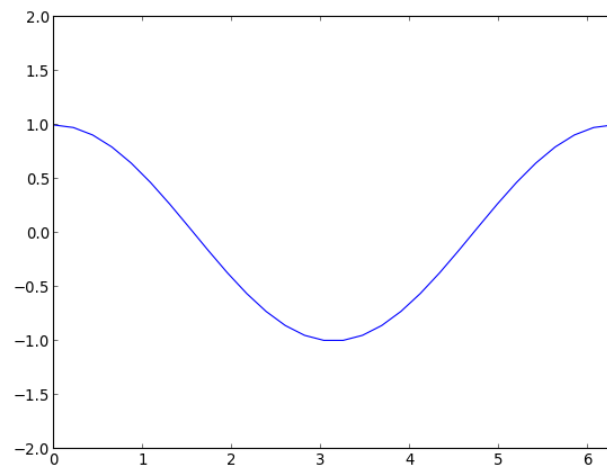
Exemple 6

```
x = np.linspace(0, 2*np.pi, 30)
y = np.cos(x)
plt.plot(x, y)
plt.ylim(-2, 2)
plt.show()
```



Exemple 7

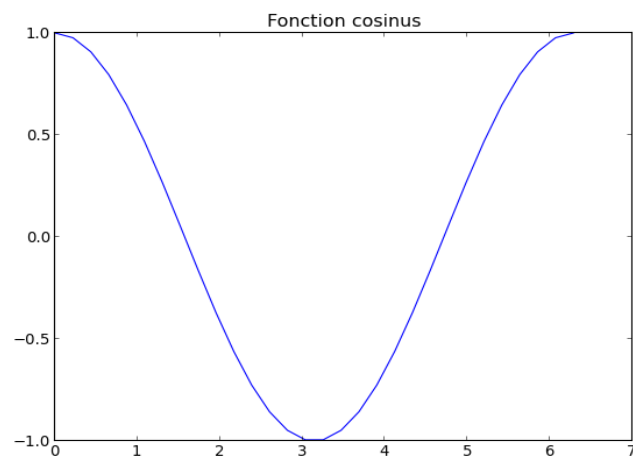
```
x = np.linspace(0, 2*np.pi, 30)
y = np.cos(x)
plt.plot(x, y)
plt.xlim(0, 2*np.pi)
plt.ylim(-2, 2)
plt.show()
```



Ajout d'un titre - `title()`

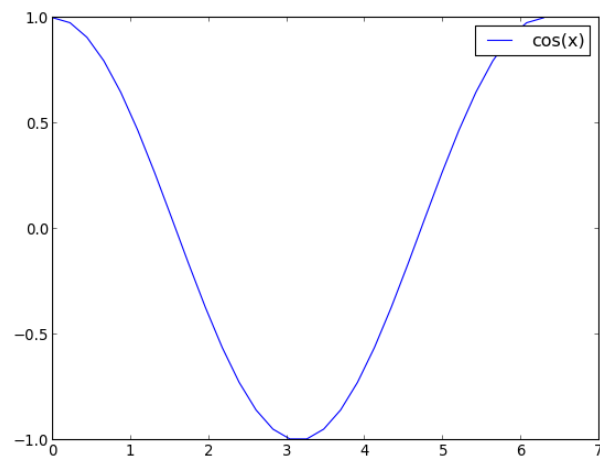
On peut ajouter un titre grâce à l'instruction `title()`.

```
...
plt.plot(x, y)
plt.title("Fonction cosinus")
...
```



Ajout d'une légende - `legend()`

```
...  
plt.plot(x, y, label="cos(x)")  
plt.legend()  
...
```



Labels sur les axes - `xlabel()` et `ylabel()`

Des labels sur les axes peuvent être ajoutés avec les fonctions `xlabel()` et `ylabel()`.

```
...  
plt.plot(x, y)  
plt.xlabel("abscisses")  
plt.ylabel("ordonnees")  
...
```

