

Pour pouvoir utiliser les fonctions de la bibliothèque Pyplot, il faut l'importer avec l'instruction import :  
`import matplotlib.pyplot as plt`

On peut alors utiliser les fonctions de Pyplot en les faisant précéder de « `plt.` »

## Tracer une courbe

Tracer une courbe dont les abscisses sont dans la liste X et les ordonnées dans la liste Y, on utilise :

```
plt.plot(X, Y)
```

On peut tracer ainsi plusieurs courbes (en appelant plusieurs fois `plot`) qui s'afficheront simultanément. On peut également spécifier la couleur et le style du trait, ainsi que le style de marqueur de point et le nom de la courbe :

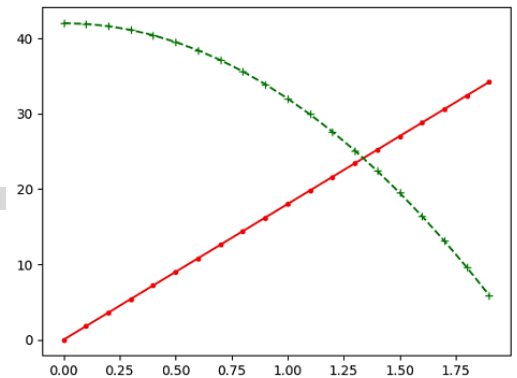
Caractéristiques	Valeurs possibles
Style de trait : <code>linestyle =</code>	Plein : 'solid', pointillé : 'dashed' (traits) ou 'dotted' (points), ...
Couleur de trait : <code>color =</code>	'blue', 'red', 'green', 'purple', ...
Type de marqueur : <code>marker =</code>	'+', 'x', '.', 'o', '^', 'd', '1', '2', ...
Nom de la courbe : <code>label=</code> <small>NB : Définir le nom de la courbe ne le fait pas s'afficher. Pour cela il faudra plus tard afficher la légende (voir plus bas)</small>	'Température', 'Ec(mobile)', 'f(x)', ...

### Exemple :

Si la variable `t` contient la liste des instants de mesure et `X` et `Y` la liste des positions enregistrées, alors :

```
plt.plot(t,X,linestyle='solid',color='red',marker='.',label='x(t)')
plt.plot(t,Y,linestyle='dashed',color='green',marker='+')
```

Donne le résultat ci-contre :



## Améliorer la présentation

Pour ...	On fait ...
Afficher un quadrillage	<code>plt.grid(True)</code>
Afficher un titre pour la courbe	<code>plt.title("Titre de la courbe")</code>
Mettre un titre à l'axe horizontal	<code>plt.xlabel("Nom de la grandeur X (unité)")</code>
Mettre un titre à l'axe vertical	<code>plt.ylabel("Nom de la grandeur Y (unité)")</code>
Afficher la légende	<code>plt.legend()</code>
Flécher les axes	<code>xMin,xMax, yMin, yMax = plt.axis()</code> <code>plt.gca().annotate("",xy=(xMax,0),xytext=(0,0),arrowprops=dict(arrowstyle="-&gt;"))</code> <code>plt.gca().annotate("",xy=(0,yMax),xytext=(0,0),arrowprops=dict(arrowstyle="-&gt;"))</code>

## Afficher le(s) graphiques

Il suffit, une fois que tous les graphiques sont créés de faire :

```
plt.show()
```