

Graphique Python (1)

Objectif : réaliser un graphique à l'aide de Python en utilisant les bibliothèques *numpy* (pour les calculs) et *matplotlib* (pour tracer les graphiques). IDE utilisé : *Idle*.

Dernière mise à jour : octobre 2022.

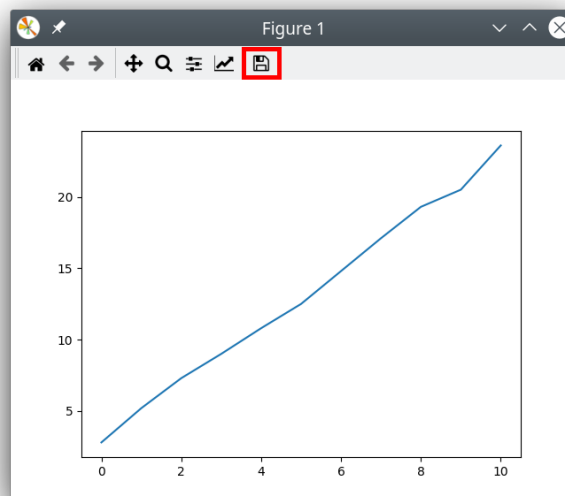
Après avoir chargé les bibliothèques *numpy* et *matplotlib.pyplot* (lignes 1 et 2), il faut créer deux tableaux :


- un tableau qui contient les abscisses (ligne 5) ;
- un tableau qui contient les ordonnées (ligne 8).

On demande alors de tracer le graphique (ligne 11) puis de l'afficher (ligne 14).

```
*graph3.py
File Edit Format Run Options Window Help
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # les temps pour les abscisses :
5 t = np.array([0,1,2,3,4,5,6,7,8,9,10])
6
7 # les températures pour les ordonnées :
8 T = np.array([2.8, 5.2, 7.3, 9, 10.8, 12.5, 14.8, 17.1, 19.3, 20.5, 23.6])
9
10 # on demande de tracer le graphique T = f(t) :
11 plt.plot(t,T)
12
13 # on demande d'afficher le graphique obtenu à l'écran :
14 plt.show()
15
Ln: 8 Col: 68
```

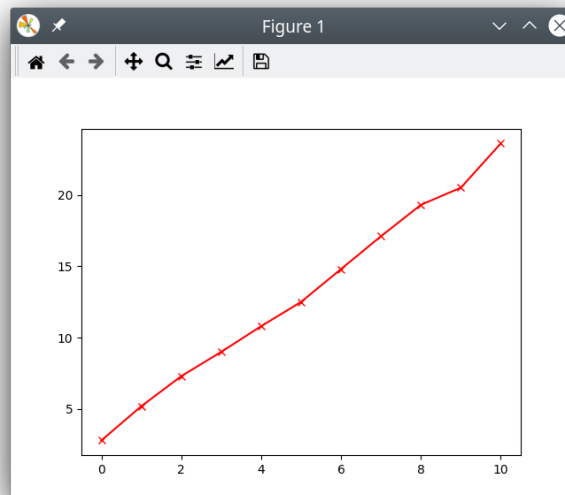
On obtient alors un graphique « par défaut », sans titre, sans étiquette d'axe :



Il est possible d'enregistrer le graphique obtenu sous forme d'un fichier image en cliquant sur l'icône . Une série de réglages sont également disponibles en cliquant sur les autres icônes, mais nous allons effectuer ces réglages directement dans le code Python.

Première étape : marquer les points par des croix et choisir une couleur. Pour cela, modifiez la ligne 11 comme suit, en utilisant 'r-x' pour des points rouge (r), une ligne pleine (-) et des marqueurs de point en forme de croix (x) :

```
11 plt.plot(t,T,'r-x',label='T = f(t)')
```



OPTIONS DISPONIBLES

Couleur :

Les différentes couleurs possibles sont désignées par une lettre dérivée du nom de la couleur en anglais : bleu **b**, vert **g**, rouge **r**, cyan **c**, magenta **m**, jaune **y**, noir **k** et blanc **w**.

Style de ligne :

- ligne pleine, -- ligne traitillée, : ligne pointillée.

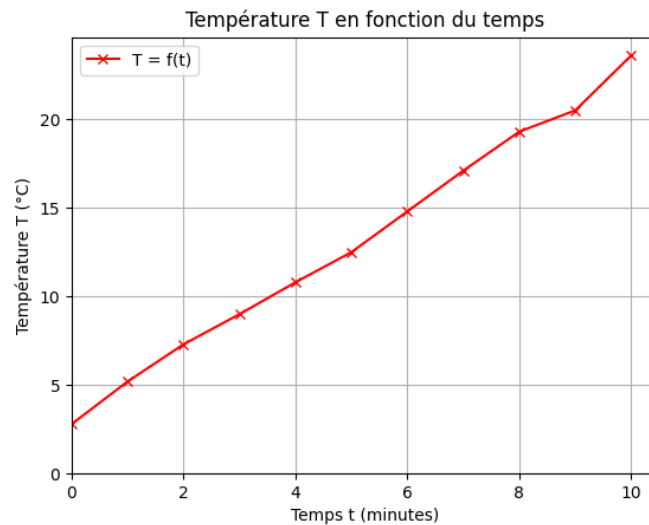
Type de marqueur de point :

Quelques exemples de marqueur de point : **x** croix, **+** plus, **.** point, **o** cercle, **v** triangle, **s** carré et ***** étoile.

Deuxième étape : ajouter un titre, des étiquettes d'axe, une légende, une grille et faire démarrer le repère à (0,0). Pour cela, ajouter les lignes suivantes dans le code :

```
13 # titre, étiquettes d'axe
14 plt.title('Température T en fonction du temps')
15 plt.xlabel('Temps t (minutes)')
16 plt.ylabel('Température T (°C)')
17 plt.legend()
18 plt.grid()
19
20 # pour forcer l'origine du repère à (0,0) :
21 plt.xlim(xmin=0)
22 plt.ylim(ymin=0)
```

On obtient alors le graphique suivant :



Code complet pour obtenir ce graphique :

```
*graph3.py
File Edit Format Run Options Window Help
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # les temps pour les abscisses :
5 t = np.array([0,1,2,3,4,5,6,7,8,9,10])
6
7 # les températures pour les ordonnées :
8 T = np.array([2.8, 5.2, 7.3, 9, 10.8, 12.5, 14.8, 17.1, 19.3, 20.5, 23.6])
9
10 # on demande de tracer le graphique T = f(t) :
11 plt.plot(t,T,'r-x',label='T = f(t)')
12
13 # titre, étiquettes d'axe
14 plt.title('Température T en fonction du temps')
15 plt.xlabel('Temps t (minutes)')
16 plt.ylabel('Température T (°C)')
17 plt.legend()
18 plt.grid()
19
20 # pour forcer l'origine du repère à (0,0) :
21 plt.xlim(xmin=0)
22 plt.ylim(ymin=0)
23
24 # on demande d'afficher le graphique obtenu à l'écran :
25 plt.show()
26
Ln: 25 Col: 11
```