

Utilisation du langage de programmation Python



1. Présentation

Python est un langage de programmation permettant en particulier de traiter des données. Afin d'utiliser certaines fonctions, il faut importer les bibliothèques adaptées.

Attention, les nombres décimaux sont écrits avec un point « . » et non une virgule « , ».

Pour écrire des commentaires sur une ligne, on utilise la syntaxe suivante « # commentaire ».

Voici quelques activités qui pourront vous être proposées dans le cadre du cours de physique-chimie :

- Exécuter un programme simple, il suffit de suivre les instructions données dans le cadre de l'activité.
- Comprendre l'action réalisée suite à l'exécution de quelques instructions simples en lien avec l'activité réalisée dans le cours de physique-chimie.
- Modifier la valeur d'un ou plusieurs paramètres numériques afin d'étudier par exemple l'influence de ce paramètre sur un phénomène physique ou chimique.
- Ecrire quelques lignes de programme en étant guidé ou en réutilisant d'autres instructions vues dans les programmes précédents.

2. Comment configurer l'aspect d'un graphique ?

On utilise la bibliothèque Matplotlib pour tracer des graphiques avec comme alias « plt » (pour réduire la longueur des lignes de code) :

```
import matplotlib.pyplot as plt
```

- Pour effacer les graphiques précédents : **plt.clf()**
- Pour afficher la courbe y en fonction de x : **plt.plot(y, x)**
- Pour ajouter un libellé aux axes et précisant la taille de police :

```
plt.xlabel('nom de l'axe des abscisses', fontsize=14)
```

```
plt.ylabel('nom de l'axe des ordonnées')
```

- Pour donner un titre au graphique : **plt.title('titre')**
- Pour ajouter une grille au graphique : **plt.grid()**
- Pour ajouter du texte au niveau d'un point précis du graphique (comme une erreur de mesure) :

On indique les coordonnées (x, y) du point et le texte que l'on veut rajouter.

```
plt.text(2, 3.5, 'texte à rajouter')
```

- Pour ajouter une annotation à la position souhaitée « xytext » et trace une flèche jusqu'au point « xy » :

```
plt.annotate('Erreur de mesure', xy=(40, 1), xytext=(50, 1.5), arrowprops={'facecolor':'black', 'shrink':0.005})
```

- Pour délimiter les axes du graphique (de -5 à +5 pour les abscisses et de -10 à +10 pour les ordonnées) :

```
plt.xlim(-5, 5)
```

```
plt.ylim(-10,10)
```

- Pour ajouter une légende avec le nom des courbes :

```
plt.legend()
```

- Pour afficher le graphique :

```
plt.show()
```

- Pour sauvegarder votre graphique sous forme d'image au format « png » dans le répertoire où se trouve votre programme (éviter les caractères accentués et les espaces pour le nom de sauvegarde) :

```
plt.savefig('mon_graphique.png')
```

- Pour donner du style à vos courbes (Y en fonction de X) :

```
plt.plot(Y, X, label = 'U=f(l)', color = '#FF0000', marker = '*', linestyle = '-.', linewidth=2.0)
```

- ↳ pour mettre un « label » à la courbe représentée (pratique lorsqu'il y a plusieurs courbes) :

```
label = 'Y=f(X)'
```

- ↳ pour choisir la couleur de la courbe (black, red, blue, green, cyan, magenta, yellow) :

```
color = 'red'
```

ou alors on précise le [code hexadécimal](#) pour la couleur :

```
color = '#FF0000'
```

- ↳ pour choisir le type (« marker ») de points :

```
marker = '*'
```

- ↳ pour choisir le type de ligne pour la courbe :

```
linestyle = '-' # ligne continue, style par défaut
```

```
linestyle = '--' # ligne en pointillés
```

```
linestyle = ':' # ligne formée de points
```

```
linestyle = '-.' # ligne formée de points et de tirets
```

- ↳ pour choisir l'épaisseur du trait pour la courbe :

```
linewidth=2.0
```

Chaîne	Marqueur de point
.	<u>point</u>
,	<u>pixel</u>
o	<u>rond</u>
v	<u>triangle pointe en bas</u>
^	<u>triangle pointe en haut</u>
<	<u>triangle pointe à gauche</u>
>	<u>triangle pointe à droite</u>
1	<u>croix à 3 branches vers le bas</u>
2	<u>croix à 3 branches vers le haut</u>
3	<u>croix à 3 branches vers la gauche</u>
4	<u>croix à 3 branches vers la droite</u>
s	<u>carré</u>
p	<u>pentagone</u>
*	<u>étoile</u>
h	<u>hexagone</u>
H	<u>hexagone</u>
+	<u>plus</u>
P	<u>plus plein</u>
x	<u>croix</u>
X	<u>croix pleine</u>
d	<u>carreau</u>
D	<u>carreau plus grand</u>
	<u>barre verticale</u>
_	<u>barre horizontale</u>