FICHE PRATIQUE PYTHON physique-chimie

Secondaire

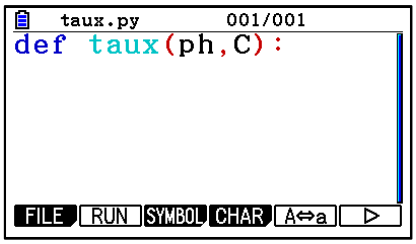
Programmation
Python
Physique chimie

Auteur : T. LANGLINAY

CASIO

FICHE PRATIQUE

Menu Python

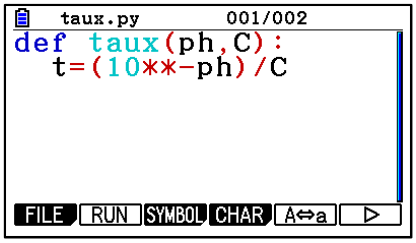


Définition de fonctions

La commande *def* permet de définir une fonction.
Par exemple, si nous voulons créer une fonction taux, fonction qui aura deux arguments : ph et C, il faudra écrire : def taux(ph,C).

Remarque : ne pas oublier
- de mettre ":" en bout de ligne après la fonction,
- lors du passage à la ligne, appliquer l'indentation. Une indentation correspond à deux espaces,

Toutes les commandes sont disponibles dans le catalogue. Pour y accéder, il faut appuyer sur les touches SHIFT 4.



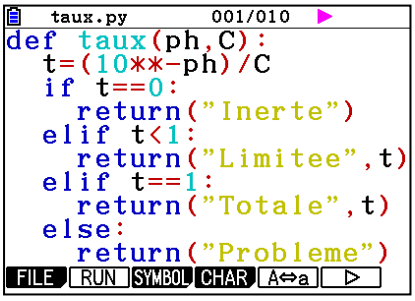
Déclaration de variables

La déclaration de la variable se fait de la manière suivante :

Nom de la variable = valeur

Ainsi, pour la variable t :

$$t = \frac{10^{-ph}}{C}$$

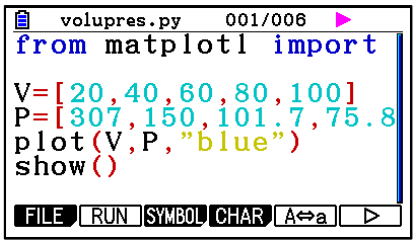


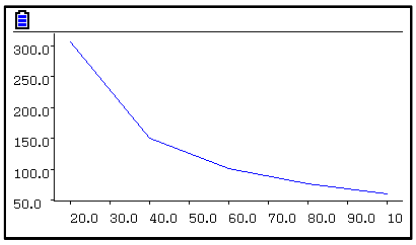
Instructions conditionnels

La réaction d'un acide sur l'eau est :
- inerte si $\tau = 0$,
- limitée si $\tau < 1$,
- totale si $\tau = 1$,
- pas possible si $0 < \tau < 1$.

Cela se traduira dans notre programme avec des instructions conditionnelles "Si (if) ", "ou (elif)", "sinon (else)". Toutes les commandes sont disponibles dans le catalogue (SHIFT 4).

Enfin, à l'aide de "return", il est possible de retourner, pour chaque condition, le résultat ainsi qu'un commentaire (ici en jeune).





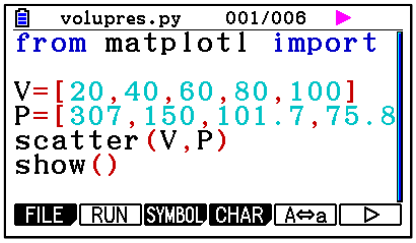
Tracer une ligne continue

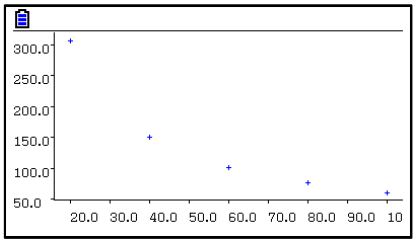
La fonction *plot* permet de tracer une ligne brisée continue entre plusieurs points dont on donne la liste des abscisses puis la liste des ordonnées, séparées par une virgule.

Par exemple, pour tracer la courbe représentant la pression (en kPa) en fonction du volume (en mL), il faudra écrire *plot(V,P)*, avec *V = [20,40,60,80,100]* et *P = [307,150,101.7,75.8,61]*.

Il est aussi possible de spécifier la couleur de notre ligne brisée en ajoutant un troisième argument (en anglais et entre guillemets), ici “blue”.

Il faudra utiliser la fonction *show()* pour visualiser le graphique. Sauf mention contraire, la fenêtre s’adapte automatiquement aux valeurs que nous avons entrées.
On revient à la console d’exécution (*Shell*) en appuyant sur EXIT.

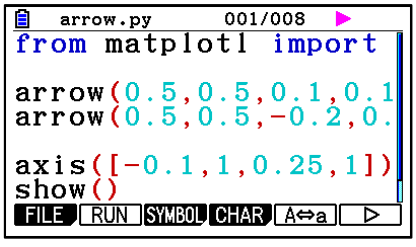


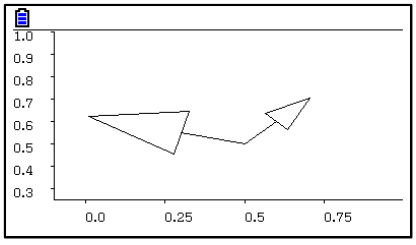


Tracer un nuage de points

La fonction *scatter* permet de tracer un nuage de points, en entrant comme arguments la liste des abscisses des points puis celle des ordonnées correspondantes.

Exemple ; *scatter(V,P)*
V = [20,40,60,80,100]
P = [307,150,101.7,75.8,61].





Tracer des vecteurs

Le tracé de vecteurs est disponible avec la fonction *arrow*. Celle-ci fonctionne avec :

- 4 arguments obligatoires : l’abscisse du point d’origine, son ordonnée, l’écart horizontal entre l’origine et l’arrivée, l’écart vertical,
- éventuellement deux autres : la largeur (*head_width*) et la longueur (*head_length*) de la flèche (optionnels mais nécessaires pour voir la flèche du vecteur).

A noter que pour la fonction *arrow*, la fenêtre ne s’adapte pas automatiquement. La fonction *axis* permet de spécifier la fenêtre graphique, ici : *axis([-0.1,1,0.25,1])*.

Retrouvez toutes nos ressources pédagogiques sur https://www.casio-education.fr/be-fr/