BIEN DÉBUTER AVEC PYTHON

par François louis LAILLIER

Date de publication : 26/03/2007

Dernière mise à jour : 26/03/2007

Le but de ce tutoriel est de vous former à la programmation d'un script **Python** simple, mais surtout de vous apprendre à l'exploiter afin d'en profiter.
Vous apprendrez au cours de ce tutoriel :

- à utiliser un IDE sous windows pour développer vos scripts
- à les sauvegarder
- à les utiliser d'une facon simple
- à éditer un script sous un environement Linux Unix
- à exécuter ce script sous linux

I - PRÉREQUIS

I-B - Prérequis

Pour effectuer ce tutoriel, si vous êtes sous Windows,seule la dernière version de **PYTHON** disponible sur **Python.org** est nécessaire. Pour les personnes sous un OS Unix, **PYTHON** est fournie avec Unix par defaut. Les connaissances à avoir sont quasiment nulles, mais ayez un livre ou un tutoriel sous la main pour comprendre la synthaxe si elle vous échappe. De bons tutoriels et livres sont disponible dans nos rubrique et

I-C - Version de PYTHON utilisée

- **PYTHON 2.5**
- Système d'exploitation: Windows XP HOME et UBUNTU Dapper Drake
- Tests effectués avec **IDLE 1.1.4**

II - SYSTEME D'EXPLOITATION

II-A - GÉNÉRALITÉS LANGAGE PYTHON

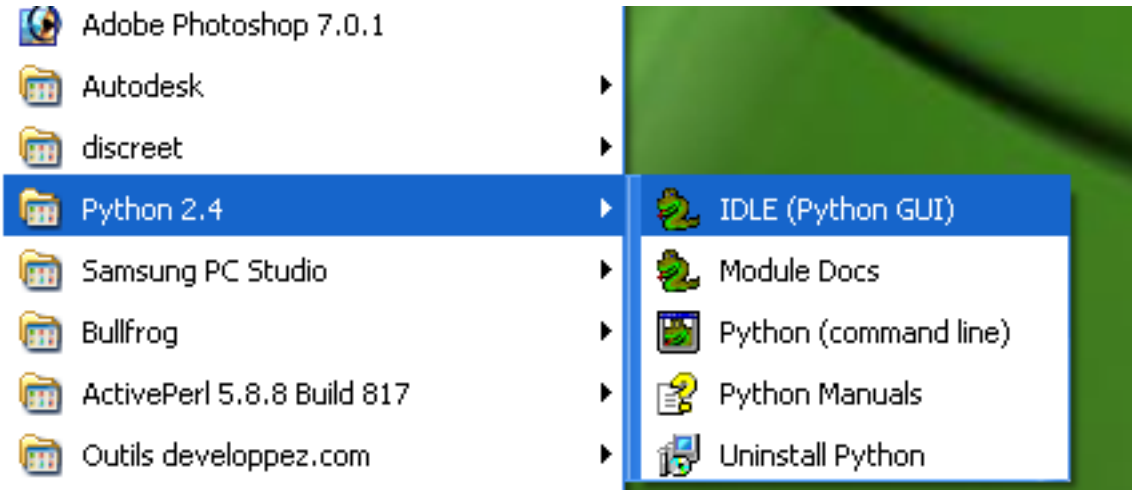
PYTHON est un langage, simple, facile à apprendre. ATTENTION Qui dit simple ne veux pas dire, impossible de créer de gros projet. Le langage Python est un language interprété, pas de compilation en Python, ce qui à mon gout est un avantage lors du développement du programme. Un language interprété est un langage qui lit une ligne de code et fait ce qu'elle lui dit de faire. Par exemple en python vous tapez cela dans le Shell.

```
>>> print "Ceci est une chaine de caractère"
>>> 'Ceci est une chaine de caractère'
>>>
```

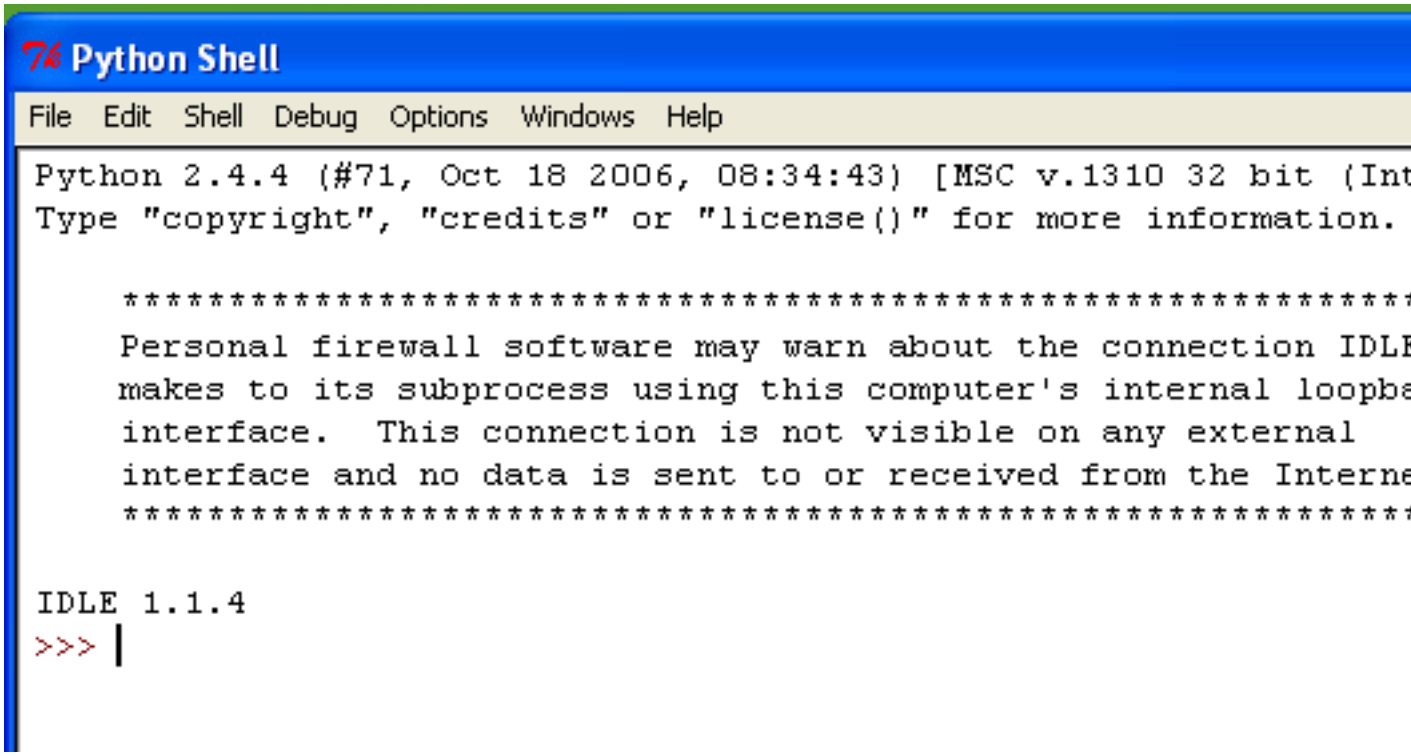
On voit très bien ce qu'est un langage interprété, je dit à la console (Shell) de m'imprimer à l'écran une chaine de caractère ensuite j'appuie sur la touche **ENTREE** et l'interpréteur execute la commande. Alors imaginez lors du developpement d'un gros projet en Python, vous ommettez une commande très simple comme ouvrir un fichier texte, un mixte entre l'oeuvre de Gerard SWINNEN et le Shell Python est souvent suffisant pour se débloquer seul et rapidement.

II-B - MICROSOFT WINDOWS

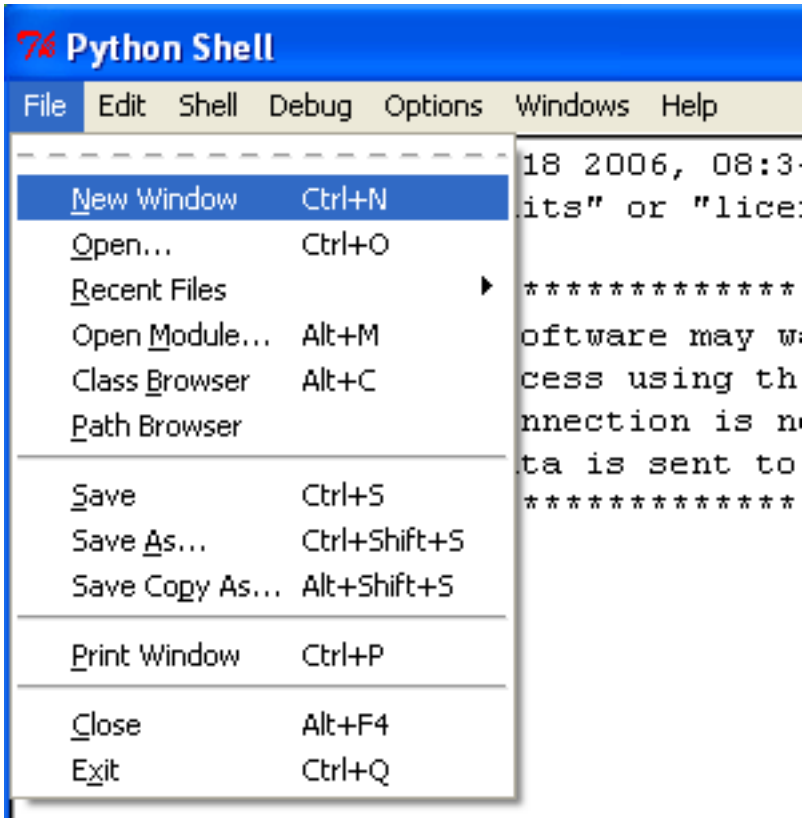
Pour pouvoir utiliser Python sous Windows il est nécessaire d'installer l'interpréteur Python. Il se trouve sur **Python.org**. Une fois celui-ci installé vous disposez de la console Python, appelée communemment Shell. Il y a aussi un IDE, IDLE. Je trouve cette IDE assez instinctive grâce à sa **coloration automatique** et grâce aux **infobulles** indiquant vos arguments lors de l'appelle d'une fonction.Pour ouvrir IDLE, rien de plus simple, allez dans "***DEMARRER/TOUS LES PROGRAMMES/Python 2.x/IDLE (Python GUI)***.



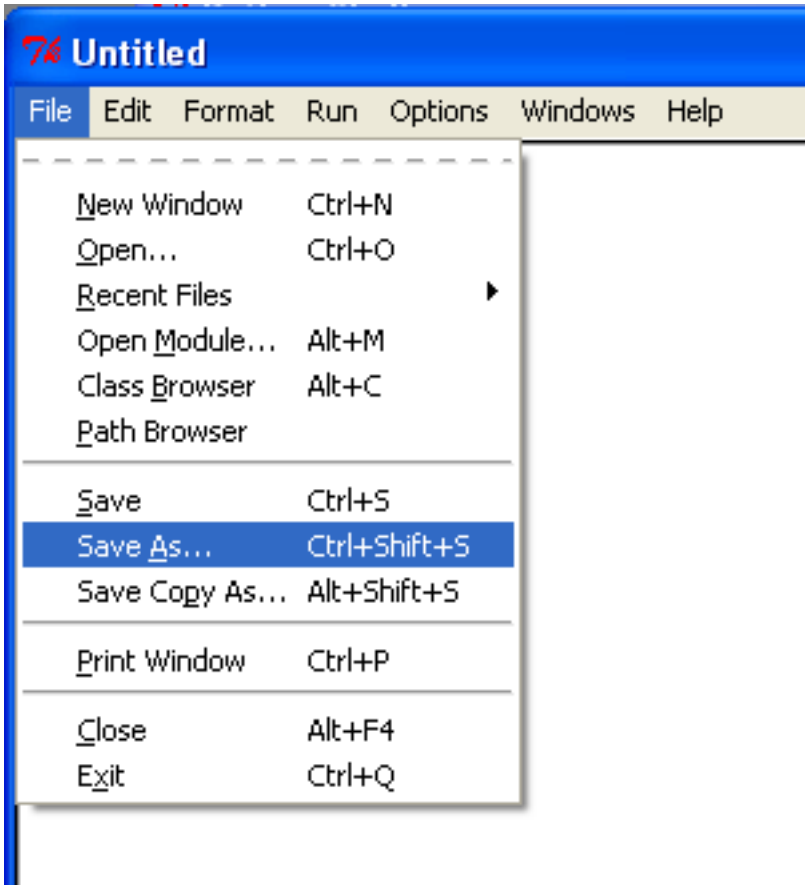
Normalement vous devriez ouvrir une fenêtre de ce type.(A la version d'IDLE prêt.)



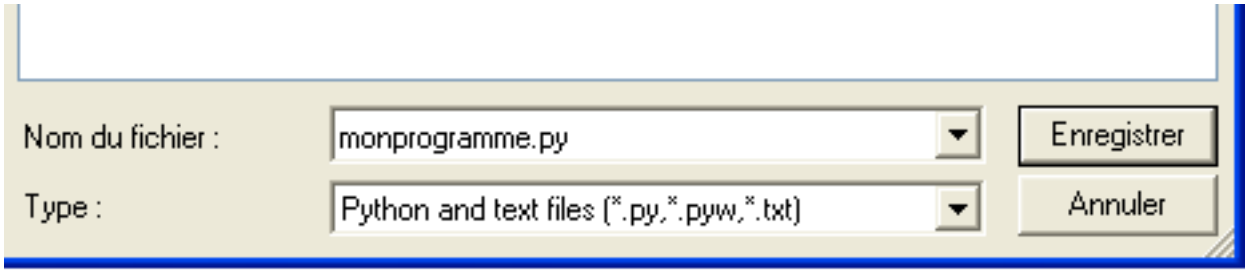
Ceci n'est que la console Python d'IDLE, la meme que si vous ouvriez Python.exe. L'intéret d'IDLE est d'intégrer un solution pour éditer nos script et de pouvoir l'exécuter " à la Visual Studio " qui d'une simple touche compile et execute le programme en cours. Ici d'une pression d'une touche nous exécuterons notre script. Pour créer un nouveau script il faut cliquer sur ***File/New Window***.



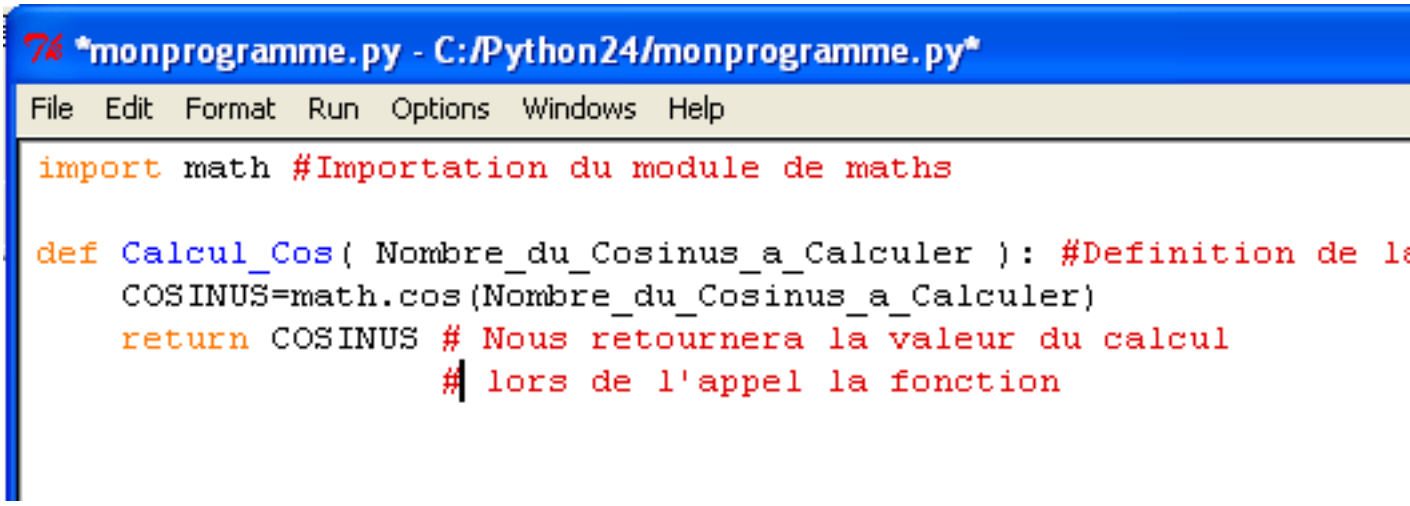
Vous devriez obtenir une nouvelle fenêtre UNTITLED. Dans celle ci cliquez sur ***File/Save as***.



Ensuite il faut renseigner les champs de la sauvegarde, il suffit de donner le nom de notre programme en y ajoutant l'extension ***.py*** afin que la coloration automatique soit active.



Bon nous allons créer une petite fonction toute bête afin de voir ce que l'on peut faire avec IDLE. Nous allons faire une fonction qui calcule le *COSINUS* d'un nombre que l'on entre comme argument de notre fonction.

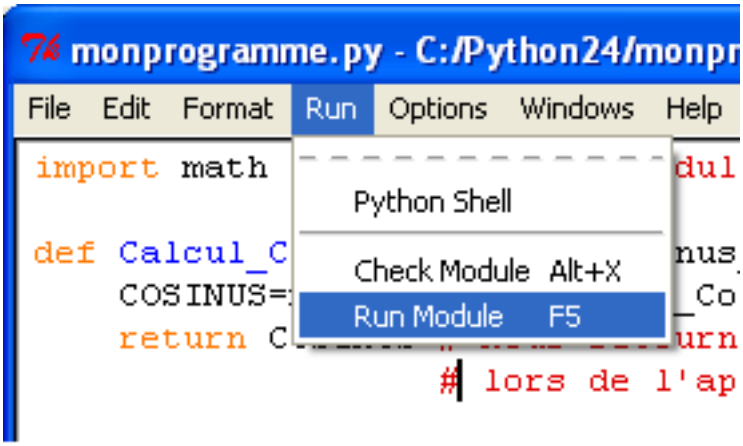


Nous aurions pu faire cela pour être encore plus rapide.

Execution de la fonction Calcul_Cos

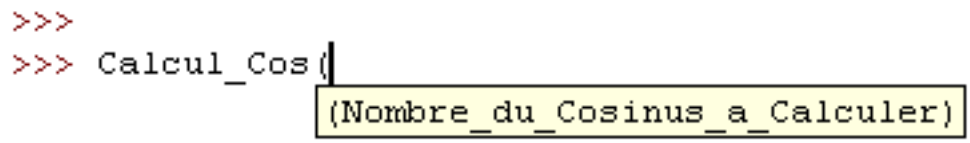
```
def Calcul_Cosinus(Nombre_du_cosinus_a_calculer):
  return math.cos(Nombre_du_cosinus_a_calculer)
```

On voie bien que les mots clés sont colorés automatiquement. Maintenant nous allons exécuter notre script soit en cliquant ***run/run module*** soit en appuyant sur la touche ***F5***.



IDLE lance le **Shell** et execute le script, puis lorsqu'il a terminé nous rend la main avec le curseur **>>>**.

Appelons maintenant notre fonction ***Calcul_Cos()***



L'info bulle nous indique combien d'argument, ici un seul, et ce que ca peut être. Par exemple le nom de l'argument que j'ai donné dans la déclaration de ma fonction est explicite lors de l'appel de celle-ci.

Execution de la fonction Calcul_Cos

```
>>>
>>> A=Calcul_Cos(25)
>>> print A
0.99120281186347359
>>>
```

Voila pour Windows, il suffit maintenant de naviguer dans les menus de IDLE, afin de trouver tous les outils existants (Comme l'indentation d'un bloc surligné). Il ne vous reste plus qu'a programmer vos programme afin de les exécuter et d'interagire avec eux.

II-C - UNIX UBUNTU Drapper Drake

Python est intégré à toutes les solutions Unix-Linux à ma connaissance. Pour exécuter python rien de plus simple sous Linux. Ouvrez un terminalX et tapez dans le terminal la commande suivante

```
mastationlinux@mastationlinux-laptop:~$ Python
Python 2.4.3 (#2, Oct  6 2006, 07:52:30)
[GCC 4.0.3 (Ubuntu 4.0.3-1ubuntu5)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Attention pour exécuter cela vous devez avoir les droits d'exécution de python. Pour plus ample informations renseignez vous auprès de votre administrateur reseau. Selon la version de linux que vous utilisez la commande peut être quelque peu différente à vous d'adapter a votre système d'exploitation.

Vous avez donc appelé la console ***Python*** si vous tapez un script Python va l'exécuter comme vu au chapitre II-A L'intérêt maintenant est d'éditer notre code et de le sauvegarder d'une telle facon à ce qu'il soit exécutable.

Pour cela ouvrez votre éditeur de texte favoris sous Linux, pour ma part j'utilise ***Gedit***, mais ***vi*** ou autre peuvent faire l'affaire.

```
mastationlinux@mastationlinux-laptop:~$ sudo gedit
PassWord
```

Ensuite une fois dans votre éditeur de texte vous avez juste à ajouter une ligne qui indique ou se trouve python. Je tiens à ajouter que ***gedit*** applique une coloration automatique si vous sauvegarder votre script avec l'extension ***.py***.

Ligne à insérer pour un script sous Linux

```
#!/usr/bin/env python
```

Le reste du script est identique à un script "Windows", la fonction de calcule du cosinus d'un nombre ressemble donc à cela.

```
#!/usr/bin/env python
```

```
import math

def Calcul_Cosinus(Nombre_du_Cosinus_a_calculer):
 COSINUS=math.cos(Nombre_du_Cosinus_a_calculer)
 return COSINUS

a=input('Entrez un nombre SVP :\n')
print Calcul_Cosinus(a)
```

Voila sauvegarder votre programme comme vous le faite comme d'habitude avec l'extension ***.py***. Changez les droits de permission de votre fichier avec la commande Chmod.

```
mastationlinux@mastationlinux-laptop:~$ chmod+x monprogramme.py
```

Naturellement si vous n'etes pas dans le repertoire courant de votre script entrez le chemin absolu. Pour exécuter votre script il ne reste plus qu'a appelez python suivi du nom de votre script.

```
mastationlinux@mastationlinux-laptop:~$ python monprogramme.py
Entrez un nombre SVP :
25
0.991202811863
mastationlinux@mastationlinux-laptop:~$
```

Voila, maintenant vous savez exécuter un script ***Python*** sous ***Linux***.

Vous vous dites: " Eh trop nul ce tutoriel, les linuxiens non même pas le droit faire F5 pour exécuter leur script, c'est quoi cette discrimination..... "

*Moi je vous dis : " On se calme les amis". Si vous voulez avoir un IDE pour éditer vos script Python, je vous conseil l'EDI . Vous pouvez installer cet EDI grâce, si vous l'avez, au gestionnaire de paquets Synaptic; Une petite recherche est c'est gagné. Juste deux petites choses, une fois sous **SCITE**, allez dans le menu View/Output et dans le menu **Language** choisissez parmis les 31 langages supportés par **SCITE**.*

III - CONCLUSION

Voila c'est terminé pour ce tutoriel, si vous pensez qu'il faut y ajouter des rubriques, ou autre améliorations faites m'en part par . Merci de votre attention et bonne ***PYTHONNADE***