





## Préambule

- ① Il faut avoir téléchargé et installé Python version 3.xx sur son ordinateur (plutôt avec windows Store)  
<https://apps.microsoft.com/store/detail/python-39/9P7QFQMJRF7?hl=fr-fr&gl=fr&rtc=1>
- ② Puis dans une fenêtre de commande (⌨+R), on tape CMD  pour obtenir la fenêtre de saisie.

Afin d'installer les bibliothèques python qui vont nous simplifier le code, on tape alors :

```
pip install numpy 
```

```
pip install pillow 
```

```
pip install matplotlib 
```

- ③ Pour charger puis utiliser les programmes python, il faut les télécharger depuis le site :  
<http://stephane.le.meteil.free.fr/an2022/> rubrique SNT puis photo numérique

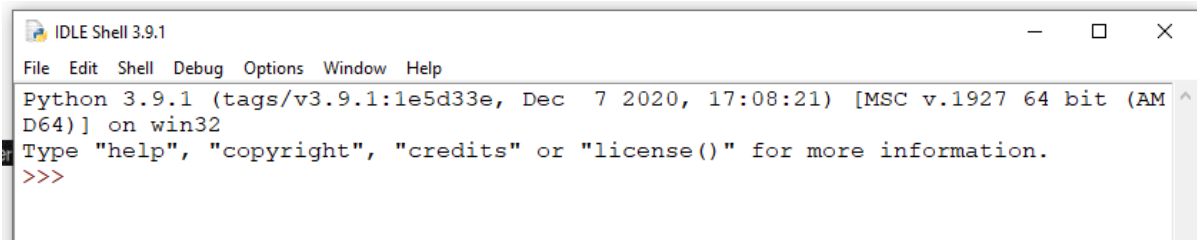
On les place dans un répertoire prévu à cet effet sur le bureau.

- ④ On téléchargera aussi les différentes photos dans ce même répertoire.

## Programme 1 : la base. Télécharger photo01.py

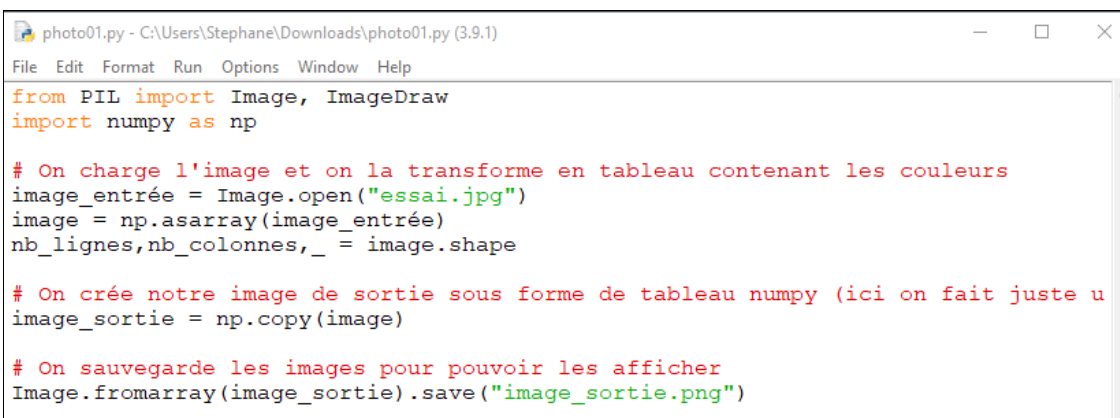
Télécharger une image de votre choix au format jpg et de taille modeste 800\*600 pixels, on la nommera **essai.jpg** et on la placera dans le répertoire contenant le programme 1

Pour accéder au programme : démarrer **idle** dans le menu démarrer de windows (c'est le nom de l'interface python)



```
IDLE Shell 3.9.1
File Edit Shell Debug Options Window Help
Python 3.9.1 (tags/v3.9.1:1e5d33e, Dec 7 2020, 17:08:21) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
```

Puis **File/Open**, retrouver votre répertoire contenant **photo01.py**, vous devriez obtenir :



```
photo01.py - C:\Users\Stephane\Downloads\photo01.py (3.9.1)
File Edit Format Run Options Window Help
from PIL import Image, ImageDraw
import numpy as np

# On charge l'image et on la transforme en tableau contenant les couleurs
image_entrée = Image.open("essai.jpg")
image = np.asarray(image_entrée)
nb_lignes,nb_colonnes,_ = image.shape

# On crée notre image de sortie sous forme de tableau numpy (ici on fait juste u
image_sortie = np.copy(image)

# On sauvegarde les images pour pouvoir les afficher
Image.fromarray(image_sortie).save("image_sortie.png")
```

On lance le programme en cliquant sur : **Run pour Run module**. Une copie de votre image, nommée **image\_sortie.png** devrait être créée et présente dans votre répertoire.

On va maintenant apprendre à manipuler cette image de différentes façons.

## Programme 2 : extraire des couleurs. Télécharger photo02.py

Ce programme extrait la teinte rouge de la photo nommée **essai.jpg** et enregistre le résultat

```
1 from PIL import Image, ImageDraw
2 import numpy as np
3
4 # On charge l'image et on la transforme en tableau contenant les couleurs
5 image_entrée = Image.open("essai.jpg")
6 image = np.asarray(image_entrée)
7 nb_lignes, nb_colonnes, _ = image.shape
8
9 # On crée notre image de sortie sous forme de tableau numpy
10 image_sortie = np.copy(image)
11 for i in range(nb_lignes):
12     for j in range(nb_colonnes):
13         R,V,B=image_sortie[i,j]
14         image_sortie[i,j]=[R,0,0]
15
16 # On sauvegarde les images pour pouvoir les afficher
17 Image.fromarray(image_sortie).save("image_sortie rouge.png")
18
```

Quel est le rôle de la ligne 13 ?

Comment modifier les lignes 14 et 17 pour obtenir la composante bleue ? La composante verte ?

Que se passe-t-il si on ne conserve que le rouge et le bleu ? le rouge et le vert ? le vert et le bleu ?

Créer un document comportant toutes les images résultats obtenues à la façon [d'Andy Wahrol](#).

Que se passe-t-il si on remplace la ligne 14 par `image_sortie[i,j]=[(R+V+B)//3, (R+V+B)//3, (R+V+B)//3]` ?

Est-ce satisfaisant ?

## Programme 3 : en niveaux de gris. Télécharger photo03.py

Ce programme propose une meilleure façon de passer en niveaux de gris l'image de base. Essayez de modifier la ligne 14 avec d'autres coefficients.

## Programme 4 : modifications. Télécharger photo04.py

Exécuter ce programme et observer l'image de sortie. Que réalise ce programme ?

Comment adapter ce programme pour que la sortie soit une symétrie horizontale de l'image de base ?



Quelle variable donne le nombre de colonne de l'image ?

Que deviendra la colonne 1 ? la 2 ?

Est-il nécessaire de traiter toutes les colonnes ?

Où s'arrête-t-on ? Pourquoi ?

