



Objectifs

- Convertir une image `.png` en nuances de gris
- Sauver l'image obtenue en niveau de gris sous forme d'un fichier `.csv`

Convertir une image en nuance de gris

fichier python : `png_couleur_pixel_gris.py`

Attention : au lycée avec les versions portables

- Créer un dossier `[leon]` dans le dossier de `PyScripteur.exe`.
- Copier l'image dans le dossier `[leon]`. Le format de l'image doit être `png`!
- Si l'image est `img.png` appeler la fonction avec le paramètre `"/leon/img"` (sans l'extension)

Exemples d'applications pour les élèves

- En 2nde, sur les fonctions : fichier `Lena`
- En SNT, travail sur les images, les couleurs...

Entraînement

- Choisir une petite image `.png` (Internet, dossier perso, ...) ou effectuer une capture d'écran d'une petite zone et la sauver en `.png`.
- À l'aide du programme suivant convertir l'image en fichier `.csv`
 - Ouvrir le fichier `.csv` dans un tableur (le séparateur est la virgule)
 - Sélectionner la plage, puis appliquer un format conditionnel avec une échelle de 2 couleurs allant de noir (min) à blanc (max).
 - Formater les cellules : onglet nombres > Catégorie : Définit par l'utilisateur, puis Code format : " "



```
1  # -*- coding:utf8 -*-
2  # python3
3
4  import matplotlib.image as mpimg
5  import numpy as np
6
7  def png2csv(ficpng):
8      """ <fic> nom du fichier image SANS extension.
9      lit les valeurs des pixels de <fic>, les convertit
10     en niveau de gris et sauve les données dans <fic.csv>
11     """
12     img = mpimg.imread(ficpng+".png")
13     fic = open(ficpng+".csv", 'w')
14     # pour être certain que l'image soit codée en entiers
15     ↔dans [0; 255]
16     if img.dtype == np.float32:
17         img = (img * 255).astype(np.uint8)
18     haut, larg, plans = img.shape
19     # chaque pixel[i, j] contient les valeurs [R, V, B]
20     # parfois 4eme plan pour transparence (canal alpha)
21     # on ne prend que les 3 composantes RVB
22     for i in range(haut):          # nb lignes
23         for j in range(larg):      # nb colonnes
24             # int(x) renvoie la partie entière de x
25             # valeur de gris 'bien pour l'oeil'
26             g = int(.299 * img[i,j][0] + 0.587 * img[i,j][1]
27             ↔+ 0.114 * img[i,j][2])
28             fic.write(str(g))
29             fic.write(",") # les données sont séparées par
30             ↔des virgules
31             fic.write("\n") # ligne suivante
32     print("ligne ",i, " fait" , end="\r") # pour voir la
33     ↔progression
```



```
30     fic.close()
31     return 1
32
33     # au lycée, avec les versions portables : copier l'image
      ↔ dans le sous-dossier
34     # <leon> créé dans le dossier de PyScripter.exe
35     # png2csv("./leon/fic_image")
```
