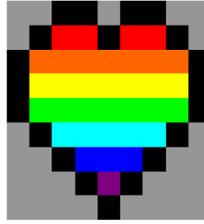


Ex 1 : Analyse d'une image pixelisée

- 1) Télécharger l'image « cœur.png » depuis mon Site ; une copie est donnée ci-contre
- 2) Recopiez le programme suivant dans une console IDLE
 - a) Que fait ce programme ?

```
from PIL import Image
img = Image.open("Coeur.png")
largeur_image=9
hauteur_image=9
print("composantes R-V-B de chaque pixel de l'image Coeur.png")
for y in range(hauteur_image):
    for x in range(largeur_image):
        r,v,b=img.getpixel((x,y))
        print("pixel (" ,x," ,",y," ) ", "RVB :",r,v,b)
```

- b) Combien de ligne de résultats ce programme affiche-t-il ? Pourquoi ?
- c) Selon les résultats du programme, quelle sont les composantes RVB du pixel de coordonnées (4,7) ?
- d) A quoi correspond cette couleur ? (donnez le nom de la couleur)

Ex 2 : Modification d'une image

- 1) Télécharger l'image « warhol.png » depuis mon Site ; une copie est donnée ci-contre
- 2) Recopiez le programme suivant dans une console IDLE
 - a) Que fait ce programme ?
 - b) Quelle ligne du programme permet d'afficher l'image ?

- c) L'image générée par le programme Python est-elle enregistrée dans votre dossier de travail ? Sinon modifier le script PYTHON
- 3) En vous inspirant du programme précédent, écrivez un programme qui inverse les valeurs des canaux bleu et rouge sans changer la valeur du vert

```
from PIL import Image
img = Image.open("warhol.png")
largeur_image=500
hauteur_image=500
for y in range(hauteur_image):
    for x in range(largeur_image):
        r,v,b=img.getpixel((x,y))
        n_r=v
        n_v=b
        n_b=r
        img.putpixel((x,y),(n_r,n_v,n_b))
img.show()
```

Ex 3 : Transformation en niveaux de gris et en noir & blanc

- 1) Télécharger l'image « joconde.png » depuis mon Site ; une copie est donnée ci-dessous
- 2) Recopiez le programme suivant dans une console IDLE
 - a) Que fait ce programme ?
 - b) Modifier ce script PYTHON afin de sauvegarder l'image obtenue

- 3) Modifiez le script pour obtenir une photo en nuances de gris
- 4) Modifiez le script pour obtenir une photo en noir en blanc (avec un seuil de séparation de 110)
- 5) Recopiez le script ci-dessous et expliquer le résultat obtenu

```
from PIL import Image
img = Image.open("joconde.png")
largeur_image=600
hauteur_image=400
for y in range(hauteur_image):
    for x in range(largeur_image):
        r,v,b=img.getpixel((x,y))
        img.putpixel((x,y),(r,v,b))
img.show()
```

- 6) Modifiez le script pour obtenir une photo par symétrie verticale puis par symétrie centrale
- 7) Modifiez le script pour obtenir une photo par rotation d'angle 90°

```
from PIL import Image
img = Image.open("joconde.png")
img2=Image.new('RGB', (600,400))
largeur_image=600
hauteur_image=400
for y in range(hauteur_image):
    for x in range(largeur_image):
        r,v,b=img.getpixel((x,y))
        img2.putpixel((x,400-1-y),(r,v,b))
img2.show()
```

Ex 4 : Duplication d'images sur une palette

On reprend l'image de l'exercice précédent : « joconde.png »

- 1) Modifier la taille de cette image en la divisant par 4 puis sauvegarder le résultat sous « joconde_réduite.png »
- 2) En vous inspirant sur les différents scripts de l'exercice précédent écrire un programme permettant la réalisation de l'image ci-dessous
- 3) Sauvegarder cette nouvelle image sous « joconde_palette.jpg »

