

Séance 2 :

Création d'une image numérique


Objectif :

- Créer une image numérique avec Python
- Traiter par programme une image pour la transformer en agissant sur les trois composantes de ses pixels

Activité 1 : Pixel par pixel

Commencez par lancer le logiciel Spyder qui est un éditeur de code Python (sur le bureau : Dossier « Informatique » > Dossier « Anaconda » > Spyder).

1. Copiez le code suivant dans l'éditeur de Spyder puis enregistrez le fichier dans un répertoire adéquat sur votre clé USB en le nommant « act1.py ».

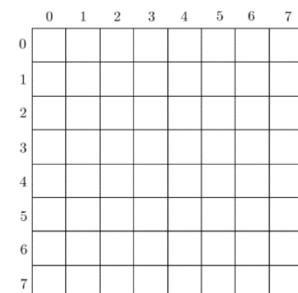
Exécutez ensuite le code en cliquant sur l'icône  dans la barre d'outils.

```
1 from PIL import Image
2
3 img = Image.new("RGB", (8,8))
4
5 for y in range(8):
6     for x in range(8):
7         img.putpixel((x,y), (255,255,255))
8
9 img.save("carreblanc.png")
```

Analysons ces quelques lignes de code :

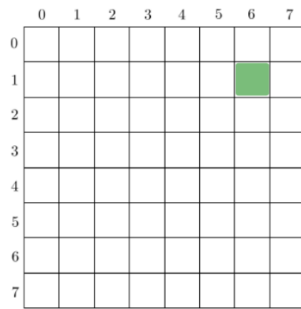
- Ligne 1 : on importe le module Image de la bibliothèque PIL qui va permettre la manipulation d'images.
- Ligne 3 : on crée une image de taille 8*8 pixels dont les pixels seront codés en RGB (Red Green Blue). Cette image est appelée « img » dans le programme.
- Lignes 5 et 6 : double boucle « for » qui va permettre de parcourir tous les pixels de cette image. Les variables y et x varient toutes les deux de 0 à 7 donc le couple (x,y) va prendre toutes les valeurs de (0,0) à (7,7).
- Ligne 7 : l'instruction permet de colorier le pixel de coordonnées (x,y) en blanc (255,255,255). Cette instruction étant à l'intérieur de la double boucle for, cette ligne sera exécutée pour tous les pixels de coordonnées (x,y). Autrement dit, tous les pixels de l'image seront colorés en blanc.
- Ligne 9 : l'image est sauvegardée sous le nom « carreblanc.png ».

2. Coloriez 6 pixels du quadrillage ci-contre en noir, en rouge, en vert et en bleu.



3. Créez votre motif en modifiant le code Python précédent.

Exemple : le pixel de coordonnées (6,1) se colorie en vert avec l'ajout de la ligne 9.



```

1 from PIL import Image
2
3 img = Image.new("RGB", (8,8))
4
5 for y in range(8):
6     for x in range(8):
7         img.putpixel((x,y), (255,255,255))
8
9 img.putpixel((6,1), (0,255,0))
10
11 img.save("motif.png")

```

Ajoutez les lignes de codes
ici pour colorier les pixels

4. Exécutez votre code Python et vérifiez que l'image créée correspond bien à votre motif de la question 2. Il faudra zoomer pour voir les pixels.

Pour trouver le code RGB de toutes les couleurs possibles, voilà deux liens :

- https://www.w3schools.com/colors/colors_rgb.asp
- <https://htmlcolorcodes.com/fr/>

Vous venez de créer une image pixel par pixel, ce qui est très fastidieux. De plus, l'image créée est très petite. Il est bien sûr possible de travailler autrement et sur des images plus grandes. Dans l'activité qui suit vous allez créer des blocs (lignes, rectangles) entiers de pixels colorés.

Activité 2 : Créer des lignes et des rectangles

Créer un nouveau fichier Python dans Spyder et enregistrez-le en le nommant « act2.py ».

1. Reprenez le code présenté en début d'activité 1 et copiez le dans ce nouveau fichier Python. Modifiez ce qui est nécessaire pour que l'image créée soit de taille 300*200 pixels.
2. Rajoutez (avant la dernière ligne de sauvegarde de l'image) le code suivant :

```

for x in range(50,150):
    img.putpixel((x,20), (255,87,51))

```

Exécutez le code complet et analysez les changements apparus dans l'image.

3. Quelles sont les coordonnées des différents pixels ayant été colorés en « orange » ?
4. Quel est le code permettant de colorier en bleu tous les pixels situés sur la ligne 100 et ayant une abscisse comprise entre 150 et 250 ?
5. Tracez plusieurs lignes de couleurs différentes dans votre image.
6. En vous inspirant des trois lignes situées au début du code permettant de remplir l'image de pixels blancs, dessinez des petits rectangles colorés au sein de votre image.

Source : Gilles Lassus, La photographie numérique, bit.ly/SNT_BDX