



Une histoire de pixels

➤ À la loupe...

1. Observation d'un écran au microscope →
2. Qu'avez-vous remarqué ?



CTRL + Clic



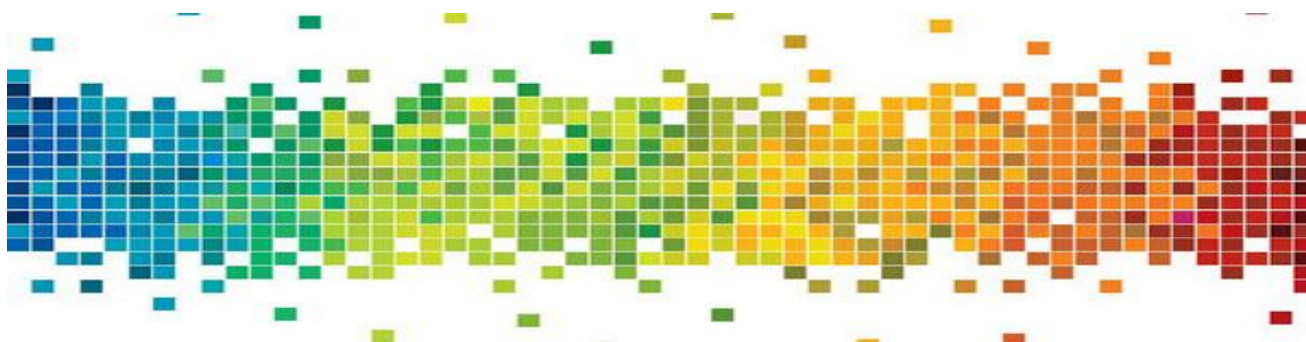
Prendre connaissance de la fiche de cours :

[Le pixel](#) (appuyez sur CTRL+Clic sur le lien)

3. Grâce au site <http://physique.ostralo.net/images/> (appuyez sur CTRL+Clic sur le lien), « les couleurs d'une image », observez comment le réglage de l'intensité des 3 sous-pixels rouge, vert et bleu peut générer une multitude de couleurs.

La perception « uniforme » pour l'œil, qui ne distingue pas (sauf à la loupe) le détail des sous-pixels mais simplement une couleur globale, est due au pouvoir de résolution limité de l'œil humain.

4. Combien de couleurs différentes peuvent-elles être générées, sachant que chaque sous-pixel (rouge, vert ou bleu) possède 256 nuances possibles ?



➤ Répartition des pixels sur l'écran, notion de définition et de résolution

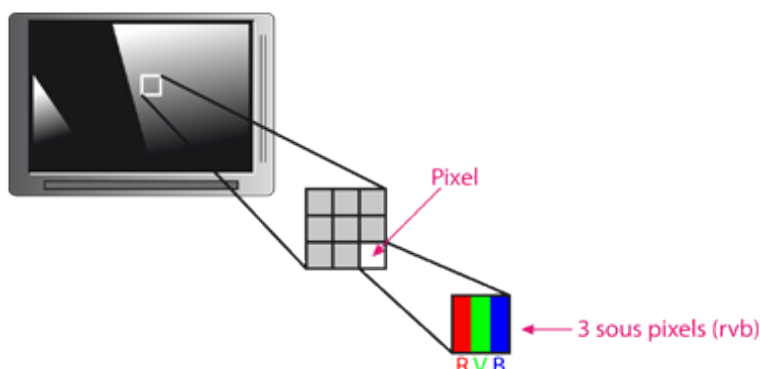
Lors de l'observation d'écrans effectuée dans la partie précédente, vous pourriez constater des différences dans l'agencement ou la forme des pixels (et sous-pixels), suivant le modèle de smartphone observé.

Prenez connaissance de la fiche de cours sur [l'image](#) concernant la résolution (appuyez sur CTRL+Clic sur le lien).

5. Combien de pixels (horizontaux * verticaux) possède un écran 4K UHD ?
6. Quelle est sa résolution (en ppp ou dpi) pour un écran de diagonale 28 pouces (inches) et au format 16/9ème ?

Ressource : [Toutcalculer](#)

(appuyez sur CTRL+Clic sur le lien)





➤ PYZO (*Bureau/Autres raccourcis/SNT/Pyzo_SNT*)



Nous allons programmer avec PYZO sous Windows, un IDE adapté pour le traitement de l'image.

Annotations de l'image :

- Pour exécuter le programme, cliquez sur « Exécuter / Démarrer le script »** (pointe vers l'icône d'exécution dans la barre d'outils).
- Les résultats d'exécution s'affichent dans cette zone. Cette zone est appelée SHELL ou console Python** (pointe vers la zone de sortie du processus terminée).
- C'est dans cette zone que vous devez saisir le code de votre programme** (pointe vers l'éditeur de code principal).
- Un explorateur de fichiers est à votre disposition** (pointe vers le volet d'explorateur de fichiers en bas à droite).
- Icône qui permet d'effacer l'écran du SHELL** (pointe vers l'icône d'effacement dans la barre d'outils).

➤ Couleur d'un pixel

Écriture du programme N° 1 : 711-carrerouge.py

Après avoir observé que les écrans recréent les couleurs à partir de trois sous-pixels rouge, vert et bleu, nous allons créer à l'aide du langage Python, une image que nous définirons pixel par pixel.

Dans zone de saisie de code, saisissez le code ci-dessous.

```
1 from PIL import Image
2 import subprocess as sp
3 img = Image.new("RGB", (8,8))
4
5 for x in range (8):
6     for y in range (8):
7         img.putpixel((x,y), (255,0,0))
8 img.save("711-carrerouge.jpg")
9 sp.Popen(("C:\Program Files (x86)\PhotoFiltre 7\PhotoFiltre7.exe", "711-carrerouge.jpg"))
```

Sauvegardez votre programme sous le nom « 711-carrerouge.py » dans votre dossier « SNT/THEME_7 » puis exécutez le script (CTRL + F5).

Vous venez de créer votre première image !

Cette image devrait s'ouvrir avec un logiciel de visualisation ; vous pouvez vérifier le résultat obtenu.

Faites voir à votre professeur.

Comment partager vos fichiers en vidéo

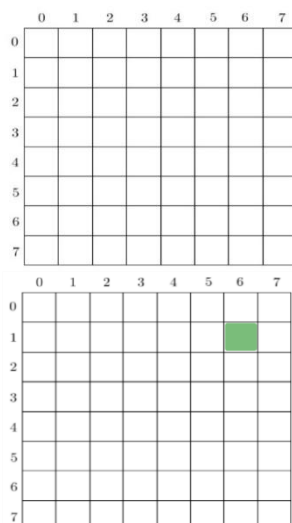


CTR + Clic



Écriture du programme N°2 : 712-quadrillage.py

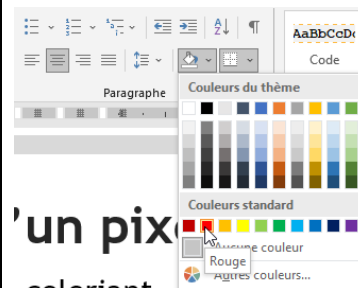
Dessinez un motif dans le tableau du document réponse, en coloriant (en noir ou en couleur) **6** des 64 cases.



Comment colorier une cellule du tableau du document réponse

Clique dans la cellule à colorier du tableau

Sélectionne la couleur souhaitée.



Reproduisez votre motif en modifiant votre programme.

Exemple : la ligne 9 colore le pixel de coordonnées x=6 et y=1 en vert.

```
1 from PIL import Image          # Import du module PIL
2 import subprocess as sp
3 img = Image.new("RGB", (8,8))  # Création image de 8x8 pixels
4
5 for x in range (8):            # Boucle de ligne
6     for y in range (8):        # Boucle de colonne
7         img.putpixel((x,y),(255,255,255)) # Coloriage pixel
8
9 img.putpixel((6,1),(0,255,0))  # Coloriage pixel
10 img.save("712-quadrillage.jpg") # Sauvegarde de l'image
11 # affichage de l'image
12 sp.Popen(("C:\Program Files (x86)\PhotoFiltre 7\PhotoFiltre7.exe", "712-quadrillage.jpg"))
```

Vous n'êtes pas obligés de saisir les commentaires !

Sauvegardez votre programme sous le nom « 712-quadrillage.py » dans votre dossier « SNT/THEME_7 » puis exécutez votre script (CTRL + F5).

Pour retrouver le code RGB de n'importe quelle couleur, vous pouvez par exemple utiliser le site <https://htmlcolorcodes.com/fr/> (appuyez sur CTRL + Clic sur le lien)

Notre méthode de création d'image possède pour l'instant deux inconvénients majeurs :

- L'image est bien trop petite ;
- Placer des pixels un par un est très fastidieux.

Nous allons y remédier !



Écriture du programme N°3 : 713-ligne.py

7. En analysant votre code ci-dessus, effectuez des modifications pour que l'image créée (*Image.new*) soit de taille 300 x 200 (300 pixels par 200 pixels).
8. Remplacez la ligne 9 du programme de la page 3/7 par les deux lignes de code suivantes :

```
for x in range(50,150) :  
    img.putpixel((x,20),(156,61,207))
```
9. Sauvez l'image avec le code : `img.save("713-ligne.jpg")` et votre programme « 713-ligne.py » dans votre dossier « SNT/THEME_7 ».
10. Exécutez votre script (CTRL + F5) et analysez les changements apparus dans l'image.
11. Tracez plusieurs lignes de couleurs différentes dans votre image.

Faites voir à votre professeur.

Écriture du programme N°3 : 714-carre.py

12. Saisissez maintenant le code suivant :

```
1 from PIL import Image  
2 import subprocess as sp  
3 img = Image.new("RGB", (300,300))  
4  
5 for x in range (300):  
6     for y in range (300):  
7         img.putpixel((x,y),(129,167,211))  
8  
9 for x in range (100,200):  
10     for y in range (100,200):  
11         img.putpixel((x,y),(255,255,255))  
12  
13 img.save("714-carre.jpg")  
14 sp.Popen(("C:\Program Files (x86)\PhotoFiltre 7\PhotoFiltre7.exe", "714-carre.jpg"))
```

Sauvegardez votre programme sous le nom « 714-carre.py » dans votre dossier « SNT/THEME_7 » puis exécutez votre script (CTRL + F5).

13. Quelle est la dimension de l'image ?
14. Que réalisent les lignes 9 à 11 de votre programme ?



1. Création de drapeaux (300px par 200px)

Activité : À l'aide du code vu dans la partie précédente, concevez 2 drapeaux parmi ceux présentés ci-dessous. Vous enregistrerez les images avec les noms indiqués sous chaque drapeau.

Astuce : vous pouvez rajouter autant de boucle « for » que vous le voulez !



715-drapeau-fr



715-drapeau-uk



715-drapeau-be



715-drapeau-lx



715-drapeau-s



715-drapeau-ch

Faites voir à votre professeur.

Comment partager vos fichiers en vidéo



CTR + Clic



➤ Modification d'une image

Nous allons maintenant insérer une image dans un programme et agir sur ses pixels.

Téléchargez l'image « [ciel.jpg](#) » depuis le site Web SNT que vous enregistrerez dans votre dossier « SNT/THEME_7 ».

Dans le code, on ouvre ensuite une image à partir d'un fichier. Ici, le fichier est « [ciel.jpg](#) ». L'image est lue par la méthode `open()` du module `Image`, et est stockée dans la variable `img`. C'est cette variable que nous utiliserons dans la suite du programme.

On récupère ensuite la largeur et la hauteur de cette image, et on affiche ces valeurs (en nombre de pixels) dans le shell.

Pour afficher l'image, il faut insérer à la fin du programme l'instruction : `sp.Popen(...)`

```
1 from PIL import Image
2 import subprocess as sp
3 #on ouvre le fichier image
4 img = Image.open("ciel.jpg")
5 colonne, ligne = img.size
6 print(colonne)
7 print(ligne)
8 sp.Popen(("C:\Program Files (x86)\PhotoFiltre 7\PhotoFiltre7.exe", "ciel.jpg"))
```

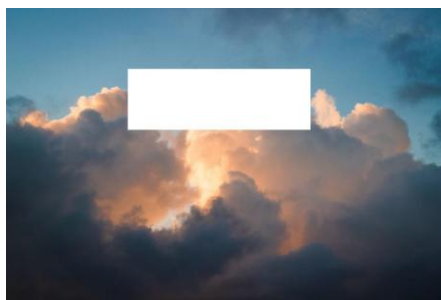
Sauvegardez votre programme sous le nom « [716-open.py](#) » dans votre dossier « SNT/THEME_7 » puis exécutez votre script (CTRL + F5).

On a vu dans la partie précédente que la fonction `img.putpixel((. , .), (... , ... , ...))` permet de changer la valeur de la couleur d'un pixel dans une image.

Par exemple nous allons créer un rectangle blanc dans l'image « [ciel.jpg](#) » :

Sauvegardez votre programme sous le nom « [717-carreciel.py](#) » dans votre dossier « SNT/THEME_7 » puis exécutez votre script (CTRL + F5).

```
1 from PIL import Image
2 import subprocess as sp
3 #on ouvre le fichier image
4 img = Image.open("ciel.jpg")
5
6 for x in range (200,500):
7     for y in range (100,200):
8         img.putpixel((x,y), (255,255,255))
9
10 img.save("717-carreciel.jpg")
11 sp.Popen(("C:\Program Files (x86)\PhotoFiltre 7\PhotoFiltre7.exe", "717-carreciel.jpg"))
```



Comment partager vos fichiers en vidéo



CTR + Clic



2. Exercices

Exercice 1 : téléchargez l'image « [voiture.jpg](#) » dans votre dossier « SNT/THEME_7 ». Programmez de façon à recouvrir la plaque d'immatriculation de la voiture par un rectangle jaune. Vous enregistrerez votre programme sous le nom « [718-plaquejaune.py](#) » et le fichier image sous le nom « [718-plaquejaune.jpg](#) » dans votre dossier « SNT/THEME_7 ».

Exercice 2 : téléchargez l'image « [charlie.jpg](#) » dans votre dossier « SNT/THEME_7 », encadrez Charlie par un rectangle rouge. Vous enregistrerez votre programme sous le nom « [719-charlie.py](#) » et le fichier image sous le nom « [719-charlie.jpg](#) » dans votre dossier « SNT/THEME_7 ».



← Charlie

Comment partager vos fichiers en vidéo



CTR + Clic