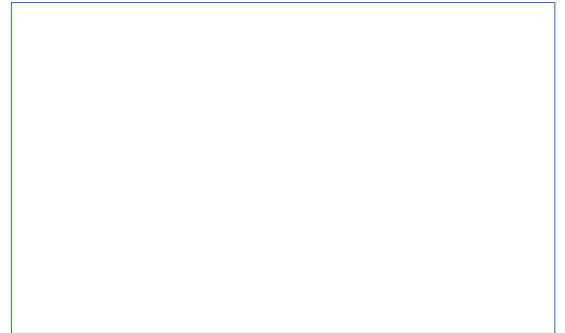


1 - Gestion des images avec le module Pillow

```

1 from PIL import Image
2
3 nom_image = "photo-montreal-palais-des-congres.jpeg"
4
5 img = Image.open(nom_image)
6 print("Image importée dans Python")
7
8 img.show()
9 print("Image affichée sur l'écran")

```

**Révisions : variables, fonctions d'entrée/sortie et if**

```

1 from PIL import Image
2
3 nom1 = "photo-montreal-palais-des-congres.jpeg"
4 nom2 = "pillow.png"
5
6 print("Tapez :")
7 print("1 pour ouvrir ", nom1)
8 print("2 pour ouvrir ", nom2)
9 n = # ATTENTION, on veut récupérer un int, pas un str
10
11 if n == 1:
12     nom_image = nom1
13 elif n == :
14     =
15
16 img = Image.open(nom_image)
17
18 nb_col = img.width # width : largeur en anglais
19 nb_lig = img.height # height : hauteur en anglais
20
21 print("Informations sur le fichier ", nom_image)
22 print("Pixels en largeur : ", nb_col)
23 print("Pixels en hauteur : ", nb_lig)
24
25 print("Voulez-vous afficher l'image ? O/N ")
26 rep = # on veut récupérer un str
27 if == "O":
28

```

Révisions : while, boucle Tant Que

```

6 # Début du rajout -----
7
8 n = # Fausse réponse pour rentrer dans la boucle au départ
9 while (n in [1, 2]) == :
10     print("Tapez :")
11     print("1 pour ouvrir ", nom1)
12     print("2 pour ouvrir ", nom2)
13     # Demande d'un entier à l'utilisateur
14
15 #Fin du rajout -----

```

Révisions : fonctions

```
1 from PIL import Image
2
3
4 # Définitions des fonctions
5
6 def choisir_un_fichier():
7     """Renvoie le nom du fichier choisi par l'utilisateur"""
8
9     nom1 = "photo-montreal-palais-des-congres.jpeg"
10    nom2 = "pillow.png"
11    n = 0 # Fausse réponse pour rentrer dans la boucle au départ
12    while (n in [1, 2]) == False:
13        print("Tapez :")
14        print("1 pour ouvrir ", nom1)
15        print("2 pour ouvrir ", nom2)
16        n = int(input())
17
18    if n == 1:
19        nom_img = nom1
20    elif n == 2:
21        nom_img = nom2
22
23    return nom_img
24
25
26
27 def obtenir_caracteristiques(img, nom_img):
28     """Affiche certaines caractéristiques de l'image, aucun renvoi"""
29
30     nb_col = img.width # width : largeur en anglais
31     nb_lig = img.height # height : hauteur en anglais
32
33     print("Informations sur le fichier ", nom_img)
34     print("Pixels en largeur : ", nb_col)
35     print("Pixels en hauteur : ", nb_lig)
36
37
38
39 def demander_affichage(img):
40     """Affiche l'image si l'utilisateur le veut, aucun renvoi"""
41
42     print("Voulez-vous afficher l'image ? O/N ")
43     rep = input()
44     if rep == "O":
45         img.show()
46
47 # Programme principal
48
49 nom_image = choisir_un_fichier() # Appel à la fonction choisir_un_fichier
50 image = Image.open(nom_image)
51 obtenir_caracteristiques(image, nom_image)
52 demander_affichage(image)
```

Afficher des informations sur une image

```
27 def obtenir_caracteristiques(img, nom_img):
28     """Affiche certaines caractéristiques de l'image, aucun renvoi"""
29
30     nb_col = img.width # width : largeur en anglais
31     nb_lig = img.height # height : hauteur en anglais
32     couches = img.getbands() # Conteneur contenant le nom des bandes
33     nb_cou = (couches) # Le nombre de bandes
34
35     print("Informations sur le fichier ", nom_img)
36     print("Pixels en largeur : ", nb_col)
37     print("Pixels en hauteur : ", nb_lig)
38     print("Nombre de pixels : ",          *          )
39     print("Couche(s) : ", couches)
40     print("Nombre de couches : ",          )
41
42     return nb_cou
```

2 - Gestion des pixels avec Python

OBTENIR LES COULEURS RGB D'UN PIXEL

```
from PIL import Image

nom1 = "photo-montreal-palais-des-congres.jpeg"
1  nom2 = "pillow.png"
2  nom_image = nom1 # Choisissez directement l'image, pas de input().
3
4  image = Image.open(nom_image) # Ouverture de l'image dans Python nb_couches = len(
5  image.getbands() )
6
7  print("Ouverture dans Python des données de ", nom_image)
8
9  10 while True:
11 12      print("\nDonner les coordonnées x puis ENTREE puis y puis ENTREE")
13 14      x = int(input())
15 16      y = int(input())
17 18      if nb_couches == 3: # RGB
19 20          r, g, b = image.getpixel( (x, y) )
21          print( (r, g, b) )
      elif nb_couches == 4:
          r, g, b, a = image.getpixel( (x, y) )
          print( (r, g, b, a) )
```

MODIFIER LES COULEURS RGB D'UN PIXEL

```
1  from PIL import Image
2
3  nom1 = "photo-montreal-palais-des-congres.jpeg"
4  nom2 = "pillow.png"
5  nom_image = nom2 # Choisissez directement l'image au démarrage.
6
7  image = Image.open(nom_image) # Ouverture de l'image dans Python
8  nb_couches = len( image.getbands() )
9
10 print("Ouverture dans Python des données de ", nom_image)
11
12 if nb_couches == 3: # RGB
13     for x in range(200):
14         image.putpixel( (x, x), (0, 255, 0) )
15
16 elif nb_couches == 4:
17     for x in range(200):
18         image.putpixel( (x, x), (0, 255, 0, 255) ) # a=255 : opaque
19
20 image.show()
```

3 - Modification et filtrage des couleurs

Atteindre tous les pixels

```
1  from PIL import Image
2
3  nom_image = "photo-montreal-palais-des-congres.jpeg"
4  image = Image.open(nom_image)
5
6  for y in range(                ): # Pour chaque ligne y de l'image
7      for      in range(                ): # et chaque colonne x de cette ligne
8
9          r, g, b = image.getpixel( (x,y) ) # On récupère les 3 valeurs
10         print( (x,y), " : ", (r,g,b) )
```