

i Dans ce cours, on traitera l'image en niveau de gris.

Ouvrir une image et la convertir en tableau

L'ouverture et la récupération des données d'une image se passe en 2 étapes :

- On ouvre le fichier image à l'aide de la bibliothèque **PIL**
- On la transforme en un tableau de nombre entier à l'aide de la bibliothèque **numpy**

```
1 import numpy as np
2 from PIL import Image
3
4 imgpil = Image.open("gateau.png")
5 # On le transforme en tableau
6 img = np.array(imgpil)
```

? Les bibliothèques **numpy** et **PIL** sont importer. Lequel de ces imports est un import avec alias

Travailler une image

L'objet `np.array img`

i L'objet `img` est un tableau contenant des tableaux de nombres

Par exemple :

```
1 [ [16, 20, 18, ..., 51, 53, 53],
2   [17, 56, 16, ..., 250, 52, 53],
3   ... ..
4   [219, 217, 215, ..., 48, 50, 51]]
```



Quelle instructions permet d'avoir le nombre de lignes de l'image

.....
.....



Quelle instructions permet d'avoir le nombre de pixels (entier) de la première ligne

.....
.....

Obtenir la valeur d'un pixel

```
1 import numpy as np
2 from PIL import Image
3
4 imgpil = Image.open("gateau.png")
5 img = np.array(imgpil)
6 print(img[10][20])
```



Le code ci-dessus permet d'obtenir la valeur de quel pixel ? Donner sa position ligne et colonne

.....
.....

Parcourir une image

```
1 import numpy as np
2 from PIL import Image
3
4 imgpil = Image.open("gateau.png")
5 # On le transforme en tableau
6 img = np.array(imgpil)
7
8 nb_lignes = len(img)
9 nb_colonnes = len(img[0])
10 for i in range(...):
11     for j in range(...):
12         print(img[...][...])
```



Complete le code ci-dessus pour qu'il affiche la valeur de chaque pixel

Modifier un pixel

```
1 import numpy as np
2 from PIL import Image
3
4 imgpil = Image.open("gateau.png")
5 # On le transforme en tableau
6 img = np.array(imgpil)
7
8 img[156][123] = 0
```



Dans le code ci-dessus, on modifie un pixel.
Quel pixel a-t'on modifier (position ligne / colonne) ? Quelle est sa nouvelle couleur ?

.....
.....

Sauvegarder une image

Pour sauvegarder une image, on passe par 3 instructions :

- `np.clip` sert à s'assurer que le tableau `img` ne contienne que des nombres entre 0 et 255.
- `Image.fromarray` transforme le tableau `img` en un objet `Image`.
- `save()` permet de copier l'image dans un fichier au format désiré.

```
1 img = np.clip(img, 0, 255)
2 imgpil = Image.fromarray(img)
3 imgpil.save("mon_image.jpg")
```