

1. Objectif

L'objectif de ces exercices est d'écrire des programmes en Python créant des drapeaux sous la forme d'image numérique.

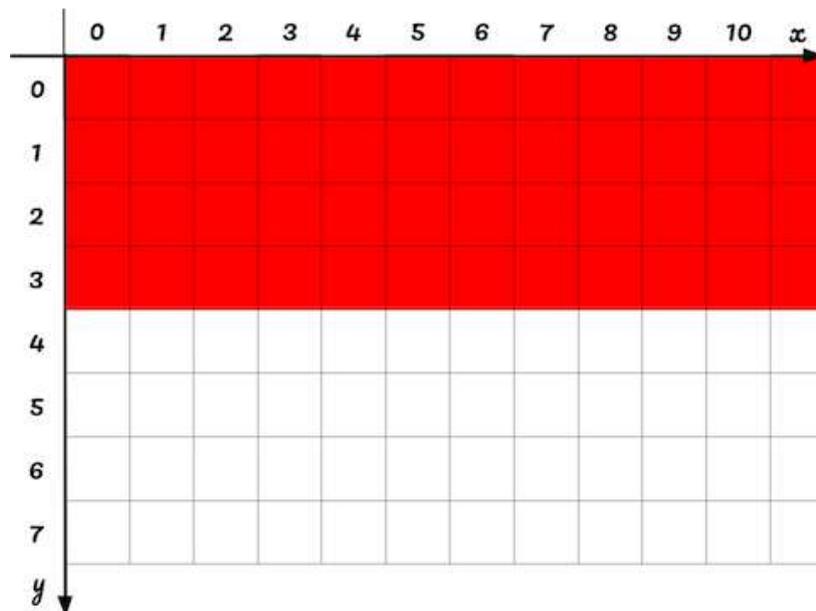
2. Contexte

Il est possible de placer un drapeau de définition *largeur* \times *hauteur* dans un repère orthonomé, où l'axe des abscisses représente la largeur du drapeau, l'axe des ordonnées, sa hauteur.

Chaque pixel du drapeau a une coordonnée (x, y) où $x \in [0; largeur[$ et $y \in [0; hauteur[$.

La couleur d'un pixel du drapeau s'exprime alors en fonction des valeurs de x et/ou y .

Exemple : Pour le drapeau de Monaco, il est possible de définir une fonction $f_{monaco}(y)$ qui associe à l'ordonnée y du pixel, sa couleur en codage RGB.



$$f_{monaco}(y) = \begin{cases} (255, 0, 0) & \text{si } y < \frac{\text{hauteur}}{2} \\ (255, 255, 255) & \text{sinon.} \end{cases} \quad \forall y \in [0; hauteur[$$

Pour rappel, en RGB (255, 0, 0) correspond à la couleur Rouge et (255, 255, 255) correspond à la couleur Blanc.

3. Exercices

Exercice 1. Drapeau de Monaco

```

from PIL import Image

LARGEUR = 600
HAUTEUR = 400

img = Image.new('RGB', (LARGEUR, HAUTEUR))

for y in range(HAUTEUR):
    for x in range(LARGEUR):
        if y < HAUTEUR // 2:
            img.putpixel((x, y), (255, 0, 0))
        else:
            img.putpixel((x, y), (255, 255, 255))

img.show()

```

1. Copier le code Python dans Thonny
2. Enregistrer le fichier avec le nom **monaco.py** et exécuter le programme
3. Répondre aux questions :

1. Quelles lignes permettent de définir les valeurs de la définition de l'image ?
2. En analysant les lignes 8 et 9, il est possible de dire que le drapeau est créé ...
 - Ligne par ligne et pour chaque ligne, colonne par colonne
 - Colonne par colonne et pour chaque colonne, ligne par ligne
3. Quelle est l'utilité de la ligne 10 ?

Exercice 2. Drapeau de Pologne



1. Compléter la fonction $f_{pologne}(y)$ qui associe à l'ordonnée y du pixel, sa couleur en codage RGB du drapeau de Pologne

$$f_{pologne}(y) = \begin{cases} \boxed{} & \text{si } y < \frac{\text{hauteur}}{2} \\ \boxed{} & \text{sinon.} \end{cases} \quad \forall y \in [0; \text{hauteur}[$$

2. Créer un programme **pologne.py** qui crée un drapeau de la Pologne de définition 800 x 600.

Exercice 3. Drapeau des Pays-Bas



1. Compléter la fonction $f_{pays-bas}(y)$ qui associe à l'ordonnée y du pixel, sa couleur en codage RGB du drapeau de Pologne

$$f_{pays-bas}(y) = \begin{cases} \boxed{} & \text{si } y \boxed{} \\ \boxed{} & \text{sinon si } y \boxed{} \\ \boxed{} & \text{sinon.} \end{cases}$$

$\forall y \in [0; hauteur[$

2. Créer un programme `pays_bas.py` qui créé un drapeau de la Pologne de définition 800 x 600.

Exercice 4. Drapeau de la France



1. Compléter la fonction $f_{france}(x)$ qui associe à l'abscisse x du pixel, sa couleur en codage RGB du drapeau de Pologne

$$f_{france}(x) = \begin{cases} (0, 0, 255) & \text{si } \boxed{} \\ \boxed{} & \text{sinon si } \boxed{} \\ \boxed{} & \text{sinon.} \end{cases}$$

$\forall x \in [0; largeur[$

2. Créer un programme `france.py` qui créé un drapeau de la France de définition 800 x 600.

Exercice 5. Drapeau de la Suède



La bande jaune verticale a les caractéristiques suivantes :

- sa largeur est de 40 pixels,
- son centre est au tiers de la largeur du drapeau.

La bande jaune horizontale a les caractéristiques suivantes :

- sa largeur est de 40 pixels,
- son centre est au tiers de la largeur du drapeau.

Cours SNT	Thème : Photographie numérique Générer des drapeaux des pays du monde en Python	PN-E3
-----------	--	-------

- Compléter la fonction $f_{suède}(x, y)$ qui associe aux coordonnées x et y du pixel, sa couleur en codage RGB du drapeau de Suède.

$$f_{suède}(x, y) = \begin{cases} \boxed{} & \text{si } x \in \left[\frac{\text{largeur}}{3} - 20; \left[\frac{\text{largeur}}{3} + 20\right] \right. \\ & \text{ou } y \in \left[\frac{\text{hauteur}}{2} - 20; \left[\frac{\text{hauteur}}{2} + 20\right] \right. \\ \boxed{} & \text{sinon.} \end{cases}$$

$$\forall x \in [0; \text{largeur}[\text{ et } \forall y \in [0; \text{hauteur}[$$

- Compléter le programme dans un fichier `suede.py` et exécuter le pour créer un drapeau de la Suède de définition 900 x 600.

```
from PIL import Image

LARGEUR = ...
HAUTEUR = ...

img = Image.new('RGB', (LARGEUR, HAUTEUR))

for y in range(HAUTEUR):
    for x in range(LARGEUR):
        if (..... < x < .....
            or ..... < y < .....):
            img.putpixel((x, y), (255, 255, 0))
        else:
            img.putpixel((x, y), (0, 0, 255))

img.show()
```

Exercice 6. Drapeau du Japon



Le cercle rouge du drapeau du Japon a les caractéristiques suivantes :

- son centre c est au centre du drapeau,
- son rayon r est de $\frac{3}{10}$ ème de la hauteur du drapeau.

Soit le pixel c , le centre du cercle, de coordonnées (x_c, y_c) , tout pixel p , dont la distance à c est inférieure ou égale à r , a la couleur rouge, sinon il a la couleur blanc.

1. Compléter la fonction $f_{japon}(x, y)$ qui associe aux coordonnées x et y du pixel, sa couleur en codage RGB du drapeau du Japon.

$$f_{japon}(x, y) = \begin{cases} \boxed{} & \text{si } distance((x, y), (x_c, y_c)) \boxed{} \\ \boxed{} & \text{sinon.} \end{cases}$$

$$\forall x \in [0; largeur[\text{ et } \forall y \in [0; hauteur[$$

On suppose qu'il existe une fonction $distance(p_1, p_2)$ qui calcule la [distance euclidienne](#) entre 2 points p_1 et p_2 de coordonnées (x_1, y_1) et (x_2, y_2) .

2. Compléter le programme dans un fichier `japon.py` et exécuter le pour créer un drapeau du Japon de définition 900 x 600.

```
from PIL import Image
from math import dist

LARGEUR = 900
HAUTEUR = 600

x_c = .....
y_c = .....

img = Image.new('RGB', (LARGEUR, HAUTEUR))

for y in range(HAUTEUR):
    for x in range(LARGEUR):
        if dist((x, y), (x_c, y_c)) < (.....):
            img.putpixel((x, y), (255, 0, 0))
        else:
            img.putpixel((x, y), (255, 255, 255))
img.show()
```

