

SNT : Les images numériques

TP3 : Dessiner avec des conditions

Objectifs • Lire et utiliser des boucles imbriquées. • Lire et utiliser des conditions pour dessiner.

Pour programmer, vous pouvez utiliser : <https://trinket.io/python3>

Des codes sont sur : <http://tableauxmaths.fr/spip/spip.php?article235>

I. Lecture et premier dessin

On donne le programme suivant :

Langage Python :

```
1 import matplotlib.pyplot as plt
2 from PIL import Image
3 #Creation d'une image blanche RGB 10x10
4 dessin = Image.new("RGB", (10,10), (255,255,255))
5 #Definitions des couleurs
6 rouge = (255, 0, 0)
7 #On parcourt tous les pixels de l'image
8 for col in range(0, 10):
9     for ligne in range(0, 10):
10        if ligne < 4:
11            dessin.putpixel((col, ligne), rouge)
12
13 plt.imshow(dessin)
14 plt.show()
```

Résultat :

	0	1	2	3	4	5	6	7	8	9
0:										
1:										
2:										
3:										
4:										
5:										
6:										
7:										
8:										
9:										

1. Comment traduire les lignes 8 et 9 de ce programme?

for col in range(0, 10):
for ligne in range(0, 10):

2. Quel est le rôle de ces lignes 8 et 9?

3. Comment traduire les lignes 10 et 11? Quel est leur rôle?
.....
.....

4. Dessiner le résultat à côté du programme puis vérifier en ligne.

5. En modifiant légèrement les lignes 8 et 9 on peut se passer de la ligne 13. Comment?

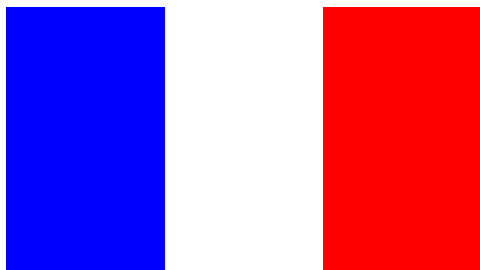
```
8 for col in range(..., ...):
9     for ligne in range(..., ...):
        dessin.putpixel((col, ligne), rouge)
```

6. Quel aurait été l'avantage?

II. Dans l'autre sens

Compléter le code Python ci-contre pour obtenir l'image ci-dessous :

Résultat :



Dimensions : 600 par 400

```
1 import matplotlib.pyplot as plt
2 from PIL import Image
3 #Creation d'une image blanche RGB 10x10
4 dessin2 = Image.new("RGB", (...), (255,255,255))
5 #Definition des couleurs
6 ...
7 ...
8 #On parcourt tous les pixels de l'image:
9 for col in range(0, ...):
10     for ligne in range(0, ...):
11         ...
12         dessin2.putpixel((col, ligne), bleu)
13         ...
14         dessin2.putpixel((col, ligne), rouge)
15 plt.imshow(dessin2)
16 plt.show()
```

III. Combiner des conditions

```
1 import matplotlib.pyplot as plt
2 from PIL import Image
3 #Creation d'une image blanche RGB 10x10
4 dessin3 = Image.new("RGB", (70,40), (255,255,255))
5 #Definitions des couleurs
6 monbleu = (0, 51, 153)
7 #On parcourt tous les pixels de l'image:
8 for col in range(0, 70):
9     for ligne in range(0, 40):
10         if ligne > 14 and ligne < 25:
11             dessin3.putpixel((col, ligne), monbleu)
12         if col > 18 and col < 30:
13             dessin3.putpixel((col, ligne), monbleu)
14 plt.imshow(dessin3)
15 plt.show()
```

Dessiner le résultat du programme ci-contre :

Résultat :

Pour dessiner le résultat de ce programme ci-dessous.

Vous prendrez 1mm pour dimensions d'un pixel.

IV. Créer des drapeaux

1. Écrire un code Python pour dessiner le drapeau de l'Allemagne.
2. Écrire un code Python pour dessiner le drapeau de la Suisse.
3. Écrire un code Python pour dessiner le drapeau de la Norvège.



V. Tu tournes en rond

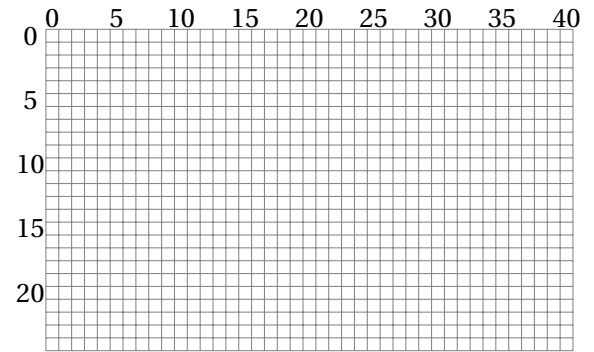
On donne le programme suivant :

```

1 import matplotlib.pyplot as plt
2 from PIL import Image
3 #Creation d'une image blanche RGB 10x10
4 dessin4 = Image.new("RGB", (41,25), (255,255,255))
5 #Definitions des couleurs
6 rouge = (255, 0, 0)
7 #On parcourt tous les pixels de l'image:
8 for col in range(0, 41):
9     for ligne in range(0, 25):
10         if (col-20)**2 + (ligne-12)**2 < 25:
11             dessin4.putpixel((col, ligne), rouge)
12 plt.imshow(dessin4)
13 plt.show()

```

Résultat :



1. Le pixel de coordonnées (col = 26 ; ligne = 14) va-t-il être colorié en rouge? Pourquoi?

.....

2. Tester plusieurs coordonnées de pixels dans le tableau ci-dessous :

col	ligne	col-20	ligne - 12	$(col - 20)^2$	$(ligne - 12)^2$	$(col - 20)^2 + (ligne - 12)^2$	< 25?
0	0	-20	-12	400	144	544	Non
20	12	0	0	0	0	0	Oui
14	2						
3	18						
19	8						
15	12						
16	9						
16	10						
16	11						
16	14						
16	15						

3. Quelle sera la figure géométrique dessinée en rouge?

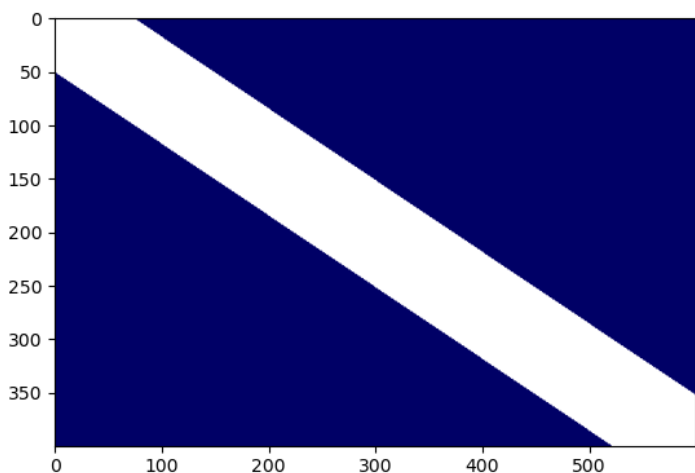
.....

VI. Dessiner en diagonale

On donne le programme ci-contre :

```
1 import matplotlib.pyplot as plt
2 from PIL import Image
3 #Creation d'une image blanche RGB 10x10
4 dessin5 = Image.new("RGB", (600, 400), (0, 0, 102))
5 #Definitions des couleurs
6 blanc = (255, 255, 255)
7 #On parcourt tous les pixels de l'image:
8 for col in range(0, 600):
9     for ligne in range(0, 400):
10         if ligne >= 0.67*col - 50 and ligne <= 0.67*col + 50:
11             dessin5.putpixel((col, ligne), blanc)
12 plt.imshow(dessin5)
13 plt.show()
```

Résultat :



Étude de la première diagonale :

1. Sur le résultat, placer les points $A(0; 0)$ et $B(600; 400)$.
2. Calculer le **coefficient directeur** de la droite (AB) :
.....
3. Combien vaut son **ordonnée à l'origine**?
4. En déduire une équation de la droite (AB) :
5. Tracer en vert la droite d'équation : $y = 0,67x - 50$.



Vous l'avez remarqué! **Les ordonnées vont de haut en bas ici**. Ce n'est pas le cas en Maths!

Étude de la seconde diagonale :

6. Sur le résultat, placer les points $C(0; 400)$ et $D(600; 0)$.
7. Calculer le **coefficient directeur** de la droite (CD) :
8. Combien vaut son **ordonnée à l'origine**?
9. En déduire une équation de la droite (CD) :
10. Quelles sont les conditions pour dessiner la deuxième bande blanche entourant la seconde diagonale :

```
if ligne >= ..... and ligne <= ..... :
    dessin5.putpixel((col, ligne), blanc)
```

11. Compléter le code Python pour dessiner l'Union Jack :
(Commencer par les bandes en diagonale)



12. Pour les plus rapides, dessiner en Python le drapeau des Seychelles :
(Commencer par déterminer les équations des droites en jeu.)

