

Une image numérique.
Pour commencer.

Voici comment il est possible de représenter par exemple un écran de 64 pixels.

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

A l'aide de Thonny, créer un nouveau fichier et recopier les deux lignes ci-dessous.

```
from PIL import Image  
monimage = Image.new("RGB", (8,8))
```

La première permet de charger la librairie et la seconde de créer une image de 64 pixels (pour l'instant tous de couleur noire).

Ajouter cette ligne qui permet de colorer le pixel en haut à gauche (de coordonnées (0,0) en rouge (on précise en effet que le sous-pixel rouge est à 255, le vert à 0 et le bleu à 0).

```
monimage.putpixel((0,0), (255,0,0))
```

Et enfin celle-ci pour visualiser l'image (ne pas hésiter à zoomer pour bien voir le pixel rouge). 4

```
monimage.show()
```

Réaliser les étapes suivantes, à faire valider par votre professeur une par une :

1. Modifier le programme pour colorer en rouge uniquement le deuxième pixel de la première ligne, puis uniquement le troisième.
2. Modifier le programme pour colorer en rouge toute la première ligne, en trouvant la méthode la plus concise.
3. Modifier à nouveau le programme pour colorer en rouge uniquement la deuxième ligne, puis uniquement la troisième ligne.
4. Trouver comment colorer en rouge toutes les lignes de l'image, de la façon la plus simple possible.
5. Afin de pouvoir facilement changer de couleur, créer une fonction `creation_image` utilisant un paramètre nommé `couleur` qui contiendra la code actuel du programme (uniquement la première ligne figure sur le code ci-dessous).

```
def creation_image(couleur):  
    monimage = Image.new("RGB", (8,8))
```

Une fois votre code exécuté, l'appel en console de `creation_image((255,0,0))` devra afficher une image rouge mais `creation_image((0,255,0))` devra afficher une image verte,

6. Ajouter deux paramètres à la fonction, longueur et largeur, pour pouvoir changer le nombre de pixels à la demande.
7. Pour aller plus loin : écrire une fonction permettant la création d'un drapeau.

```
>>> drapeau((0,0,255), (255,255,255), (255,0,0))
```



```
>>> drapeau((0,255,0), (255,255,255), (255,0,0))
```

