

- Cree una imagen de un canal, es decir con una forma (alto, ancho) del tamaño que desee, donde la diagonal principal hacia arriba sea blanca y desde la diagonal hacia abajo sea negra.

Para comenzar, se importan las librerías `matplotlib.pyplot` y `random`, la primera con el objetivo de crear una representación gráfica del resultado, y la segunda solo para darle un valor aleatorio a la diagonal principal de la matriz. Luego se define una imagen, la cual es una matriz, la cual estará compuesta de filas, que a su vez estarán compuestas por un conjunto de píxeles.

Ahora definimos un for dentro de otro for, los cuales para dos índices (los cuales representan el número de fila y columna respectivamente), esto con el fin de construir la matriz/imagen. Ahora dentro de este loop, se ponen unas condiciones, siendo la primera que si el número de fila es menor al de columna, el color del píxel va a ser blanco, es decir, sus valores de rojo, verde y azul serán todos 255. La siguiente condición es que si el número de columna es igual al número de fila, el píxel tendrá un color aleatorio. Por último, si el numero de fila es mayor al de columna, el color del píxel sera negro, es decir; rojo, verde y azul tendran todos el valor de 0.

- Use una imagen de la librería [skimage](#), en la base de datos “data”.  
e implemente alguno de los siguientes procesos:([link](#))
  - Mejoramiento del contraste

Considerando el mejoramiento como un aumento del contraste, ya que en sí mejorarlo implica algo subjetivo y mucho más complejo.

De la librería `Skimage`, se importa `data` y `exposure`, además, se añade `matplotlib` (como `plt`), y luego de esto, usando `data`, se sube la imagen del gato.

Ahora bien, ecualizamos la imagen, por medio de `exposure`, y luego de esto, con la variable `for`, se introduce al comando `plt.imshow` tanto la imagen original como la que ya fue ecualizada, y por último, se imprimen ambas imágenes.