INFORME ACTIVIDAD GRUPAL

EJERCICIO #1

```
import matplotlib.pyplot as plt
```

-Importamos el módulo pyplot y lo renombramos como plt, el cual nos ayuda para la creación de una imagen

```
ancho=200
```

-Definimos una variable llamada ancho y le damos el valor de 200

```
imagen = []
```

-Así mismo creamos otra variable llamada imagen en donde creamos una lista que a medida que el algoritmo avance,se llenará de datos

```
contador=0
```

-Una variable más con el fin de crear un contador para saber la cantidad de pixeles que existen en la imagen

```
for i in range(0, ancho):

  fila = []
```

-Con ayuda de un for, recorremos la variable ancho y creamos una nueva variable (fila) que serán nuestras filas en la matriz, como resultado obtendremos 200 filas

```
  for j in range(0, ancho):
```

-Dentro del primer for, creamos otro que serán nuestras columnas para así separar la imagen por pixeles, al ser un cuadrado, recorre la misma distancia que fila (200) pero de forma horizontal.

```
    if j<i:

     rojo = 0

     verde = 0

     azul = 0
```

-Con ayuda de la función if le damos los parametros al programa, si el numero de columna es menor al numero de la fila, con el sistema RGB pintar color negro. (R=0, G=0, B=0)

```
    else:
      rojo=255
      verde=255
      azul=255
```

-Si no cumple la condición dada, pintara cualquier otro de color blanco (el cual es la combinacion de los colores RGB al máximo (255))

```
    pixel = (rojo, verde, azul)
```

-Creamos una variable nombrada ”pixel” el cual nos ayuda a pintar cada elemento de la fila, ya que pixel será el conjunto de los colores RGB.

```
    fila.append(pixel)
```

-A la lista fila, le agregamos cada “pixel” o columna trabajada para organizar la fila de manera horizontal con la función append

```
    contador+=1
```

-A contador le sumamos 1, y así sucesivamente para contar la cantidad total de píxeles en la imagen

```
  imagen.append(fila)
```

-Organizamos los datos de la variable “imagen” los cuales son cada “fila” en forma de lista (horizontalmente, sin saltos)

```
plt.imshow(imagen)
```

-Mostramos la imagen creada con ayuda del módulo pyplot

```
print(imagen)
```

-Le mostramos al usuario una lista con las características de cada pixel (RGB, por ende tres valores para cada pixel)

```
print('contador queda con : '+str(contador))
```

-Finalmente mostramos el resultado de “contador” de forma string (caracteres); con el cual observamos la cantidad de pixeles usados en la imagen.

Ejercicio #2

MEJORAMIENTO DEL CONTRASTE:

```
from skimage import exposure
```

-Se importa desde la librería skimage el algoritmo de exposición.

```
from skimage import data
```

-Se importa de la librería skimage la base de datos (data) que es donde se encuentra una de las imágenes que podemos seleccionar.

```
import numpy as np
```

-Se importa la librería numpy y se renombra como np para facilitar su uso.

```
import matplotlib.pyplot as plt
```

-Se importa la librería matplotlib, que es numérica y el módulo pyplot y se renombra plt para facilitar su uso, esta librería permitirá hacerle cambios a la figura.

```
imagen = data.retina()
```

-Se declara la variable imagen y ponemos que tipo de imagen queremos, en este caso la imagen de una retina.

```
p2, p98 = np.percentile(imagen, (2,98))
```

-Calcula los dos percentiles (p2, p98) que van a ser utilizados en el siguiente procedimiento para lograr el estiramiento de contraste.

```
img_rescale = exposure.rescale_intensity(imagen, in_range=(p2,p98))
```

-Se realiza el estiramiento del contraste con ayuda del algoritmo exposure y en el rango de percentiles anteriormente calculados.

```
img_eq = exposure.equalize_hist(imagen)
```

-Se realiza la ecualización de la imagen con el algoritmo exposure.

```
img_adapteq = exposure.equalize_adapthist(imagen, clip_limit=0.03)
```

-Se realiza la ecualización adaptativa de la imagen con el algoritmo exposure y le añade su límite de contraste.

```
for eq in (img_eq, img_adapteq):
```

```
plt.imshow(eq)
```

-Funciona para mostrar los datos obtenidos como imagen.

```
 plt.show()
```

-Mostrar las dos imágenes.