

✓ Inteligência Artificial Embarcada - Aula 01

✓ Entendendo a Estrutura do Google Colab

```
!ls
```

```
!pwd
```

```
!pip freeze
```

```
!nvidia-smi
```

✓ Manipulando Imagens

```
#importando as bibliotecas
```

```
import cv2
```

```
from google.colab.patches import cv2_imshow
```

```
'''
```

```
DisabledFunctionError: cv2.imshow() is disabled in Colab'  
because it causes Jupyter sessions  
to crash; see https://github.com/jupyter/notebook/issues/3935.  
As a substitution, consider using
```

```
'''
```

```
import numpy as np
```

```
img3 = cv2.imread('/content/james.jpeg')
```

```
altura, largura, canal = img3.shape  
# cv2.shape --> (linha, coluna, canal)
```

```
print("""A Imagem Possui {} Canais, com uma resolução de {} px por {} px""").\n      format(canal, altura, largura))
```

```
cv2.imshow(img3)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
img3_ajustada = cv2.resize(img3, (int(0.5*largura), int(0.5*altura)),
                             cv2.INTER_AREA)
```

```
cv2.imshow(img3_ajustada)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
#res = cv.resize(img,(2*width, 2*height), interpolation = cv.INTER_CUBIC)
```

```
altura_ajustada,largura_ajustada,canal_ajustado = img3_ajustada.shape
```

```
print("""
A Imagem ajustada Possui {} Canais, com uma resolução de {} px por {} px""".\
      format(canal_ajustado,altura_ajustada,largura_ajustada))
```

```
from IPython.display import display
from PIL import Image
```

```
img3_RGB = cv2.cvtColor(img3, cv2.COLOR_BGR2RGB)
display(Image.fromarray(img3_RGB))
```

```
# Opencv --> Espaço de Cores Default – BGR (Blue Green Red)
# Opencv --> Espaço de Cores Default – RGB (Red Green Blue)
```

```
import os
```

```
randomByteArray = bytearray(os.urandom(120000))
print(type(randomByteArray))
print(randomByteArray)
```

```
NumpyArray = np.array(randomByteArray)
print(type(NumpyArray))
print(NumpyArray)
print(NumpyArray.shape)
print(NumpyArray.min())
print(NumpyArray.max())
```

```
ImagemCinza = NumpyArray.reshape(300, 400)
```

```
alturaCinza,larguraCinza = ImagemCinza.shape
```

```
print("""
```

```
  A Imagem ajustada Possui 1 Canal, com uma resolução de {} px por {} px""").\
    format(alturaCinza,larguraCinza))
```

```
cv2_imshow(ImagemCinza)
```

```
cv2.waitKey(0)
```

```
cv2.destroyAllWindows()
```

```
ImagemBGR = NumpyArray.reshape(100, 400, 3)
```

```
alturaBGR,larguraBGR,canalBGR = ImagemBGR.shape
```

```
print("""
```

```
  A Imagem ajustada Possui {} Canais, com uma resolução de {} px por {} px""").\
    format(canalBGR,alturaCinza,larguraCinza))
```

```
cv2_imshow(ImagemBGR)
```

```
cv2.waitKey(0)
```

```
cv2.destroyAllWindows()
```