

```

##app.py (vs code)
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image
import requests
from io import BytesIO

# Helper function to load image from a URL
def load_image_from_url(url):
    response = requests.get(url)
    return Image.open(BytesIO(response.content))

# Elephant image URL
elephant_url =
"https://upload.wikimedia.org/wikipedia/commons/3/37/African\_Bush\_Elephant.jpg"

# Load elephant image
elephant = load_image_from_url(elephant_url)

# Display original image
plt.figure(figsize=(6, 4))
plt.imshow(elephant)
plt.title("Elephant")
plt.axis("off")
plt.show()

# Convert to NumPy array and print shape
elephant_np = np.array(elephant)
print("Elephant image shape:", elephant_np.shape)

# Convert to grayscale
elephant_gray = elephant.convert("L")

# Display grayscale image
plt.figure(figsize=(6, 4))
plt.imshow(elephant_gray, cmap="gray")
plt.title("Elephant \(Grayscale\)")
plt.axis("off")
plt.show()

```



Elephant



Elephant image shape: (3888, 2592, 3)

Elephant (Grayscale)



```
##mutlicolor.py
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image
import requests
from io import BytesIO

# Load image from URL
def load_image_from_url(url):
    response = requests.get(url)
    return Image.open(BytesIO(response.content))

# Load Elephant image
elephant_url =
"https://upload.wikimedia.org/wikipedia/commons/3/37/African\_Bush\_Elephant.jpg"
elephant = load_image_from_url(elephant_url).convert("RGB")
elephant_np = np.array(elephant)

# Split RGB channels
R, G, B = elephant_np[:, :, 0], elephant_np[:, :, 1],
elephant_np[:, :, 2]

# Create images emphasizing each channel
red_img = np.zeros_like(elephant_np)
green_img = np.zeros_like(elephant_np)
blue_img = np.zeros_like(elephant_np)

red_img[:, :, 0] = R
```

```

green_img[:, :, 1] = G
blue_img[:, :, 2] = B

# Display original and RGB color-emphasized images
plt.figure(figsize=(12, 8))

plt.subplot(2, 2, 1)
plt.imshow(elephant_np)
plt.title("Original Image")
plt.axis("off")

plt.subplot(2, 2, 2)
plt.imshow(red_img)
plt.title("Red Channel Emphasis")
plt.axis("off")

plt.subplot(2, 2, 3)
plt.imshow(green_img)
plt.title("Green Channel Emphasis")
plt.axis("off")

plt.subplot(2, 2, 4)
plt.imshow(blue_img)
plt.title("Blue Channel Emphasis")
plt.axis("off")

plt.tight_layout()
plt.show()

# Optional: Apply a colormap to grayscale
elephant_gray = elephant.convert("L")
elephant_gray_np = np.array(elephant_gray)

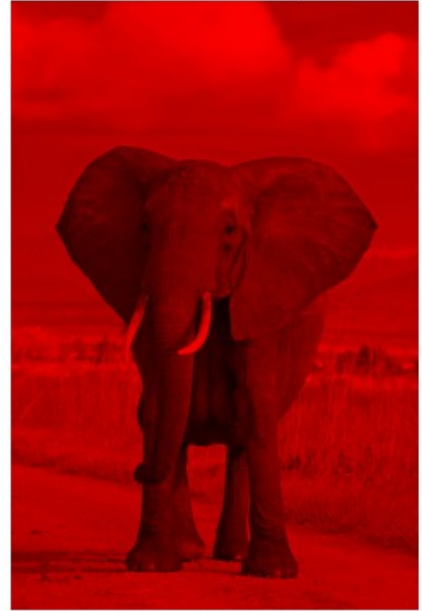
plt.figure(figsize=(6, 5))
plt.imshow(elephant_gray_np, cmap="viridis") # Change cmap to 'hot',
'cool', etc.
plt.title("Colormapped Grayscale")
plt.axis("off")
plt.colorbar()
plt.show()

```

Original Image



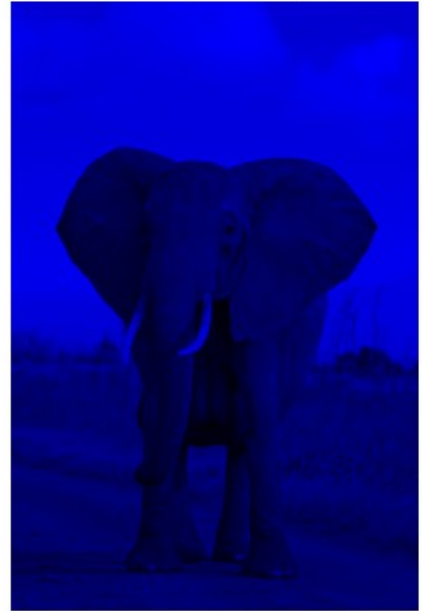
Red Channel Emphasis



Green Channel Emphasis



Blue Channel Emphasis



Colormapped Grayscale

