1. Read an image from a stored file and display the image using OpenCV.

```
import numpy as np
import cv2 as cv
from google.colab.patches import cv2_imshow
import urllib

url = "https://images.unsplash.com/photo-1540648656402-35b0295b2ef0?q=80&w=1740&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8M

resp = urllib.request.urlopen(url)
image = np.asarray(bytearray(resp.read()), dtype="uint8")
img = cv.imdecode(image, cv.IMREAD_COLOR)

cv2_imshow(img)
```

2. Read an image from a stored file and display the image using PIL.

```
import requests
from PIL import Image
from io import BytesIO
import matplotlib.pyplot as plt


url = "https://images.unsplash.com/photo-1540648656402-35b0295b2ef0?q=80&w=1740&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MH
response = requests.get(url)
img_data = response.content

img = Image.open(BytesIO(img_data))


plt.imshow(img)
```

```
plt.axis('off')
plt.show()
plt.show()
```

3. Read an image from a stored file and display the image using Matplotlib.

```
from google.colab import files
from PIL import Image
import matplotlib.pyplot as plt


uploaded = files.upload()


image_path = list(uploaded.keys())[0]
img = Image.open(image_path)

plt.imshow(img)
plt.axis('off')
plt.show()
```

Choose Files IMG_20231…163605.jpg

- **IMG_20231003_163605.jpg**(image/jpeg) - 3468263 bytes, last modified: 3/10/2023 - 100% done

```
Saving IMG_20231003_163605.jpg to IMG_20231003_163605.jpg
```

4. Read an image from a stored file and display the image using scikit-image.

```
from google.colab import files
from skimage import io
import matplotlib.pyplot as plt


uploaded = files.upload()

image_path = list(uploaded.keys())[0]
img = io.imread(image_path)


plt.imshow(img)
# plt.axis('off')
plt.show()
```

Choose Files IMG_20231…163605.jpg

- **IMG_20231003_163605.jpg**(image/jpeg) - 3468263 bytes, last modified: 3/10/2023 - 100% done

```
Saving IMG_20231003_163605.jpg to IMG_20231003_163605 (1).jpg
```

5. Extract information like size, shape, data type of an image using OpenCV.

```
print(img.shape)
print("Size of Image");
print(img.size)
print("datatype of Image")
print(img.dtype)
```

```
(3417, 5056, 3)
Size of Image
51829056
datatype of Image
uint8
```

6. Split an RGB image into three channels and display each channel using OpenCV.

```
import cv2 as cv
from google.colab.patches import cv2_imshow
b, g, r = cv.split(img)

cv2_imshow(b)
cv2_imshow(g)
cv2_imshow(r)
```

Show hidden output

7. Split an RGB image into three channels and merge the channels as [B,G,R] and [G,B,R]

```
image_merge = cv.merge([b, g, r])
cv2_imshow(image_merge)
image_merge2 = cv.merge([g, b, r])
cv2_imshow(image_merge2)
```

Show hidden output

8.Extract information like size, format, mode of an image using PIL.

```
from google.colab import files
from PIL import Image


uploaded = files.upload()

image_path = list(uploaded.keys())[0]
img = Image.open(image_path)

print(f"Format: {img.format}")
print(f"Size: {img.size}")
print(f"Mode: {img.mode}")
```

Choose Files IMG_20231...162812.jpg

- **IMG_20231003_162812.jpg**(image/jpeg) - 8309894 bytes, last modified: 3/10/2023 - 100% done

```
Saving IMG_20231003_162812.jpg to IMG_20231003_162812.jpg
Format: JPEG
Size: (6144, 8192)
Mode: RGB
```

9. Read an image and rotate the image in 30, 60, 90, 120, 150 degree using PIL. Display all rotated images.

```
from PIL import Image
import matplotlib.pyplot as plt

image_path = list(uploaded.keys())[0]


img = Image.open(image_path)

angles = [30, 60, 90, 120, 150]
rotated_images = [img.rotate(angle) for angle in angles]


plt.figure(figsize=(15, 10))

plt.subplot(2, 3, 1)
plt.imshow(img)
plt.title("Original Image")
plt.axis('off')

for i, angle in enumerate(angles):
    plt.subplot(2, 3, i + 2)
    plt.imshow(rotated_images[i])
    plt.title(f"Rotated {angle} degrees")
    plt.axis('off')

plt.show()
```

Show hidden output

10. Read an image and flip the image Horizontally and Vertically. Display all result images.

```
from google.colab import files
from PIL import Image
import matplotlib.pyplot as plt


uploaded = files.upload()

image_path = list(uploaded.keys())[0]
img = Image.open(image_path)


img_flipped_horizontally = img.transpose(Image.FLIP_LEFT_RIGHT)
img_flipped_vertically = img.transpose(Image.FLIP_TOP_BOTTOM)


plt.figure(figsize=(15, 5))


plt.subplot(1, 3, 1)
plt.imshow(img)
plt.title("Original Image")
plt.axis('off')


plt.subplot(1, 3, 2)
plt.imshow(img_flipped_horizontally)
plt.title("Horizontally Flipped")
plt.axis('off')


plt.subplot(1, 3, 3)
plt.imshow(img_flipped_vertically)
plt.title("Vertically Flipped")
plt.axis
```

Show hidden output