NUMPY , OPEN CV
APLIKASI PENGOLAHAN CITRA

Arranged By:

Rajendra Rakha Arya Prabaswara

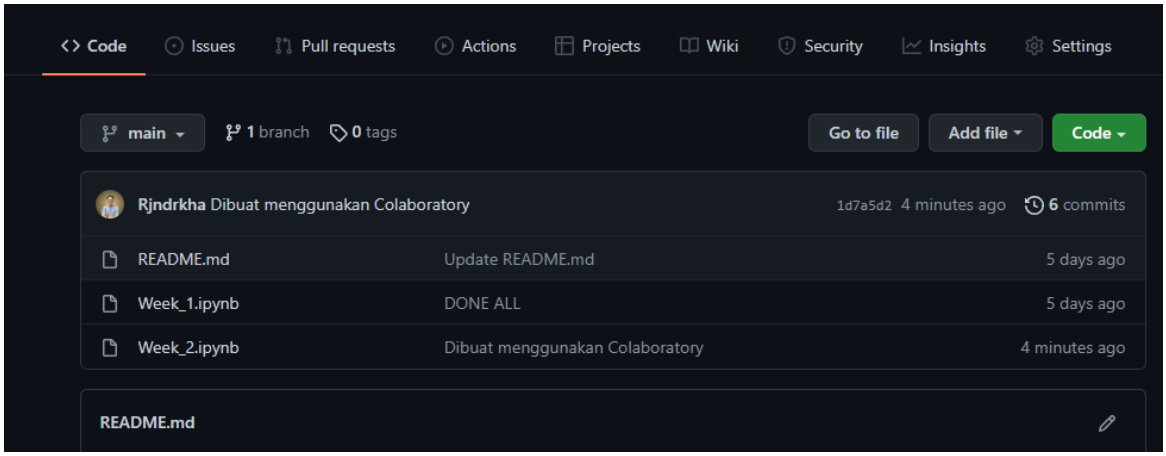
(1941720080/19)

PROGRAM STUDI D-IV TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

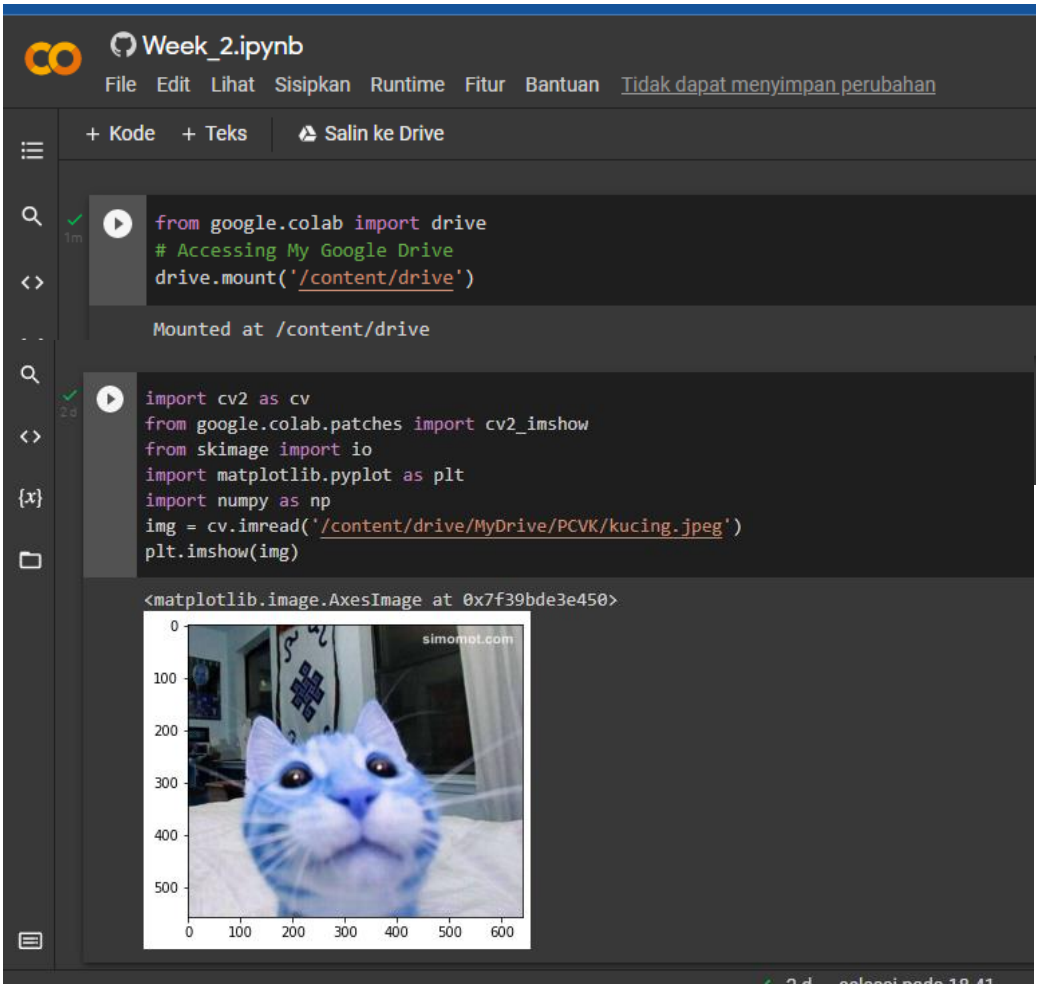
POLITEKNIK NEGERI MALANG

1. Save Project To Github



https://github.com/Rjndrkha/PCVK_Genap_2022

2. Importing File & Image from Google Drive



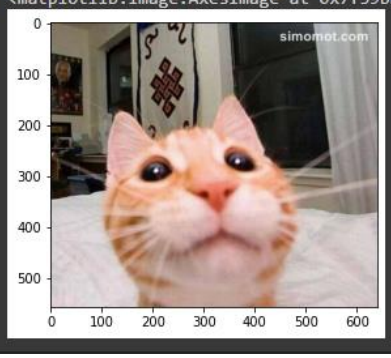
S

Image with BGR Channel Preview

3. Image With RGB Channel Prevew

```
img2 = img
img3 = cv.cvtColor(img,cv.COLOR_BGR2RGB) #konversi channel BGR -> RGB
plt.imshow(img3)
```

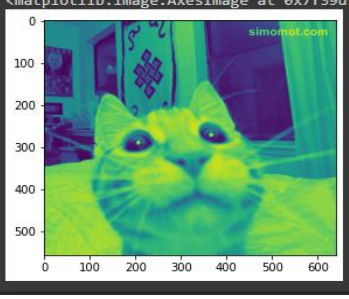
```
<matplotlib.image.AxesImage at 0x7f39bb8f0290>
```



4. Image With Grayscale Channel Preview

```
img_gray = cv.imread('/content/drive/MyDrive/PCVK/kucing.jpeg', cv.IMREAD_GRAYSCALE)
plt.imshow(img_gray)
```

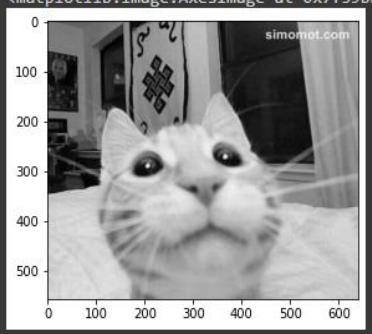
```
<matplotlib.image.AxesImage at 0x7f39d7077a90>
```



5. Image With Grayscale Mapping

```
plt.imshow(img_gray, cmap='gray')
```

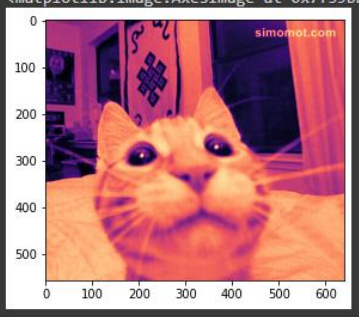
```
<matplotlib.image.AxesImage at 0x7f39bb7d3c10>
```



6. **Image With Magma Mapping**

```
plt.imshow(img_gray, cmap='magma')
```

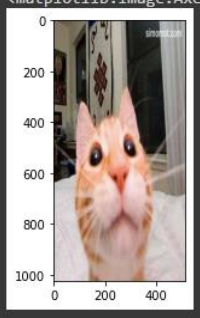
```
<matplotlib.image.AxesImage at 0x7f39bb7c5810>
```



7. **Resized image in 512 X 1024 px**

```
img4 = cv.resize(cv.cvtColor(img,cv.COLOR_BGR2RGB), (512,1024))
plt.imshow(img4)
```

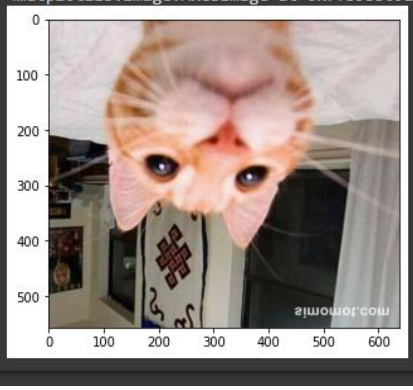
```
<matplotlib.image.AxesImage at 0x7f39bb735690>
```



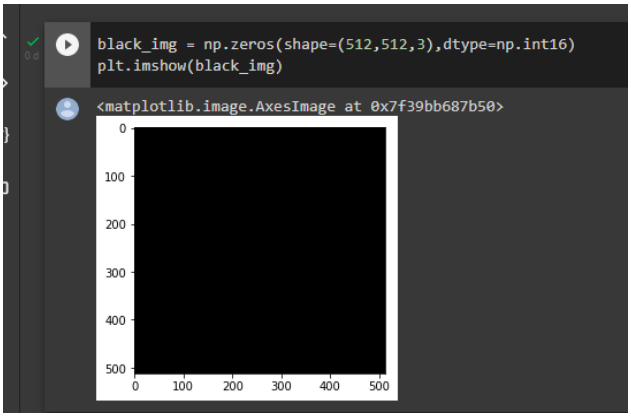
8. **Image Is Mirrored Upside Down**

```
img5 = cv.flip(cv.cvtColor(img,cv.COLOR_BGR2RGB),0)

fig = plt.figure(figsize=(5,5))
ax = fig.add_subplot(111)
ax.imshow(img5)
```

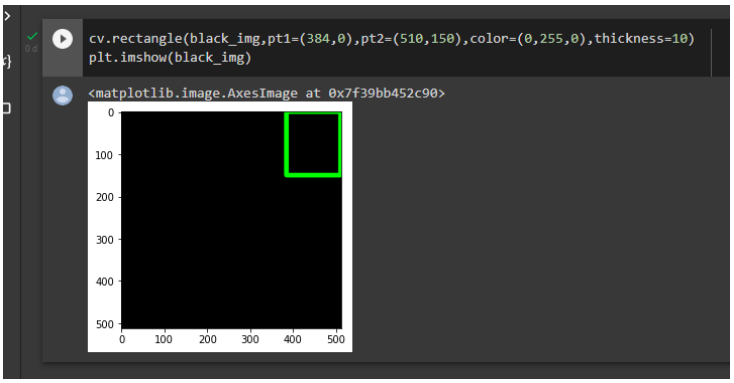
```
<matplotlib.image.AxesImage at 0x7f39bb691b90>
```



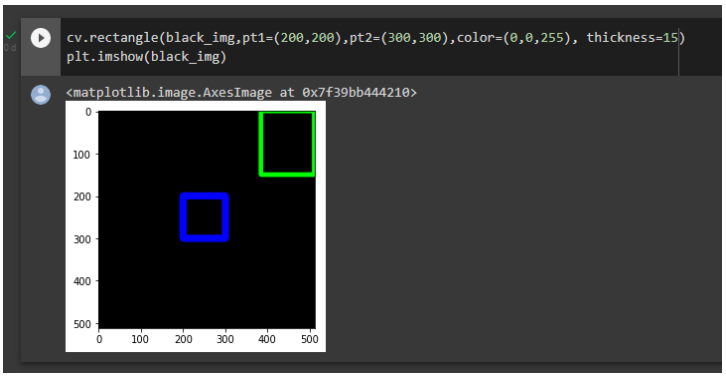
9. **Creating 2D Geometry Image using Open CV**



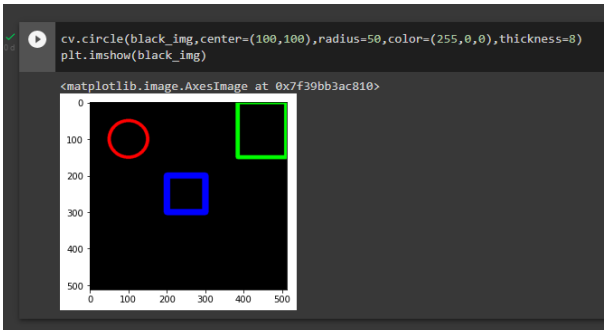
10. **Add rectangle defining coordinat pt1 and pt2**



11. **Add a square shape according to the pt1 and pt2 coordinates written in the program code.**



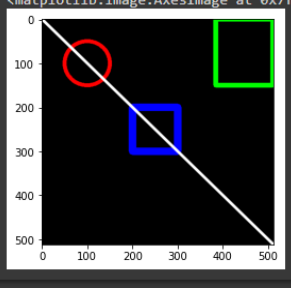
12. **Add a circle shape according to the radius written in the program code.**



13. Add Diagonal Line in Image

```
cv.line(black_img,pt1=(0,0),pt2=(512,512),color=(255,255,255),thickness=5)
plt.imshow(black_img)
```

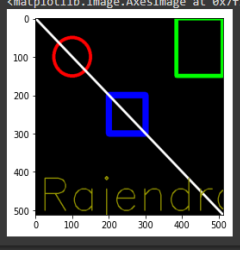
```
<matplotlib.image.AxesImage at 0x7f39bb315ed0>
```



14. Adding text in image

```
font = cv.FONT_HERSHEY_SIMPLEX
cv.putText(black_img,text='Rajendra',org=(10,500),fontFace=font,fontScale=4,color=(155,155,0),thickness=2,lineType=cv.LINE_AA)
plt.imshow(black_img)
```

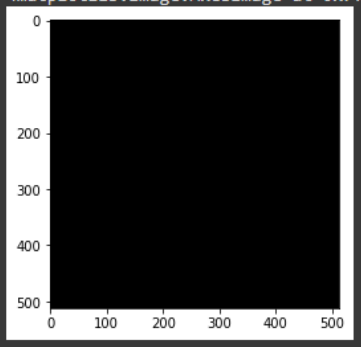
```
<matplotlib.image.AxesImage at 0x7f39bb6a1190>
```



15. Creating new Black Image

```
black_img2=np.zeros(shape=(512,512,3),dtype=np.int32)
plt.imshow(black_img2)
```

```
<matplotlib.image.AxesImage at 0x7f39bb0d8a50>
```



16. **This is the program code to initialize a NumPy array with the data type int32**

```
vertices = np.array([[100,300],[200,200],[400,300],[200,400]],dtype=np.int32)
vertices
```

```
array([[100, 300],
       [200, 200],
       [400, 300],
       [200, 400]], dtype=int32)
```

17. **Reshaped the array**

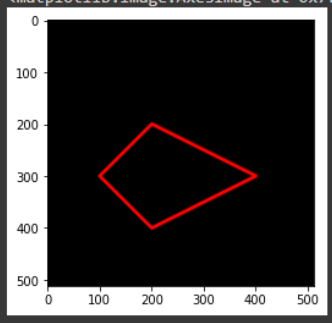
```
pts = vertices.reshape((-1,1,2)) # nilai 2 untuk menunjukkan bahwa tiap titik dibuat 3 channel yang mewakili R,G,B
pts
```

```
array([[[100, 300]],

       [[200, 200]],

       [[400, 300]],

       [[200, 400]]], dtype=int32)
```

18. **Add a polyline to the second black image**

```
cv.polylines(black_img2,[pts],isClosed=True,color=(255,0,0),thickness=5)
plt.imshow(black_img2)
```

```
<matplotlib.image.AxesImage at 0x7f39bb052210>
```



Question

1. What is the difference between the images displayed without and with matplotlib?
2. What is the difference and effect of creating a black image between int16 and int32 data types?
3. What is the use of "google.colab.patches import cv2_imshow" in the following code snippet

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

4. What is the use of "skimage import io" in the code snippet for question number 3

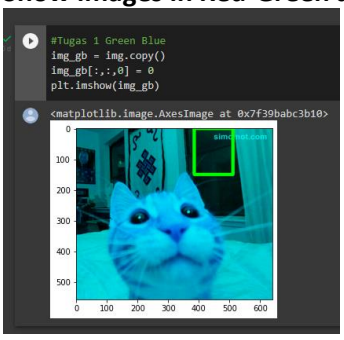
1. Images displayed with matplotlib can represent data into colors and glyphs such as markers (circles), lines (lines) and polygons. While the image displayed without matplotlib cannot represent data like the image displayed using matplotlib.
2. Difference in image size, depending on int. the smaller the int data type given, the less the capacity and the larger the int data type, the greater the capacity. It's just a different capacity from int16 and int32
3. The code in the line above adds one patch because cv.imshow doesn't run on Google Colab or Jupyter Notebook. The img variable is the original image, while the grayImg has been converted to black and white (gray).
4. To load images, read and write images

TASK

1. By using figsize, notice whether the image pixel size also changes?

Yes the image pixel also changes

2. Show images in Red-Green and Green-Blue channels only!



3. **Show image row 10-100, column 10-100!**

Number 3

```
img2 = img
img3 = cv.cvtColor(img,cv.COLOR_BGR2RGB)
plt.imshow(img3[10:100, 10:100])
```

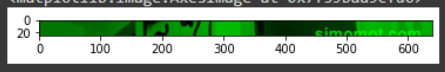
```
<matplotlib.image.AxesImage at 0x7f39bab32d50>
```

4. **Show image rows 5-30, all columns, channel Green only!**

Number 4

```
plt.imshow(img5[5:30])
```

```
<matplotlib.image.AxesImage at 0x7f39baa9cfd0>
```

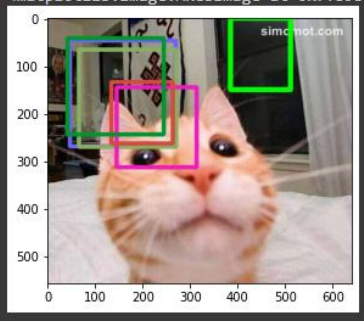


5. **Create 5 boxes of different sizes and colors in one image. it is recommended to use random numbers!**

+ Kode + Teks Salin ke Drive

```
end_point = (endp, endp)

color = (c1, c2, c3)

thickness = 7

img6 = cv.rectangle(img3, start_point, end_point, color, thickness)

# Displaying the image
plt.imshow(img6)
```

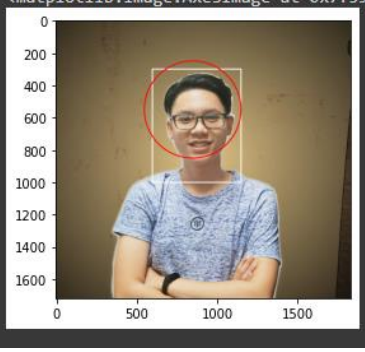
```
<matplotlib.image.AxesImage at 0x7f39baa80f90>
```



6. **Make rectangles and circles on the face of your photo image when you are active (not a passport photo).**

```
[67] img_sc7 = imgme1
     cv.rectangle(img_sc7,pt1=(600,1000),pt2=(1150,300),color=(255,255,255),thickness=5)
     cv.circle(img_sc7,center=(850,550),radius=300,color=(255,0,0),thickness=5)
     plt.imshow(img_sc7)
```

```
<matplotlib.image.AxesImage at 0x7f39b9dcde90>
```

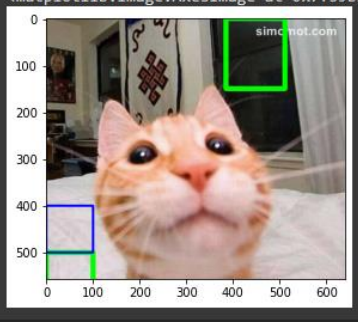


7. **Create a rectangle in the lower left corner of channel B in the RGB color space from the kitten / lena / mandrill / male / female / couple / sailboat / peppers image !**

Number 7

```
img_sc10 = cv.cvtColor(img, cv.COLOR_BGR2RGB)
cv.rectangle(img_sc10,pt1=(0,400),pt2=(100,500),color=(0,0,255),thickness=3)
plt.imshow(img_sc10)
```

```
<matplotlib.image.AxesImage at 0x7f39b9ca54d0>
```



8. **Complete the writing of the file name on the image file from question no.6. use the font, font size, and font color according to your wishes.**

Number 8

```
font = cv.FONT_HERSHEY_SIMPLEX
cv.putText(imgme1,text='Rajendra',org=(10,1600),fontFace=font,fontScale=10,color=(255,0,0),thickness=10,lineType=cv.LINE_AA)
plt.imshow(imgme1)
```

```
<matplotlib.image.AxesImage at 0x7f39b7ff0a10>
```