**Reading And Displaying Images**

import cv2

from google.colab.patches import cv2\_imshow

img1 =cv2.imread("download.jfif",1)

cv2\_imshow(img1)

Graphical user interface, text, application, Teams

Description automatically generated

img1.shape

(255, 198, 3)

img1.size

151470

img1.dtype

dtype('uint8')



import cv2

import numpy as np

image=cv2.rotate(img1,cv2.ROTATE\_90\_COUNTERCLOCKWISE)

cv2\_imshow(image)

Text

Description automatically generated

import cv2

import numpy as np

image=cv2.rotate(img1,cv2.ROTATE\_90\_CLOCKWISE)

cv2\_imshow(image)

Graphical user interface, text

Description automatically generated

import numpy as np

#create a black image

img=np.zeros((512,512,3),np.uint8)

cv2\_imshow(img)

Shape

Description automatically generated

img=np.zeros((512,512,3),np.uint8)

img=cv2.line(img,(0,0),(511,511),(255,0,0),5)

cv2\_imshow(img)

Shape

Description automatically generated

img=np.zeros((512,512,3),np.uint8)

img=cv2.line(img,(0,0),(511,511),(255,0,0),5)

img=cv2.rectangle(img,(384,0),(510,128),(0,255,0),3)

cv2\_imshow(img)

Shape

Description automatically generated

img=np.zeros((512,512,3),np.uint8)

img=cv2.line(img,(0,0),(511,511),(255,0,0),5)

#rectangle takes 2 diagonal co-ordinates

img=cv2.rectangle(img,(384,0),(510,128),(0,255,0),3)

# circle requires center co-ordinates (x,y) and radius

img=cv2.circle(img,(447,63),25,(0,0,255),-1) #-1 to fill the color

cv2\_imshow(img)

cv2\_imshow(img)

A picture containing shape

Description automatically generated

import numpy as np

import cv2

from google.colab.patches import cv2\_imshow

#create a black image

img=np.zeros(512,512,3),np.uint8)

draw a diagonal blue line with thickness of 5px

img=cv2.line(img,(0,0),(511,511),(255,0,0),5)

img=cv2.rectangle(img,(384,0),(510,128),(0,255,0),3)

img=cv2.circle(img,(447,63),25(0,0,255),-1)

img=cv2.ellipse(img,(256,256),(100,50),0,0,180,255,-1)

cv2\_imshow(img)

cv2.waitkey(1)

cv2.destroyAllWindows()

A picture containing icon

Description automatically generated

img=np.zeros((512,512,3),np.uint8)

pts=np.array([[10,5],[20,30],[70,20],[50,10]],np.int32)

pts=pts.reshape((-1,1,2))

img=cv2.polylines(img,[pts],True,(0,255,255)) #True

cv2\_imshow(img)

#cv2.waitkey(0)

cv2.destroyAllWindows()

Shape

Description automatically generated

from google.colab.patches import cv2\_imshow

img=np.zeros((512,512,3),np.uint8)

pts=np.array([[10,5],[20,30],[70,20],[50,10]],np.int32)

pts=pts.reshape((-1,1,2))

img=cv2.polylines(img,[pts],True,(0,255,255)) #True indicates closed polygon

font=cv2.FONT\_HERSHEY\_SIMPLEX

cv2.putText(img,'OpenCV',(10,500),font,4,(255,255,255),2,cv2.LINE\_AA) #

cv2\_imshow(img)

#cv2.waitkey(0)

cv2.destroyAllWindows()

A picture containing application

Description automatically generated

img = np.ones((512,512,3),np.uint8)

img.fill(255)

img = cv2.ellipse(img, (256, 80), (60,60), 120,0,300,(0,0,255),-1)

img = cv2.ellipse(img, (256, 80), (20,20), 120,0,300,(255,255,255),-1)

img = cv2.ellipse(img, (176, 200), (60,60), 0,0,300,(0,255,0),-1)

img = cv2.ellipse(img, (176, 200), (20,20), 0,0,300,(255,255,255),-1)

img = cv2.ellipse(img, (336, 200), (60,60), 300,0,300,(255,0,0),-1)

img = cv2.ellipse(img, (336, 200), (20,20), 300,0,300,(255,255,255),-1)

font = cv2.FONT\_HERSHEY\_SIMPLEX

img = cv2.putText(img, "OpenCV", (196,296), font, 1, (0,0,0), 4, cv2.LINE\_AA)

cv2\_imshow(img)

Logo

Description automatically generated with medium confidence

img=np.zeros((420,400,3),np.uint8)

img=cv2.rectangle(img,(50,50),(350,450),(105,105,105),-1)

img=cv2.rectangle(img,(100,100),(300,400),(169,169,169),-1)

img=cv2.rectangle(img,(150,150),(250,350),(255,255,255),-1)

cv2\_imshow(img)

Logo

Description automatically generated