```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
from google.colab import drive
drive.mount('/content/drive')
import cv2

from google.colab.patches import cv2_imshow


img = cv2.imread("/content/drive/My Drive/Colab Notebooks/ajit.JPG",cv2.IMREAD_UNCHANGED)
cv2_imshow(img)
print(img)
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/dri
```

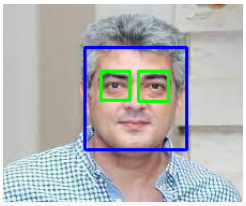
```
[[[195 198 196]
  [195 198 196]
  [195 198 196]
  ...
  [176 188 194]
  [178 190 196]
  [179 191 197]]

 [[195 198 196]
  [195 198 196]
  [195 198 196]
  ...
  [175 187 193]
  [177 189 195]
  [178 190 196]]

 [[194 197 195]
  [194 197 195]
  [194 197 195]
  ...
  [176 187 191]
  [177 188 192]
  [177 188 192]]

 ...

 [[255 237 197]
  [238 209 170]
  [124  95  56]
  ...
  [255 255 247]
  [255 255 250]
  [249 244 241]]

 [[255 228 189]
  [255 244 205]
  [211 180 141]
  ...
  [255 250 241]
  [255 255 251]
  [247 242 239]]

 [[234 202 161]
  [151 119  78]
  [122  90  49]
  ...
  [250 243 234]
  [249 243 236]
  [255 252 247]]]
```

```
from google.colab import drive
drive.mount('/content/drive')
import cv2

from google.colab.patches import cv2_imshow


img = cv2.imread("/content/drive/My Drive/Colab Notebooks/ajit.JPG",cv2.IMREAD_UNCHANGED)
cv2_imshow(img)
print(img)

cv2.waitKey(1000)

cv2.destroyAllWindows()

gray_img=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
cv2_imshow(gray_img)
print(gray_img)
```

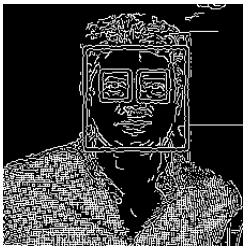
```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/dri
```

```
[[[195 198 196]
  [195 198 196]
  [195 198 196]
  ...
  [176 188 194]
  [178 190 196]
  [179 191 197]]

 [[195 198 196]
  [195 198 196]
  [195 198 196]
  ...
  [175 187 193]
  [177 189 195]
  [178 190 196]]

 [[194 197 195]
  [194 197 195]
  [194 197 195]
  ...
  [176 187 191]
  [177 188 192]
  [177 188 192]]

 ...

 [[255 237 197]
  [238 209 170]
  [124  95  56]
  ...
  [255 255 247]
  [255 255 250]
  [249 244 241]]

 [[255 228 189]
  [255 244 205]
  [211 180 141]
  ...
  [255 250 241]
  [255 255 251]
  [247 242 239]]

 [[234 202 161]
  [151 119  78]
  [122  90  49]
  ...
  [250 243 234]
  [249 243 236]
  [255 252 247]]]
```

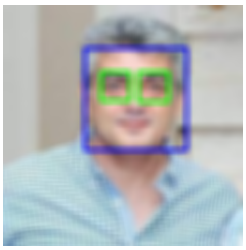
```
import numpy as np
import cv2

face_cascade = cv2.CascadeClassifier("/content/drive/My Drive/Colab Notebooks/haarcascade_frontalface_default.xml")
eye_cascade = cv2.CascadeClassifier("/content/drive/My Drive/Colab Notebooks/haarcascade_eye.xml")

img = cv2.imread("/content/drive/My Drive/Colab Notebooks/ajit.JPG",cv2.IMREAD_UNCHANGED)

gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

faces = face_cascade.detectMultiScale(gray, 1.3, 5)
#faces = face_cascade.detectMultiScale(gray)

for (x,y,w,h) in faces:
    cv2.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)
    roi_gray = gray[y:y+h, x:x+w]
    roi_color = img[y:y+h, x:x+w]
    eyes = eye_cascade.detectMultiScale(roi_gray)
    for (ex,ey,ew,eh) in eyes:
        cv2.rectangle(roi_color,(ex,ey),(ex+ew,ey+eh),(0,255,0),2)

cv2_imshow(img)

k = cv2.waitKey(0)
if k == 27:         # wait for ESC key to exit
    cv2.destroyAllWindows()
elif k == ord('s'): # wait for 's' key to save and exit
    cv2.imwrite('messigray.png',img)
    cv2.destroyAllWindows()
```



```
edges=cv2.Canny(img,100,150)
cv2_imshow(edges)
```



```
dst=cv2.GaussianBlur(img,(9,9),cv2.BORDER_DEFAULT)
cv2_imshow(dst)
```



```
from google.colab import drive
drive.mount('/content/drive')
import cv2

from google.colab.patches import cv2_imshow


img = cv2.imread("/content/drive/My Drive/Colab Notebooks/ajit.JPG",cv2.IMREAD_UNCHANGED)
cv2_imshow(img)
scale = 50
w = int(img.shape[1]*scale/100)
h = int(img.shape[0]*scale/100)
dsize = (w,h)
output = cv2.resize(img,dsize)
cv2_imshow(output)
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive",
```