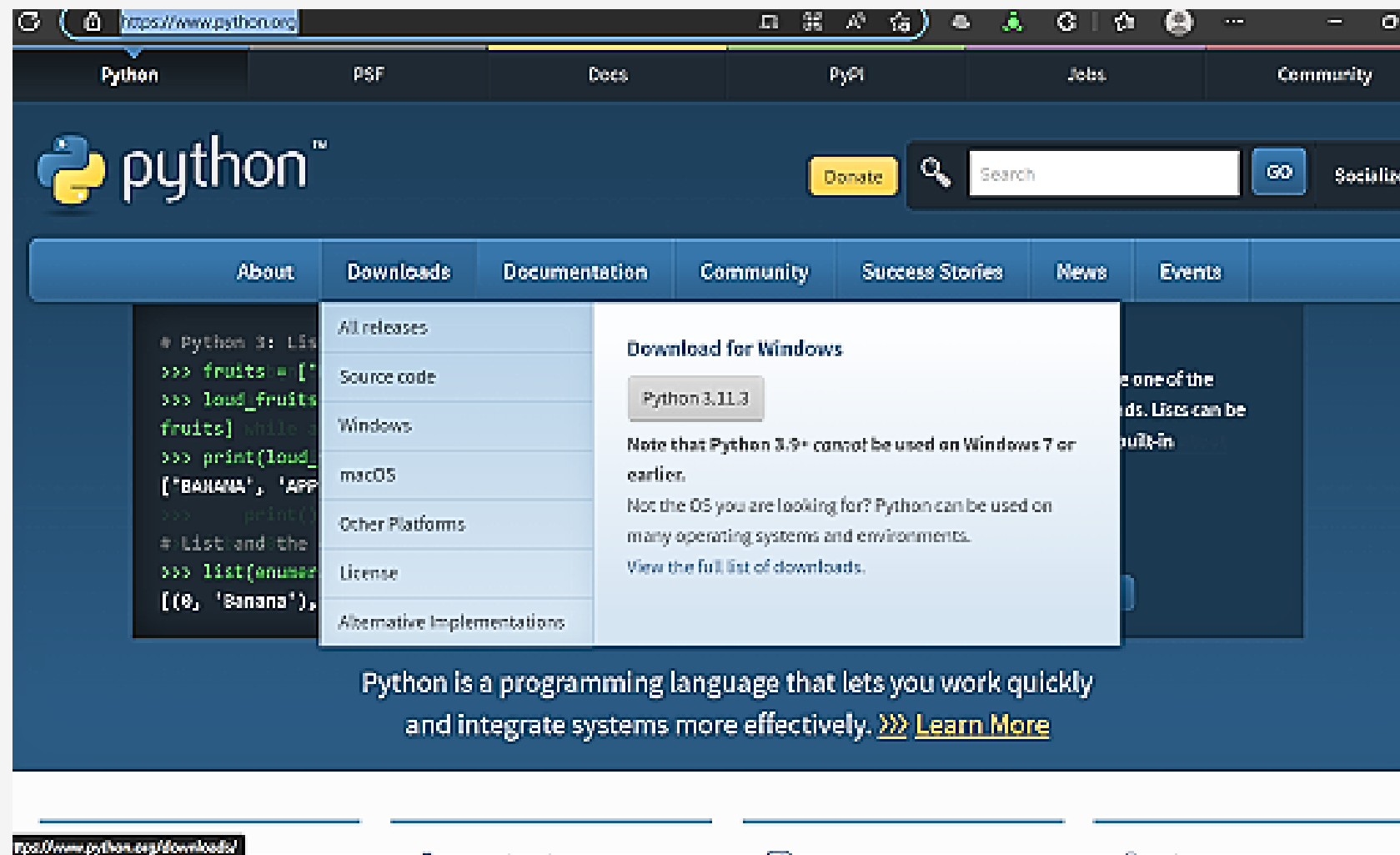


PYTHON INSTALLATION

<https://www.python.org/>

- Visit the site given above.



- After installing python open Terminal/Command Prompt.
- Check your python version

```
C:\Windows\system32\cmd.exe

Microsoft Windows [Version 10.0.19044.2728]
(c) Microsoft Corporation. All rights reserved.

C:\Users\vedan>python --version
Python 3.10.6

C:\Users\vedan>
```

READING AN IMAGE

```
1 import cv2
2 ## LOAD AN IMAGE USING 'IMREAD'
3 img = cv2.imread("coffee.jpg")
```

BLUR IMAGE

```
import cv2

# Load an image
img = cv2.imread("coffee.jpg")
img = cv2.resize(img, (640, 480))
# Apply Gaussian blur with kernel size (5, 5)
blurred_img = cv2.GaussianBlur(img, (5, 5), 0)

# Display the blurred image
cv2.imshow("Blurred Image", blurred_img)
cv2.waitKey(0)
```

IMAGE.SHAPE

```
import cv2

# Load an image
img = cv2.imread('coffee.jpg')

# Get the shape of the image
print(img.shape)
```

RESIZING THE IMAGE

```
import cv2

## LOAD AN IMAGE USING 'IMREAD'
img = cv2.imread("coffee.jpg")
#RESIZING THE IMAGE
img = cv2.resize(img, (640, 480))
```

RESIZING THE IMAGE

- Another way to resize the image

```
import cv2

## LOAD AN IMAGE USING 'IMREAD'
img = cv2.imread("coffee.jpg")
#RESIZING THE IMAGE
img = cv2.resize(img,(0,0),fx=0.1,fy=0.1)
```

DISPLAYING THE IMAGE

```
import cv2

## LOAD AN IMAGE USING 'IMREAD'
img = cv2.imread("coffee.jpg")
#RESIZING THE IMAGE
# img = cv2.resize(img,(640,480))
img = cv2.resize(img,(0,0),fx=0.1,fy=0.1)
#DISPLAYING A IMAGE
cv2.imshow("coffee",img)
cv2.waitKey(0)
```

CANNY IMAGE

```
import cv2

# Load an image
img = cv2.imread("coffee.jpg")
img = cv2.resize(img,(640,480))

# Convert to grayscale
gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

# Apply Canny edge detection with threshold values of 100 and 200
canny_img = cv2.Canny(gray_img, 100, 200)

# Display the Canny image
cv2.imshow("Canny Image", canny_img)
cv2.waitKey(0)
```

CROPPING A IMAGE

```
import cv2

# Load an image
img = cv2.imread('coffee.jpg')
img = cv2.resize(img, (640, 480))

img = img[0:200, 440:640]
##height first then width
cv2.imshow("cropped_img", img)
cv2.waitKey(0)
```

VIDEO CAPTURE

BGR TO GRAYSCALE IMAGE

```
import cv2

# Load an image in BGR format
img = cv2.imread("coffee.jpg")
img = cv2.resize(img, (640, 480))

# Convert BGR to grayscale
gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

# Display the grayscale image
cv2.imshow("Grayscale Image", gray_img)
cv2.waitKey(0)
```

```
import cv2
```

```
#VIDEO CAPTURING
```

```
cap = cv2.VideoCapture('video.mp4') # Use video file
#cap = cv2.VideoCapture(0) # Use camera with index 0
# for external camera you may use 1,2...
```


VIDEO CAPTURE

```
import cv2

#VIDEO CAPTURING

cap = cv2.VideoCapture('video.mp4') # Use video file
#cap = cv2.VideoCapture(0) # Use camera with index 0
# for external camera you may use 1,2...

while cap.isOpened():
    success, frame = cap.read()
    if not success:
        break
    cv2.imshow('Video', frame)

    key = cv2.waitKey(1)
    if key == ord('q'):
        break
```

LINE, SHAPE & TEXT

```
1  import cv2
2  import numpy as np
3  img=cv2.imread("coffee.jpg")
4  image=cv2.namedWindow("image", cv2.WINDOW_NORMAL)
5  image=cv2.resizeWindow("image", 1920, 1080)
6  img=cv2.resize(img,(1920,1080))
7
8  cv2.line(img,(0,0),(1920,1080),(255, 0,0),10)
9  cv2.rectangle(img,(0,0),(250,350),(0,0,255),2)
10 cv2.circle(img,(400,50),30,(0,255,0),5)
11 cv2.putText(img," OPENCV ",(300,200),cv2.FONT_HERSHEY_COMPLEX,1,(0,150,0),3)
12
13 cv2.imshow("image",img)
```

FACE DETECTION

```
1  import cv2
2
3  # Load the cascade
4  face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
5
6  # To capture video from webcam.
7  cap = cv2.VideoCapture(0)
8  # To use a video file as input
9  # cap = cv2.VideoCapture('filename.mp4')
10
11 while True:
12     # Read the frame
13     _, img = cap.read()
14     # Convert to grayscale
15     gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
16     # Detect the faces
17     faces = face_cascade.detectMultiScale(gray, 1.1, 4)
18     # Draw the rectangle around each face
19     for (x, y, w, h) in faces:
20         cv2.rectangle(img, (x, y), (x+w, y+h), (255, 0, 0), 2)
21     # Display
22     cv2.imshow('img', img)
23     # Stop if escape key is pressed
24     k = cv2.waitKey(30) & 0xff
25     if k==27:
26         break
27 # Release the VideoCapture object
28 cap.release()
```