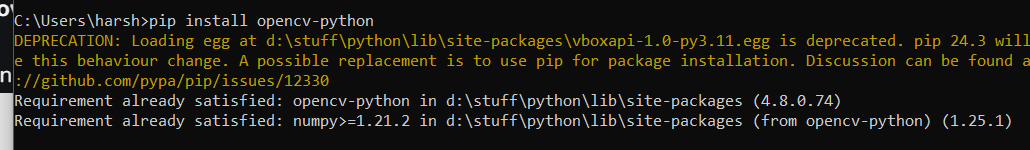
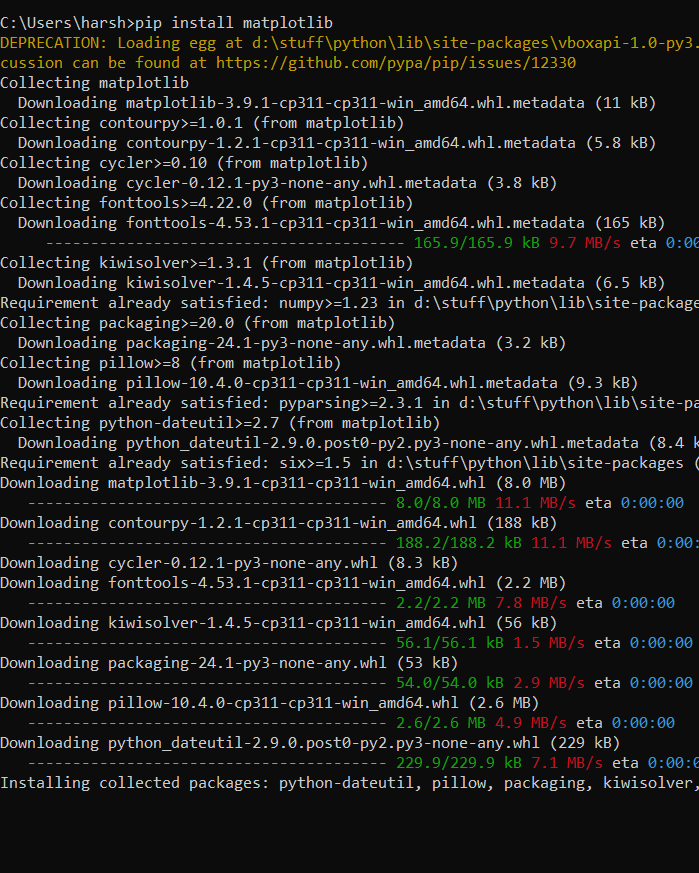
**Name –** Harshika Jadhav

**Installing opencv**



**Installing matplotlib**

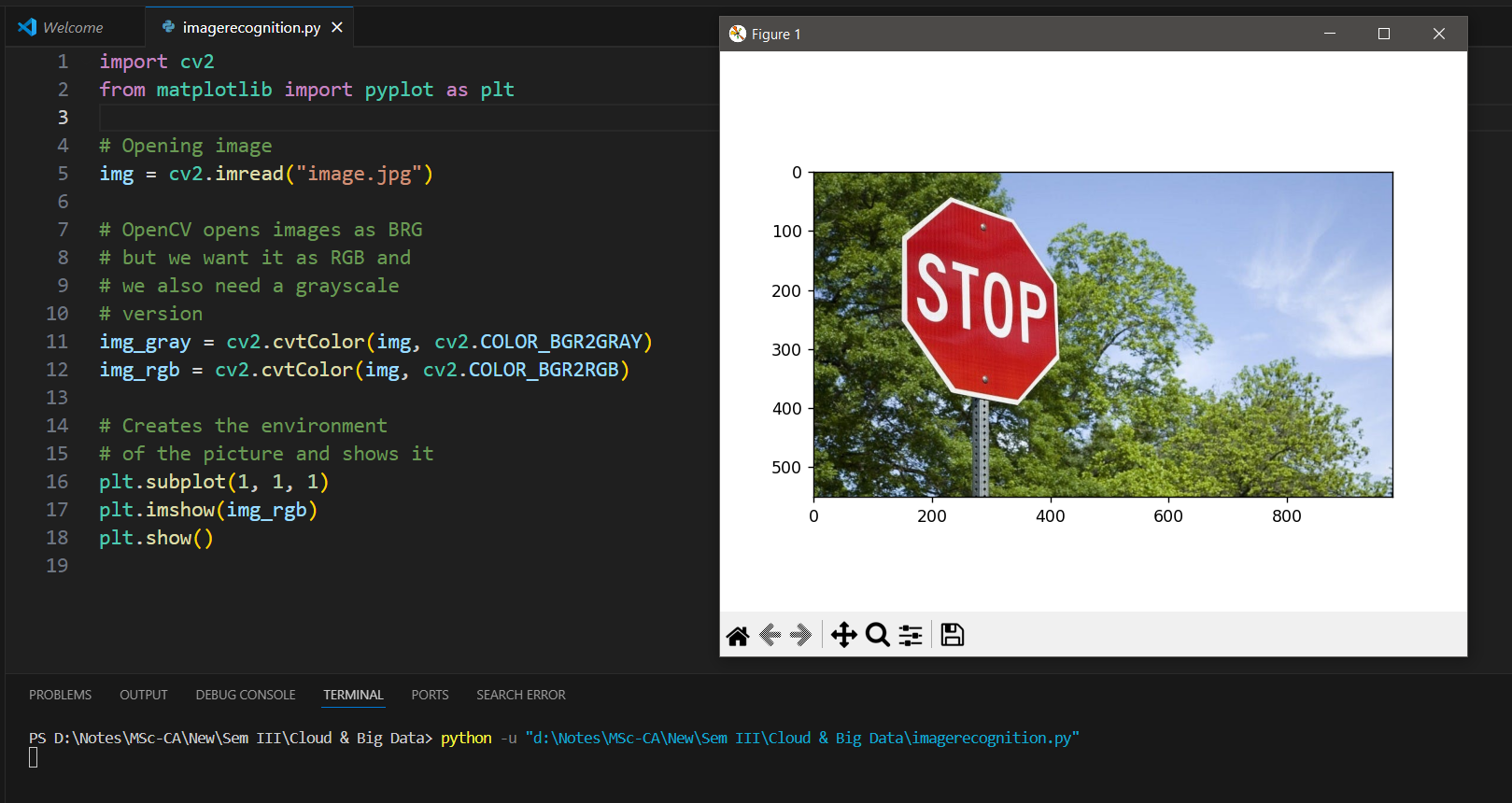


**Download img from the website** - <https://www.geeksforgeeks.org/detect-an-object-with-opencv-python/>

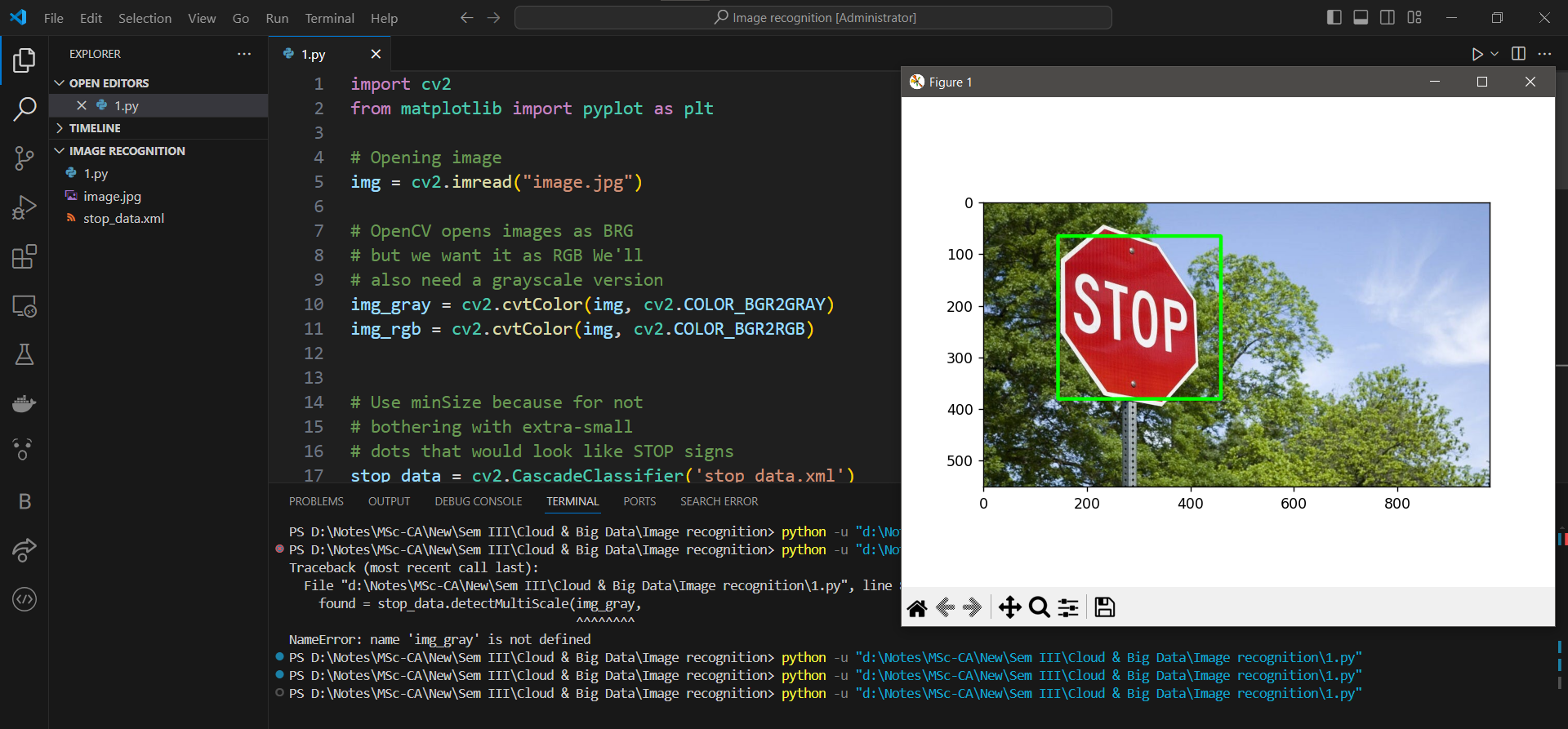
**Image:**



**Open an Image:**



**Final Output**



**Code 1:**

**import cv2**

**from matplotlib import pyplot as plt**

**# Opening image**

**img = cv2.imread("image.jpg")**

**# OpenCV opens images as BRG**

**# but we want it as RGB and**

**# we also need a grayscale**

**# version**

**img\_gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)**

**img\_rgb = cv2.cvtColor(img, cv2.COLOR\_BGR2RGB)**

**# Creates the environment**

**# of the picture and shows it**

**plt.subplot(1, 1, 1)**

**plt.imshow(img\_rgb)**

**plt.show()**

**Code 2:**

**# Use minSize because for not**

**# bothering with extra-small**

**# dots that would look like STOP signs**

**stop\_data = cv2.CascadeClassifier('stop\_data.xml')**

**found = stop\_data.detectMultiScale(img\_gray,**

**minSize =(20, 20))**

**# Don't do anything if there's**

**# no sign**

**amount\_found = len(found)**

**if amount\_found != 0:**

**# There may be more than one**

**# sign in the image**

**for (x, y, width, height) in found:**

**# We draw a green rectangle around**

**# every recognized sign**

**cv2.rectangle(img\_rgb, (x, y),**

**(x + height, y + width),**

**(0, 255, 0), 5)**