

PCV Lab-1

190030059

A.R.Snehita

Prelab

1. What is computer vision?

A. Computer vision is a process by which we can understand the images and videos how they are stored and how we can manipulate and retrieve data from them. Computer Vision is the base or mostly used for Artificial Intelligence. Computer-Vision is playing a major role in self-driving cars, robotics as well as in photo correction apps.

OpenCV is the huge open-source library for the computer vision, machine learning, and image processing areas and plays a major role in real-time operation which is critical in today's systems. By using it, one can process images and videos for various purposes. Can be integrated with various libraries, such as NumPy, Scipy etc.

2. Give some industrial applications of computer vision?

A. IBM used computer vision to create My Moments for the 2018 Masters golf tournament. IBM Watson watched hundreds of hours of Masters footage and could identify the sights (and sounds) of significant shots. it curated these key moments and delivered them to fans as personalized highlight reels.

Google Translate lets users point a smartphone camera at a sign in another language and almost immediately obtain a translation of the sign in their preferred language.

The development of self-driving vehicles relies on computer vision to make sense of the visual input from a car's cameras and other sensors. it's essential to identify other cars, traffic signs, lane markers, pedestrians, bicycles and all of the other visual information encountered on the road.

IBM is applying computer vision technology with partners like Verizon to bring intelligent AI to the edge, and to help automotive manufacturers identify quality defects before a vehicle leaves the factory.

3. Why should we deal with images in computer vision?

- A. If you were told to name certain things that you'd find in a park, you'd casually mention things like grass, bench, trees, etc. This is a very easy task that any person can accomplish in the blink of an eye. However, there is a very complicated process that takes place in the back of our minds. Human vision involves our eyes, but it also involves all of our abstract understanding of concepts and personal experiences through millions of interactions we have had with the outside world. Up until recently, computers had very limited abilities to think on their own. Computer vision is a recent branch of technology that focuses on replicating this human vision to help computers identify and process things the same way humans do.

Computer vision is one of the fields of artificial intelligence that trains and enables computers to understand the visual world. Computers can use digital images and deep learning models to accurately identify and classify objects and react to them.

4. What is Bayer colour filter array?

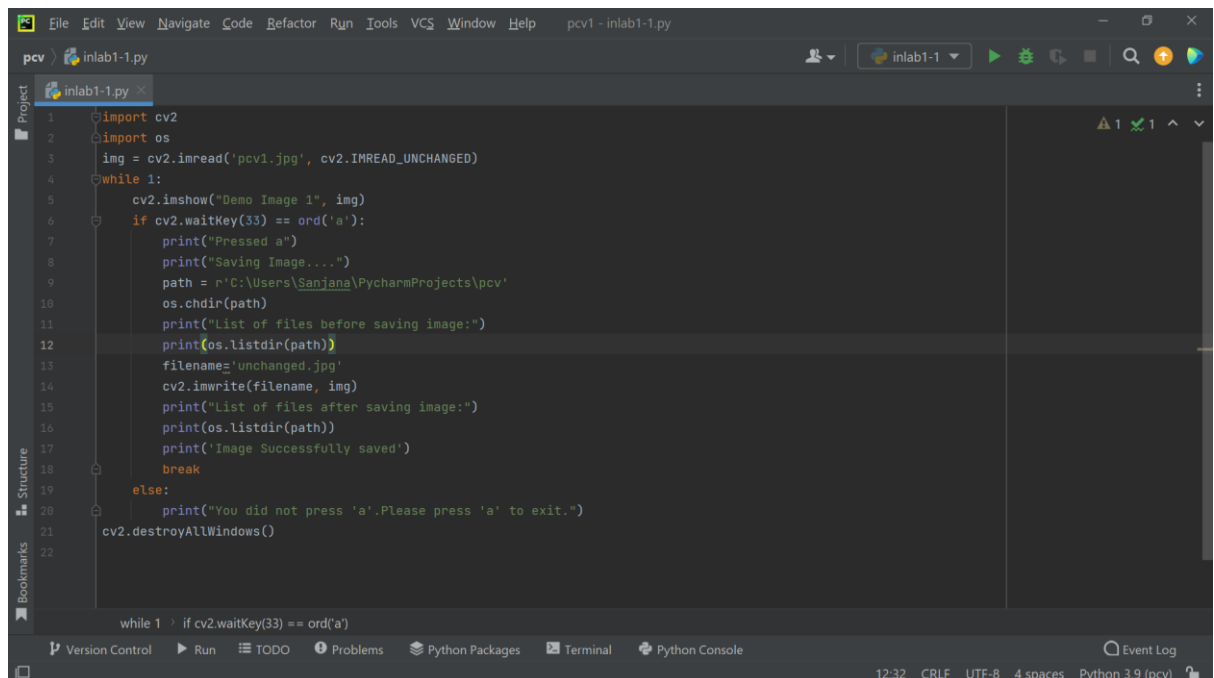
- A. A Bayer filter mosaic is a colour filter array (CFA) for arranging RGB colour filters on a square grid of photo sensors. Its arrangement of colour filters is used particularly in most single-chip digital image sensors used in digital cameras, camcorders, and scanners to create a colour image. The filter pattern is half green, one quarter red and one quarter blue, hence is also called BGGR, RGBG, GRBG, or RGGB.

Inlab

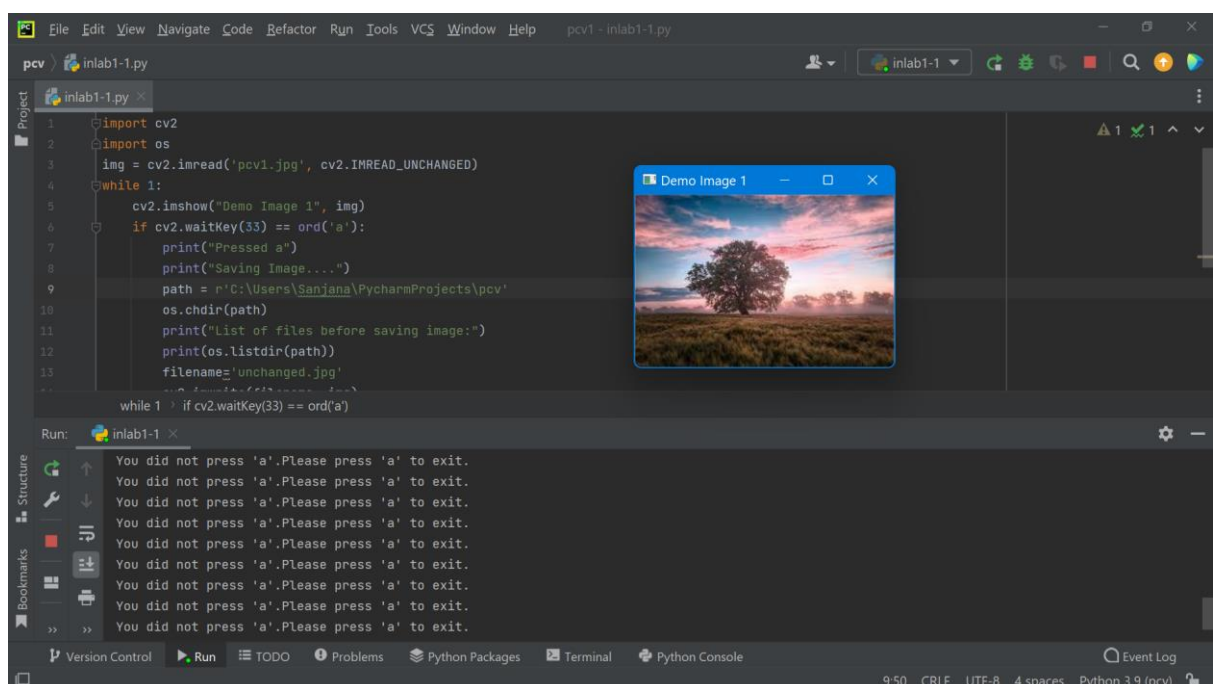
1. Perform a program for loading an image in an unchanged, colour, gray mode and display them until you press 'a' using OPENCV. Also Write a code to save the three images using OPENCV.

A.

Program for loading an image in an unchanged mode and saving it



```
1 import cv2
2 import os
3 img = cv2.imread('pcv1.jpg', cv2.IMREAD_UNCHANGED)
4 while 1:
5     cv2.imshow("Demo Image 1", img)
6     if cv2.waitKey(33) == ord('a'):
7         print("Pressed a")
8         print("Saving Image...")
9         path = r'C:\Users\Sanjana\PycharmProjects\pcv'
10        os.chdir(path)
11        print("List of files before saving image:")
12        print(os.listdir(path))
13        filename='unchanged.jpg'
14        cv2.imwrite(filename, img)
15        print("List of files after saving image:")
16        print(os.listdir(path))
17        print('Image Successfully saved')
18        break
19    else:
20        print("You did not press 'a'.Please press 'a' to exit.")
21 cv2.destroyAllWindows()
```

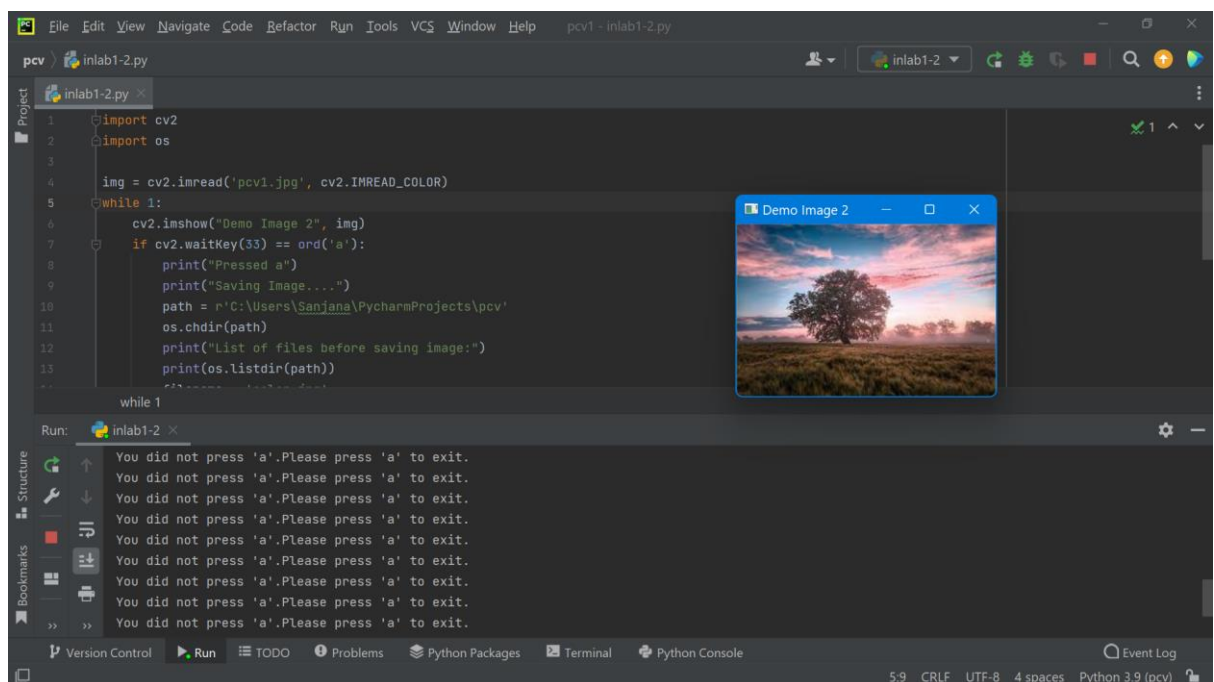
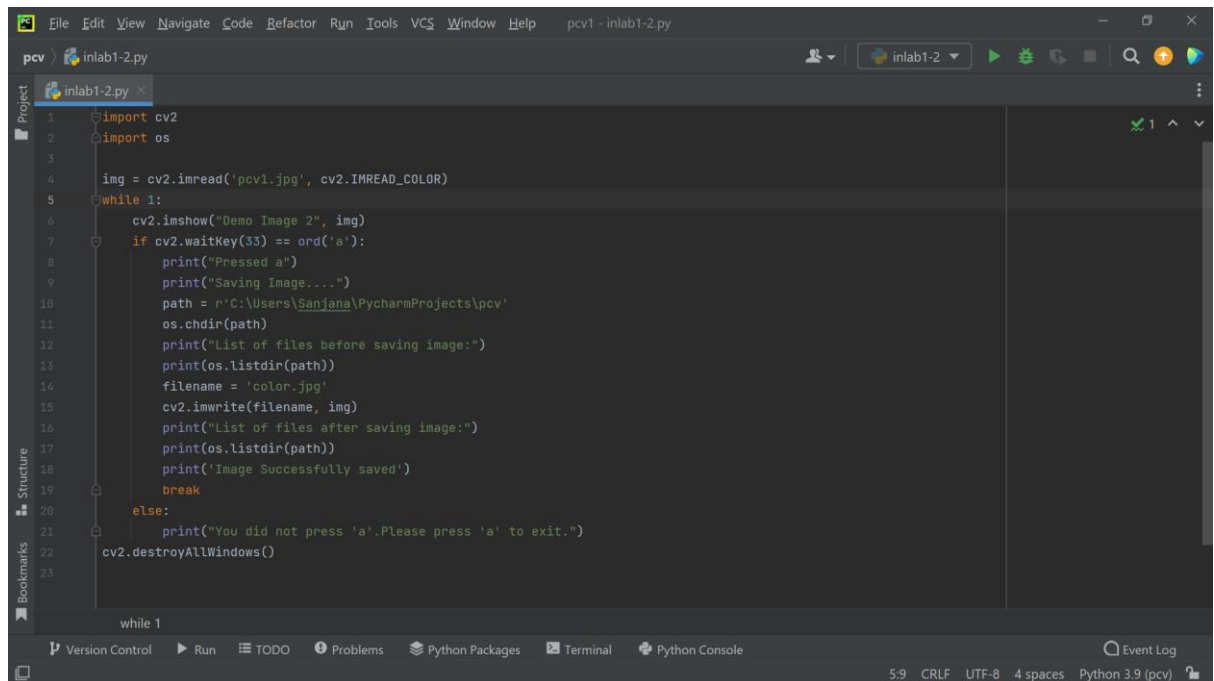


```
1 import cv2
2 import os
3 img = cv2.imread('pcv1.jpg', cv2.IMREAD_UNCHANGED)
4 while 1:
5     cv2.imshow("Demo Image 1", img)
6     if cv2.waitKey(33) == ord('a'):
7         print("Pressed a")
8         print("Saving Image...")
9         path = r'C:\Users\Sanjana\PycharmProjects\pcv'
10        os.chdir(path)
11        print("List of files before saving image:")
12        print(os.listdir(path))
13        filename='unchanged.jpg'
14        cv2.imwrite(filename, img)
15        print("List of files after saving image:")
16        print(os.listdir(path))
17        print('Image Successfully saved')
18        break
19    else:
20        print("You did not press 'a'.Please press 'a' to exit.")
21 cv2.destroyAllWindows()
```

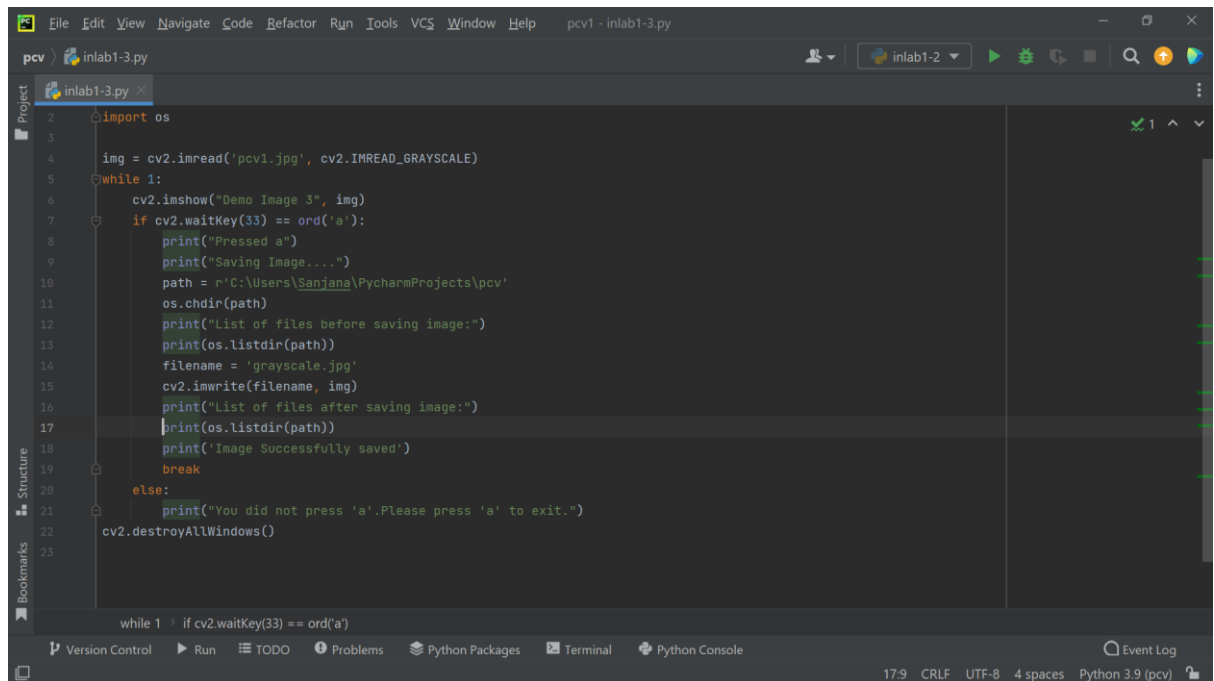
Run: inlab1-1

```
You did not press 'a'.Please press 'a' to exit.
You did not press 'a'.Please press 'a' to exit.
You did not press 'a'.Please press 'a' to exit.
You did not press 'a'.Please press 'a' to exit.
You did not press 'a'.Please press 'a' to exit.
You did not press 'a'.Please press 'a' to exit.
You did not press 'a'.Please press 'a' to exit.
You did not press 'a'.Please press 'a' to exit.
```

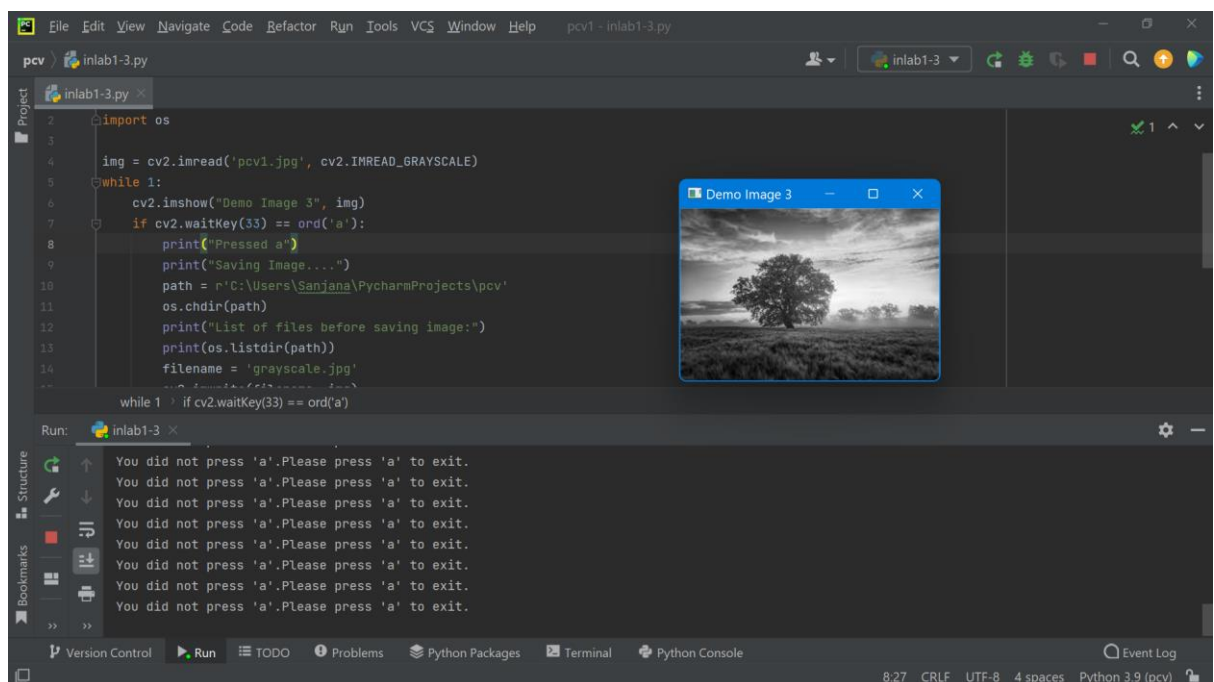
Program for loading an image in colour mode and saving it



program for loading an image in gray mode and saving it



```
1 import os
2
3
4 img = cv2.imread('pcv1.jpg', cv2.IMREAD_GRAYSCALE)
5 while 1:
6     cv2.imshow("Demo Image 3", img)
7     if cv2.waitKey(33) == ord('a'):
8         print("Pressed a")
9         print("Saving Image...")
10        path = r'C:\Users\Sanjana\PycharmProjects\pcv'
11        os.chdir(path)
12        print("List of files before saving image:")
13        print(os.listdir(path))
14        filename = 'grayscale.jpg'
15        cv2.imwrite(filename, img)
16        print("List of files after saving image:")
17        print(os.listdir(path))
18        print('Image Successfully saved')
19        break
20    else:
21        print("You did not press 'a'. Please press 'a' to exit.")
22 cv2.destroyAllWindows()
```



```
1 import os
2
3
4 img = cv2.imread('pcv1.jpg', cv2.IMREAD_GRAYSCALE)
5 while 1:
6     cv2.imshow("Demo Image 3", img)
7     if cv2.waitKey(33) == ord('a'):
8         print("Pressed a")
9         print("Saving Image...")
10        path = r'C:\Users\Sanjana\PycharmProjects\pcv'
11        os.chdir(path)
12        print("List of files before saving image:")
13        print(os.listdir(path))
14        filename = 'grayscale.jpg'
```

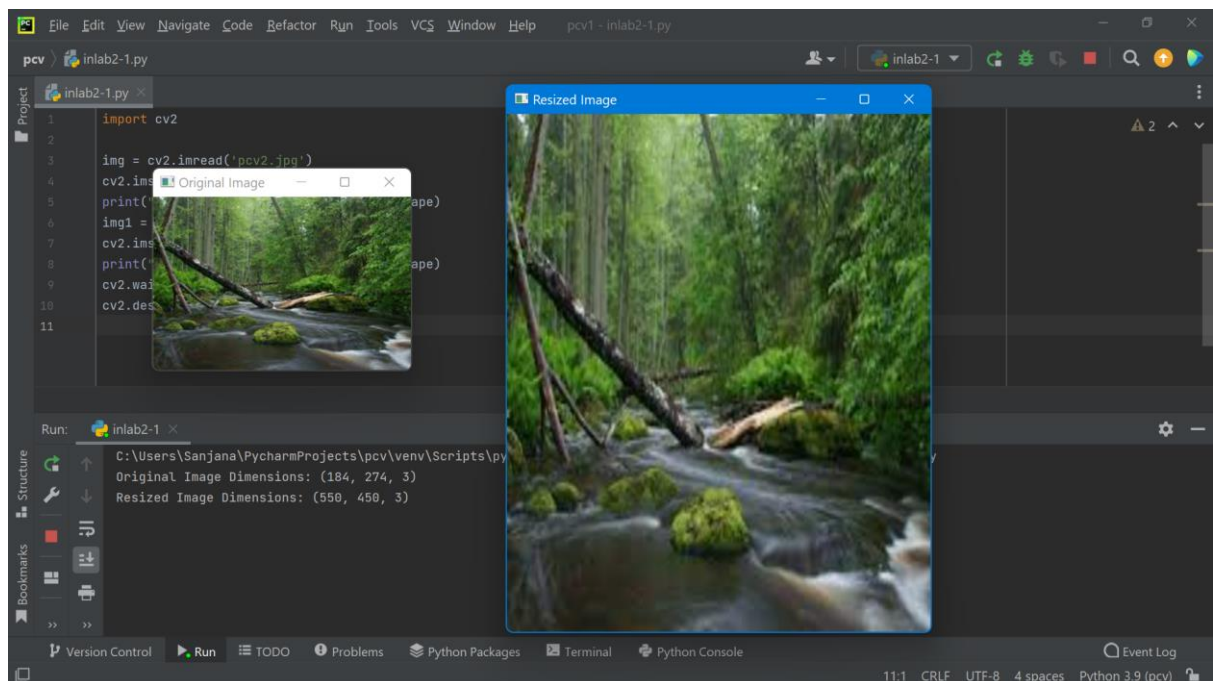
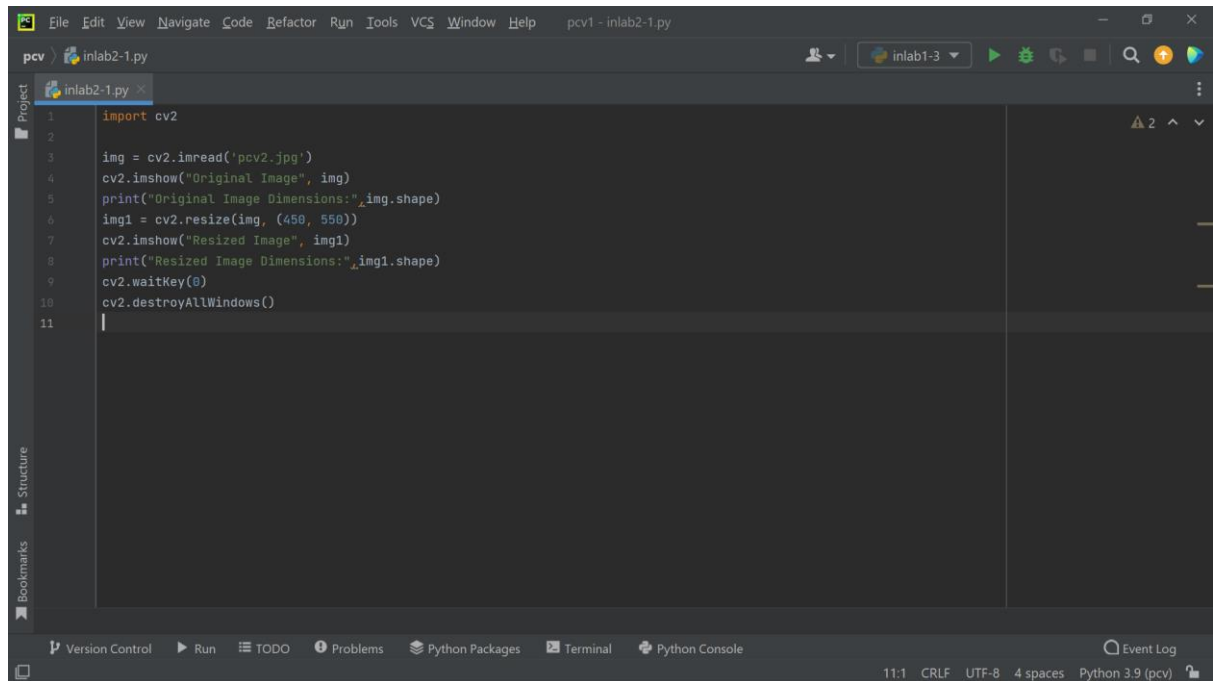
2. Harry is applying to university and to follow the admission process he needs to upload three types of certificates for which the conditions given below. So, help harry to do his task. First, select an image of a certificate and then perform the following operations:

- Resize to specific width (450) and height (550)
- Resize only height (450)

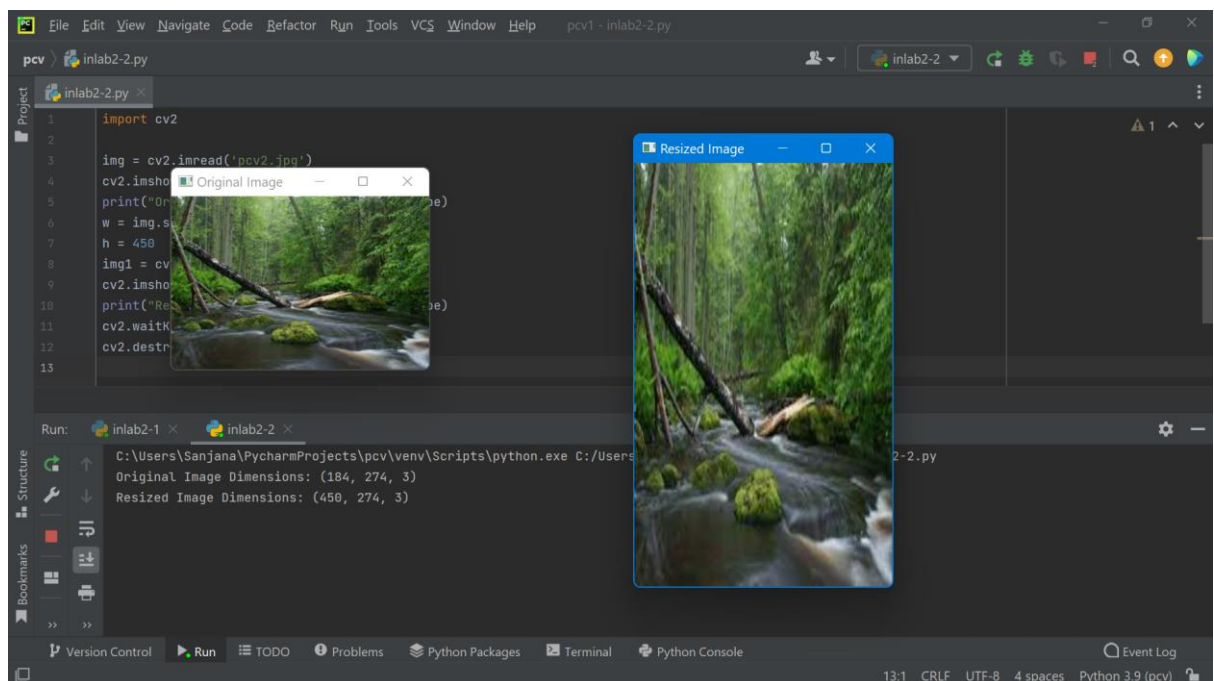
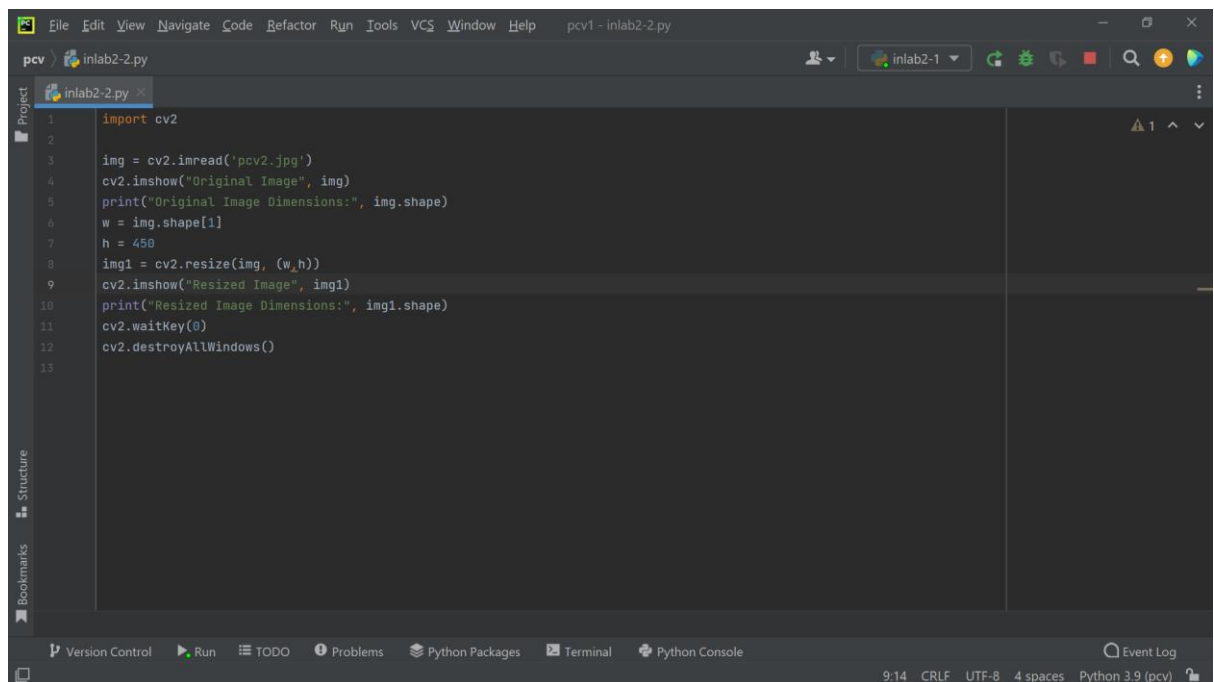
- Downscale with `resize()`

A.

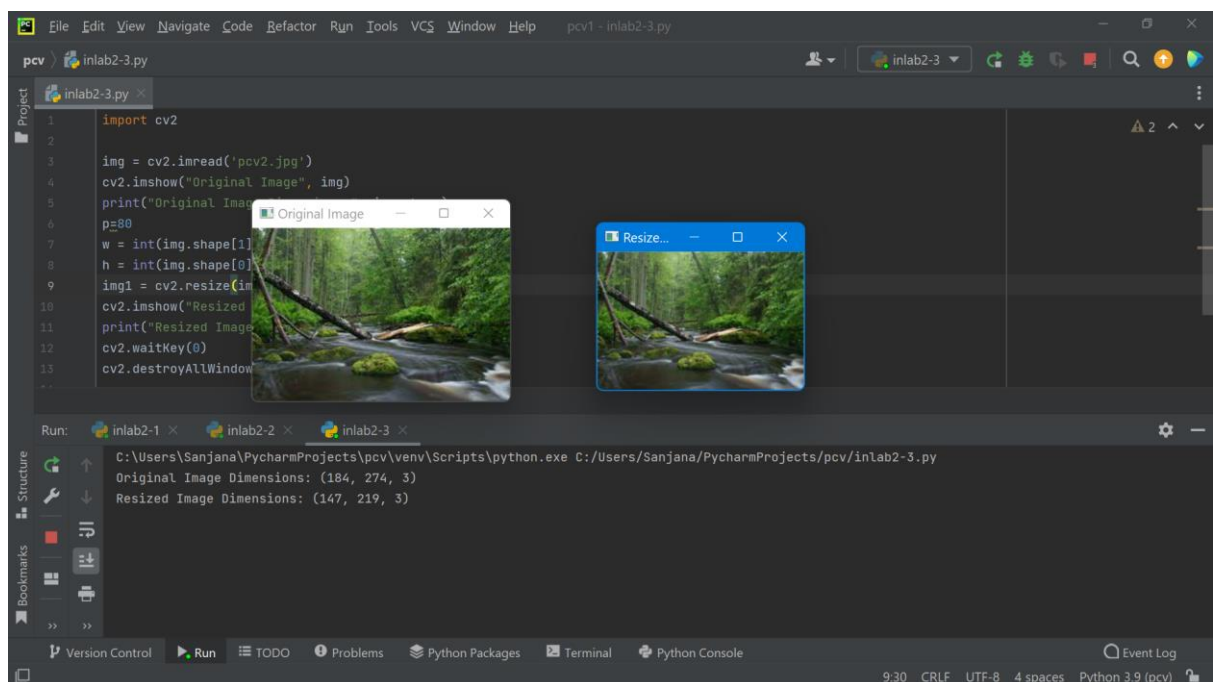
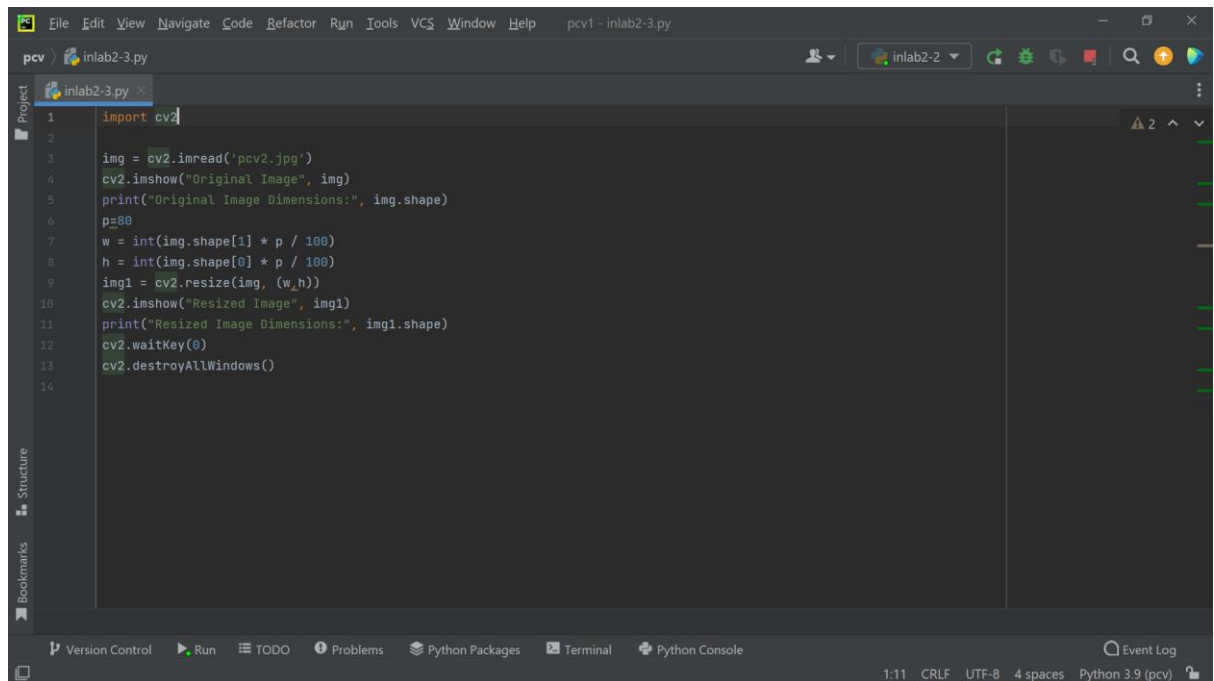
Resize to specific width (450) and height (550)



Resize only height (450)

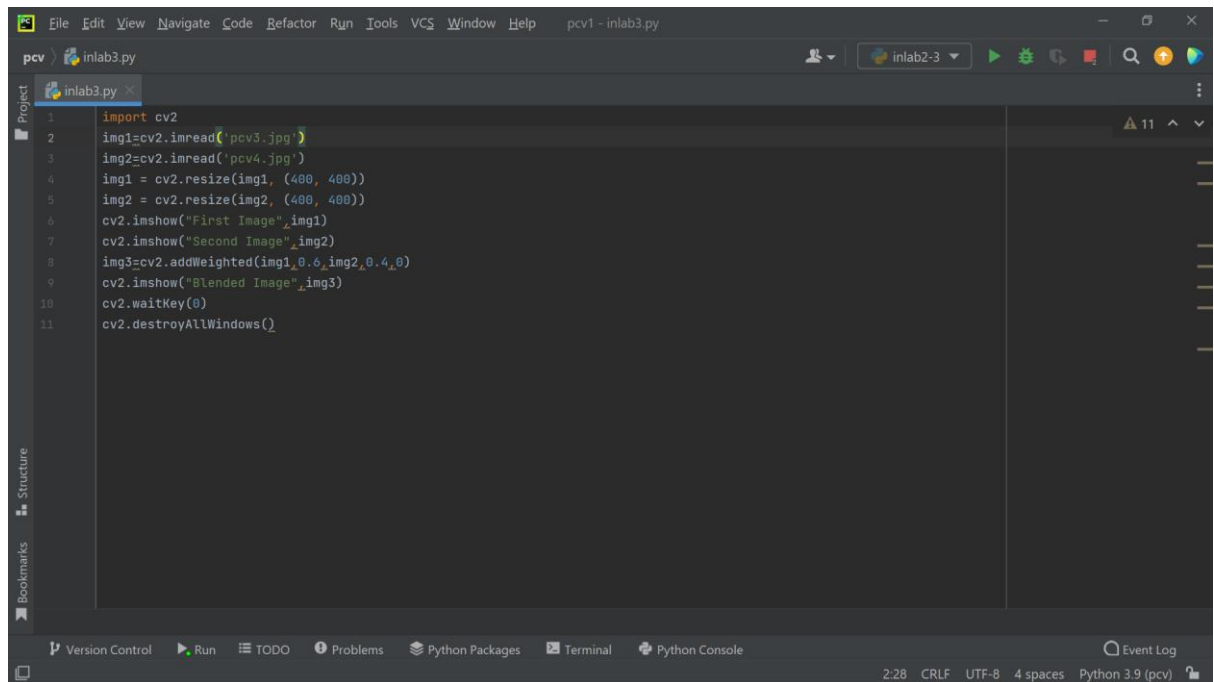


Downscale with resize ()



3. Implement a code to perform blending operation on two images in which have 60% of image1 and 40% of image2.

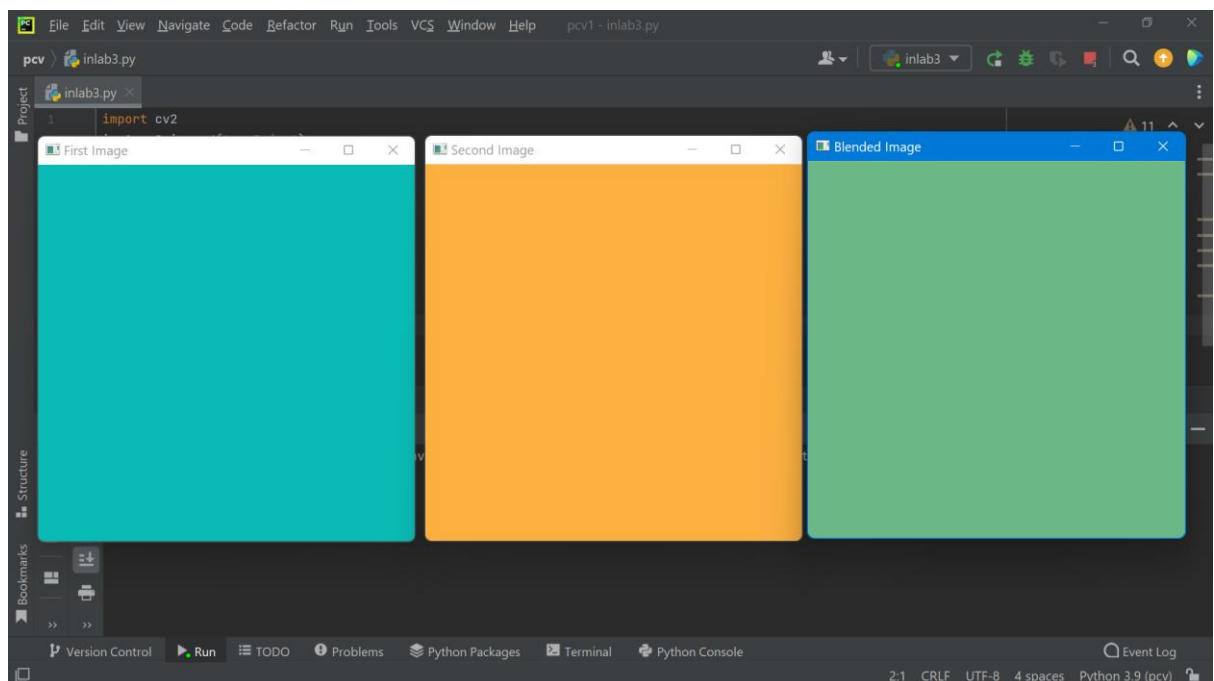
A.



The screenshot shows an IDE window titled 'pcv1 - inlab3.py'. The code in the editor is as follows:

```
1 import cv2
2 img1=cv2.imread('pcv3.jpg')
3 img2=cv2.imread('pcv4.jpg')
4 img1 = cv2.resize(img1, (400, 400))
5 img2 = cv2.resize(img2, (400, 400))
6 cv2.imshow("First Image",img1)
7 cv2.imshow("Second Image",img2)
8 img3=cv2.addWeighted(img1,0.6,img2,0.4,0)
9 cv2.imshow("Blended Image",img3)
10 cv2.waitKey(0)
11 cv2.destroyAllWindows()
```

The IDE interface includes a menu bar (File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, Help), a toolbar with icons for running and debugging, and a bottom status bar showing '2:28 CRLF UTF-8 4 spaces Python 3.9 (pcv)'.



Postlab

1. Danil wants to create an interactive Chatbot of colour filter in which it must perform the

following operations and the chatbot must ask whether the image is required to be saved

after performing the corresponding task.

[1] Hue

[2] Saturation

[3] HSV Image

[4] Value

[5] Green Channel

[6] Doubled image

A.

```
import numpy as np
import os
import cv2

def hue(img):
    hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)

    for (name, chan) in zip(("H", "S", "V"), cv2.split(hsv)):
        hu = chan
        break
    cv2.imshow('Original Image', img)
    while 1:
        cv2.imshow('Hue Image', hu)
        if cv2.waitKey(0):
            print("Pressed 0 for exiting.")

            print('Do you want to save the image?')

            s=input()
            if(s.lower()=='yes'):

                print("Saving Image...")

                path = r'C:\Users\Sanjana\PycharmProjects\pcv'
                os.chdir(path)

                print("List of files before saving image:")
                print(os.listdir(path))

                filename = 'pcv1_hue.png'
                cv2.imwrite(filename, hu)

                print("List of files after saving image:")
                print(os.listdir(path))

                print('Image Successfully saved.Exiting...')
            else:
                print('Image is not saved.Exiting!!')
                break
        else:
            print("You did not press '0'.Please press '0' to exit.")
```

```

cv2.destroyAllWindows()

return

def sat(img):
    original = img.copy()
    hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
    # loop over each of the individual channels and display them
    c=0
    for (name, chan) in zip(("H", "S", "V"), cv2.split(hsv)):
        st = chan
        c+=1
        if(c==2):
            break
    cv2.imshow("Original Image", original)
    while 1:
        cv2.imshow('Saturated Image', st)
        if cv2.waitKey(0):
            print("Pressed 0 for exiting.")

            print('Do you want to save the image?')

            s = input()
            if (s.lower() == 'yes'):

                print("Saving Image....")

                path = r'C:\Users\Sanjana\PycharmProjects\pcv'
                os.chdir(path)

                print("List of files before saving image:")
                print(os.listdir(path))

                filename = 'pcv1_sat.png'
                cv2.imwrite(filename, st)

                print("List of files after saving image:")
                print(os.listdir(path))

                print('Image Successfully saved.Exiting....')
            else:
                print('Image is not saved.Exiting!!')
                break
        else:
            print("You did not press '0'.Please press '0' to exit.")
    cv2.destroyAllWindows()
    return

def hsv(img):
    hsvImage = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
    cv2.imshow("Original Image", img)
    while 1:
        cv2.imshow('HSV Image', hsvImage)
        if cv2.waitKey(0):
            print("Pressed 0 for exiting.")

            print('Do you want to save the image?')

            s = input()

```

```

        if (s.lower() == 'yes'):

            print("Saving Image...")

            path = r'C:\Users\Sanjana\PycharmProjects\pcv'
            os.chdir(path)

            print("List of files before saving image:")
            print(os.listdir(path))

            filename = 'pcv1_hsv.png'
            cv2.imwrite(filename, hsvImage)

            print("List of files after saving image:")
            print(os.listdir(path))

            print('Image Successfully saved.Exiting...')
        else:
            print('Image is not saved.Exiting!!')
            break
    else:
        print("You did not press '0'.Please press '0' to exit.")
cv2.destroyAllWindows()
return

def val(img):
    hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
    # loop over each of the individual channels and display them
    for (name, chan) in zip(("H", "S", "V"), cv2.split(hsv)):
        vl=chan
    cv2.imshow('Original Image', img)
    while 1:
        cv2.imshow('Value Image', vl)
        if cv2.waitKey(0):
            print("Pressed 0 for exiting.")

            print('Do you want to save the image?')

            s = input()
            if (s.lower() == 'yes'):

                print("Saving Image...")

                path = r'C:\Users\Sanjana\PycharmProjects\pcv'
                os.chdir(path)

                print("List of files before saving image:")
                print(os.listdir(path))

                filename = 'pcv1_val.png'
                cv2.imwrite(filename, vl)

                print("List of files after saving image:")
                print(os.listdir(path))

                print('Image Successfully saved.Exiting...')
            else:
                print('Image is not saved.Exiting!!')
                break
    else:

```

```

        print("You did not press '0'.Please press '0' to exit.")
cv2.destroyAllWindows()
return

def grn(img):
    green_channel = img[:, :, 1]
    green_img = np.zeros(img.shape)
    green_img[:, :, 1] = green_channel
    cv2.imshow('Original Image',img)
    while 1:
        cv2.imshow('Green Image', green_img)
        if cv2.waitKey(0):
            print("Pressed 0 for exiting.")

            print('Do you want to save the image?')

            s = input()
            if (s.lower() == 'yes'):

                print("Saving Image....")

                path = r'C:\Users\Sanjana\PycharmProjects\pcv'
                os.chdir(path)

                print("List of files before saving image:")
                print(os.listdir(path))

                filename = 'pcv1_grn.png'
                cv2.imwrite(filename, green_img)

                print("List of files after saving image:")
                print(os.listdir(path))

                print('Image Successfully saved.Exiting....')
            else:
                print('Image is not saved.Exiting!!')
                break
        else:
            print("You did not press '0'.Please press '0' to exit.")
cv2.destroyAllWindows()
return

def dbl(img):
    out = cv2.normalize(img.astype('float'), None, 0.0, 1.0,
cv2.NORM_MINMAX)
    cv2.imshow('Original Image', img)
    while 1:
        cv2.imshow('Doubled Image', out)
        if cv2.waitKey(0):
            print("Pressed 0 for exiting.")

            print('Do you want to save the image?')

            s = input()
            if (s.lower() == 'yes'):

                print("Saving Image....")

                path = r'C:\Users\Sanjana\PycharmProjects\pcv'

```

```

        os.chdir(path)

        print("List of files before saving image:")
        print(os.listdir(path))

        filename = 'pcv1_dbl.png'
        cv2.imwrite(filename, out)

        print("List of files after saving image:")
        print(os.listdir(path))

        print('Image Successfully saved.Exiting....')
    else:
        print('Image is not saved.Exiting!!')
        break
    else:
        print("You did not press '0'.Please press '0' to exit.")
cv2.destroyAllWindows()
return

img = cv2.imread('pcv1.jpg', cv2.IMREAD_UNCHANGED)

while True:
    print('Welcome to Opencv chatbot for color filter')
    print('Do you want to perform an operation on the image?')
    s = input()
    if s.lower() == 'no' or s == '0':
        print('Thank you for coming here. Have a good day!!')
        break
    print('You can choose any of the operations to perform on the given image ')
    print('-----')
    print('1.Hue')
    print('2.Saturation')
    print('3.HSV Image')
    print('4.Value')
    print('5.Green Channel')
    print('6.Doubled Image')
    print('-----')
    print('Enter a number:')
    n = int(input())
    if n == 1:
        print('Hue')
        hue(img)
    elif n == 2:
        print('Saturation')
        sat(img)
    elif n == 3:
        print('HSV Image')
        hsv(img)
    elif n == 4:
        print('Value')
        val(img)
    elif n == 5:
        print('Green Channel')
        grn(img)
    elif n == 6:
        print('Doubled Image')
        dbl(img)

```

```
else:  
    print('Please Enter a Valid Number!!!')
```

