MAJOR PROJECT 2:

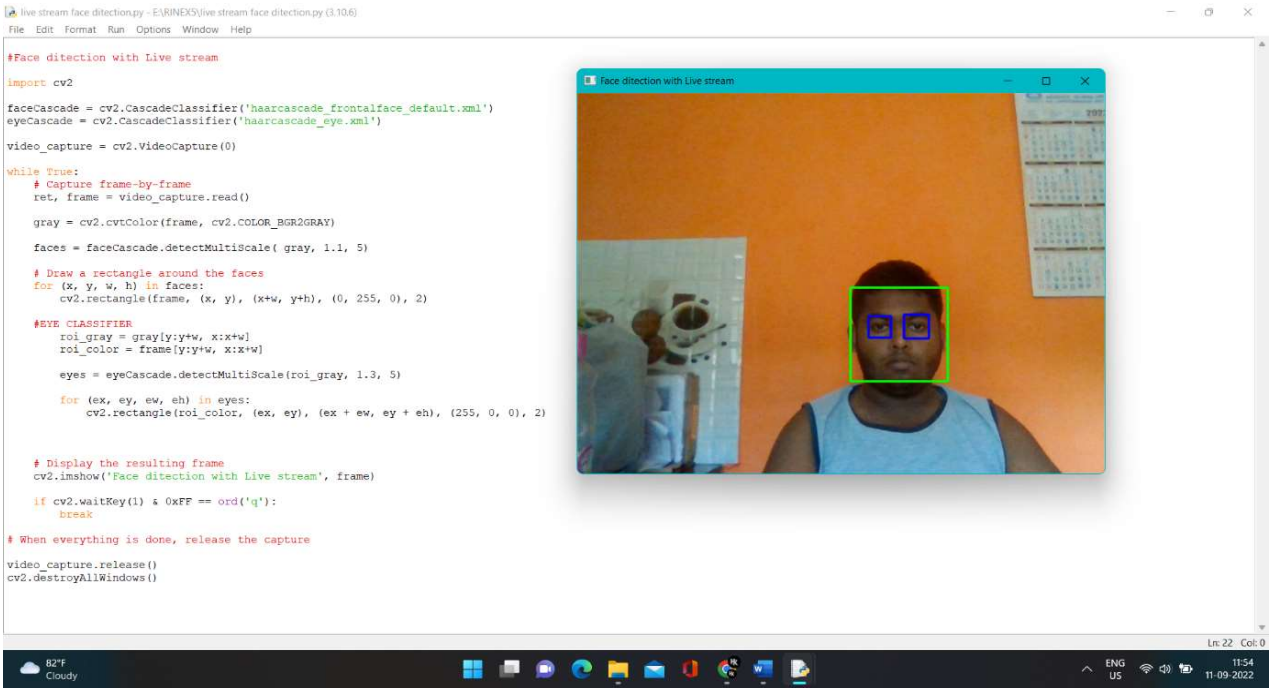
#Face with eye ditection with Live stream

```
import cv2

faceCascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
eyeCascade = cv2.CascadeClassifier('haarcascade_eye.xml')
video_capture = cv2.VideoCapture(0)

while True:
    # Capture frame-by-frame
    ret, frame = video_capture.read()

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

    faces = faceCascade.detectMultiScale( gray, 1.1, 5)

    # Draw a rectangle around the faces
    for (x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)

    #EYE CLASSIFIER
        roi_gray = gray[y:y+w, x:x+w]
        roi_color = frame[y:y+w, x:x+w]

        eyes = eyeCascade.detectMultiScale(roi_gray, 1.3, 5)

        for (ex, ey, ew, eh) in eyes:
            cv2.rectangle(roi_color, (ex, ey), (ex + ew, ey + eh), (255, 0, 0), 2)



    # Display the resulting frame
    cv2.imshow('Face ditection with Live stream', frame)

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

```
    break

# When everything is done, release the capture
video_capture.release()
cv2.destroyAllWindows()
```

## OUTPUT of the above code:

### Close-up view from the web cam:




### Distant view from the web cam: