This code is used for playing alarm only……The kar.98.wav is the audio file

```
def sound_alarm():
    pygame.mixer.init()
    # path to music file
    music_file_path = "C:\\Users\\Shreyas\\Desktop\\Extra\\Sir WOrk\\dowseness
Detechction\\Kar98.wav"
    pygame.mixer.music.load(music_file_path)
    pygame.mixer.music.play()
```

For detecting Eyes

```
def eye_aspect_ratio(eye):
    A = dist.euclidean(eye[1], eye[5])
    B = dist.euclidean(eye[2], eye[4])  # vertical distance
    C = dist.euclidean(eye[0], eye[3])  # horizontal distance

    ear = (A + B) / (2.0 * C)
    return ear
```

To detect left and right eyes

```
EYE_AR_THRESH = 0.3
EYE_AR_CONSEC_FRAMES = 40

COUNTER = 0
ALARM_ON = False

predictor_path = 'shape_predictor_68_face_landmarks.dat'  # path to dat file
detector = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor(predictor_path)


# Grab the indexes of the facial landmarks for the left and
# right eye,respectively
(lStart, lEnd) = face_utils.FACIAL_LANDMARKS_IDXS["left_eye"]
(rStart, rEnd) = face_utils.FACIAL_LANDMARKS_IDXS["right_eye"]
```

It will start camera from laptop, take frame on face that is will detect the face depending on nose, ear, lips.
After that it will focus on only eyes and separate left and right eyes and will see if any eyes is closed. If eyes closed then will play alarm.

“q” key press to quit from code.

```
cap = cv2.VideoCapture(0)
while True:
   ret, frame = cap.read()
   if ret == False:
      print('Failed to capture frame from camera. Check camera index in cv2.VideoCapture(0) \n')
      break
      # cv2.imshow(frame)

   frame = imutils.resize(frame, width=450)
   gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
   rects = detector(gray, 0)

   for rect in rects:
      # determine the facial landmarks for            #face region
      shape = predictor(gray, rect)
      shape = face_utils.shape_to_np(shape)  # converting to numpy array

      leftEye = shape[lStart:lEnd]
      rightEye = shape[rStart:rEnd]

      leftEAR = eye_aspect_ratio(leftEye)
      rightEAR = eye_aspect_ratio(rightEye)

      ear = (leftEAR + rightEAR) / 2.0

      leftEyeHull = cv2.convexHull(leftEye)
      rightEyeHull = cv2.convexHull(rightEye)
      cv2.drawContours(frame, [leftEyeHull], -1, (0, 255, 0), 1)
      cv2.drawContours(frame, [rightEyeHull], -1, (0, 255, 0), 1)

      if ear < EYE_AR_THRESH:
         COUNTER += 1

         if COUNTER >= EYE_AR_CONSEC_FRAMES:
            if not ALARM_ON:
               ALARM_ON = True
               d = threading.Thread(target=sound_alarm)
               d.setDaemon(True)
               d.start()

            cv2.putText(frame, "DROWSINESS ALERT!", (10, 30),
                        cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

      else:
         COUNTER = 0
         ALARM_ON = False

      cv2.putText(frame, "EAR: {:.2f}".format(
         ear), (300, 30), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
```

```
    cv2.imshow("Frame", frame)
    key = cv2.waitKey(1) & 0xFF

    if key == ord("q"):  # Press 'q' to stop
        break

cv2.destroyAllWindows()
cap.release()
```