LAB 8

As shown in below code , I had read the mat file using h5py library,then converted it to proper numpy array format of 400*10304.Using np.mean , row wise mean was calculated and then subtracted from the array.Covariance matrix was calculated using np.cov, after that eigenvectors and eigenvalues were been calculated.As cov matrices are symmetric which deduces that eigenvectors are mutually orthogonal to eachother.Hence matrix dot would be such that dot[i][j] would store dot product of eigenvector[i] and eigenvector[j].Hence, dot would be an identity matrix.

In [31]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import scipy.io
import h5py
#question 1
face_images = {}
with h5py.File('faceimages.mat', 'r') as f: #reading mat file using h5py librar
y
    data = f.keys()
    for i, j in f.items():
        face_images[i] = np.array(j)
face_array = face_images['data'][:-1]# facearray will be 10304*400 matrix
face_array=np.transpose(face_array) # transpose will give 400*10304 indicating
400 images (rows) having 10304 pixels(cols)
#question 2
mean = np.mean(face_array, axis = 1) # finding mean row wise
mean=np.transpose([mean] * 10304) # expanding the mean by repeating 10304 times
to make its dimension identical to face_array
face_array = face_array - mean #mean subtract
face_cov = np.cov(face_array) # calculating covariance
#question 3
eigenValues, eigenVectors = np.linalg.eig(face_cov) #calculating eigenvectors a
nd eigenvalues
dot=eigenVectors.T@eigenVectors # dot[i][j] will store dot product of ith eigen
vector with jth eigenvector
dot[np.where(abs(dot)<=0.00001)]=0# as python will not give exact 0 so any elem
ent with absolute value less than 0.00001 will approx to 0
print(dot)#dot will be identity matrix showing all the eigenvectors are orthogo
nal to easch other
```

```
[[1. 0. 0. ... 0. 0. 0.]
 [0. 1. 0. ... 0. 0. 0.]
 [0. 0. 1. ... 0. 0. 0.]
 ...
 [0. 0. 0. ... 1. 0. 0.]
 [0. 0. 0. ... 0. 1. 0.]
 [0. 0. 0. ... 0. 0. 1.]]
```