## Learnings from the assignment

From file p_norm.py:
Writing the code for this file helped me refresh some basic Python concepts.

From file row_manipulation.py:
Writing the code helpmed me refresh Numpy concepts. In this file, there is a 2D array on which some straightforward manipulations are performed. These operations were cropping the array, creating a 0.5 padded version of the cropped array and concatenating together two copies of this padded array.

From file shape_manipulation.py:
This part was easy to solve after refreshing Numpy concepts. The output after running the program is consistent with the sample run given in the problem statement.

From file pca.py:
This part helped me refresh the concept of PCA and Matplotlib library.

From file image_conversion.py:
Learned how images are read using Matplotlib and OpenCV library. Matplotlib uses the ordering RGB to store the image and OpenCV uses BGR ordering for storing the image. The question also illustrated that when an image is read by the 'cv2.imread()' method, it gets stored by default in the 'uint8' integer format with a pixel in the range of 0 to 255. Even when the pixels in these images are normalized to float values in the 0 to 1 range, both matplotlib and opencv continue to display the images correctly using their respective 'imshow()' methods.

From file display_images.py:
Learned how to use the key press events in OpenCV. Pressing 'n' key would display the next image and pressing 'p' key would display the previous image in the sequence of images in the directory with circular wrap around. Pressing the "ESC" key would close the OpenCV window.

From file display_video.py:
In this task, I learned how to read videos in OpenCV, create a bounding box and display text in a bounding box. The grayscale version of the video is displayed simultaneously to the right of the original video. Pressing "q" key will complete the execution of the program.

**Sample Run**

All the screenshots of running of the scripts are stored in "convincingDirectory/" directory which would give the idea of sample run.

**Can this be a blog post?**

No, it can't be a blog post. In this PDF, I have given a very high level overview of the things which I have learned. Someone who has previous knowledge of Matplotlib, OpenCV will be able to understand this PDF. For someone who has no idea about Matplotlib and OpenCV, it would be hard for them to understand this.