#### Step1: Download and Install Anaconda

First, go to the anaconda website. Then go to Products, Individual Edition,  and here you can see download bitton for python 3.8, although we would be showcasing notebooks running python 3.9, python 3.8 is also fine. Go ahead and click download and wait for it to complete downloading. This would then download as an .exe executable. You can open and run this .exe to start the process.

You can follow the installer steps. In this, we come to a tricky situation if this is a clean installation and there are no existing python versions installed on your system, then you can select the **Add Anaconda3 to my PATH environment variable**. But if you have another distribution of python installed, then it would override the python path with the anaconda path if you select the first option. That means if you type **python** on the command prompt, it would invoke the anaconda python and not the default python. So if you don't have a previous install or if you don't mind not using your previous install, click the above-mentioned box and go ahead and install.

Next, we must check that python is installed correctly. Open up the command prompt and type:

        conda

Most notably, we do not get an error message which means that Anaconda was installed properly.

#### Step2: Create Virtual Environment

Now, let's see the flags that we can pass with conda. The main one that we would use is **create**, which would create a new virtual environment.

        conda create --name opencv-env

This would show where it would create the virtual environment and ask for confirmation. Press y to continue.

Now our virtual environment has been created.

We'll activate the environment

        conda activate opencv-env

This command will activate our virtual environment as we can see our virtual environment name as a prefix on the command line prompt.

        (opencv-env)

This tells us that everything that we run and install, takes effect in our virtual environment.

#### Step3: Install Packages

We will use pip package manager to install all packages. You can install any package by simply typing: "pip install" followed by the name of the package. You can install multiple packages simultaneously by typing on the same line.

The key package to install is:

        pip install opencv-contrib-python streamlit moviepy jupyter matplotlib ipykernel

This would install all the mentioned packages.  If you run into any errors, we would recommend following up by installing one at a time and also upgrading pip itself.

Now let's install a couple of packages that we would use in specific notebooks later in this course. If you have any issue installing these packages, you can use google colab when you get to those notebooks.

        pip install pyautogui mediapipe mime

Now lets open python interpreter and import OpenCV

        python

        >>> import cv2

        >>> cv2.\_\_version\_\_

We can see that this does not give an error and we have successfully installed OpenCV.

Now exit the interpreter.

        >>> exit()

#### Step4: Validate Jupyter

We can start up jupyter using the command:

        jupyter notebook

This would then open the Jupyter hub in your default web browser. We can navigate to '**new**' and select python3.

This would open a jupyter notebook and we can test it out by importing OpenCV.

        In[ ]: import cv2  
               cv2.\_\_version\_\_

Executing this cell would give us the version of OpenCV installed in our virtual environment. This shows that jupyter has been installed correctly and our environment has been setup correctly so we can load in the python packages.

Let's shut down our jupyter notebook in the command line by hitting **Ctrl+C**interrupt key. As a final step, we would deactivate the virtual environment:

        deactivate

 Note that jupyter will only use the virtual environment if it is activated before launching jupyter, therefore, make sure to do that each time you launch jupyter. At this point, we are all setup and ready to go with our python OpenCV environment.