Chapter 2: Anaconda & Pip

Objectives

To set up anaconda environment

- Download and install anaconda
- Management of package and environment with Pip

Introduction

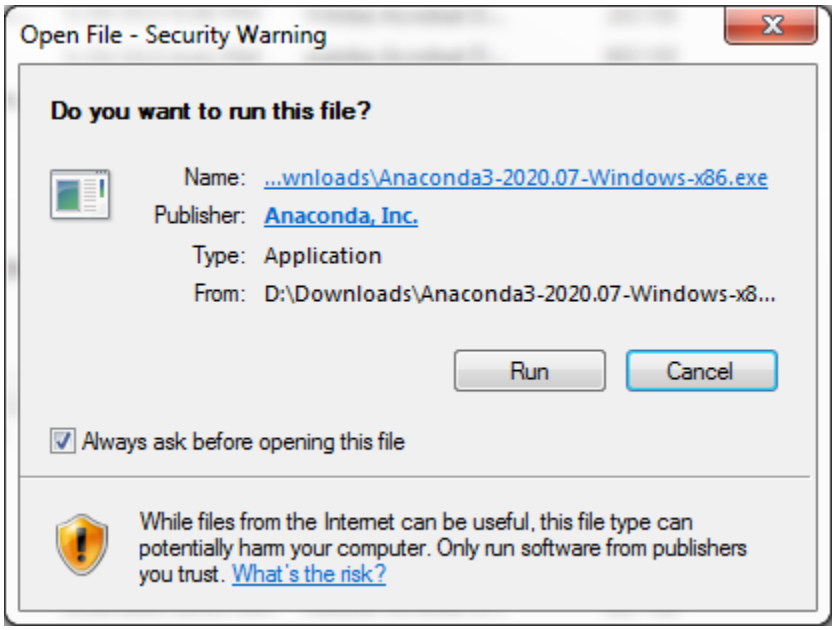
When starting to learn Python, it is useful to install Python’s latest version along with the packages needed for performing different tasks. But once the needs of the project become more versatile, structured, or complex, there is a requirement for a virtual environment for organizing different codebases. The virtual environment is simply a networked application that helps the user in interacting with other users' work and computing environment simultaneously. A virtual environment enables control of as many packages as needed. Therefore, a more reproducible, portable, and stable environment is available for coding. Some of the popular packages are

Virtualenv, Anaconda (popularly known as Conda), and Pipenv. Among them, Anaconda is the most popular open-source virtual environment which consists of Spyder, Jupyter, or other notebooks which are required for scientific computing, data analytics, or large data processing. This chapter examines the Conda installation process along with discussing the packages used for environmental management to ensure the availability of a base for working with complex data.

Installation of Anaconda

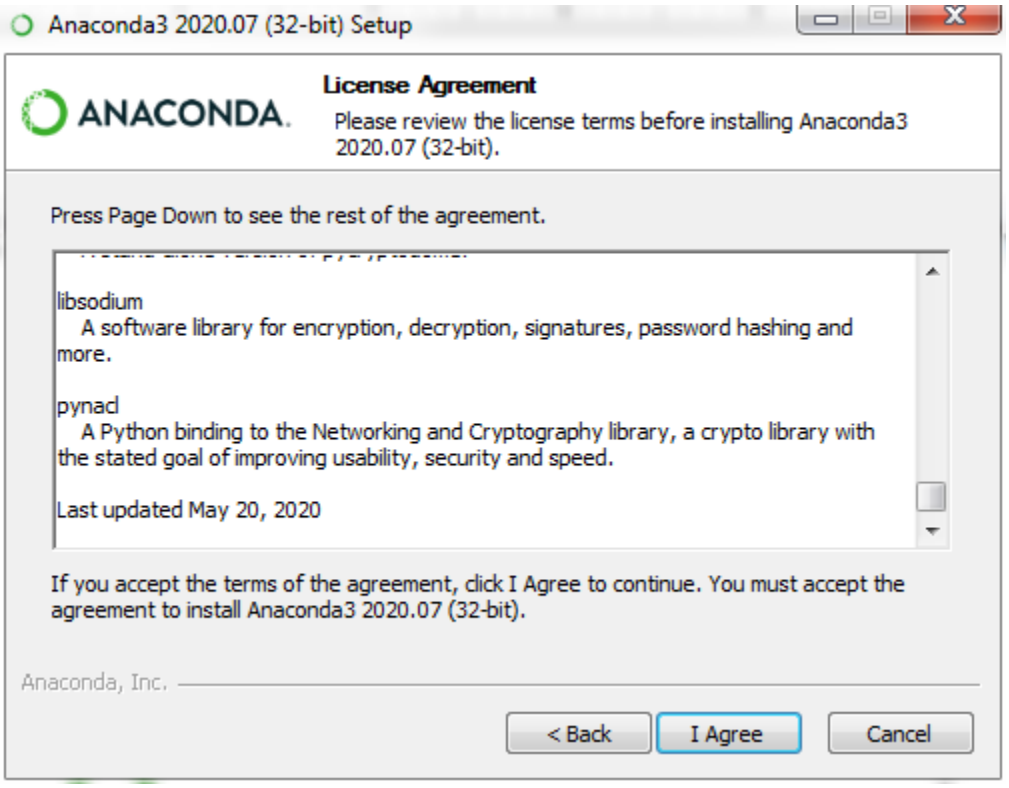
To start working with Python, first install Anaconda's latest version. For this, the requirement is to ensure that the 3.7 version of Python is at least installed. The steps for installing the Anaconda software are as follows.

- Download Anaconda installer for Windows or Mac as per the requirement
- In the 'Downloads' folder, double-click on the downloaded exe file for Anaconda and click on it to launch

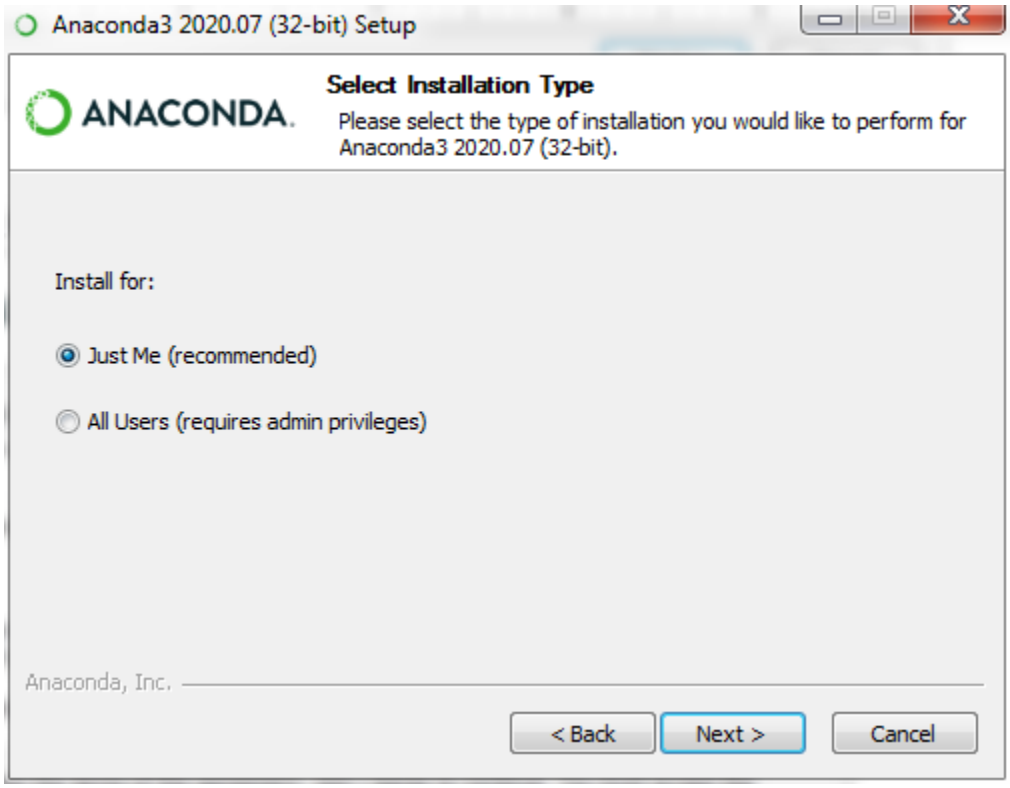


- The 'Setup wizard' window will appear. Click on 'Next'

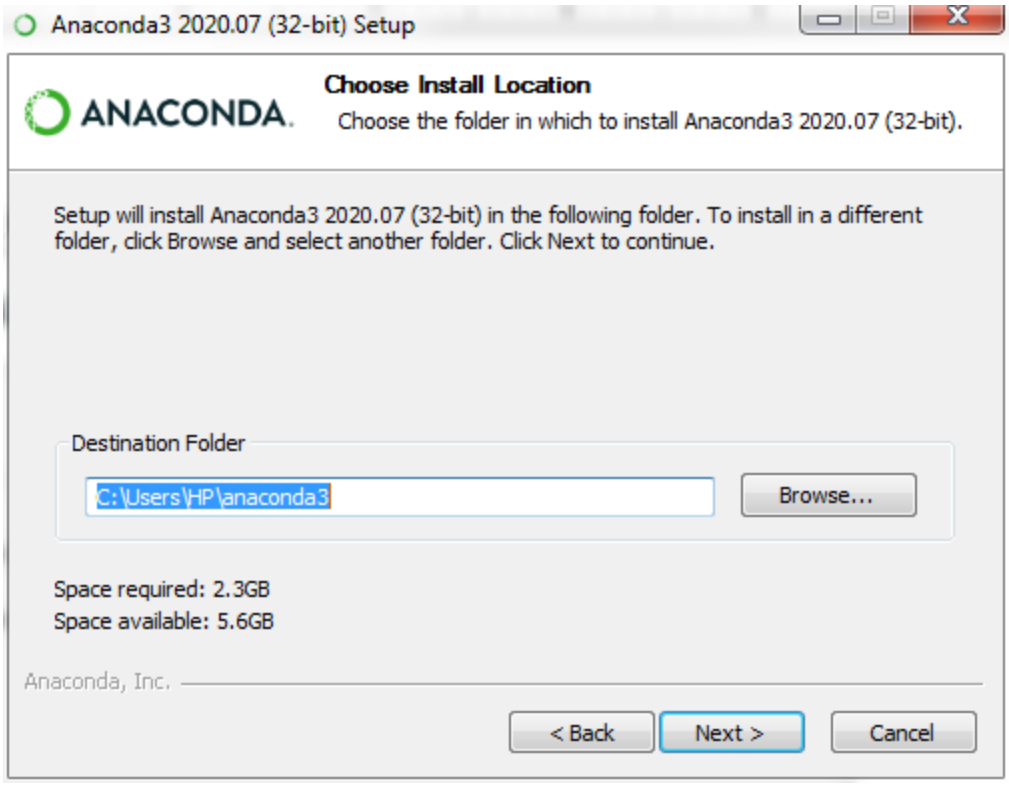
- The 'License agreement' window will appear. Read all the licensing agreement terms and then click on 'I agree'.



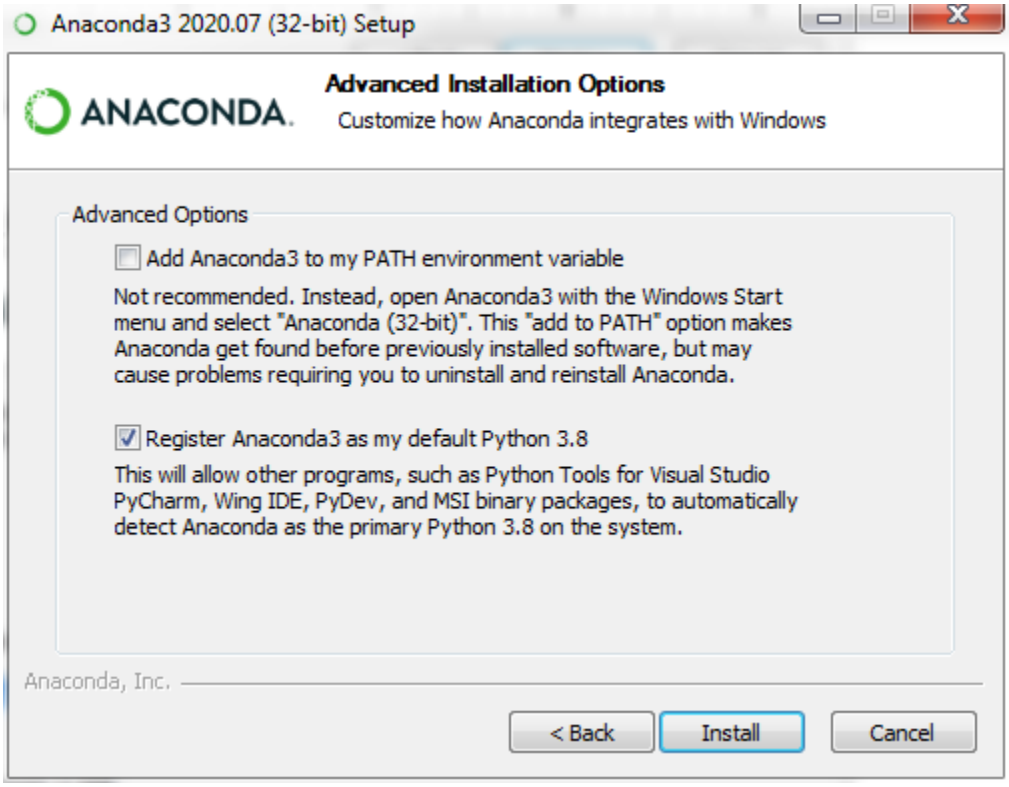
- Next, select the installation type you need i.e. for only yourself or for others too. It's recommended to install it for yourself. Select all users if you want to install it for all users on the computer. After this selection click on ‘Next’.



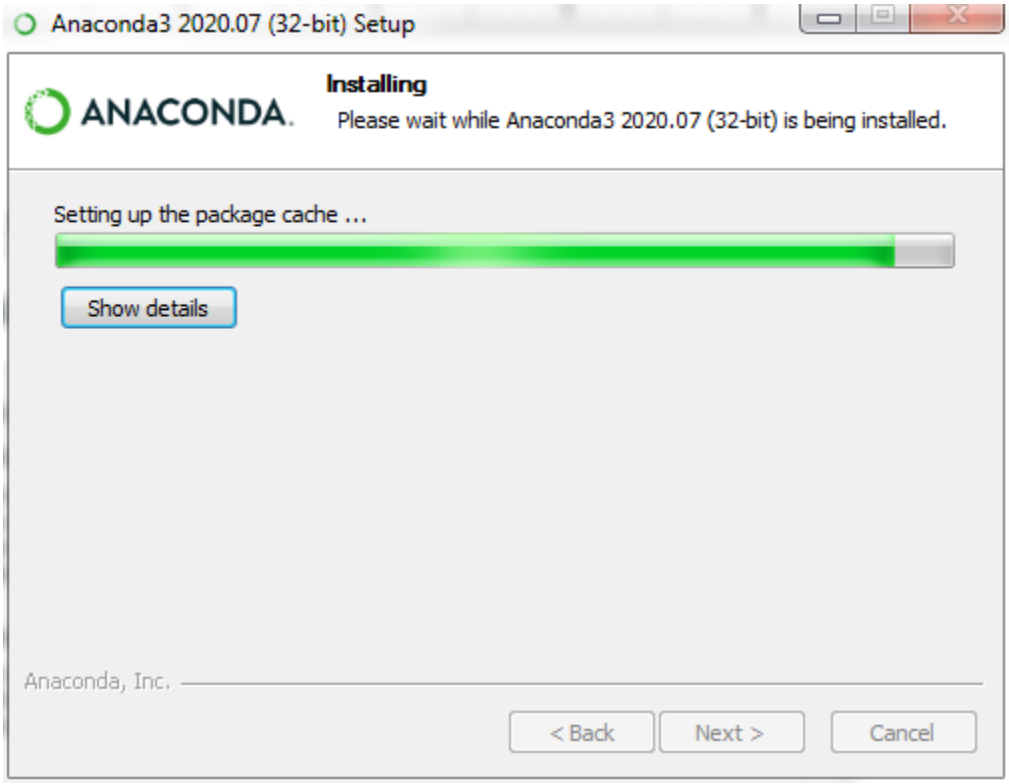
- Select the destination folder for the installation and then click on ‘Next’.



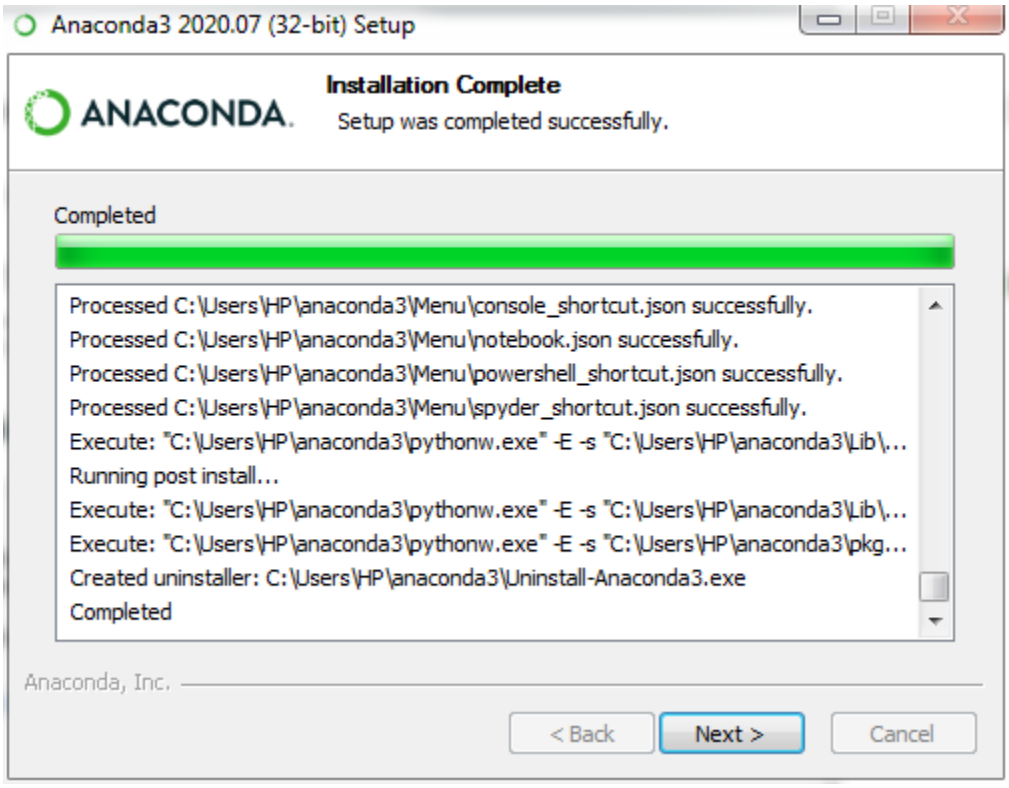
- There is an option to install Anaconda2 in the path environment, however, it's not recommended. So, just click on ‘Install’.



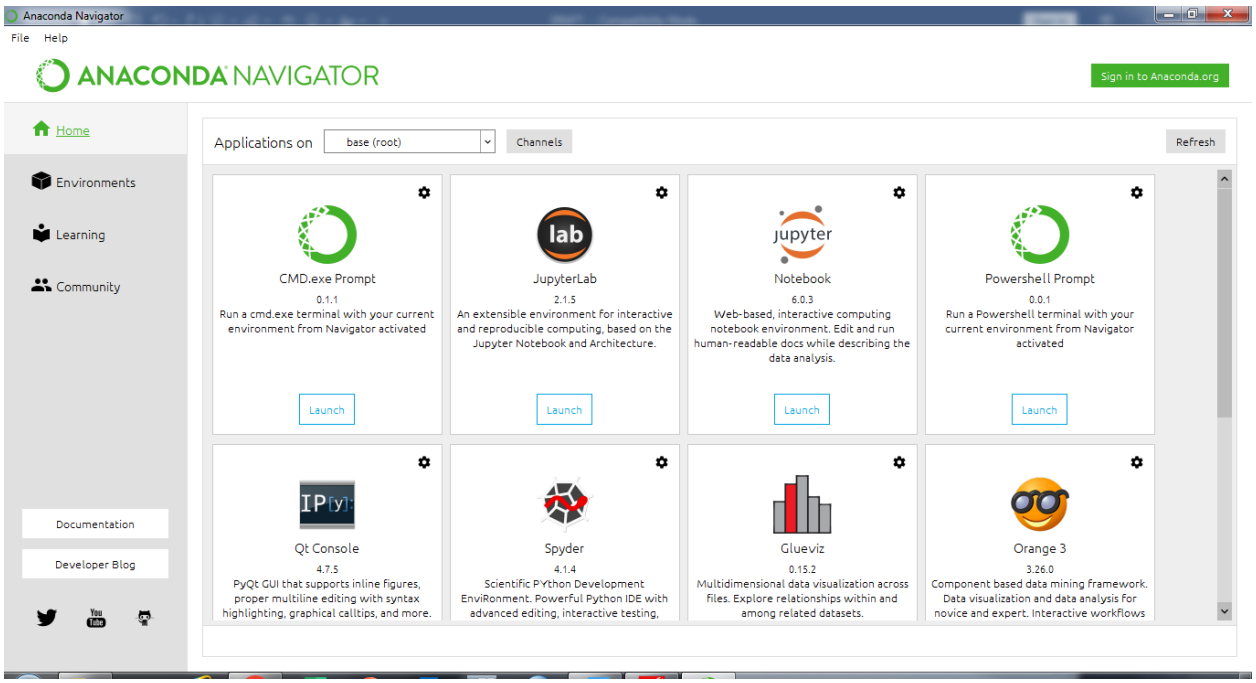
The installation will begin.



- Once the installation is completed, the below window will appear. Click on ‘Next’.



- After completing these steps a window will open showing ‘Installation complete’. There click on ‘Finish’.
- Open the Anaconda navigator. The virtual environment with the anaconda could be seen as:



Following these steps, the Anaconda environment is installed on Windows.

Packages

With programming, it can be difficult to decide which package to use for multiple works. There are multiple packages available like Anaconda. Conda is a powerful package and environment manager which uses Anaconda prompt command line in Windows or a terminal window in Linux or MacOS. In Python, pip is a standard package manager which can be used outside and inside of Anaconda for installing and managing additional packages which are not part of the Python package index. Although Conda and pip seem identical package management tools, they are designed for different purposes. Pip is an official and recommended tool for installing packages from the Python package index (PyPI) and GitHub but Conda is used for installing packages from the Anaconda cloud and Anaconda repository. Conda can’t be used for installing

packages from GitHub but we can use Conda for pip installation. There are 150000 packages with PyPI while only 1500 with the Anaconda repository. In case a package is needed and not available on the Anaconda repository then it can be installed with pip. Pip even installs Python packages while conda installs software written in any language. Thus, before using pip, a Python interpreter needs to be installed.

For example – The available libraries in the anaconda environment could be checked using conda list –explicit.

```
conda list --explicit
```

```
https://repo.anaconda.com/pkgs/main/win-32/libtiff-4.1.0-h56a325e_1.conda
https://repo.anaconda.com/pkgs/main/win-32/libxslt-1.1.34-he774522_0.conda
https://repo.anaconda.com/pkgs/msys2/win-32/m2w64-gcc-libs-5.3.0-7.tar.bz2
https://repo.anaconda.com/pkgs/main/win-32/python-3.8.3-he1778fa_3.conda
https://repo.anaconda.com/pkgs/main/win-32/qt-5.9.7-vc14h73c81de_0.conda
https://repo.anaconda.com/pkgs/main/win-32/curl-7.71.1-h2a8f88b_1.conda
https://repo.anaconda.com/pkgs/main/win-32/pywin32-227-py38he774522_1.conda
https://repo.anaconda.com/pkgs/main/win-32/menuinst-1.4.16-py38he774522_1.conda
https://repo.anaconda.com/pkgs/main/win-32/argh-0.26.2-py38_0.conda
https://repo.anaconda.com/pkgs/main/win-32/asn1crypto-1.3.0-py38_0.conda
https://repo.anaconda.com/pkgs/main/win-32/bitarray-1.4.0-py38he774522_0.conda
https://repo.anaconda.com/pkgs/main/win-32/boto-2.49.0-py38_0.conda
https://repo.anaconda.com/pkgs/main/win-32/certifi-2020.6.20-py38_0.conda
https://repo.anaconda.com/pkgs/main/win-32/chardet-3.0.4-py38_1003.conda
https://repo.anaconda.com/pkgs/main/win-32/comtypes-1.1.7-py38_1001.conda
https://repo.anaconda.com/pkgs/main/win-32/console_shortcut-0.1.1-4.conda
https://repo.anaconda.com/pkgs/main/win-32/docutils-0.16-py38_1.conda
https://repo.anaconda.com/pkgs/main/win-32/entrypoints-0.3-py38_0.conda
https://repo.anaconda.com/pkgs/main/win-32/fastcache-1.1.0-py38he774522_0.conda
```

Once the list is checked, the installation for pip and other pip packages could be done like herein for NumPy

```
conda install pip
```

```
Collecting package metadata (current_repodata.json): ...working... done
Solving environment: ...working... done

# All requested packages already installed.


Note: you may need to restart the kernel to use updated packages.
```

```
conda install NumPy
```

```
Collecting package metadata (current_repodata.json): ...working... done
Note: you may need to restart the kernel to use updated packages.

Solving environment: ...working... done

# All requested packages already installed.
```

Thus, it is beneficial to use a virtual environment as it helps in managing dependencies but ensures that the package remains restricted to the environment and doesn’t affect other scripts. There is a need to install both pip and conda requirements so that packages that can’t be installed with conda could be used.

Chapter Summary

- The virtual environment is simply a networked application that helps the user in interacting with other users' work and computing environment simultaneously.
- Some of the popular packages are Virtualenv, Anaconda (popularly known as Conda), and Pipenv.
- Anaconda is the most popular open-source virtual environment which consists of Spyder, Jupyter, or other notebooks which are required for scientific computing, data analytics, or large data processing.
- Conda and pip seem to be identical package management tools but they are designed for different purposes.