**Dev Setups: Installing Python and Data Science basic stack**[**¶**](#gjdgxs)

This document is iterated on every session, and a lot of work goes into making sure it is clear and as simple as possible. Please do not share outside of Insight. Thanks!

**Part 1: Install Python and packages**[**¶**](#30j0zll)

Most operating systems other than Windows already have Python installed by default. However, we recommend using Anaconda as the easiest, and most useful method to get going with python and the libraries you will need. In addition, Anaconda allows (forces) you to create work environments for your projects. The benefits of using environments are many, allowing your install dependencies to live in harmony.

### **Anaconda**[**¶**](#1fob9te)

Anaconda is an open source distribution of the Python programming languages for data processing, predictive analytics, and scientific computing. It aims to simplify package installation and management while integrating (virtual) environment management.



*Definition*: **Python Package**

*A package is a collection of Python modules. While a module is a single Python file which may contain functions, methods, and classes, a package is a directory of Python modules. Packages contain an additional initialization file* (\_\_init\_\_.py)*, to distinguish it from a directory that just happens to contain a bunch of Python scripts.*

*Definition*: **Virtual environment**

*A virtual environment is a named, isolated, working copy of Python that maintains its own files, directories, and paths so that you can work with specific versions of libraries or Python itself without affecting other Python projects. Virtual environments make it easy to cleanly separate different projects and avoid problems with different dependencies and version requirements across components.*

We strongly encourage the use of virtual environments. The benefits of using environments are many, allowing your install dependencies to live in harmony. Environments can also be created to [run specific versions of Python](http://conda.pydata.org/docs/py2or3.html)!

`Anaconda = package manager + environment manager`

**Part 1: Install Python and packages**[**¶**](#30j0zll)

Let's set up our Python environment via Anaconda. We'll be running through the instructions for Mac OS, but the instructions for Windows and Linux systems should be very similar. Simply follow the same installation steps as described below, and follow Anaconda's recommended installation procedures.

Mac OS specific installation and setup: <https://docs.anaconda.com/anaconda/install/mac-os>

Windows specific installation and setup: <https://docs.anaconda.com/anaconda/install/windows>

Linux specific installation and setup: <https://docs.anaconda.com/anaconda/install/linux>

**Mac OS setup**[**¶**](#3znysh7)

**Step 1:** Go to the Anaconda [download page](https://www.anaconda.com/download) and download the appropriate installer. We'll be installing the Python 3 version.

Installation requires about 600M of space. If you're tight on space, you can try miniconda.

**Python 2 vs Python 3**

In the past, we have advised Insight Fellows and industry partners to use Python 2.x to ensure support for modern packages used in industry. However, analysis and modelling packages are increasingly supporting Python 3.x. Additionally, Python 2.x now has an End-of-Life date set for 2020. We are thus recommending Python 3.x moving forward.

**Step 2:** Follow Anaconda's recommended installation instructions, making note of the directory where Anaconda lives (you should be prompted to specify this, your home directory is a good choice).

**Step 3:** Now's a good time to make sure we have all the latest Anaconda updates. You can check for updates in the command line,

conda update --prefix [path to anaconda] anaconda

**Step 4**: Create your first [environment](https://conda.io/docs/user-guide/tasks/manage-environments.html). We'll install Python 3 into it.

conda create -n insight python=3

This is a barebones approach which only loads Python. We'll install packages later.

**Part 2: Start your virtual environment**[**¶**](#2et92p0)

To enter an Anaconda environment, do:

source activate insight

Where insight is the desired name for your environment.

In your command line you should now see the name of your environment in brackets. Welcome to your virtual workspace! Think of this as a bunker where you can put everything you'll need for your project development, and which remains isolated from the outside world.

For Windows, you may need to use the 'Anaconda Prompt' in place of the command line. Once in the Anaconda prompt, enter your virtual environment with: "activate insight" (in place of "source activate insight" for Mac OS and Linux).

**Part 4: Installing packages**[**¶**](#tyjcwt)

Installing packages with Anaconda is easy. The basic format is:

conda install <list of packages>

For example, let's go ahead and install a few packages we'll need for this workshop.

conda install numpy scipy pandas matplotlib flask scikit-learn jupyter

Here we've installed:

* Jupyter notebooks for exploratory and prototyping work
* pandas for working with tabular data
* numpy for manipulating numeric types
* matplotlib for generating plots and visualizations
* scipy for scientific computing
* scikit-learn for machine learning
* flask for developing web applications

We'll talk a lot more about these packages later on.

**Here are a few useful Anaconda commands:**

"conda update <list of packages>", to update packages.

"conda info --envs", gives you a list of all the virtual environments you've created.

"conda list", gives you a list of all packages you've installed in the current environment, including which versions.

"source deactivate", exits you from the virtual environment.

For now, let's test things out by launching a Jupyter notebook. In the terminal type,

jupyter notebook

This will open up a browser window where you'll see a directory tree from your current working directory. We'll be using Jupyter notebooks heavily in this workshop. Open up DS\_dev\_setup\_part\_2.ipynb to launch the next part and test your setup.