

# JMM Short Course:

## Python Installation and Introduction

This document provides some instructions and links for installing Python and getting started with Jupyter Notebooks, which we will be using for a couple of the sessions during this course. If you already have a Python setup you are happy with, then you don't need to read any further and the second page provides links to the two cloud services that can also be used to run the notebooks for the class if you don't want to install anything on your computer.

### 1 Installing Python

- You can install python through Anaconda by going to [anaconda.com](https://anaconda.com) and selecting individual edition under products in the top menu bar. That page has a download link where you can select the installer for your particular operating system. I recommend simply accepting the defaults throughout the installation process unless you really know what you are doing.
- You can also see a step by step guide for your operating system by going to <https://docs.anaconda.com/anaconda/install/> (feel free to skip the recommended step #2)
- Once you have gotten through the installation process the first step is to create a new environment for the class.
- Begin by opening up a terminal (on Mac or Linux this is your regular terminal - on Windows, Anaconda installs a new 'anaconda prompt' that you can search for in the start menu).
- You should see the word (base) in parentheses to the left of your path and cursor.
- Now type:

```
conda create --name Network_Intro
```

and then hit the enter key.

- Next, we need to activate the environment so type

```
conda activate Network_Intro
```

and hit enter again. Now the text to the left of your path should say (Network\_Intro).

- Finally, try to launch a jupyter notebook in your browser by typing the following in the command line (and then hitting enter):

```
jupyter notebook
```

- If this works, great!
- If not, you will have to install jupyter by typing:

```
>>> conda install jupyter
```

it will think for a while and then ask you for permission to proceed. Type 'y' and hit enter and it will take care of the rest for you.

- You can read more about how jupyter notebooks interact with the python ecosystem here: <https://problemsolvingwithpython.com/02-Jupyter-Notebooks/02.00-Introduction/>

## 2 Notebooks in the cloud

If you are unable to install python on your computer, you can still run python programs and notebooks in the cloud using services provided by CoCalc or CoLab. Having Python installed on your computer means that you won't be reliant on internet access to be able to run programs and that you won't need to worry about space limits or computation time limits that the service providers place on the free accounts but particularly if you are just starting out running the notebooks in the cloud can be a convenient starting point.

- <https://cocalc.com> provides a notebook interface that you can access without even making an account. Just click the green 'Run CoCalc Now' button on the main page. This will take you to a new page where you need to choose a 'kernel' for you notebook - for this class you want to select the one that says python3. If you do register for an account you will be able to save your notebooks in the cloud instead of needing to start over from scratch every time.
- <https://colab.research.google.com> is a Google hosted option that is also free as long as you have a google/gmail account. No need to select a kernel here (they are all python) you can just click file and then new to get started. One advantage to colab for this material is that you can directly import the notebooks from GitHub. If you click File → Open Notebook there is a tab at the top of the interface labelled GitHub where you can enter:

`https://github.com/drdeford/JMM2021NetworkX`

which will show you a list of the notebooks for today's session to start working with.

## 3 Notebooks for the tutorial

The following is a brief description of the main notebooks we will be using to explore the basic features of the networkx package.

- **0\_Jupyter\_Basics.ipynb** This notebook walks through the basic usage of Jupyter notebooks and how to interact with Python cells. It also includes examples of basic arithmetic, variable assignment, data types, data structures, and imports work in Python as well as a couple of exercises to check your understanding. If you already have some familiarity with Python, this one is safe to skip ☺
- **1\_Ego\_Networks.ipynb** This notebook explores the basic properties of the Graph object in networkx by directly constructing ego networks. We will see how to access properties of the corresponding nodes and edges and visualize the networks using the 'draw' function.
- **2\_Social\_Networks.ipynb** In this notebook we will examine how to work with larger and more complex networks and evaluate the results of graph algorithms, including centrality calculations. The main motivating examples will show some of the common properties of social networks and how they differ from structured combinatorial graphs and random network models.
- **3\_Network\_Dynamics.ipynb** In the final notebook we will put everything together, looking at how to model random walks and epidemiological disease spreading using networkx.
- Additional example notebooks for common network analysis tasks can be found here: [https://github.com/drdeford/Math\\_581\\_05](https://github.com/drdeford/Math_581_05), the course repository for Math 581: Computational Tools for Complex Networks. These notebooks don't have as much expository material as the ones for this tutorial session but do include more detailed computations.
- If you are interested in trying out the GerryChain package for analyzing districting plans, you can find tutorial notebooks and sample data here: <https://github.com/drdeford/GerryChain-Templates> along with additional documentation.