1. **Set up jupyter notebooks** You will need to have a way to run jupyter notebooks. Many participants utilize [Anaconda](https://www.anaconda.com/) or [Visual Studio Code](https://code.visualstudio.com/). Other participants use [Google Colab](https://colab.google/) to run all of their notebooks. Whatever you choose, the set up will be somewhat different depending on the computer and operating system you are using so we cannot provide detailed instructions for all cases. Please reach out to the instructors and/or your fellow participants for help getting set up! (If you already know how to use jupyter notebooks, you can skip this step)
   1. *Install Anaconda (For those using a computer that is not an Apple with an M1/M2 chip)*: Follow the steps at Anaconda Guide.
   2. *Set up a conda environment (ONLY if you are not using Anaconda Navigator, i.e. for those not following step a.)*: Follow the steps at Setting Up a Conda Environment.
   3. *Learn how to open a jupyter notebook:* Follow the steps at Opening a Jupyter Notebook.
2. **Learn how to install a python package**: Follow the instructions at Installing a Python Package

Once you have completed steps 1 through 4 (as needed) you should be good to start working through our Python Prep content. Here are a couple of additional guides that you may or may not need while working through the content.

* **Basic Terminal Commands (Linux or MacOS):** Follow the guide at Basic Terminal Commands
* **Basic Windows Command Prompt (Windows):** Follow the guide at Basic Windows Command Prompt

An Anaconda Guide

Last Edited Sep 9, 2023

*Note: This guide is for computers without an Apple M1/M2 chip. If you have a Mac product you can see how to check if you have an Apple M1/M2 chip here,* [*https://www.howtogeek.com/706226/how-to-check-if-your-mac-is-using-an-intel-or-apple-silicon-processor/*](https://www.howtogeek.com/706226/how-to-check-if-your-mac-is-using-an-intel-or-apple-silicon-processor/)*.*

*If you have an Apple M1/M2 chip, do a web search for the most up to date instructions on how to use jupyter notebooks for your particular computer. If you struggle, consider asking the cohort slack channel for help. There are likely fellow participants that have encountered your tech issue!*

**Introduction**

Anaconda is an open source distribution of Python (and R, another programming language). It integrates a number of features that are useful for the Erdős Institute’s educational content, including a straightforward user interface for running jupyter notebooks and installing Python packages.

In this brief tutorial we will review how to install Anaconda, and show you what to check to see that you have correctly installed it.

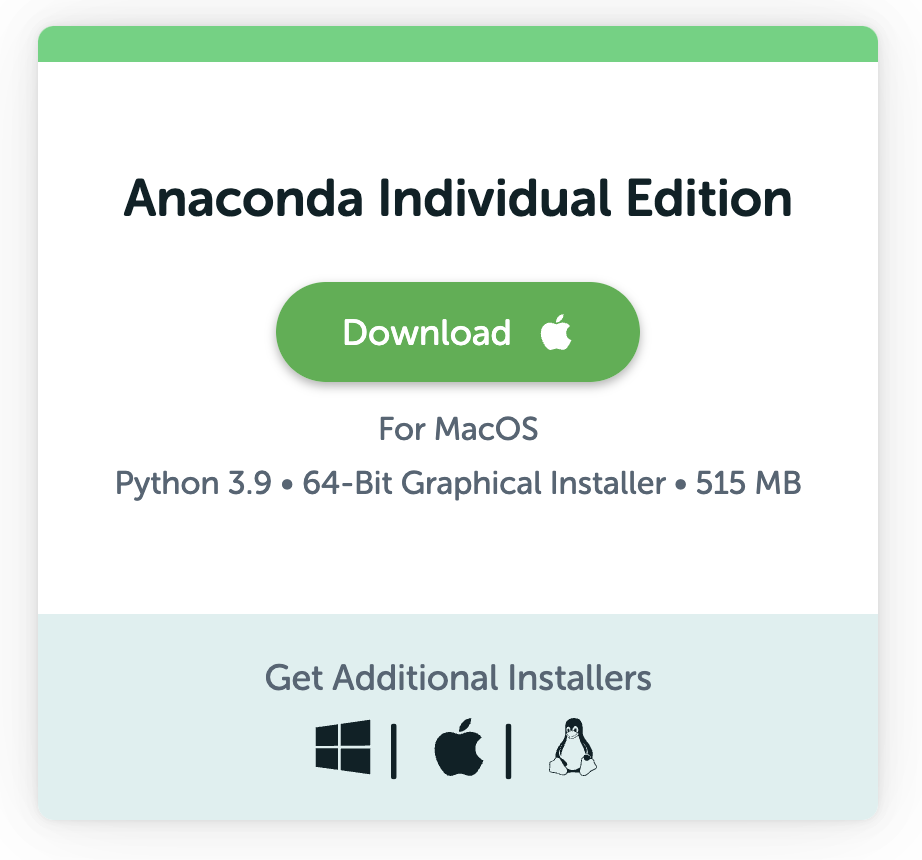
*Note: If you already have a preferred Python setup that allows you to run a jupyter notebook, you are welcome to use that rather than installing a new one. Just be sure to check the package versions you are using to compare with what versions with which the content was written.*

**Instructions**

*Note: If you already have a python setup that allows you to open and run jupyter notebooks you can skip this tutorial.*

*Installation*

Installation of Anaconda is relatively straightforward, just go to this link, <https://www.anaconda.com/products/individual>, and look for the following box.



It should automatically detect what operating system you are running and provide you with the appropriate download.

*Opening the Navigator*

Following a successful Anaconda installation you should be able to open the Anaconda navigator.

Windows Machine

Click on the Windows Icon and search through the applications installed on your computer and open via the Anaconda Icon, it should look something like this:

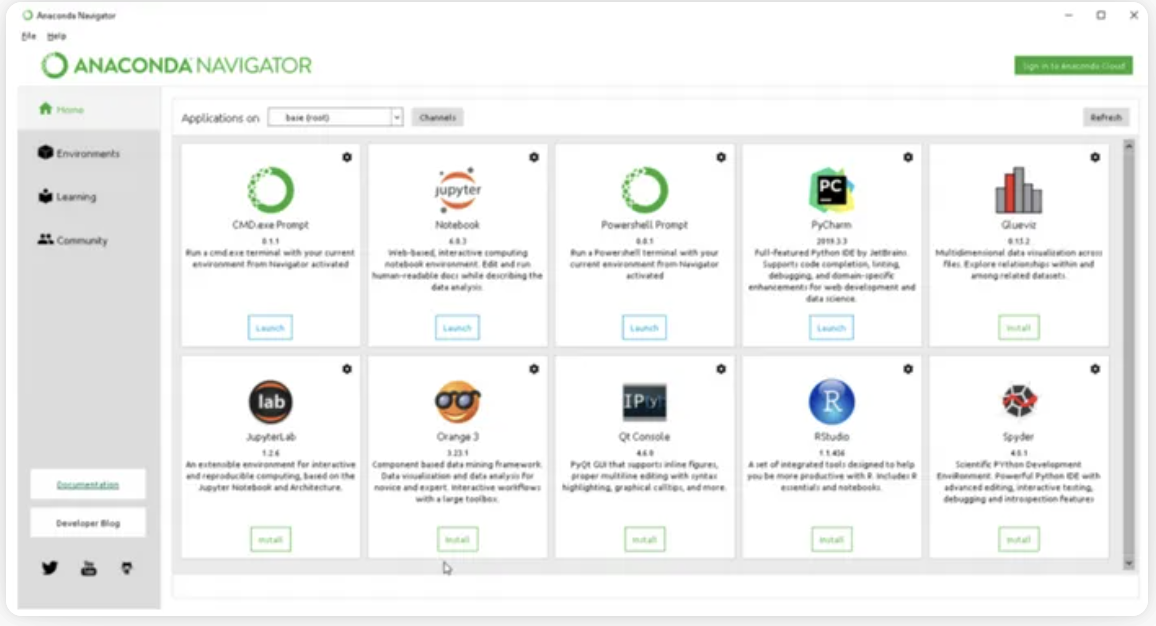
MacOS

Open your Finder and go to the “Applications” Folder, look for “Anaconda-Navigator”, with the same icon you see above.

Linux

Open a terminal window and type anaconda-navigator.

Upon successfully opening the navigator you should see something resembling the following:



**Conclusion**

In this tutorial we installed the Anaconda python distribution. We will learn more about how to use this to open *jupyter notebooks* and install Python packages in coming tutorials. If you find yourself having trouble, or with additional navigator questions try looking at the official Anaconda documentation, <https://docs.anaconda.com/anaconda/navigator/getting-started/>.

Setting Up a Conda Environment

Last Edited Jan 31, 2022

*Note: This tutorial is for those* ***not*** *using the Anaconda Navigator.*

**Introduction**

In this tutorial we will review how to set up and activate a *conda environment* with a desired set of python packages.

To get started with conda, you can either read through this tutorial or go directly to the conda documentation page found here, <https://docs.conda.io/projects/conda/en/latest/user-guide/getting-started.html#starting-conda>.

**Conda Environments**

*What is a Conda Environment?*

A conda environment is a directory on your computer that will contain a specific subset of python packages. Importantly, each unique environment you set up will have no impact on any other environment in your system. This is nice because you may want to have multiple versions of the same python packages installed on your computer. You may want such a set-up while working on multiple projects with different teams.

*How to Create an Environment*

In order to create a conda environment you must first have conda installed on your machine. If you have installed the Anaconda distribution, <https://docs.google.com/document/d/1d0Lixamip0Jj5EkjO-GXYb3F22jxnw40YsGgqHqv0GY/edit?usp=sharing>, or directly installed conda using miniforge, <https://docs.google.com/document/d/1AP7vUyxIok6bJwMWKJa12PPWOYbSbNrNWseIDCPEYxE/edit?usp=sharing>, then you should already have conda installed.

Checking for Conda Installation - Windows

Search for the “Anaconda Prompt” Desktop application in your system. If it is there you have conda installed.

Checking for Conda Installation - MacOS/Linux

1. Open the terminal
2. Execute where conda
   1. If it is installed, you should see a list of directories corresponding to conda
   2. If it is not installed, you should get the reply conda not found.

*Creating an Environment*

The following instructions will work in the “Anaconda Prompt” on Windows and the Terminal on MacOS/Linux.

Instructions

1. Open a new Anaconda Prompt or Terminal Window
2. Execute the command conda create --name insert\_name python\_packages, where:
   1. insert\_name should be replaced with your preferred name for the environment and
   2. python\_packages should be replaced with a list of the python packages that you want installed in the environment. *Note: You do not need to know all of these when you create the environment. Additional packages can be installed later.*
      1. For our boot camp python\_packages should be replaced with scikit-learn pandas matplotlib numpy scipy seaborn jupyter statsmodels.
3. Follow any prompts you receive in your terminal/command prompt.

*Activating an Environment*

The following instructions will work in the “Anaconda Prompt” on Windows and the Terminal on MacOS/Linux.

Instructions

1. Open a new Anaconda Prompt or Terminal Window.
2. Execute the command conda activate environment\_name where you should replace environment\_name with the desired environment’s name.

*Deactivating an Environment*

The following instructions will work in the “Anaconda Prompt” on Windows and the Terminal on MacOS/Linux.

Instructions

1. In the Anaconda Prompt or Terminal Window with the active conda environment execute the command conda deactivate.

**Conclusion**

You now know how to set up, activate and deactivate a conda environment.

Opening a Jupyter Notebook

Last Edited Jan 21, 2022

**Introduction**

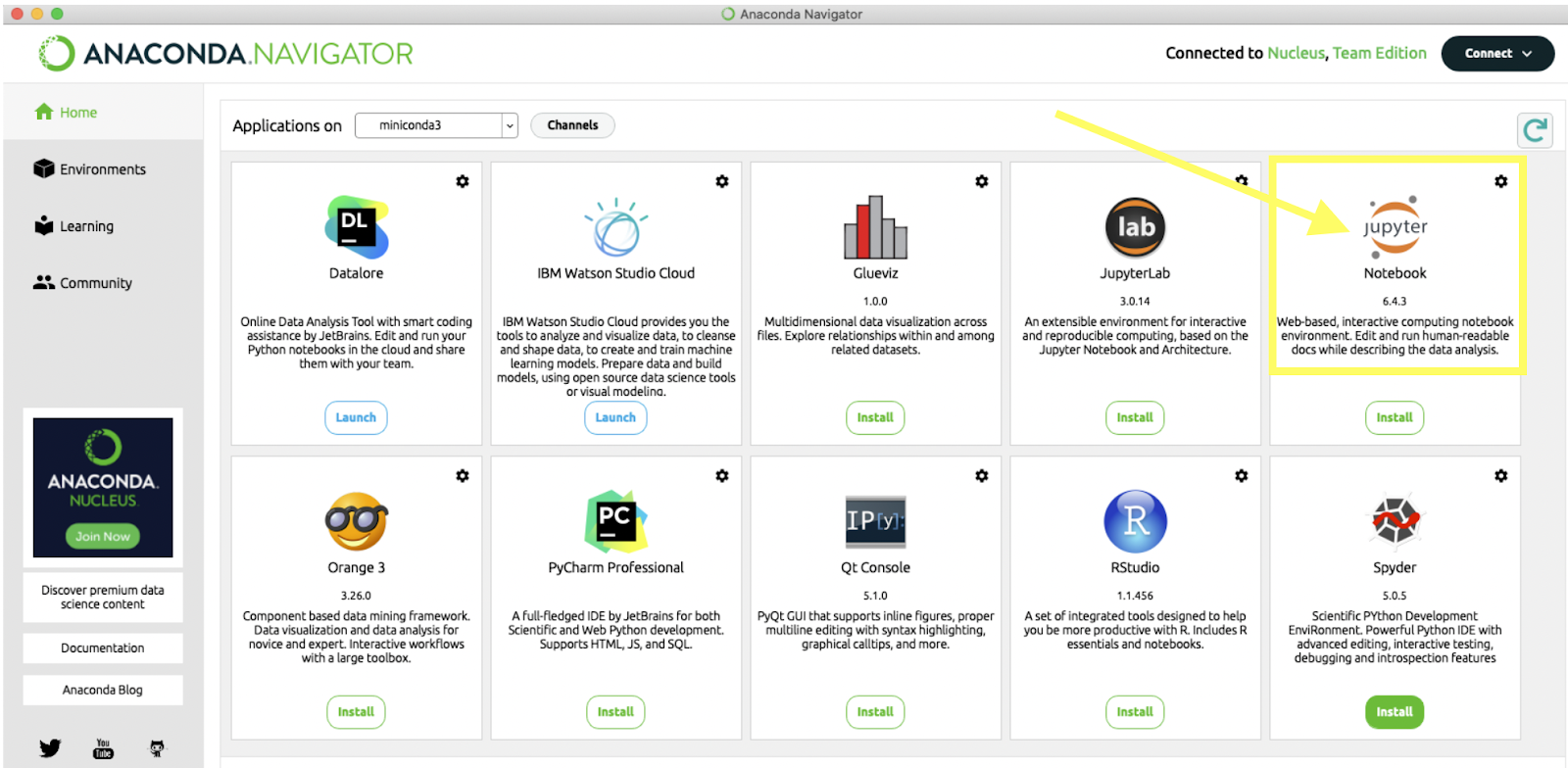
*Jupyter notebooks* are a web-based interactive page that combines document style writing with live code. They are particularly useful for teaching python because you can demonstrate code while explaining it with a chunk of well formatted text. In this tutorial we will demonstrate how to open a jupyter notebook on your computer.

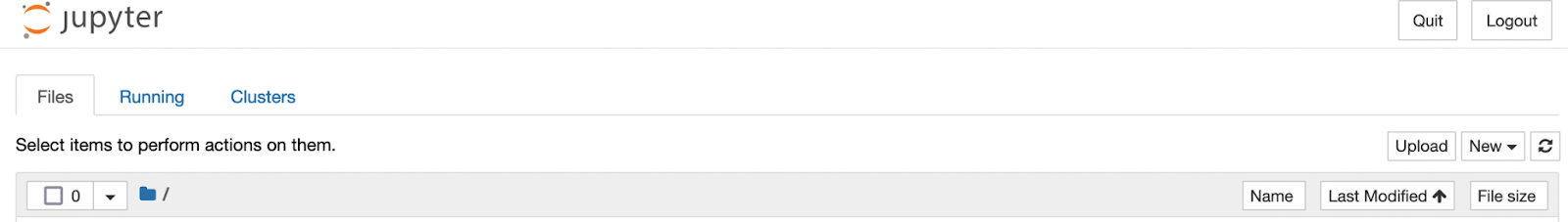
**Opening the Jupyter Notebook Home Page**

*Note: If you are already familiar/comfortable with opening a jupyter notebook, you can skip this tutorial.*

Using Anaconda Navigator

*Note: You must have Anaconda Navigator installed to follow these instructions.*

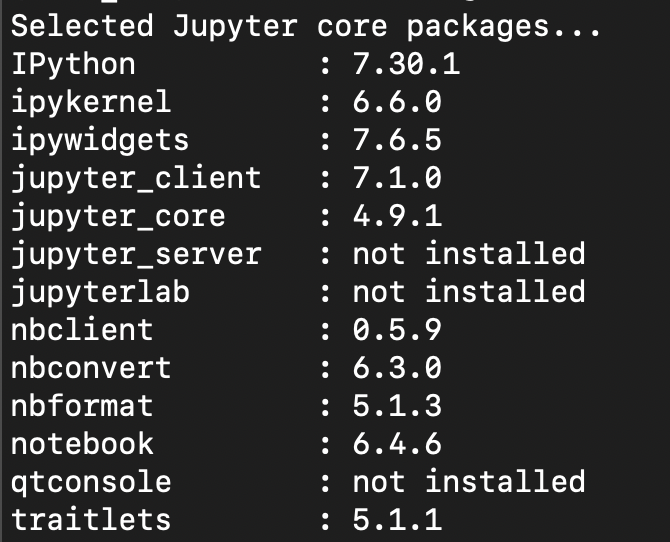
1. Open the Anaconda Navigator following the instructions outlined in our Anaconda Guide, <https://docs.google.com/document/d/1d0Lixamip0Jj5EkjO-GXYb3F22jxnw40YsGgqHqv0GY/edit?usp=sharing>.
2. Once the Navigator is open double click on the Jupyter Notebook Icon, see the highlighted Navigator image below:
3. Your default web browser should then open up a tab, the top of which should look like this:



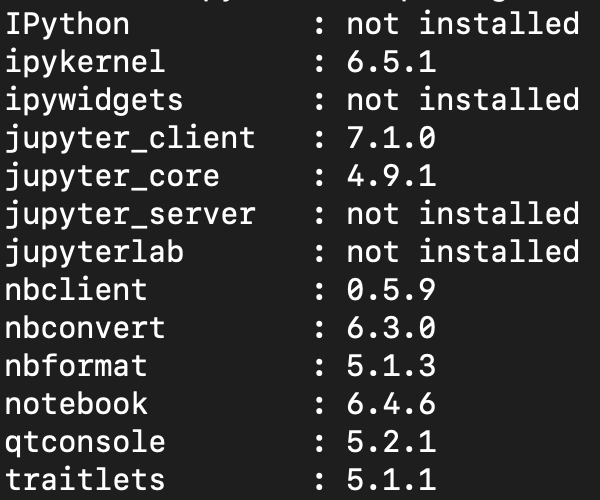
Using a Conda Environment

*Note: These instructions are for someone that has already set up a conda environment following these instructions,* [*https://docs.google.com/document/d/1YZjcvmGPSTce0gG90rpYLcNzuupM7ZpoP4ulKNFq1z4/edit?usp=sharing*](https://docs.google.com/document/d/1YZjcvmGPSTce0gG90rpYLcNzuupM7ZpoP4ulKNFq1z4/edit?usp=sharing)*.*

1. Open a new terminal or prompt and navigate to your desired directory.
2. Activate your desired conda environment by executing the command conda activate environment\_name.
3. Execute jupyter --version, and:
   1. If you see something like the following with a series of numbers next to IPython, then proceed to step 4.

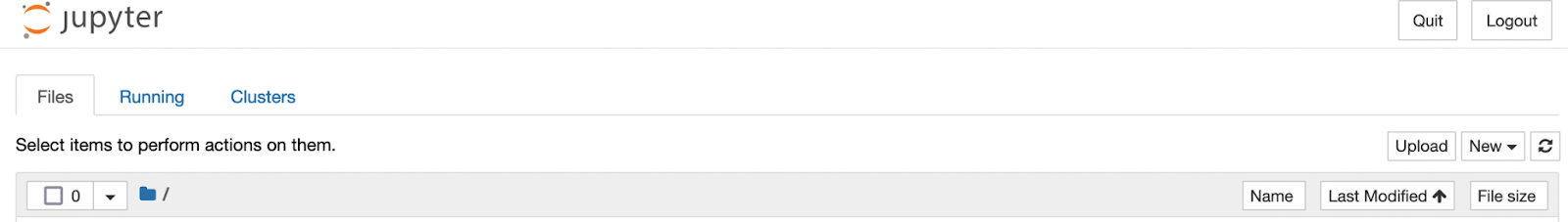


1. If you see something like the following with not installed to the right of IPython:



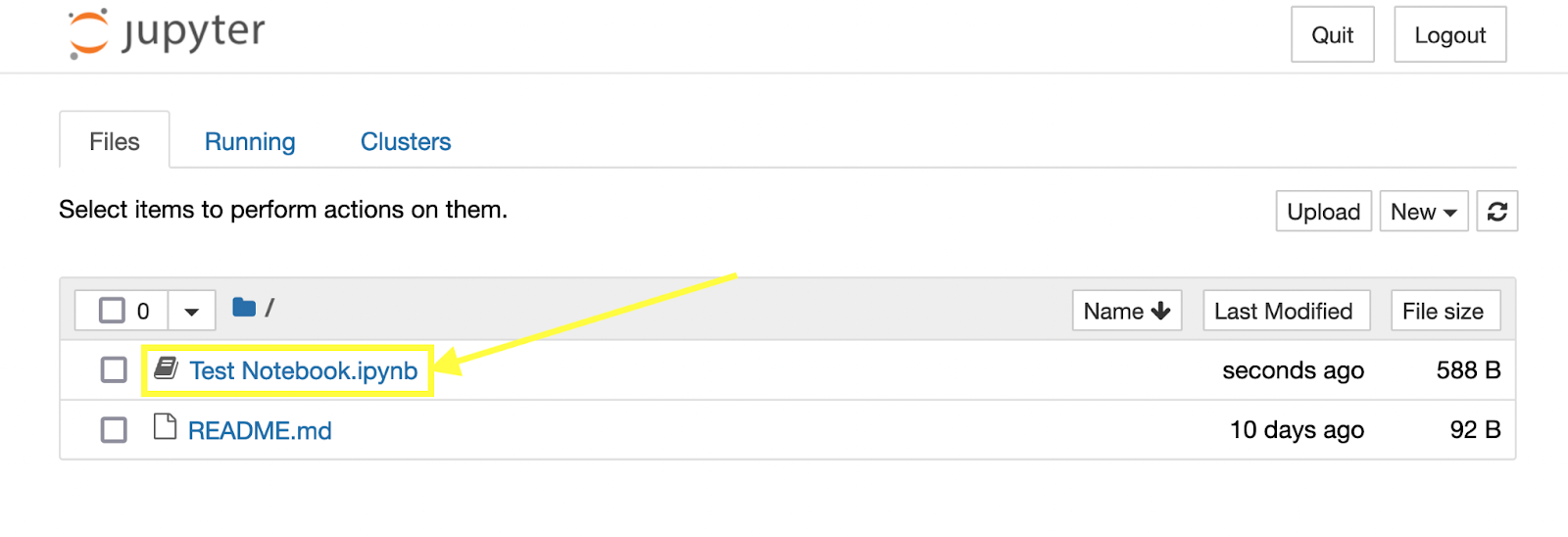
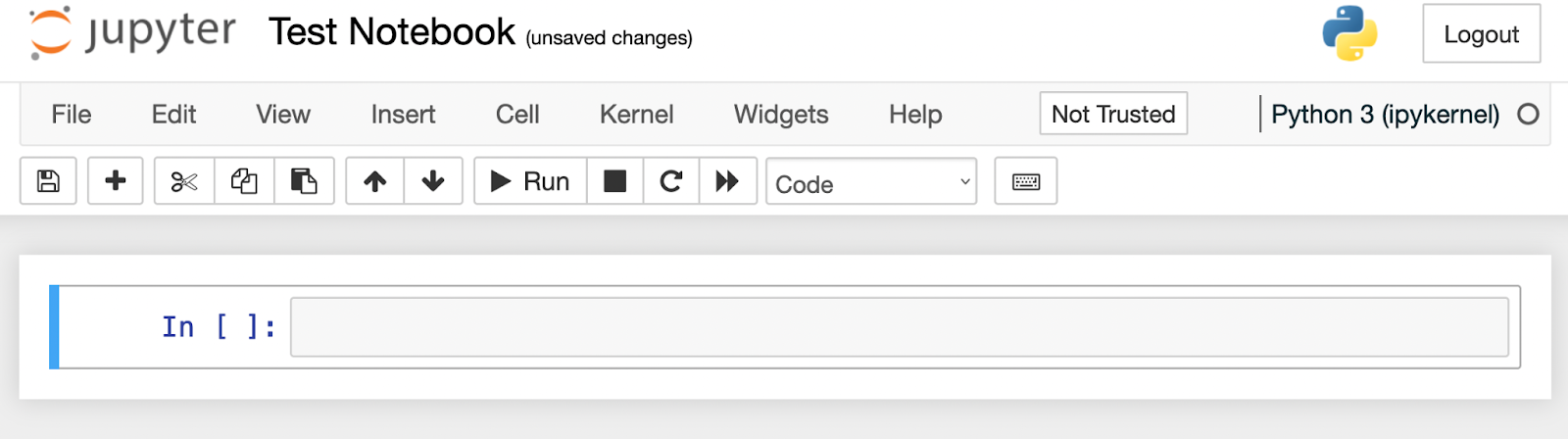
Execute conda install jupyter, then move to step 4.

1. Execute the command jupyter notebook.
2. Your default web browser should then open up a tab, the top of which should look like this:



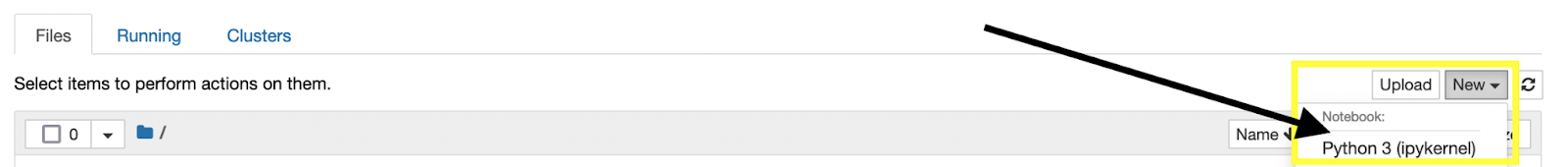
**Opening an Existing Jupyter Notebook from the Home Page**

Instructions

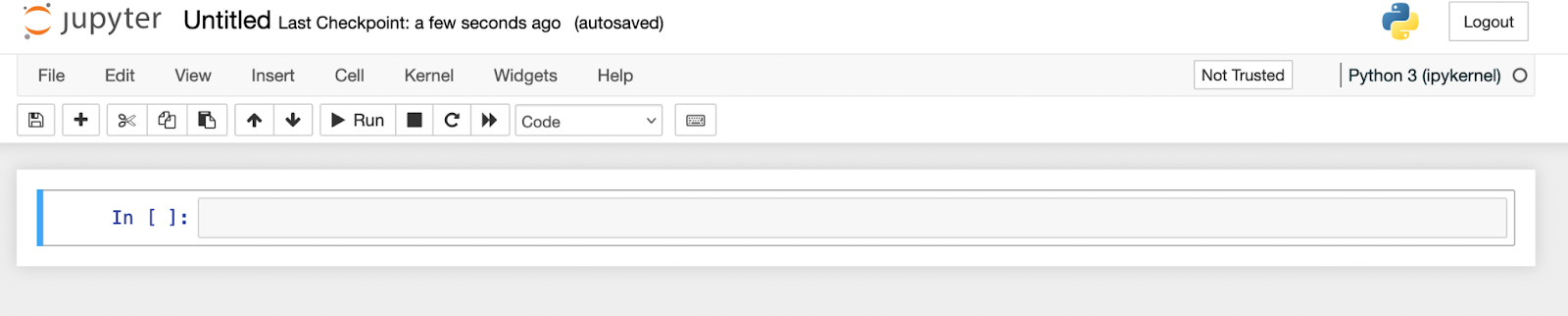
1. Open the home page following the instructions above.
2. A jupyter notebook file is denoted on the homepage with a notebook icon, and a file ending of .ipynb. For example, if you opened the jupyter notebook homepage in the ABlankRepositoryForCloning, you should see the Test Notebook.ipynb file highlighted in the image below.
3. Click on the desired .ipynb file. This should open the notebook in a new tab in your web browser. For example the Test Notebook.ipynb, looks like this when open:

**Opening a New Jupyter Notebook from the Home Page**

1. Open the home page following the instructions above.
2. Click on the “New” dropdown button in the upper right corner of the home page.
3. Click “Python 3 (ipykernel)” as shown below:



1. You should then see something like this in a new tab:



**Conclusion**

In this tutorial we demonstrated how to open the jupyter notebook home page and subsequently how to open a jupyter notebook. We are now ready to start our python prep materials.

Installing a Python Package

Last Edited Jan 31, 2022

**Introduction**

Often we will want to use various python packages. If you installed the Anaconda distribution or set up a conda environment following our instructions, you should already have most of the packages we utilize in the boot camp. However, there will come a time when you want to install a python package. In this written tutorial we will demonstrate how to do so.

**Instructions**

Installation Using the Anaconda Navigator

It is possible to install (and upgrade) python packages using the Anaconda Navigator. If you would like to do so, follow the instructions in the Anaconda Navigator documentation found here, <https://docs.anaconda.com/anaconda/navigator/tutorials/manage-packages/#installing-a-package>.

Installation Using Conda in the Terminal or Anaconda Prompt

1. Open your terminal or prompt.
2. If you have a conda environment, activate it.
3. Find the conda installation instructions for the package you are interested in:
   1. For example, here is a link to the seaborn installation guide, <https://seaborn.pydata.org/installing.html>.
4. Execute the conda command provided by the relevant installation guide:
   1. For example, conda install seaborn, for seaborn.
5. Follow any instructions given in the terminal or prompt.
6. If no conda installation instructions are provided look at the pip instructions given below.

Installation Using pip in the Terminal or Prompt

Conda installation is not always possible. In such cases you may have to use pip, the package installer for Python, <https://pip.pypa.io/en/stable/>. Prior to using pip you will need to have it installed. Check for installation by executing pip –-version. If you have pip installed the version you have should appear. If not, install following pip’s instructions, <https://pip.pypa.io/en/stable/installation/>.

To install a package with pip:

1. Open a terminal or prompt window.
2. If you have a conda environment, activate it.
3. Find the pip installation instructions for the package you are interested in:
   1. For example, here is a link to the seaborn installation guide, <https://seaborn.pydata.org/installing.html>.
4. Execute the pip command provided by the relevant installation guide:
   1. For example, pip install seaborn, for seaborn.
5. Follow any instructions shown in the terminal or prompt window.

**Conclusion**

In this tutorial we have provided instructions on how to install a new python package.

Basic Terminal Commands

Last Edited Jan 24, 2022

*This tutorial is for those using MacOS or Linux, if you use a Windows machine see this tutorial instead:* [*https://docs.google.com/document/d/1xZJjETAeBdnKcKCmO5jbwQp2w\_GuDt793q58hx1lAnk/edit?usp=sharing*](https://docs.google.com/document/d/1xZJjETAeBdnKcKCmO5jbwQp2w_GuDt793q58hx1lAnk/edit?usp=sharing)*.*

**Introduction**

In this tutorial we will cover the basic terminal commands you may need to know for the Erdős Institute’s boot camp materials. Your computer’s terminal is an interface that allows you to directly type commands for your computer to execute. Terminal is the standard name for this interface on MacOS or Linux machines. If you use a Windows computer you should look at the command prompt tutorial found here, <https://docs.google.com/document/d/1xZJjETAeBdnKcKCmO5jbwQp2w_GuDt793q58hx1lAnk/edit?usp=sharing>. We will use the terminal to navigate *directories* and install python packages. You may also use the terminal to manage git repositories on your computer instead of the GitHub desktop application.

**Terminal Basics**

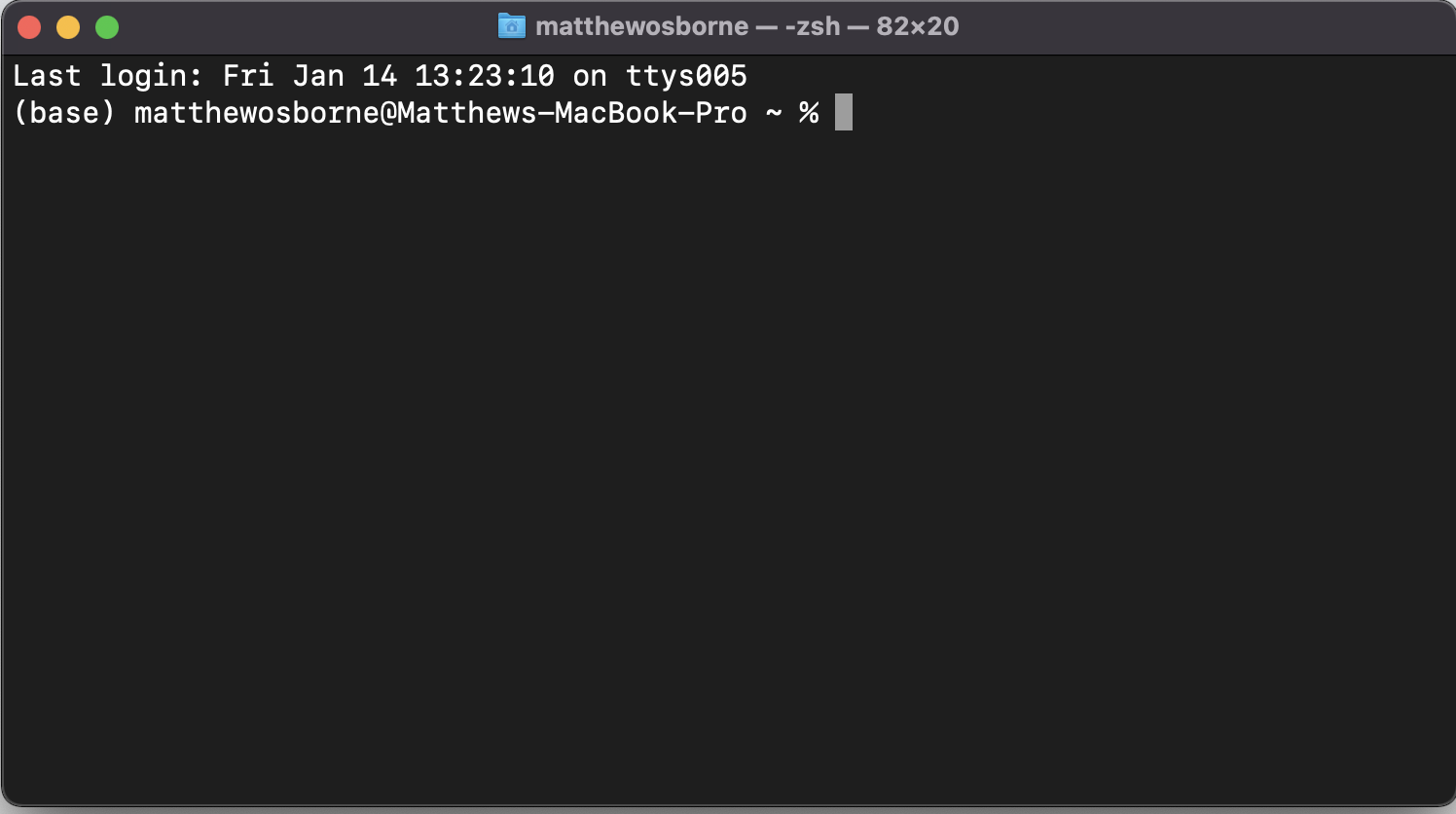
*Opening a Terminal Window*

MacOS

1. Open your Finder
2. Go to the “Utilities” folder in the “Applications” section
3. Open the application labeled “Terminal”

Linux

1. Hit ctrl+alt+T simultaneously

Once opened you should see something like the following:

*Directories*

*Directory* may be an unfamiliar term, but it is just another way to refer to a folder on your computer. It is a “location” on your computer’s memory in which files or other folders are stored. In the above image of a terminal, the default directory is the folder /Users/matthewosborne on my MacBook. If you opened a fresh terminal you would likely see something similar. So when you see or hear “directory”, this should just mean the same thing as folder.

*Basic Commands*

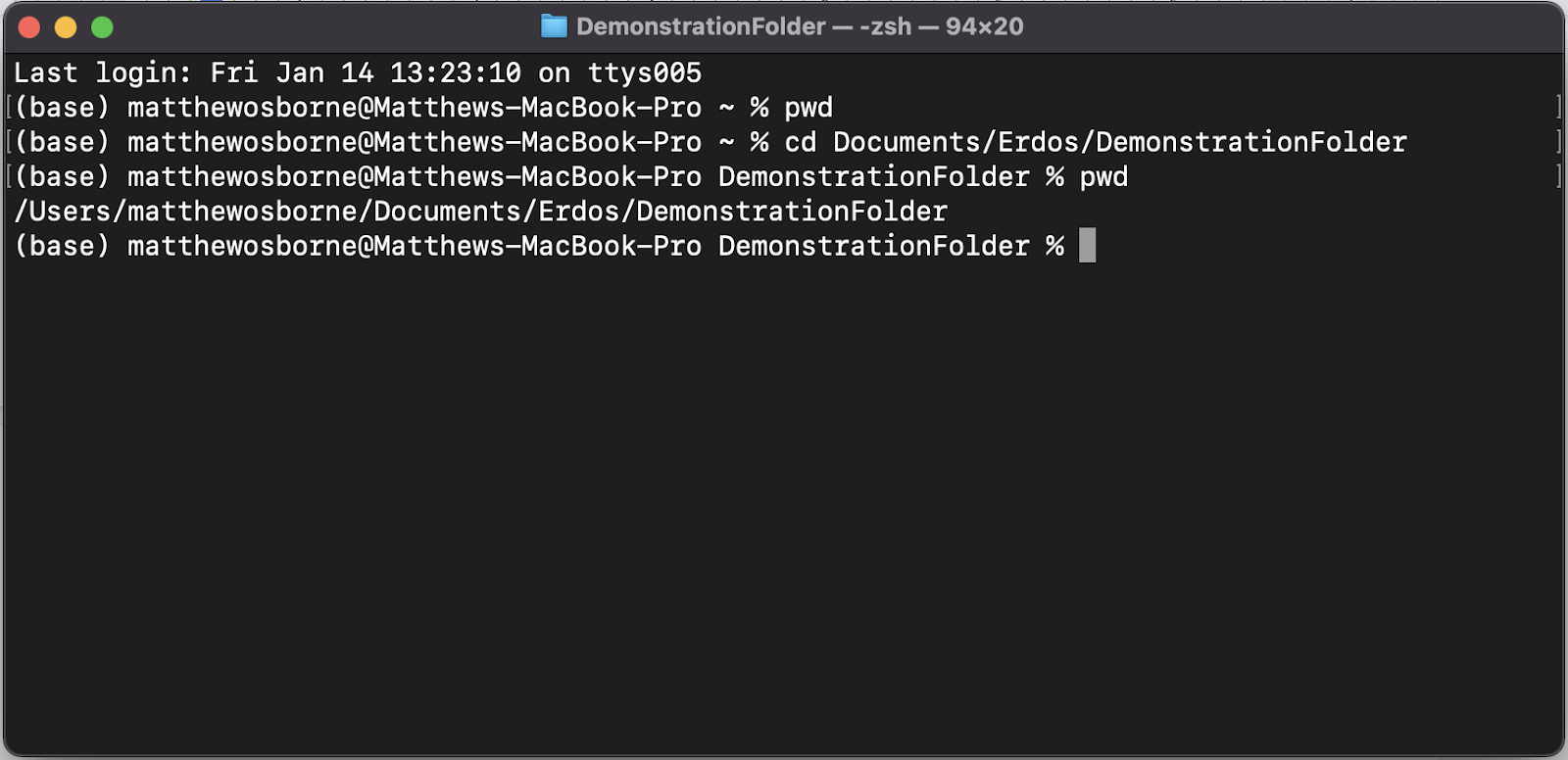
Here are a few of the most common commands for the terminal. These commands should be all you need to know in order to work through our educational content.

Below we will use the phrase “execute the command”. What this means in practical terms is to type the command into your terminal and hit Enter. Doing so tells your computer what activities you would like it to undertake.

1. Finding where you currently are in the terminal - pwd

The command pwd stands for *present working directory*. When you execute this command the computer will spit out what directory your terminal interface is currently in.

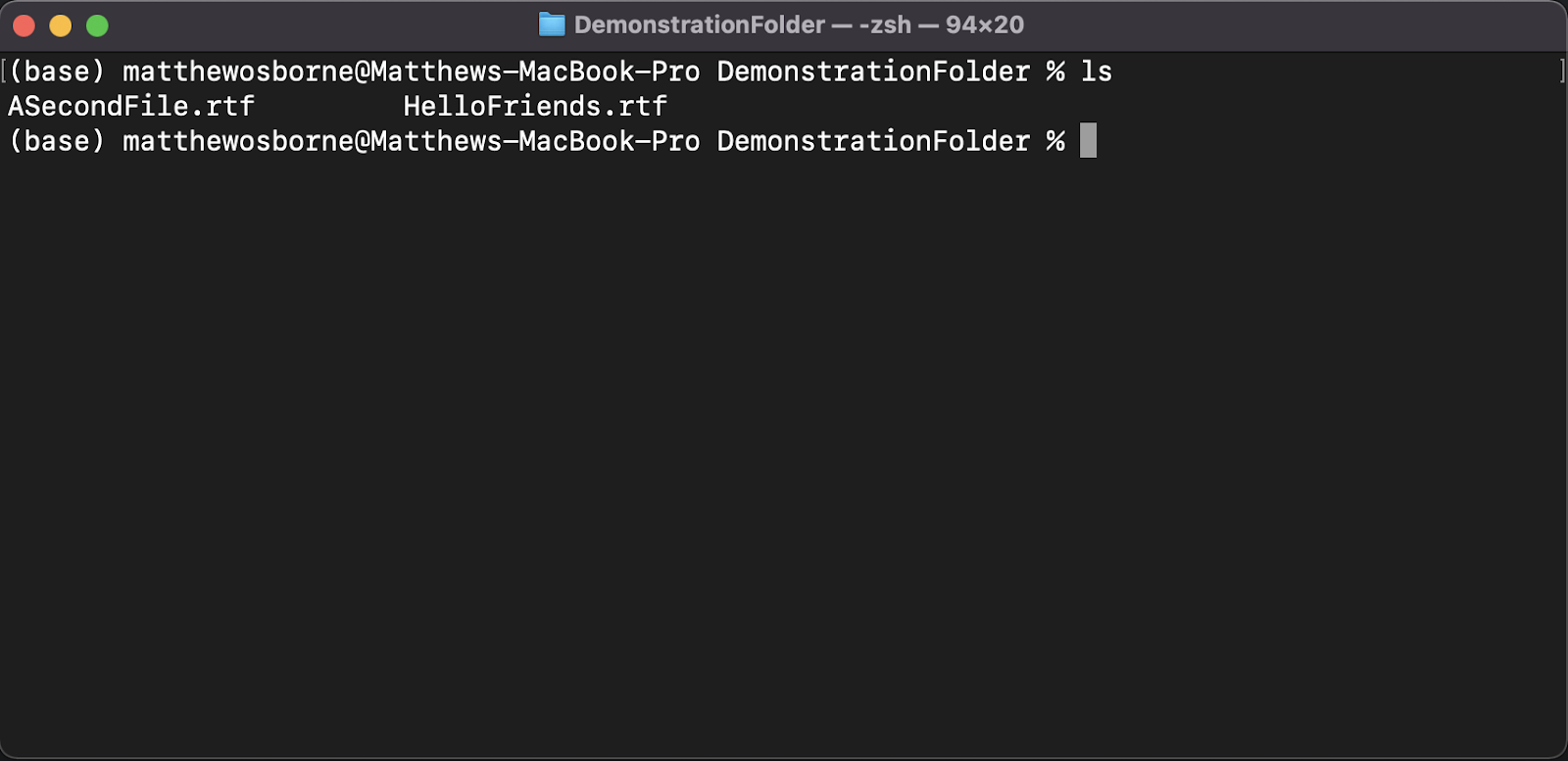
1. Changing what directory you are working within - cd

cd stands for *change directory.* Executing cd directory\_name tells your computer that you would like to go from whatever directory you are currently in to the directory you have given in place of directory\_name. Below we will see an example where I go from /Users/matthewosborne to /Users/matthewosborne/Documents/Erdos/DemonstrationFolder.

Notice that I did not have to type out the entire directory path, only Documents/Erdos/DemonstrationFolder. This is because the folder Documents/Erdos/DemonstrationFolder is contained within my current working directory, /Users/matthewosborne.

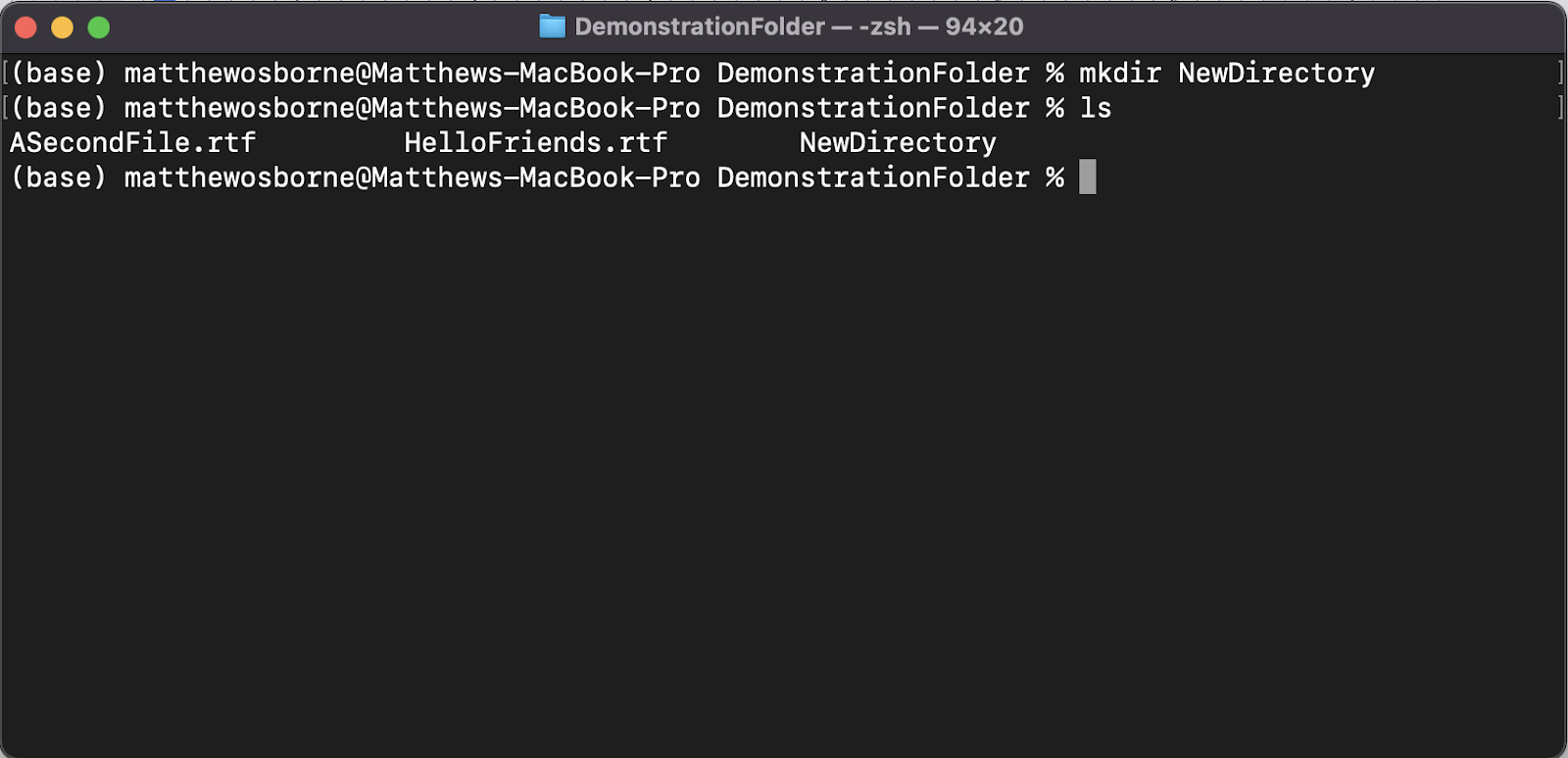
A common command involving cd is cd .., which will take you back one folder. If I had executed this while working in /Users/matthewosborne I would have ended up in /Users.

1. Listing the contents of your present working directory - ls

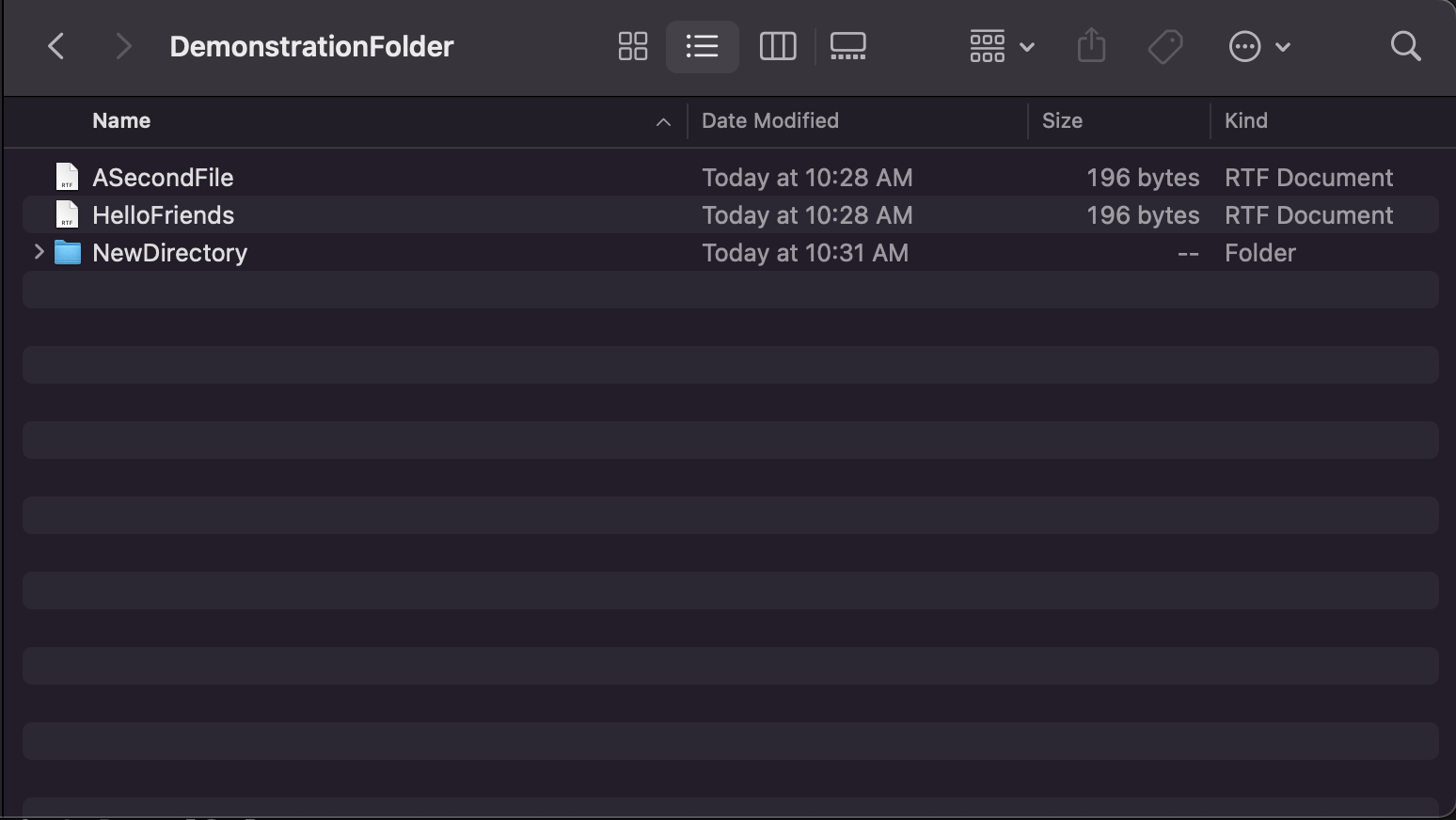
Executing ls tells your computer to list the contents of your present working directory. When this is done your terminal will print all of the files and folders contained in that directory. 

In this example we can see that DemonstrationFolder contains two files HelloFriends.rtf and ASecondFile.rtf.

1. Making a new directory - mkdir

Executing mkdir directory\_name tells your computer to make a new directory with the given name. Below we see an example where I make a new directory with the name NewDirectory within the DemonstrationFolder directory.

Even though I created this directory in the terminal I can also see it in my Finder when I open DemonstrationFolder.



**Conclusion**

In this tutorial we gave a brief introduction to the terminal. In particular we reviewed a few of the most common terminal commands.

Basic Windows Command Prompt

Last Edited Jan 24, 2022

*This tutorial is for those using a Windows machine, if you run MacOS or Linux see this tutorial instead:* [*https://docs.google.com/document/d/1npTQ7SQVGXVi7hfL6M8yXSlgvhdQ9OH-NA5of7m817k/edit?usp=sharing*](https://docs.google.com/document/d/1npTQ7SQVGXVi7hfL6M8yXSlgvhdQ9OH-NA5of7m817k/edit?usp=sharing)*.*

**Introduction**

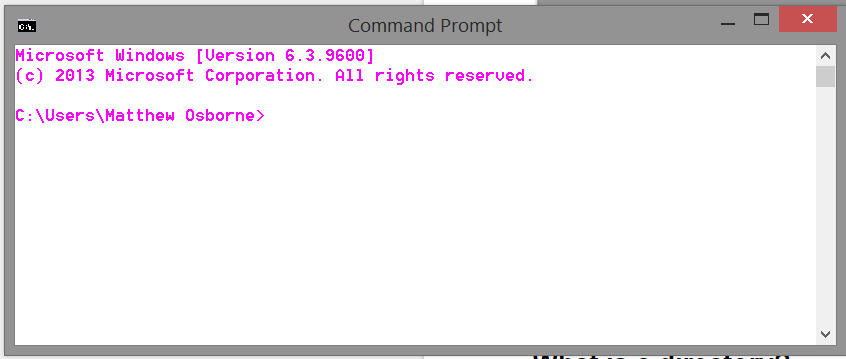
In this tutorial we will cover the Windows command prompt basics you may need to know for the Erdős Institute’s boot camp materials. The command prompt is an interface that allows you to directly type commands for your computer to execute. We will use the command prompt to navigate *directories* and install python packages. You may also use it to manage git repositories on your computer instead of the GitHub desktop application.

**Command Prompt Basics**

*Opening a Command Prompt Window*

1. Click on the Windows Icon
2. In the search bar type “Command Prompt” hit Enter
3. Assuming “Command Prompt” appears click to open this program

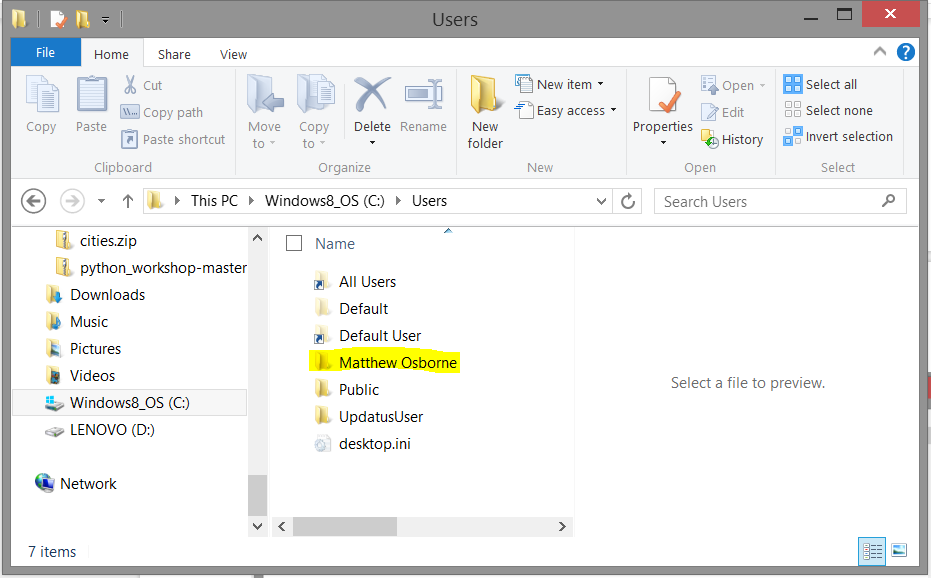
Once opened you should see something like the following:



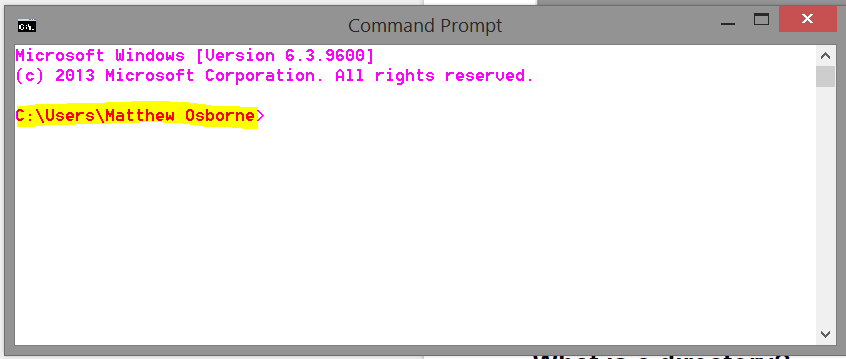
Note that the font and background color may be different. The default background is usually black and the default font is usually white.

*Directories*

*Directory* may be an unfamiliar term, but it is just another way to refer to a folder on your computer. It is a “location” on your computer in which files or other folders are stored. For example, the highlighted folder Matthew Osborne, can be accessed as the directory C:\Users\Matthew Osborne.



When we are in the command prompt we are always working in a directory. In the command prompt I opened I am working in the C:\Users\Matthew Osborne directory. This is highlighted below:



If you opened a fresh terminal you would likely see something similar. So when you see or hear “directory”, this should just mean the same thing as folder.

*Basic Commands*

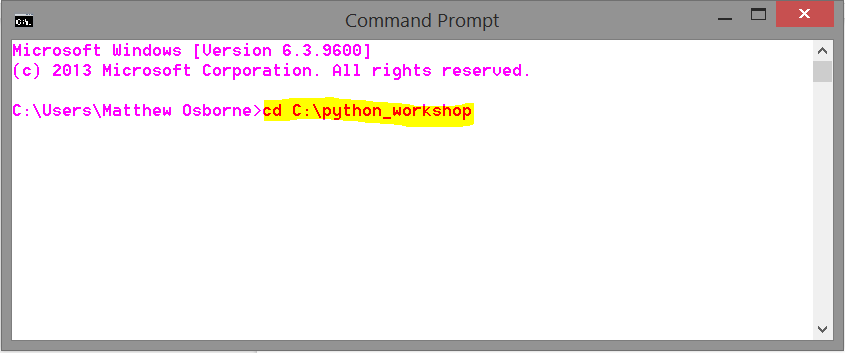
Here are a few of the most common commands for the command prompt. These commands should be all you need to know in order to work through our educational content.

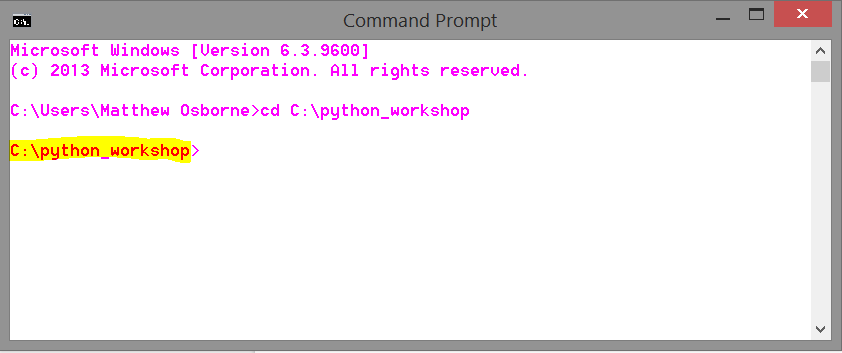
Below we will use the phrase “execute the command”. What this means in practical terms is to type the command into your command prompt and hit Enter. Doing so tells your computer what activities you would like it to undertake.

1. Finding where you currently are in the terminal - %cd%

When you execute the %cd% command, the computer will spit out what directory your terminal interface is currently in. For example, if I were to execute %cd% in the command prompt picture above, I should see C:\Users\Matthew Osborne.

1. Changing what directory you are working within - cd

cd stands for *change directory.* To do this type in: **cd\** and then type in the name of the directory you’d like to go to. Executing cd directory\_name tells your computer that you would like to go from whatever directory you are currently in to the directory you have given in place of directory\_name. Below we will see an example where I go from C:\Users\Matthew Osborne to C:\python\_workshop.

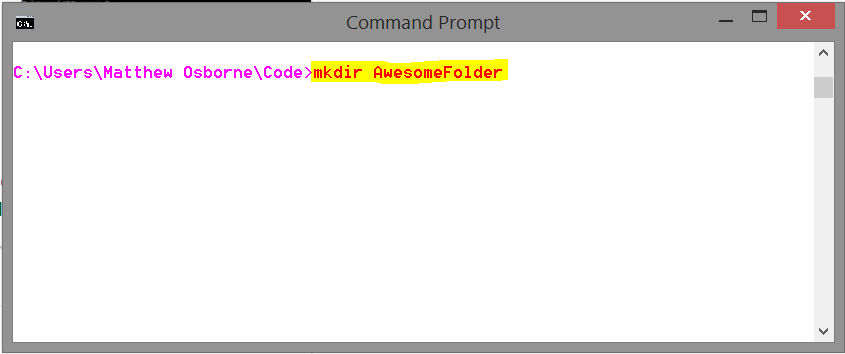


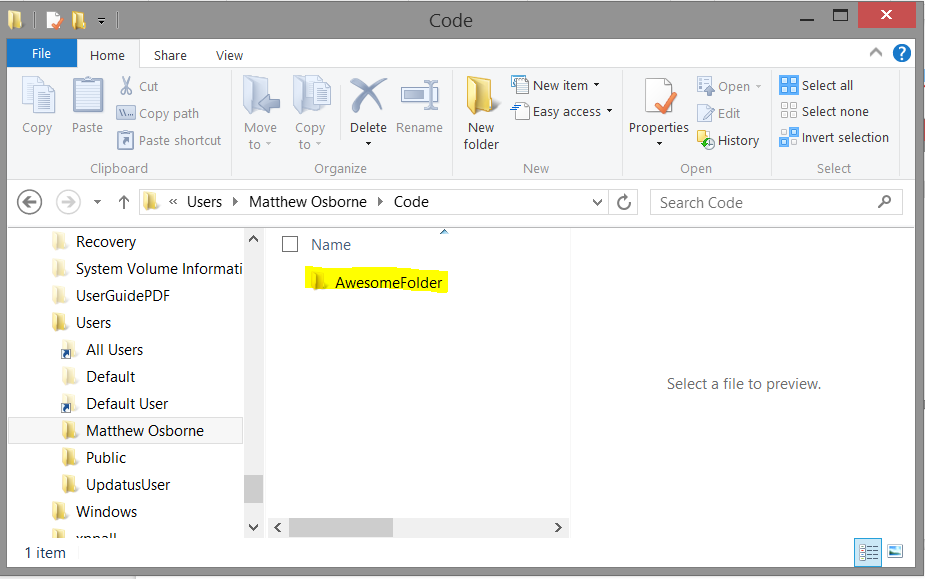
A common command involving cd is cd .., which will take you back one folder. If I had executed this while working in C:\Users\Matthew Osborne I would have ended up in C:\Users.

1. Listing the contents of your present working directory - dir

Executing dir tells your computer to list the contents of your present working directory. When this is done your command prompt will print all of the files and folders contained in that directory.

1. Making a new directory - mkdir

Executing mkdir directory\_name tells your computer to make a new directory with the given name. Below we see an example where I make a new directory with the name AwesomeFolder within the current directory, C:\Users\Matthew Osborne\Code. 

Now if I check the AwesomeFolder folder will be on my computer inside of C:\Users\Matthew Osborne\Code. 

**Conclusion**

In this tutorial we gave a brief introduction to the command prompt. In particular we reviewed a few of the most common command prompt commands.