**CHAPTER 7 INTRODUCTION TO ANACONDA PYTHON pp. 302 - 316**

**(7.1) GETTING STARTED WITH ANACONDA PYTHON p. 305**

**(7.2) PYTHON CHEATSHEETS p. 311**

**(7.3) JUPYTER NOTEBOOKS p. 312**

**Ch. 7 requires the following Python packages**

conda

jupyter

**Ch. 7 Jupyter/IPython notebooks to be downloaded from the Class Blackboard -> Course Materials folder**

py\_exploratory\_comp\_1\_sol.ipynb

1. **INTRODUCTION TO ANACONDA PYTHON**

Go to <https://www.anaconda.com/download/> and select your OS then the appropriate version of Anaconda and finally choose the latest **Python Version which is currently 3.6 (64 bit is preferable).** For Mac OS choose Graphical Installer.

Download and install the software

**For Windows install**

Double click the installer to launch.

NOTE: To prevent permission errors, do not launch the installer from the [Favorites folder](https://docs.anaconda.com/anaconda/user-guide/troubleshooting.html#distro-troubleshooting-favorites-folder).

NOTE: If you encounter issues during installation, temporarily disable your anti-virus software during install, then re-enable it after the installation concludes. If you installed for all users, uninstall Anaconda and re-install it for your user only and try again.

Click Next.

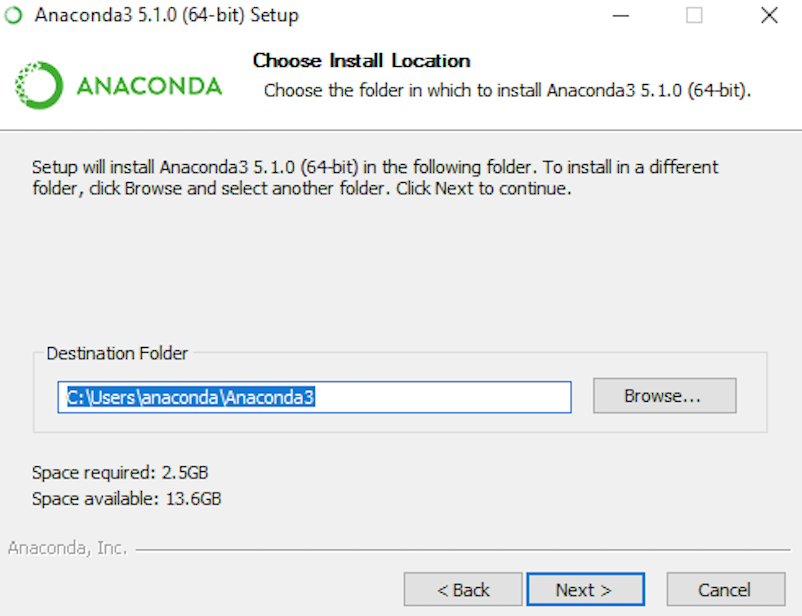
Read the licensing terms and click “I Agree”.

Select an install for “Just Me” unless you’re installing for all users (which requires Windows Administrator privileges) and click Next.

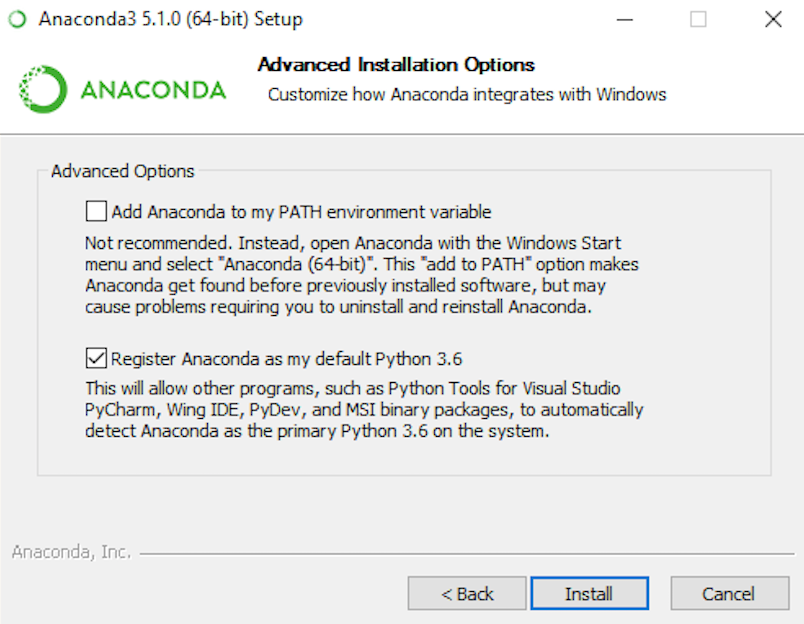
Select a destination folder to install Anaconda and click the Next button.

NOTE: Install Anaconda to a directory path that does not contain spaces or unicode characters.

NOTE: Do not install as Administrator unless admin privileges are required.

[](https://docs.anaconda.com/_images/install-win-destination.png)

**Check the Add Anaconda to my PATH environment variable, but note this can interfere with other software**

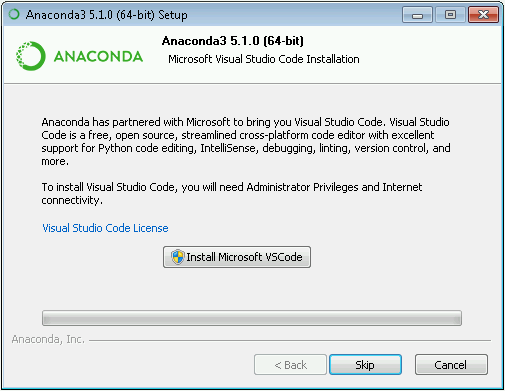
[](https://docs.anaconda.com/_images/install-win-path.png)

Recommend registing Anaconda as your default Python 3.6. so leave this box checked.

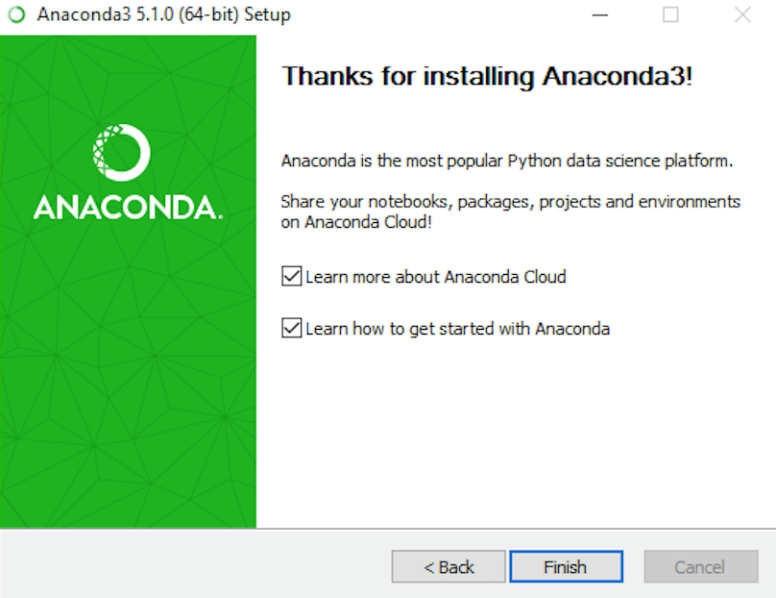
Click the Install button.

Click the Next button.

Optional: To [install VS Code](https://docs.anaconda.com/anaconda/user-guide/tasks/integration/vscode.html), click the Install Microsoft VS Code button. After the install completes click the Next button.

[](https://docs.anaconda.com/_images/vscode-install.png)

After a successful installation you will see the “Thanks for installing Anaconda” dialog box:

[](https://docs.anaconda.com/_images/anaconda-install-win.png)

If you wish to read more about Anaconda Cloud package management service and Anaconda support, check the boxes “Learn more about Anaconda Cloud” and “Learn how to get started with Anaconda”. Click the Finish button.

* 1. **GETTING STARTED WITH ANACONDA PYTHON**

<https://docs.anaconda.com/anaconda/user-guide/getting-started.html>

**Getting started with Anaconda**

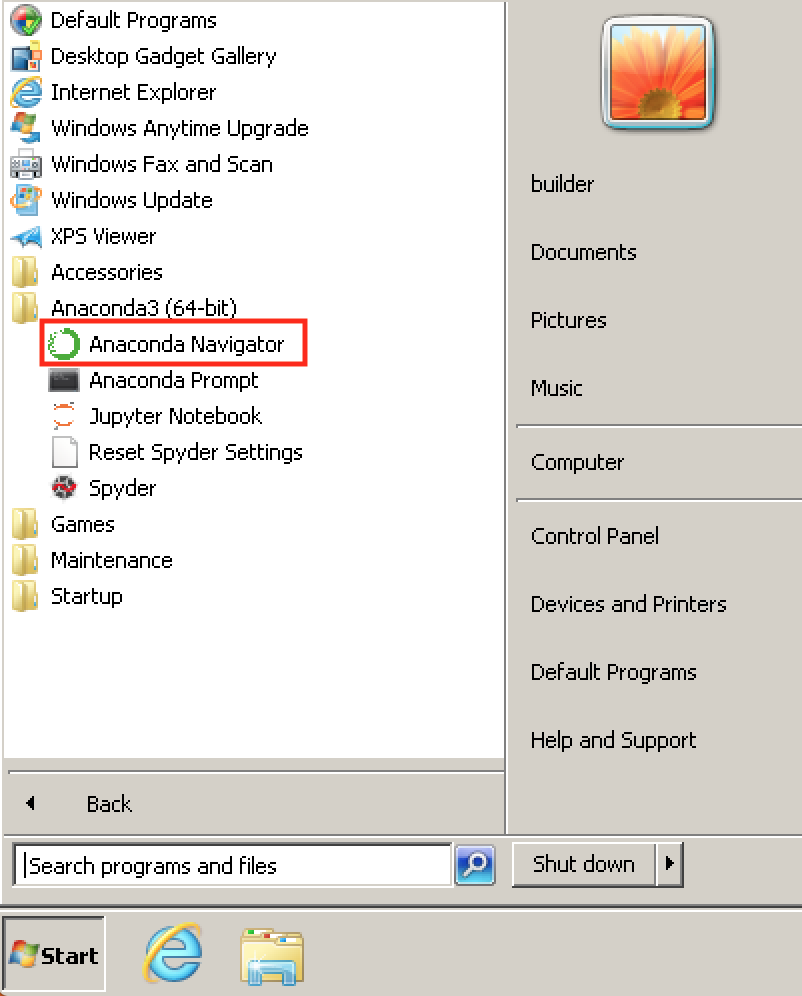
Anaconda Distribution contains **conda** and **Anaconda Navigator**, as well as Python and hundreds of scientific [packages](https://docs.anaconda.com/anaconda/packages/index.html). When you installed Anaconda, you installed all these too. You can try both conda and Navigator to see which is right for you to manage your packages and environments. You can even switch between them, and the work you do with one can be viewed in the other.

Now, try this simple programming exercise two ways, with Navigator and a terminal, to help you decide which approach is right for you.

**Write and run a Python program using Anaconda Navigator.**

**Windows**

From the Start menu, click the Anaconda Navigator desktop app or create a shortcut on your Desktop (look for the green circle)

*[](https://docs.anaconda.com/_images/navigator-anaconda-prompt.png)*

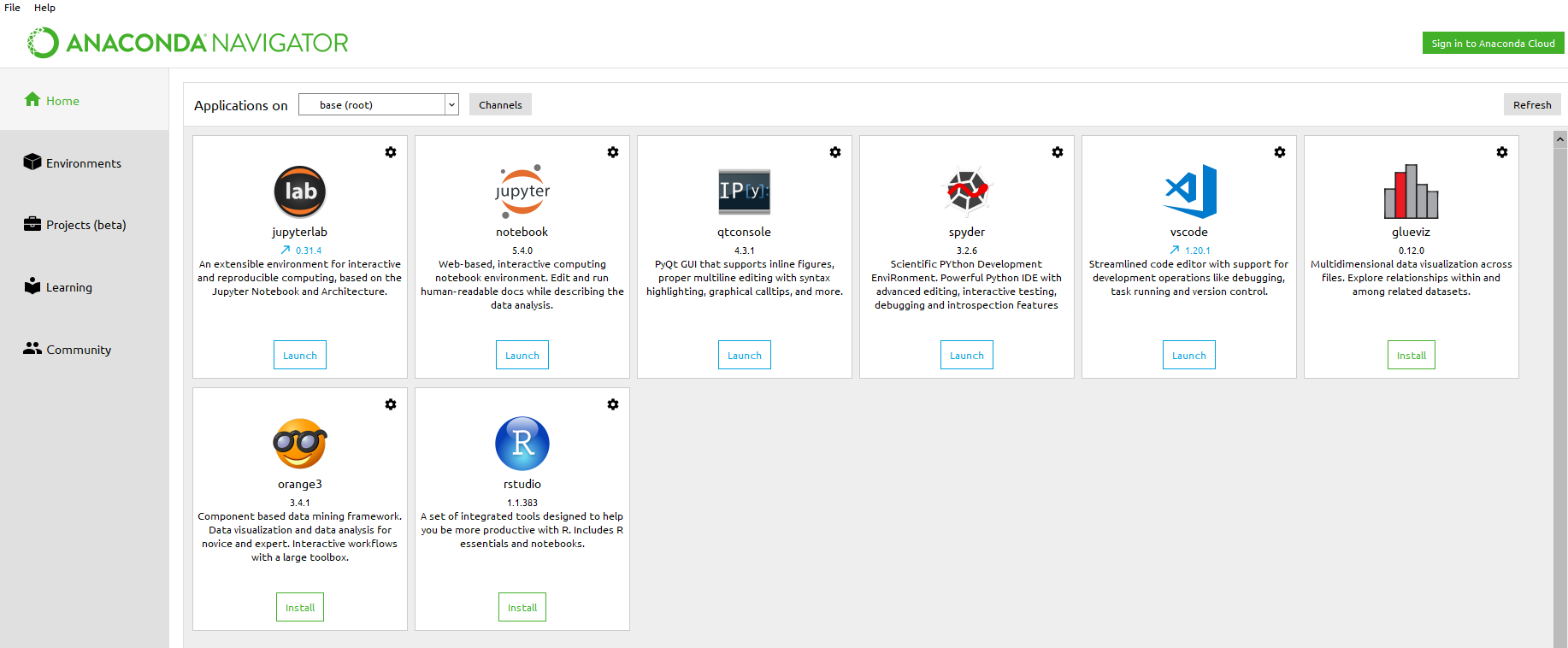
**MacOS**

Open Launchpad, then click the Anaconda-Navigator icon.

**Linux**

Open a Terminal window and type anaconda-navigator.

Your first screen should look something like below.



We will be primarily using Juptyer Notebooks and the Spyder IDE.

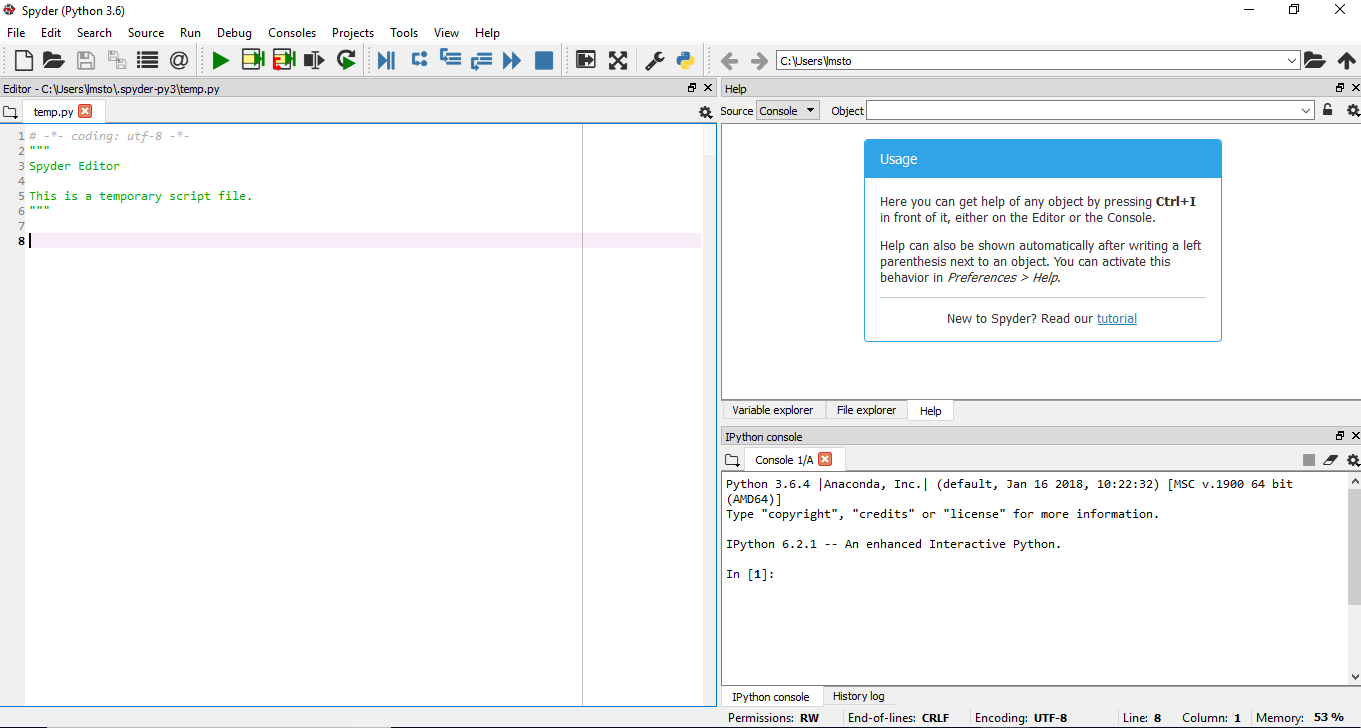
**Run Python in Spyder IDE (integrated development environment)**

On Navigator’s Home tab, in the Applications pane on the right, scroll to the Spyder tile and click the Install button to install Spyder.

NOTE: If you already have Spyder installed, you can jump right to the Launch step.

Launch Spyder by clicking Spyder’s Launch button.

**Note the warning regarding upgrading to the latest version of Spyder.**



In the new file on the left, delete any placeholder text, then type or copy/paste print("Hello Anaconda")

In the top menu, click File - Save As and name your new program hello.py

Run your new program by clicking the triangle Run button.

You can see your program’s output in the bottom right Console pane.

**Close Spyder**

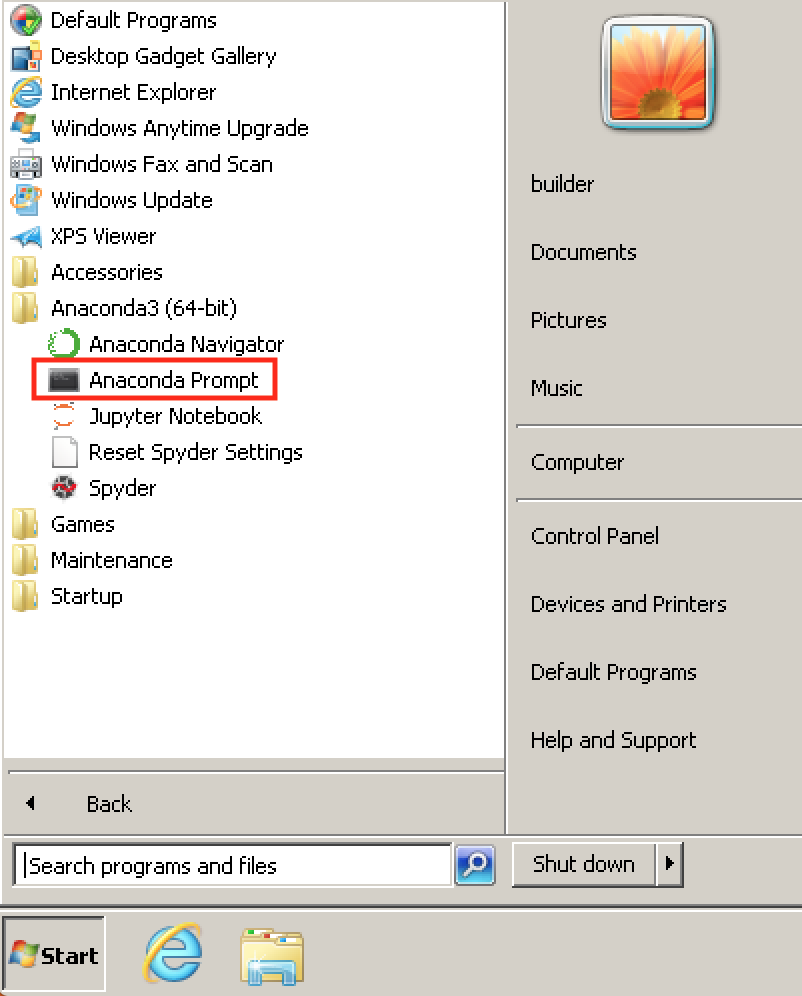
From Spyder’s top menu bar, select Spyder - Quit Spyder (In macOS, select Python - Quit Spyder).

**Next write a Python program using Anaconda Prompt or Terminal**

**1. Open Anaconda Prompt**

**Windows**

From the Start menu, search for and open “Anaconda Prompt”:

[](https://docs.anaconda.com/_images/anaconda-prompt.png)

**MacOS**

Open Launchpad, then click the Terminal icon.

**Linux**

Open a Terminal window.

**2. Start Python**

At Anaconda Prompt (Terminal on Linux or macOS), type python and press Enter.

The >>> means you are in Python.

**3. Write a Python program**

At the >>>, type print("Hello Anaconda!") and press Enter.

When you press enter, your program runs. The words “Hello Anaconda!” print to the screen. You’re programming in Python!

Next we will introduce Jupyter notebooks but before we begin we will download the latest version of a couple of packages from the Command Window which you have just opened.

First in the Command Prompt exit from your Python environment by entering exit() at the >>> prompt.

Next determine your version of conda which is used for installing and updating packages

conda --version

conda 4.4.10

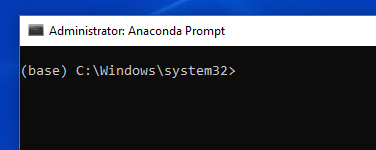
**To upgrade or install packages in conda you will probably need to restart Anaconda Prompt with admin privileges.**

**Exit existing Anaconda Prompt**

exit()

**then go back to Anaconda Prompt in your menu but Right Click on it and select More -> Run as Administrator** **and check Yes on pop-up screen**

Now you will a window called Administrator Ananconda Prompt – see below



While you are in Adminstrator, go ahead and upgrade conda entering the following after >

conda update -n base conda

And also Spyder

conda update spyder

And install 2 other packages required by Jupyter notebook plus one so that Spyder can open a Jupyter notebook, then exit

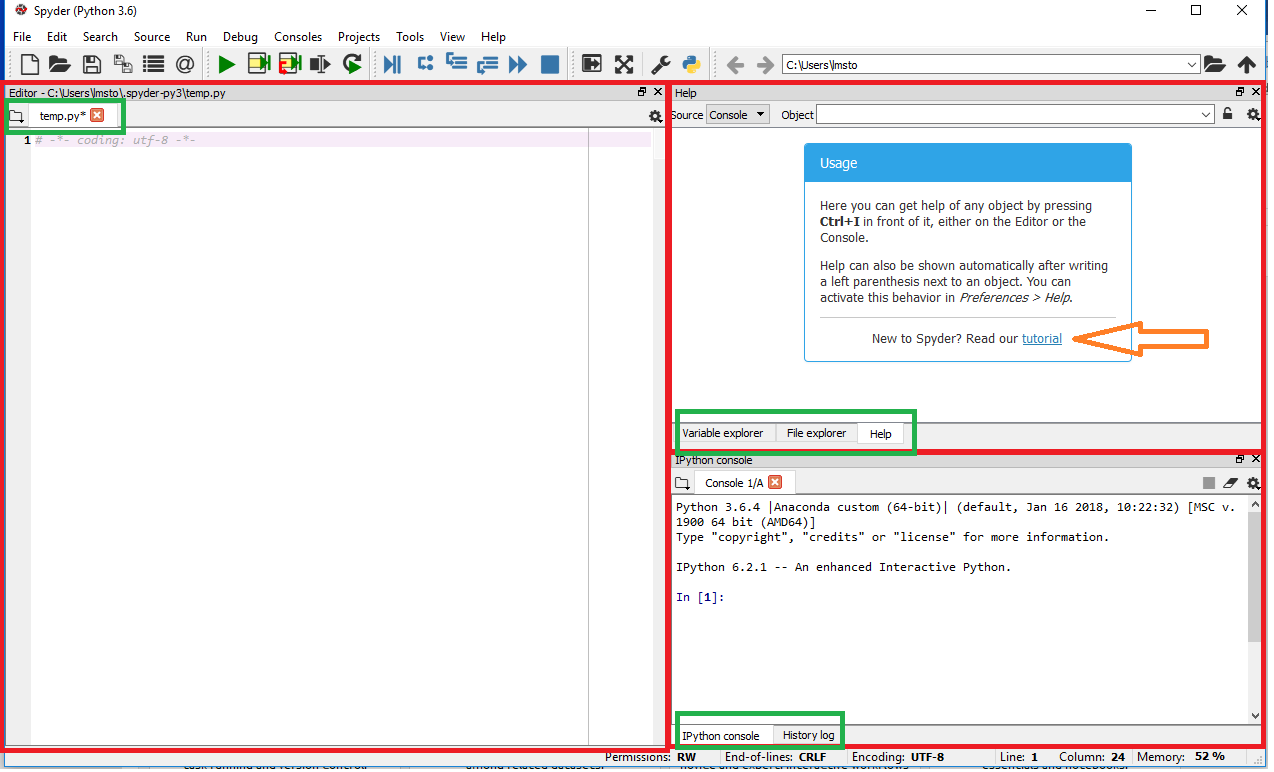
conda install nodejs

conda install -c cpcloud npm

conda install spyder-notebook -c spyder-ide

exit()

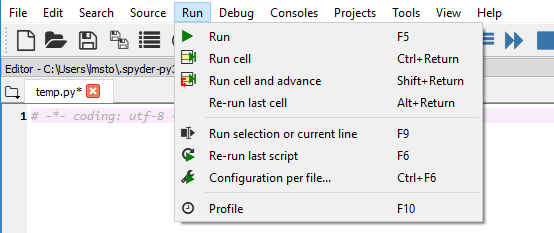
Now open up Anaconda Navigator and your updated verion of Spyder.



The first thing to note is how the Spyder app is organized. The application includes multiple separate windows (marked with red rectangles), each of which has its own tabs (marked with green rectangles). You can change which windows you prefer to have open from the **View -> Windows and Toolbars** option.

The **default configuration has the Editor, Object inspector/Variable explorer/File explorer, and Console/History log windows** open as shown above (except that the Variable explorer tab is missing in the screenshot).   
  
The **Console** (right) is where python is interactively (i.e., IPython)waiting for you to type commands, which tell it to load data, do math, plot data, etc. After every command is entered after the In [X] prompt, you need to hit the enter key (return key), and then python may or may not give some output.

The **Editor** (left) allows you to write sequences of commands, which together make up a program. Groups of them can be run from the selections in the top Run menu.



The **History Log** stores the last 100 commands you've typed into the Console.

**The Object inspector/Variable explorer/File explorer windows** are purely informational -- if you watch what the first two display as we go through the tutorial, you'll see that they can be quite helpful.   
  
**Entering Data**  
Type "x=5" in the Console -- this is the command to create a variable named x and give it the value 5. If you raise the "Variable explorer" tab you will see that x has been added to the list of variables in python's memory. You can also type "print x" or even just "x" in the Console to see the value of x. Now type "y=4" and then "x+y". Notice that this last command does not create a variable, although it does produce an output from the calculation.   
  
**Arrow Keys**  
If you use the arrow keys in the Console, you can bring back a previous command so that you can edit and re-execute it. Go back to the command "x+y" and change it to "junk=x+y". You've now created the variable junk. What can you type to see its value in the Console? 

Now click on the Tutorial indicated in the top right screen and complete it.

* 1. **PYTHON CHEATSHEETS**

Next download some Python cheatsheets that will be useful the rest of the semester from

<https://www.datacamp.com/community/data-science-cheatsheets>.

**The ones you will want include;**

* Importing Data: Python Cheat Sheet
* Python Data Visualization: Bokeh Cheat Sheet
* Jupyter Notebook Cheat Sheet
* Python Seaborn: Statistical Data Visualization
* Pandas Cheat Sheet: Data Wrangling in Python
* Matplotlib Cheat Sheet: Plotting in Python
* SciPy Cheat Sheet: Linear Algebra in Python
* NumPy Cheat Sheet: Data Analysis in Python
* Scikit-Learn Cheat Sheet: Python Machine Learning

**Advanced Sheets:**

* PySpark Cheat Sheet: Spark DataFrames in Python
* PySpark Cheat Sheet: Spark in Python
* Keras Cheat Sheet: Neural Networks in Python

**An additional one from**

<https://www.datacamp.com/community/tutorials/python-data-science-cheat-sheet-basics>

* 1. **JUPYTER NOTEBOOKS**

**Next we will introduce Jupyter notebooks.**

<https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/>

Jupyter/IPython Notebook Quick Start Guide

This document is a brief step-by-step tutorial on installing and running Jupyter (IPython) notebooks on local computer for new users who have no familiarity with python.

Briefly, if someone gave you a notebook to run and you don’t know what a notebook is, this document is for you.

**Jupyter Notebook App** (formerly **IPython Notebook**) is an application running inside the browser. This guide describes how to install and use Jupyter Notebook App as normal desktop application, without using any remote server.

For other use-cases, please refer to the [Official Jupyter Documentation](http://jupyter.readthedocs.org/).

[**What is the Jupyter Notebook?**](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#id4)

In this page briefly introduce the main components of the Jupyter Notebook environment. For a more complete overview see [References](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#references).

[**Notebook document**](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#id5)

Notebook documents (or “notebooks”, all lower case) are documents produced by the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app), which contain both computer code (e.g. python) and rich text elements (paragraph, equations, figures, links, etc…). Notebook documents are both human-readable documents containing the analysis description and the results (figures, tables, etc..) as well as executable documents which can be run to perform data analysis.

**Jupyter Notebook App**

The Jupyter Notebook App is a server-client application that allows editing and running [notebook documents](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#notebook-document) via a web browser. The Jupyter Notebook App can be executed on a local desktop requiring no internet access (as described in this document) or can be installed on a remote server and accessed through the internet.

In addition to displaying/editing/running notebook documents, the Jupyter Notebook App has a “Dashboard” ([Notebook Dashboard](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#dashboard)), a “control panel” showing local files and allowing to open notebook documents or shutting down their [kernels](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#kernel).

A notebook kernel is a “computational engine” that executes the code contained in a [Notebook document](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#notebook-document). The ipython kernel, referenced in this guide, executes python code. Kernels for many other languages exist ([official kernels](http://jupyter.readthedocs.org/en/latest/#kernels)).

When you open a [Notebook document](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#notebook-document), the associated kernel is automatically launched. When the notebook is executed (either cell-by-cell or with menu Cell -> Run All), the kernel performs the computation and produces the results. Depending on the type of computations, the kernel may consume significant CPU and RAM. Note that the RAM is not released until the kernel is shut-down.

[**Notebook Dashboard**](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#id8)

The Notebook Dashboard is the component which is shown first when you launch [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app). The Notebook Dashboard is mainly used to open [notebook documents](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#notebook-document), and to manage the running [kernels](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#kernel) (visualize and shutdown).

The Notebook Dashboard has other features similar to a file manager, namely navigating folders and renaming/deleting files.

**Running the Jupyter Notebook**

**Launching Jupyter Notebook App**

The [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app) can be launched by clicking on the Jupyter Notebook icon installed by Anaconda in the start menu (Windows) or by typing in a terminal (cmd on Windows):

jupyter notebook

This will launch a new browser window (or a new tab) showing the [Notebook Dashboard](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#dashboard), a sort of control panel that allows (among other things) to select which notebook to open.

When started, the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app) can access only files within its start-up folder (including any sub-folder). No configuration is necessary if you place your notebooks in your home folder or subfolders. Otherwise, you need to choose a [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app) start-up folder which will contain all the notebooks.

**Change Jupyter Notebook startup folder (Windows)**

Copy the Jupyter Notebook launcher from the menu to the desktop.

Right click on the new launcher and change the Target field, change %USERPROFILE% to the full path of the folder which will contain all the notebooks.

Double-click on the Jupyter Notebook desktop launcher (icon shows [IPy]) to start the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app). The notebook interface will appear in a new browser window or tab. A secondary terminal window (used only for error logging and for shut down) will be also opened.

**Change Jupyter Notebook startup folder (Mac OS)**

To launch [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app):

Click on spotlight, type terminal to open a terminal window.

Enter the startup folder by typing cd /some\_folder\_name.

Type jupyter notebook to launch the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app) The notebook interface will appear in a new browser window or tab.

**Shut down the Jupyter Notebook App**

Closing the browser (or the tab) **will not close** the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app). To completely shut it down you need to **close the associated terminal**.

In more detail, the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#notebook-app) is a server that appears in your browser at a default address (http://localhost:8888). Closing the browser will not shut down the server. You can reopen the previous address and the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app) will be redisplayed.

You can run many copies of the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app) and they will show up at a similar address (only the number after “:”, which is the port, will increment for each new copy). Since with a single [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app) you can already open many notebooks, we do not recommend running multiple copies of [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app).

**Close a notebook: kernel shut down**

When a notebook is opened, its “computational engine” (called the [kernel](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#kernel)) is automatically started. Closing the notebook browser tab, will not shut down the [kernel](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#kernel), instead the kernel will keep running until is explicitly shut down.

To shut down a kernel, go to the associated notebook and click on menu File -> Close and Halt. Alternatively, the [Notebook Dashboard](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#dashboard) has a tab named Running that shows all the running notebooks (i.e. kernels) and allows shutting them down (by clicking on a Shutdown button).

**Executing a notebook**

Download the notebook you want to execute and put it in your notebook folder (or a sub-folder of it).

Then follow these steps:

* Launch the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app) (see [previous section](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/execute.html#launching-notebook)).
* In the [Notebook Dashboard](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#dashboard) navigate to find the notebook: clicking on its name will open it in a new browser tab.
* Click on the menu Help -> User Interface Tour for an overview of the [Jupyter Notebook App](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html" \l "notebook-app)user interface.
* You can run the notebook document step-by-step (one cell a time) by pressing shift + enter.
* You can run the whole notebook in a single step by clicking on the menu Cell -> Run All.
* To restart the [kernel](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#kernel) (i.e. the computational engine), click on the menu Kernel -> Restart. This can be useful to start over a computation from scratch (e.g. variables are deleted, open files are closed, etc…).

**Note**

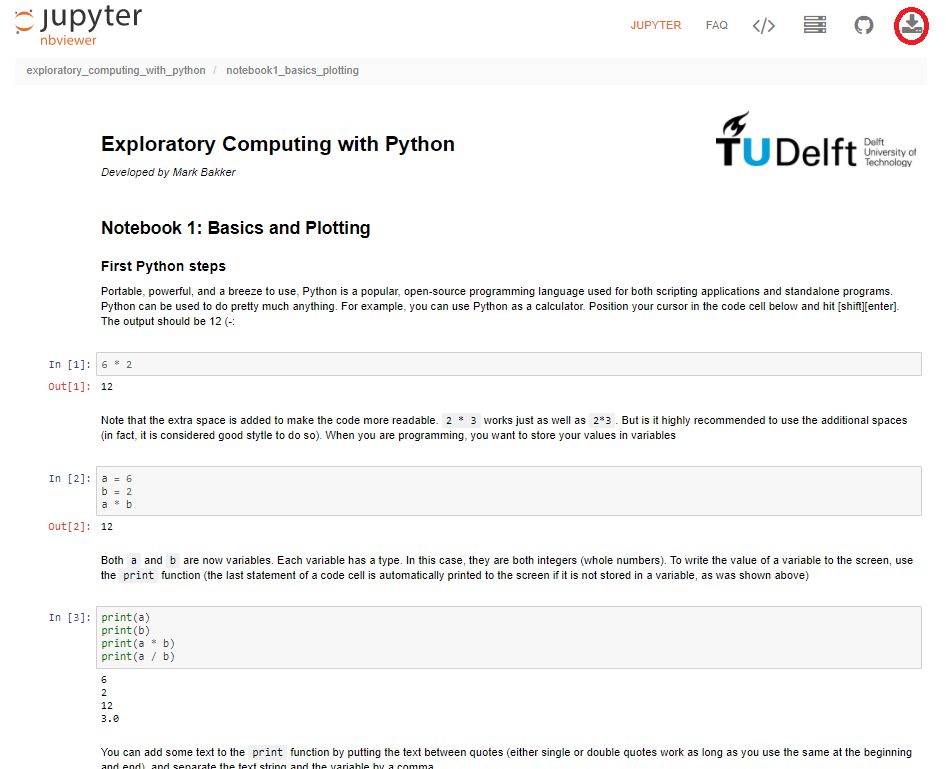
**Save notebooks**: modifications to the notebooks are automatically saved every few minutes. To avoid modifying the original notebook, make a copy of the notebook document (menu File -> Make a copy …) and save the modifications on the copy.

**Warning**

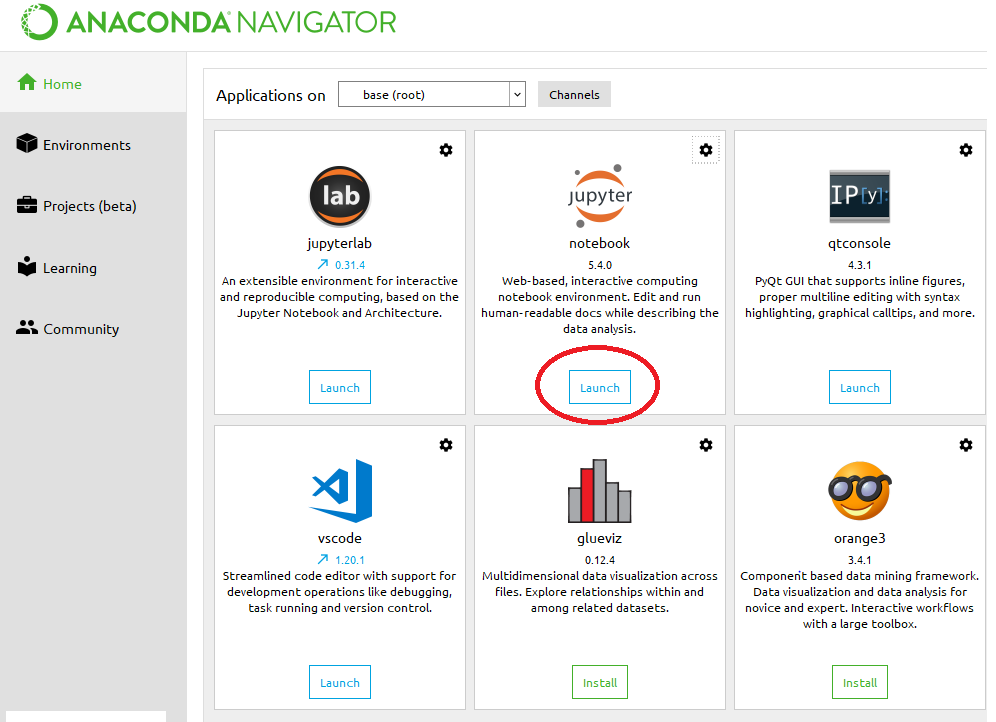
Pay attention to not open the **same** notebook document on **many tabs**: edits on different tabs can overwrite each other! To be safe, make sure you open each notebook document in only one tab. If you accidentally open a notebook twice in two different tabs, just close one of the tabs.

**Example**

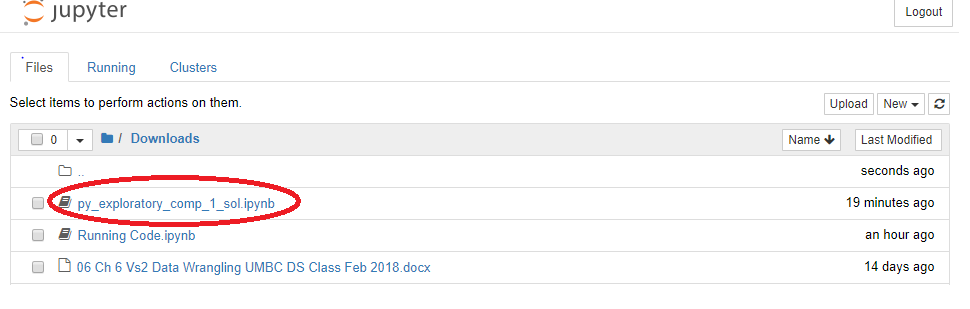
* Go to the following website <https://mbakker7.github.io/exploratory_computing_with_python/>
* Open the first example: Exploratory Computing with Python and you should see the following:



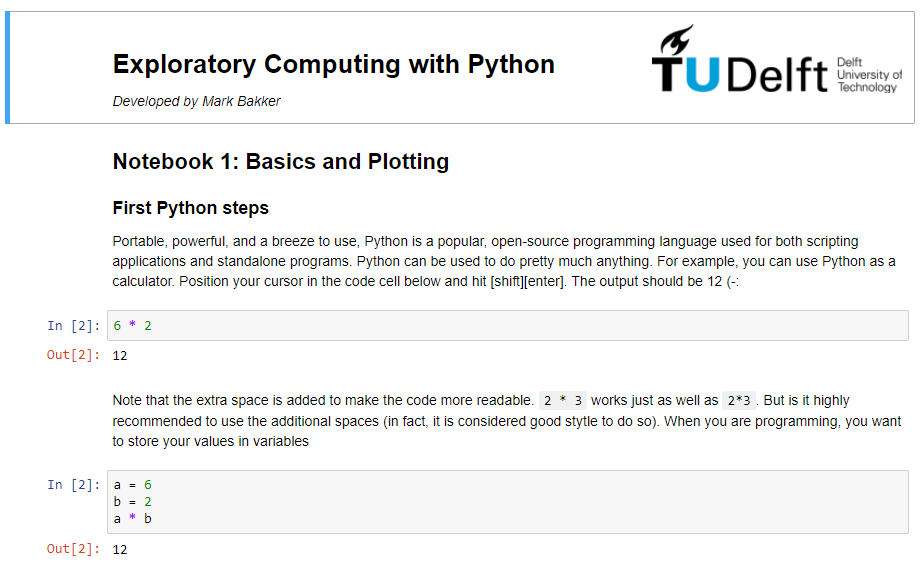
* Then select the top right icon (circled in red) and save to your default folder (usually Download in Windows).
* Open up Anaconda Navigator and select Jupyter (Launch)



* You should see a view of your laptop working space and navigate to the location of the ipynb file you just downloaded and select it



You should end up with a working Jupyter notebook



You can run each of the In [X] blocks of code or copy and paste them into Spyder in the interactive Console.

**END CHAPTER 7**