

1 ESS245 Computational Geology: Interacting with Jupyter 2

Since Jupyter notebooks will be the primary tool we use in this course, and the dominant format for your assignment submissions, we will explore their use in greater detail in the following.

A Jupyter notebook has three principal elements see Fig.1

1. Text areas (often referred to as cells). You can edit these text cells.
2. Code blocks (or code cells). These contain computer code that can be executed. You can edit and execute these cells.
3. Result Blocks, which contain the output produced by a code block. Results blocks cannot be edited. However, you can select and copy them.

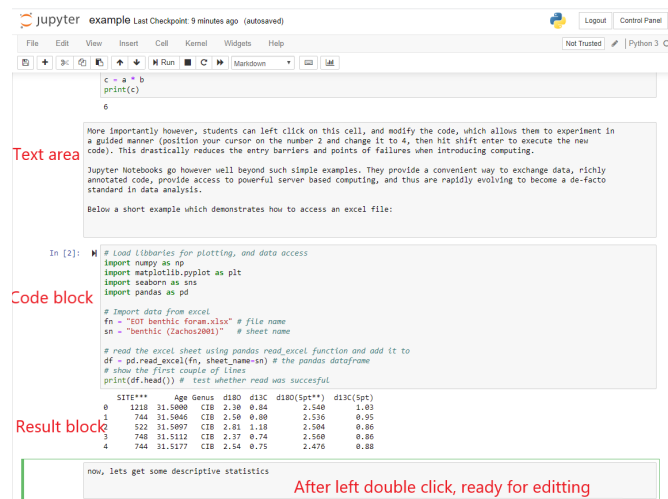


Figure 1: An example of a Jupyter notebook (.ipynb file) including text areas, code blocks and result blocks. After double left click on the cell, the cell will be ready for editing (and the bar will change to green)

If you are still reading this as a pdf file, please switch to the notebook version of this very text by following this link

<https://utoronto.syzygy.ca/jupyter/hub/user-redirect/git-pull?repo=https://github.com/ulw/PNTA-Notebooks&urlpath=lab/tree/PNTA-Notebooks/>

[Interacting_with_Jupyter/Interacting_with_jupyter-2.ipynb&branch=main](#)

If you do a single left click with your mouse on this text, you will see that this cell is now drawn with a blue border. This indicates that the cell has been selected. (see Fig.2).

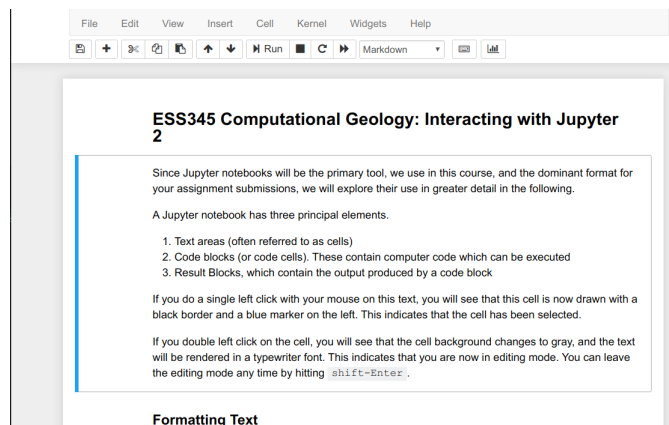


Figure 2: A single click will select a notebook cell, which is indicated by the blue border to the left.

If you double-click on this cell, you will see that the cell background changes to gray, rendering the text in a typewriter font. This indicates that you are now in editing mode. You can leave the editing mode any time by hitting **Shift-Enter**.

Also, most of the usual keyboard shortcuts will work:

- Ctrl-z = undo
- Ctrl-c = copy
- Ctrl-x = cut
- Ctrl-v = insert
- Ctrl-s = save

1.1 Potential Problems

Sometimes, you may lose the connection to the Jupyter server. This is indicated by the red **Not Connected** icon in the status bar (see Fig.4). This

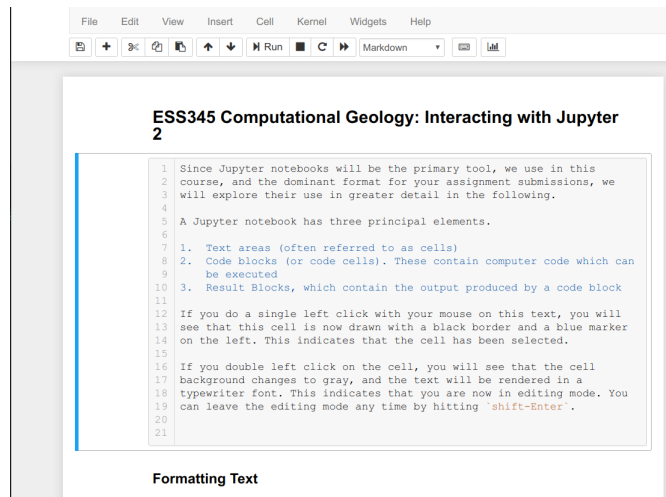


Figure 3: A double click on any cell it will activate the editing mode. You can leave the editing mode any time by hitting **shift-Enter**.

is usually caused by a firewall rule. Please speak to a TA or the instructor to resolve this issue.

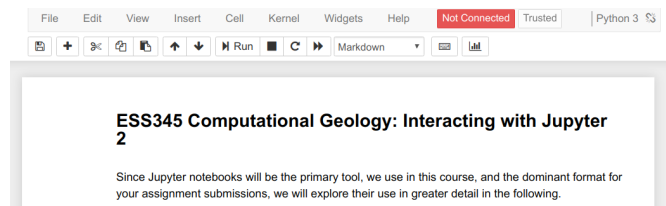


Figure 4: The red **not connected** icon indicates that you lost connection to your Jupyter Server. This is usually caused by a firewall rule. Please speak to a TA or the instructor to resolve this issue.

1.2 Formatting Text

Before exploring how-to-use Jupyter notebooks to run python code, let's explore how to format the text in the text cells. Create a new notebook in your `my_stuff` folder. You can do this via the drop-down list called **New** in the upper right corner, and then select **New Notebook**, and rename the new file. These steps should now pose no problem to you. If they do, please

speak up!

Once your new notebook is open, it will look like Fig. 5. Notice the text `In []:` on the left side. This tells you that this is a code cell that expects python code. So before we start playing with text, we need to change the cell type (see Fig. 6)

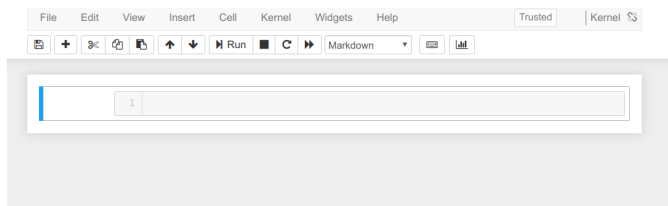


Figure 5: The default view when opening a new notebook. Note the text `In []:` on the left side. This tells you that this is a code cell that expects python code.

Use the dropdown menu to change the cell type to **markdown**.

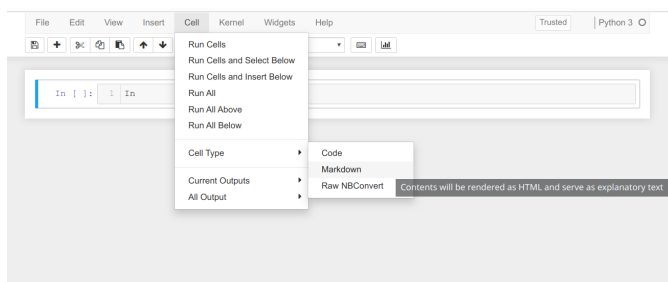


Figure 6: Use the dropdown menu to change the cell type to **markdown**

Once you have changed the cell to markdown, you will note that the `In []:` text on the left is now gone.

Next, copy the following lines into your new notebook cell:

1.3 This is a second-level heading

This is **bold** text

1. This is a simple list

- First list entry
 - Second list entry
2. This is the 3rd entry in the enumerated list and has no other purpose than just being the 3rd entry.

You will notice that the formatting has been lost. Go back to these instructions, activate editing mode, and explore how to create the formatting you see (see Fig.7) . Now go to your new notebook, and apply the `mark down` syntax in your own notebook. . Remember that you can leave the edit mode any time by hitting **Shift+Enter**.

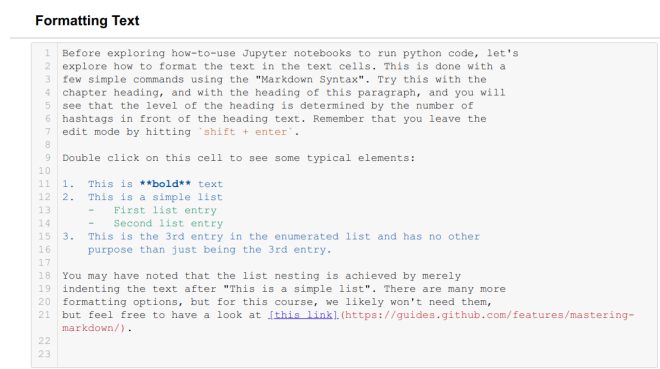


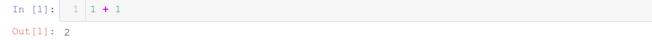
Figure 7: Here, you can see how to format your text with `mark down` syntax

You may have noted that the list nesting is achieved by indenting the text after "This is a simple list". There are many more formatting options, but for this course, we likely won't need them, but feel free to have a look at [this link](https://guides.github.com/features/mastering-markdown/).

Last but not least save your edits by clicking on the floppy disk icon in the upper left corner. This will create a checkpoint so that you can return to this version at any time!

1.4 Adding Code cells

Now let's try adding a code cell below the text cell and enter a trivial statement like `1 + 1` and hit **Shift+Enter** and you should see the result displayed below the code cell (hopefully, it is 2). Go ahead and edit your code cell (e.g., `1+3`) and hit **Shift+Enter** again. The result should change accordingly (see Fig.8)



```
In [1]: 1 + 1
Out[1]: 2
```

Figure 8: Example of a trivial python statement

Note that your edits are not auto-saved! You need to explicitly save your notebooks File-> Save All" The

1.5 Downloading your notebook

All of your assignments will have to be submitted on Quercus. In order to mark your assignments, you need to submit a **pdf copy**, as well as the **actual notebook file**. For the pdf-copy, use File -> Print and select save as pdf. For the notebook, use File -> Download

1.6 Recap

In this module, you learned how-to:

1. Create a notebook
2. Save a notebook
3. Download a notebook
4. Create text cells
5. Edit and format text cells
6. Create code cells
7. Execute code in a code cell
8. Add a new code cell