

# Jupyter Notebook Tutorials

## Create a working folder in the Jupyter Notebook

By default the Jupyter notebook will open in your current directory. However, you can create an other working folder for your notebooks.

The screenshot displays the Jupyter Notebook web interface. At the top, the 'jupyter' logo and a 'Logout' button are visible. Below the logo, there are tabs for 'Files', 'Running', and 'Clusters'. The 'Files' tab is active, showing a list of files and folders. A red circle labeled '1)' highlights the 'New' button in the top right corner. A dropdown menu is open, showing options: 'Notebook: Python 3', 'Other: Text File', 'Folder', and 'Terminal'. A red circle labeled '2)' highlights the 'Folder' option. Below the file list, there is a section for 'Untitled Folder' with a checkbox and a folder icon. At the bottom, there are buttons for 'Rename', 'Move', and a trash icon. A red circle highlights the 'Rename' button. A modal dialog titled 'Rename directory' is open, showing a text input field with the text 'Working Folder' and buttons for 'Cancel' and 'Rename'.

# Jupyter Notebook Tutorials

## Create a new Jupyter Notebook

The screenshot displays the Jupyter Notebook web interface. At the top, the Jupyter logo is on the left, and a 'Logout' button is on the right. Below the logo, there are tabs for 'Files', 'Running', and 'Clusters'. A message states 'Select items to perform actions on them.' Below this, a file browser shows a 'Working Folder' with a dropdown menu set to '0'. A 'New' button is visible, which has opened a dropdown menu showing 'Notebook: Python 3' and 'Other:'. In the foreground, a 'Rename Notebook' dialog box is open, prompting the user to 'Enter a new notebook name:'. The text 'Untitled' is entered in the input field. The dialog box has 'Cancel' and 'Rename' buttons at the bottom right. The background shows the Jupyter Notebook editor with a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with various icons. The notebook title is 'Untitled' and it indicates 'Last Checkpoint: a few seconds ago (unsaved changes)'. The kernel is set to 'Python 3'.

# Jupyter Notebook Tutorials

## Working with Markdown Cells

Select a **cell** in an open Jupyter Notebook  
In the toolbar select “Markdown” from the cell type dropdown



### Create Headers Cells

# Markdown Cell H1  
## Markdown Cell H2  
### Markdown Cell H3  
#### Markdown Cell H4

### Markdown Cell H1

### Markdown Cell H2

### Markdown Cell H3

Markdown Cell H4

### Un-numbered Lists

- un-numbered list item 1
- un-numbered list item 2

- un-numbered list item 1
- un-numbered list item 2

### Numbered List

1. numbered list item 1
2. numbered list item 2

1. numbered list item 1
2. numbered list item 2

### Sub-Numbered List

1. numbered list item 1
  1. sub numbered list
  2. sub numbered list
2. numbered list item 2

1. numbered list item 1
  - A. sub numbered list
  - B. sub numbered list
2. numbered list item 2

# Jupyter Notebook Tutorials

## Working with Markdown Cells

### Formatted Text

*\*italic text\**

**\*\*bold text\*\***

~~strikethrough text~~

*italic text*

**bold text**

~~strikethrough text~~

### Links

[Link Example](http://google.com)

[Link Example](#)

### Local and Online Images

![local image](images/logo.png)

### Code Examples

```
```python
s = "Python Programming"
print s
```
```

```
s = "Python Programming"
print s
```

### Tables

|      |      |      |
|------|------|------|
| Col1 | Col2 | Col3 |
| ---  | ---  | ---  |
| 1234 | 5678 | 9124 |
| abcd | efgh | ijkl |

| Col1 | Col2 | Col3 |
|------|------|------|
| 1234 | 5678 | 9124 |
| abcd | efgh | ijkl |

### HTML

```
<strong>Hello</strong>
<hr>
```

**Hello**

# Jupyter Notebook Tutorials

## Working with Markdown Cells

### Formatted Text

*\*italic text\**

**\*\*bold text\*\***

~~strikethrough text~~

*italic text*

**bold text**

~~strikethrough text~~

### Links

[Link Example](http://google.com)

[Link Example](#)

### Local and Online Images

![local image](images/logo.png)

### Code Examples

```
```python
s = "Python Programming"
print s
```
```

```
s = "Python Programming"
print s
```

### Tables

|      |      |      |
|------|------|------|
| Col1 | Col2 | Col3 |
| ---  | ---  | ---  |
| 1234 | 5678 | 9124 |
| abcd | efgh | ijkl |

| Col1 | Col2 | Col3 |
|------|------|------|
| 1234 | 5678 | 9124 |
| abcd | efgh | ijkl |

### HTML

```
<strong>Hello</strong>
<hr>
```

**Hello**

# Jupyter Notebook Tutorials

## Working with Code Cells

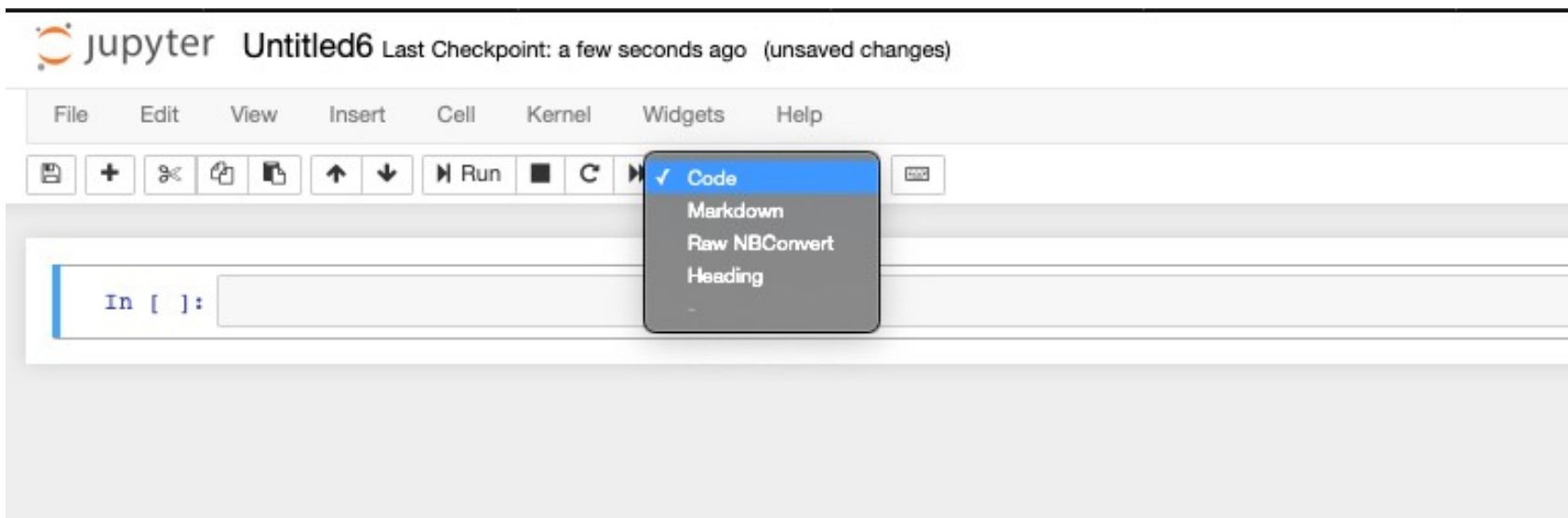
Code Cells have 2 modes:

**Edit Mode:** Edit mode is the mode that allows you to edit the code in the cell. When in edit mode keyboard input is interpreted as cell content. Edit mode can be identified by the green outline on the cell. To enter command mode click in a cell.

**Command Mode:** Command mode is the mode where you can execute keyboard shortcuts. Keyboard input is interpreted as keyboard shortcuts. **To enter command mode click outside the cells or press the escape key.**

### Create a code cell

- 1 Select a **cell** in an open Jupyter Notebook
- 2 In the toolbar select “**Code**” from the cell type dropdown



# Jupyter Notebook Tutorials

## Jupyter Notebook Keyboard Shortcuts

View all available keyboard shortcuts

To view the entire list of keyboard shortcuts by going into Command Mode (Press Escape) and then pressing the “H” key.

### Some useful Command mode keyboard shortcuts

“A” : Insert cell above  
“B” : Insert cell below  
“C” : Copy selected cell  
“X” : Cut selected cell  
“S” : Save and Checkpoint  
Shift + Enter : Run current cell and select cell below”  
“M” : Change cell to Markdown  
“Y” : Change cell to Code

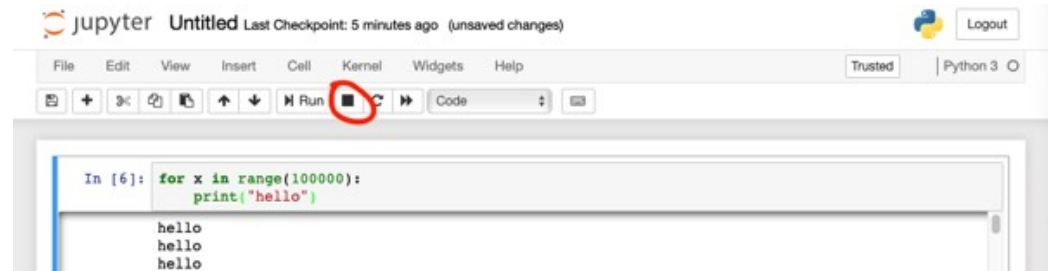
### Some useful Edit mode keyboard shortcuts

Command/Control ] : Indent  
Command /Control [ : Decent  
Command/Control / : Comment  
Command/Control A : Select All  
Command/Control up : Go to cell Start

# Jupyter Notebook Tutorials

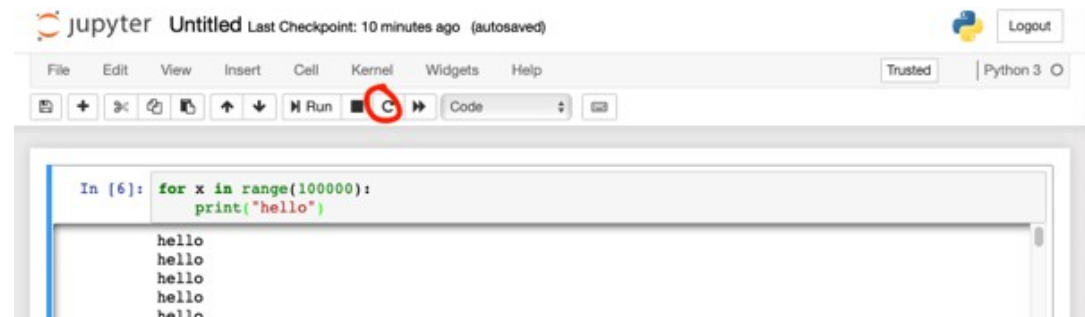
## How to Interrupt the Kernel (Stop code from running)

To stop code from running press the STOP button



## How to reset the Kernel (Clear all in memory objects and stop the code)

To reset the kernel press the reset button



## How to hide a output

Double click on the left of result

## How to remove a output

Esc + R + Y

## How to remove all output

select the Cell -> All Output -> Clear menu item.



# Jupyter Notebook Tutorials

## Using Magic Commands

Magic Commands are powerful shortcuts exposed by the kernel that allow you to solve many common problems.

List all available magic commands

Use `%lsmagic`

```
In [6]: %lsmagic
Out[6]: Available line magics:
%alias %alias_magic %autocall %automagic %autosave %bookmark %cat %cd %clear %colors %config %connect_inf
o %tcp %debug %dhist %dirs %doctest_mode %ed %edit %env %gui %hist %history %killbgscripts %ldir %less
%lf %lk %ll %load %load_ext %loadpy %logoff %logon %logstart %logstate %logstop %ls %lsmagic %lx %macr
o %magic %man %matplotlib %mkdir %more %mv %notebook %page %pastebin %pdb %pdef %pdoc %pfile %pinfo %
pinfo2 %popd %pprint %precision %profile %prun %psearch %psource %pushd %pwd %pycat %pylab %qtconsole %
quickref %recall %rehashx %reload_ext %rep %rerun %reset %reset_selective %rm %rmdir %run %save %sc %se
t_env %store %sx %system %tb %time %timeit %unalias %unload_ext %who %who_ls %whos %xdel %xmode

Available cell magics:
%%! %%HTML %%SVG %%bash %%capture %%debug %%file %%html %%javascript %%js %%latex %%markdown %%perl %%p
run %%pypy %%python %%python2 %%python3 %%ruby %%script %%sh %%svg %%sx %%system %%time %%timeit %%writ
efile
```

Run External Code using the `%run` magic command

Create a python file outside of the Jupyter Notebook.

To run an external file input the command `%run <python file name>`

Press shift + enter

```
timesfive.py x
1 def timesfive(x):
2     """multiply by 5"""
3     return x * 5
4
5 for N in range(1, 5):
6     print(N, "Times five is", timesfive(N))
```

```
File Edit View Insert Cell Kernel Widgets Help
+ -> <-> Run C >> Code
In [12]: %run timesfive.py
1 Times five is 5
2 Times five is 10
3 Times five is 15
4 Times five is 20
In [ ]:
```

# Jupyter Notebook Tutorials

## Using Magic Commands

Magic Commands are powerful shortcuts exposed by the kernel that allow you to solve many common problems.

To use the shell use the `%system` magic command

```
In [21]: %system date
Out[21]: ['Mon Mar 11 22:54:49 CDT 2019']
```

```
In [28]: %system pwd
Out[28]: ['/Users/henrypalma']
```

```
In [23]: %system ls
Out[23]: ['2D Game',
          'AdditionTutor',
          'AdditionTutorIOS',
```

Measure the execution time of code using `%timeit` magic command

To measure a single line of code use `%timeit` magic command before the line of code you want to measure.  
To measure multiple lines of code use the `%%timeit` magic command before the lines of code you want to measure.

```
In [18]: %timeit TestData = [n ** 5 for n in range(1000)]
294 µs ± 10.4 µs per loop (mean ± std. dev. of 7 runs, 1000 loops each)
```

```
In [20]: %%timeit
TestData = []
for n in range(1000):
    TestData.append(n ** 5)
333 µs ± 16.4 µs per loop (mean ± std. dev. of 7 runs, 1000 loops each)
```

# Jupyter Notebook Tutorials

## Triks

Convert your notebook in html, under a terminal type :

```
jupyter nbconvert --to html notebook.ipynb
```

Or from menu of jupyter : File/Downloads/html

Convert your notebook in pdf :

```
jupyter nbconvert --to pdf notebook.ipynb
```

Also pass the --execute flag to generate the output cells :

```
jupyter nbconvert --execute --to html notebook.ipynb  
jupyter nbconvert --execute --to pdf notebook.ipynb
```

To insert a youtube video ( in a Code cell ):

```
from IPython.display import YouTubeVideo  
YouTubeVideo("gKQvQG8FwQk",560,315,rel=0)
```