

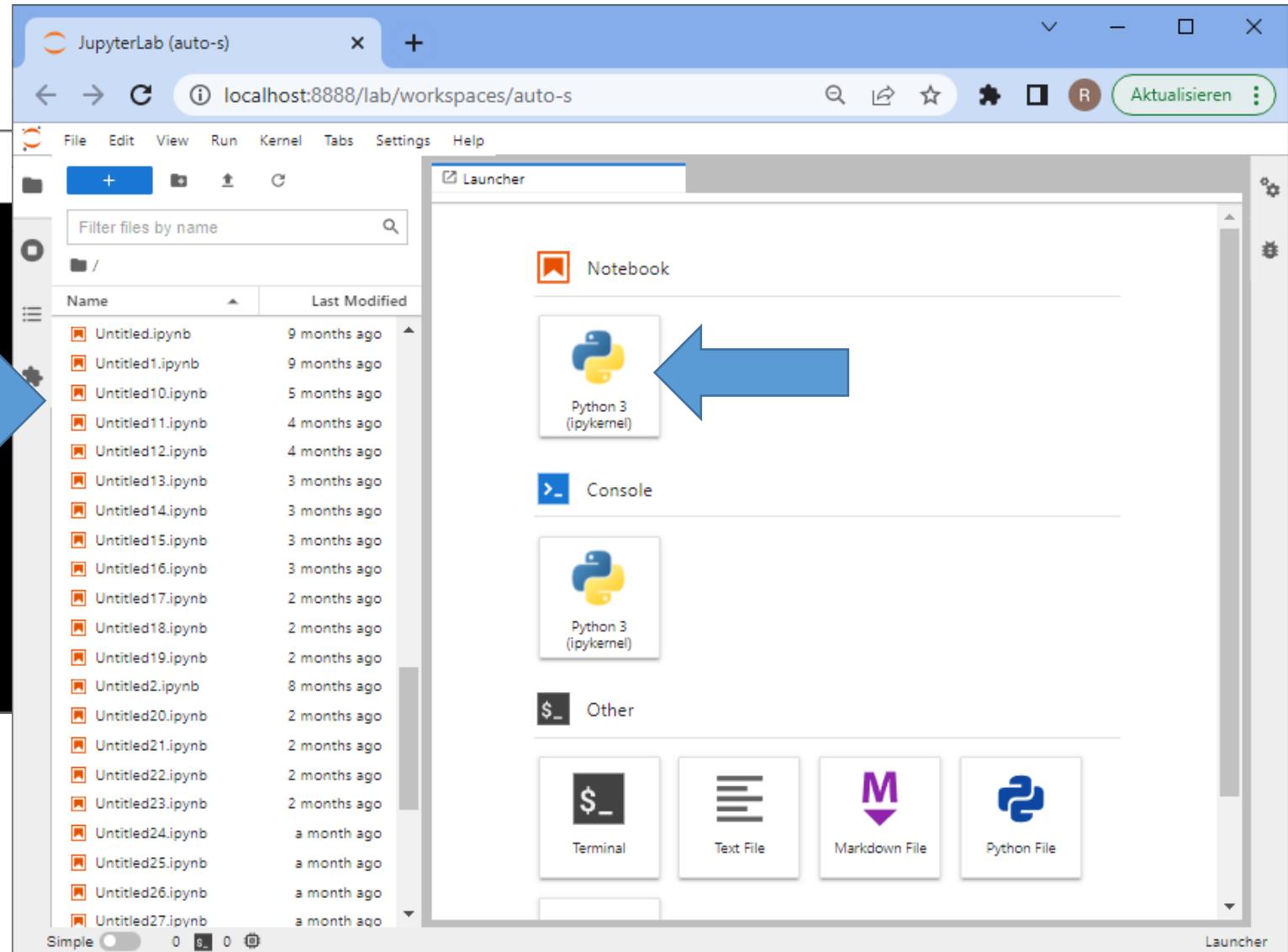


Robert Haase

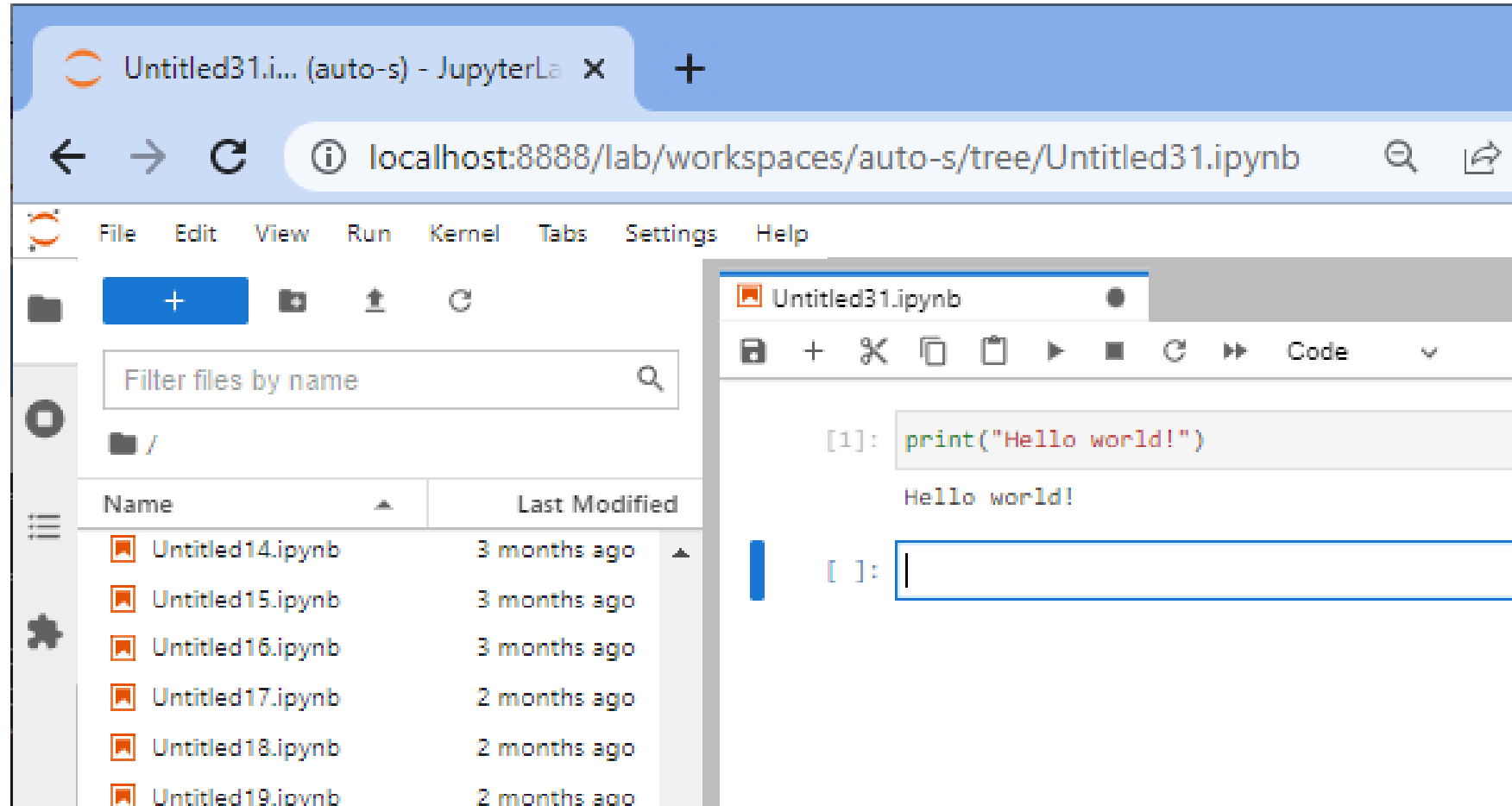
September 2022

- Our programming environment for this course

```
C:\Users\rober>conda deactivate - cond...  
  
c:\Users\rober>conda activate bio_39  
(bio_39) c:\Users\rober>jupyter lab
```



- Execute code cell-by-cell and see results instantaneously

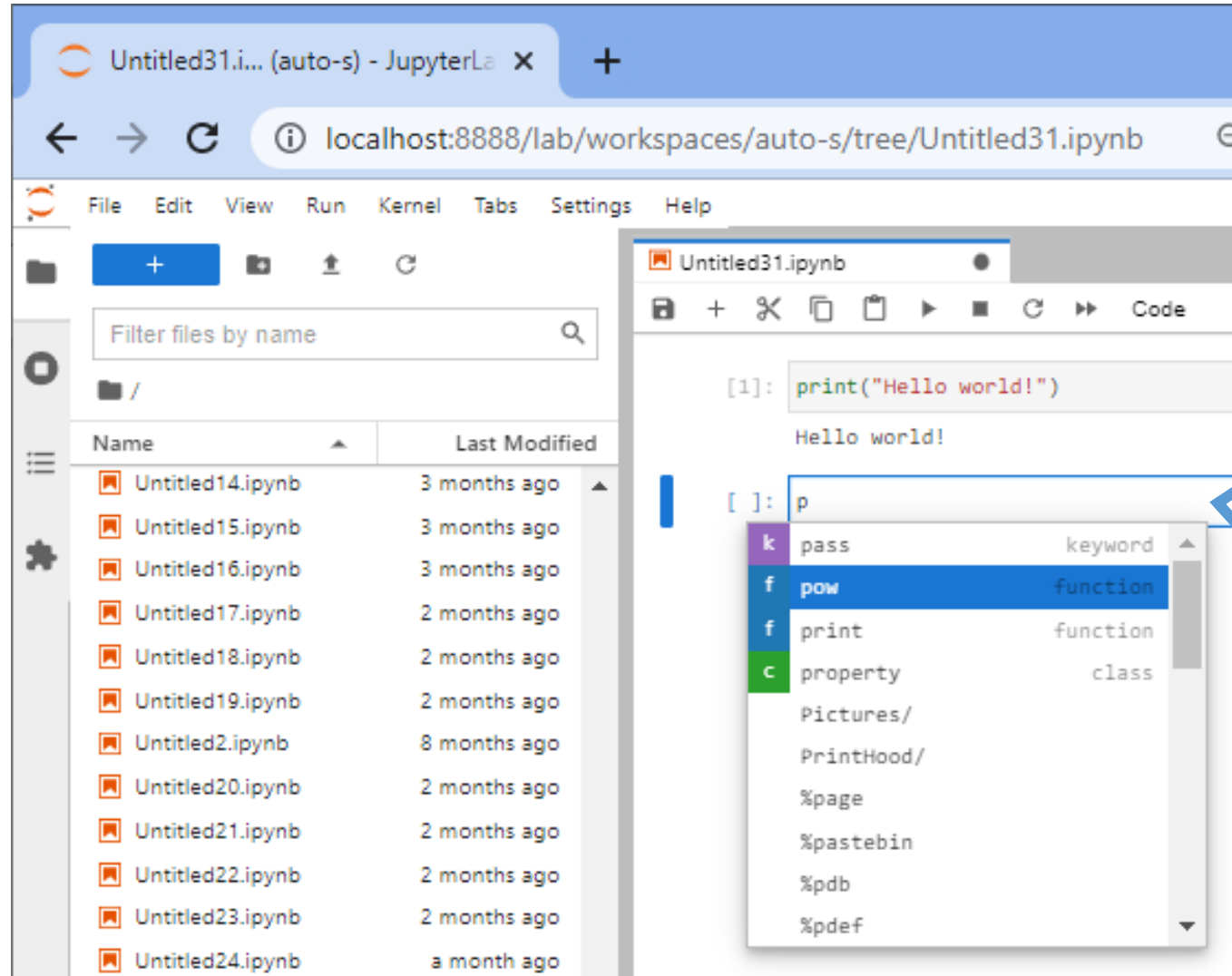


The screenshot shows the JupyterLab web interface. The top bar indicates the current workspace is 'Untitled31.i... (auto-s) - JupyterLa'. The browser address bar shows 'localhost:8888/lab/workspaces/auto-s/tree/Untitled31.ipynb'. The left sidebar contains a file browser with a search bar and a list of files. The main area displays the 'Untitled31.ipynb' file, which is a code cell containing the following code and output:

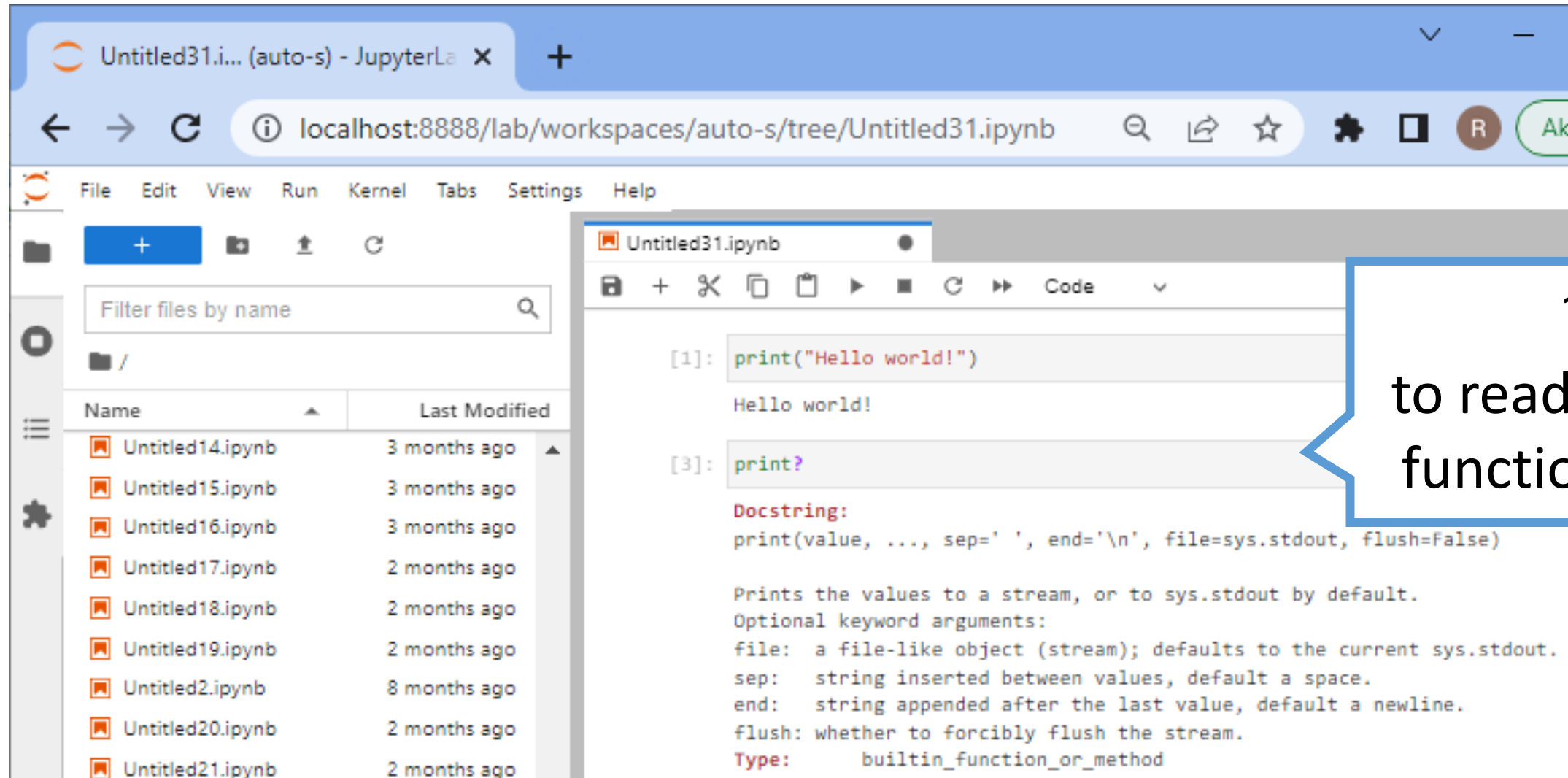
```
[1]: print("Hello world!")  
Hello world!
```

A blue box highlights the code cell, and a callout bubble points to it with the text "SHIFT + ENTER to execute a code cell".

- Context-specific help, auto-completion



- Help / “docstrings”



The screenshot shows the JupyterLab interface. On the left is a file browser with a search bar and a list of files. On the right is a code editor for 'Untitled31.ipynb'.

File Browser:

Name	Last Modified
Untitled14.ipynb	3 months ago
Untitled15.ipynb	3 months ago
Untitled16.ipynb	3 months ago
Untitled17.ipynb	2 months ago
Untitled18.ipynb	2 months ago
Untitled19.ipynb	2 months ago
Untitled2.ipynb	8 months ago
Untitled20.ipynb	2 months ago
Untitled21.ipynb	2 months ago

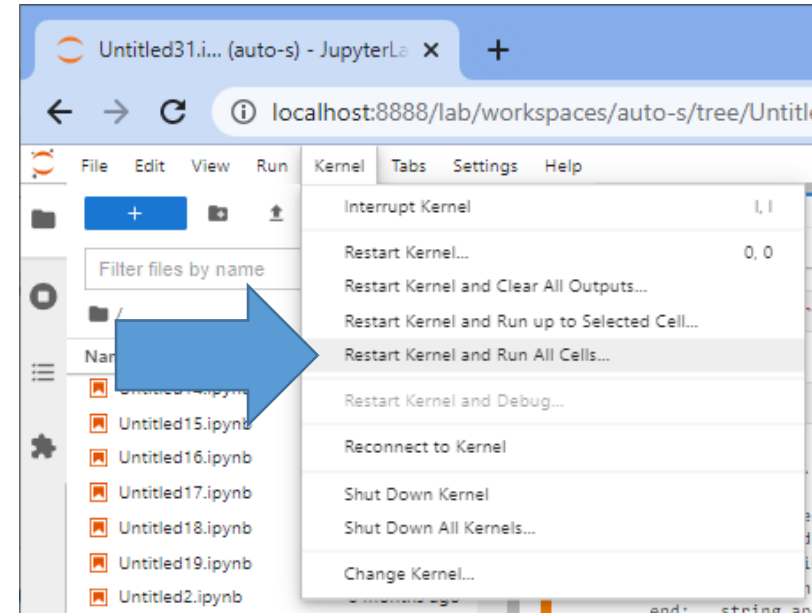
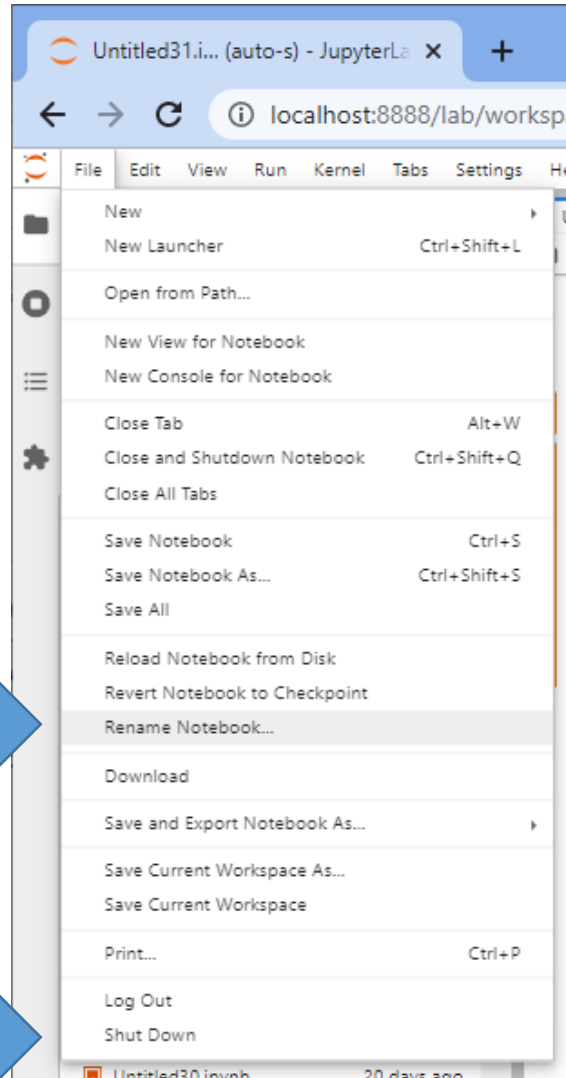
Code Editor:

```
[1]: print("Hello world!")  
Hello world!  
  
[3]: print?  
  
Docstring:  
print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)  
  
Prints the values to a stream, or to sys.stdout by default.  
Optional keyword arguments:  
file: a file-like object (stream); defaults to the current sys.stdout.  
sep: string inserted between values, default a space.  
end: string appended after the last value, default a newline.  
flush: whether to forcibly flush the stream.  
Type: builtin_function_or_method
```

?

to read what a
function does

- Saving / renaming / closing



Enforcing a “clean” execution state is important for ensuring reproducibility and repeatability