

# JUPYTER NOTEBOOKS:

*An Introduction*

*Kristian Garza*

*DataCite, @kriztean*

# OVERVIEW

- ▶ This workshop will introduce you to *Jupyter Notebooks*. You will learn what they are, what they do and why you might like to use them.
- ▶ It is an introductory set of lessons for those who are brand new, have little or no knowledge of coding and computational methods in research.
- ▶ By the end of the tutorial you will have a good understanding of what notebooks can do, how to open one up, *perform some basic tasks* and save it for later.

# INTRODUCTION

- ▶ A notebook interface, the concept is that it is a virtual notebook environment used for literate programming.
- ▶ *Literate programming* pairs the functionality of word processing software with both the shell and kernel of that notebook's programming language.
- ▶ Notebooks are documents that contain both code and rich text elements, such as links, equations and different ways of visualising data via graphs, tables and figures.

# JUPYTER NOTEBOOKS

- ▶ Jupyter is named after three computer programming languages - *Julia*, *Python* and *R*.
- ▶ They offer a hybrid environment in which you can perform computational tasks while also using text to annotate or describe what you and your code blocks are doing.
  - ▶ It's a like a mix between the *command line* and a *word processor*.

# WHAT CAN JUPYTER NOTEBOOKS DO?

- ▶ Data cleaning
- ▶ Data transformation
- ▶ Numerical simulation
- ▶ Statistical modeling
- ▶ *Visualisation*

These tools are very helpful for exploration as well as demonstration.

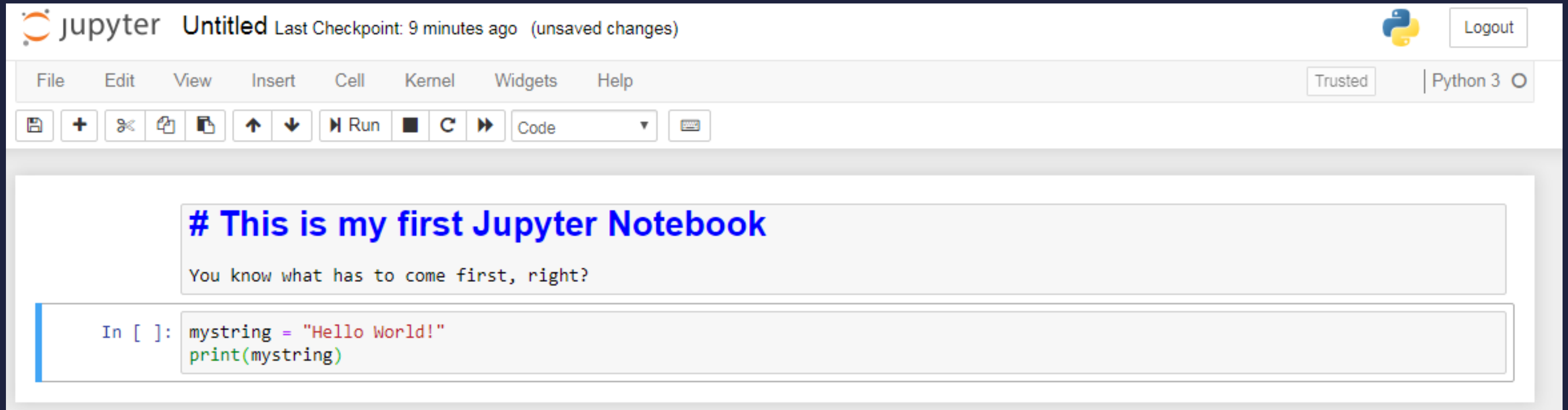
# WHY USE JUPYTER NOTEBOOKS?

- ▶ They are great for exploration in data analysis, presenting results, and sharing ideas.
- ▶ Jupyter Notebooks are also *great at performing rapid visualisations that you can test out, change and share easily.*
- ▶ They are also freely available and you can use them in a *normal browser.*
- ▶ Offer a way to experiment with data processing *without having to be a*

# WHAT MAKES THEM DIFFERENT TO OTHER APPLICATIONS?

- ▶ **Code:** *Running code* means making the computer do what you are telling it to do. "Executing code" is the same thing.
- ▶ **Output:** is *the result* of the computational process, such as a visualisation, graph, model, equation and so on.
- ▶ Jupyter notebooks are a series of *cells* containing executable code and outputs.

# THE NOTEBOOK INTERFACE



The screenshot shows the Jupyter Notebook interface. At the top, the Jupyter logo is on the left, followed by the text "jupyter Untitled Last Checkpoint: 9 minutes ago (unsaved changes)". On the right, there is a Python logo and a "Logout" button. Below this is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". To the right of the menu bar are "Trusted" and "Python 3" buttons. Below the menu bar is a toolbar with icons for saving, adding, deleting, copying, pasting, undo, redo, and running. A dropdown menu is set to "Code". The main area contains two code cells. The first cell has a blue comment: "# This is my first Jupyter Notebook" followed by the text "You know what has to come first, right?". The second cell contains a Python code snippet: "In [ ]: mystring = 'Hello World!'" and "print(mystring)".

SHIFT + ENTER



# SHORT SUMMARY

- ▶ A notebook can either run *on your desktop* with no internet or on a remote server *via the internet*
- ▶ A notebook runs and stores the code and output, with markdown notes
  - ▶ A notebook is an *editable document* with input and output cells

# LET'S START

- ▶ **Tutorial 1: Overview of nodes and connections in the PID Graph.**
  - ▶ **Go to: <https://doi.org/10.14454/3bpw-w381>**
  - ▶ **Extra activities: <https://tinyurl.com/y24jojbm>**
    - ▶ **Coffee break (15 min)**
- ▶ **Tutorial 2: Graph of a specific researcher (30 min, Najko)**

# ACTIVITIES

<https://tinyurl.com/y24jojbm>