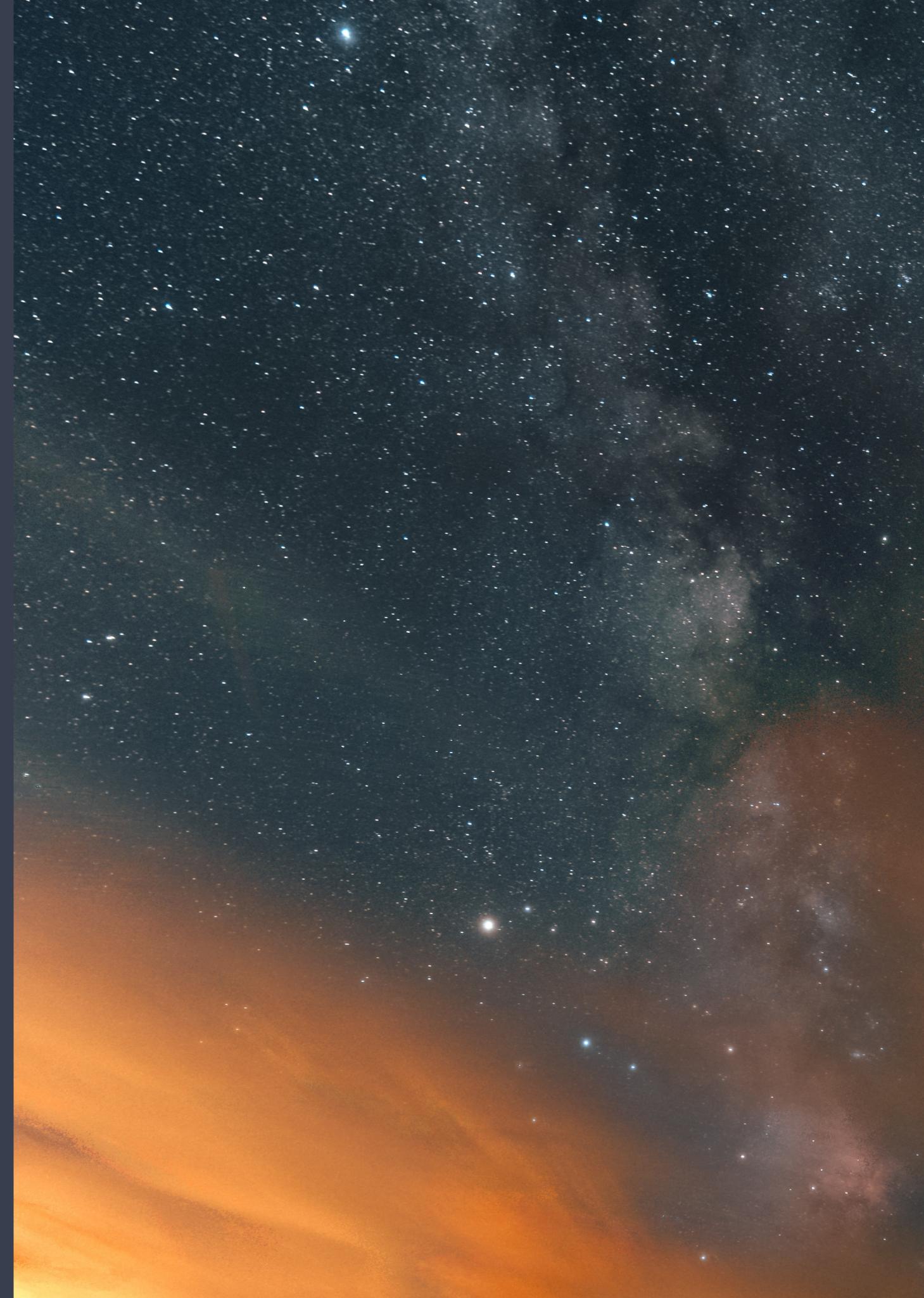


Gav McClary

# CODING FOR DATA ANALYSIS AND VISUALISATION

An intro to working with Python, Jupyter  
Notebook and Pandas



# OBJECTIVES

- How to use Jupyter Notebook for data analysis
- How to write Python code
- Using Pandas library to explore and visualise data

The aim of this course is to provide you with some fundamental skills to enable you to utilise modern open-source (free!) tools to read, clean, transform and visualise data.

We will start with an introduction to Jupyter Notebook, an open-source web application that allows the creation and sharing of documents that contain live code, equations, visualisations and more.

Next, we will explore the Python programming language and work with some of its basic data structures such as: lists, dictionaries, sets and files.

Finally, we will use the Pandas library to explore and visualise the data.

I hope you enjoy the course!

Gav

# DATA ANALYSIS

**Data analysis** is a process of inspecting, cleansing, transforming and modeling data with the goal of discovering useful information, informing conclusion and supporting decision-making.

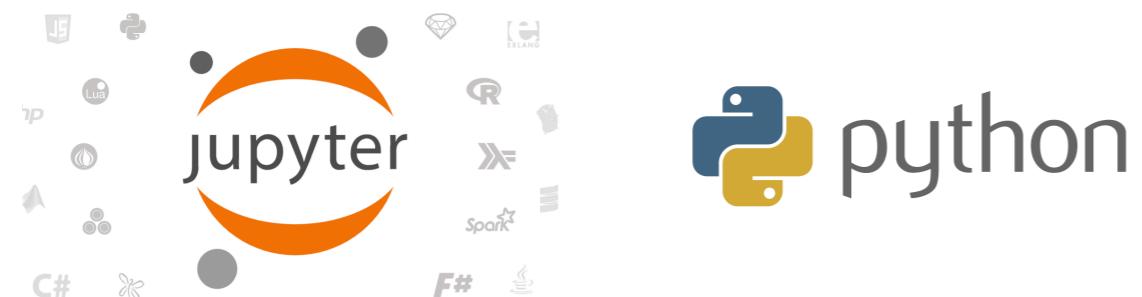
During this course you will learn how to create sample data and also how to load existing data such as **csv** files.

You will then **clean** your data, and transform data in pandas **DataFrames**.

After exploring DataFrames you will **visualise** your data and test some of your new skills!



## Introduction



pandas

$$y_i t = \beta' x_{it} + \mu_i + \epsilon_{it}$$



## **Content**

# HOW TO USE JUPYTER NOTEBOOK FOR DATA ANALYSIS

## What is Jupyter Notebook?

*The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more (<https://jupyter.org>).*

Throughout the course we will be using Jupyter Notebook to execute (run) all the Python code and pandas code so as to keep all our experiments in one place.

We can use Jupyter's **markdown** cells to add comments and descriptions and we can place **visualisations** such as **Box Plots** and **Histograms** inline with our code experiments.



## How to use Jupyter Notebook for data analysis

### Installation Instructions

To install Jupyter Notebook there are two paths we can take :

Path A: Install Jupyter using Anaconda and conda

**Download [Anaconda \(Python 3.7 version\)](#)**

**Following the instructions on the download page to install**

**Run Jupyter Notebook with command:**

**<= Anaconda Navigator**

```
jupyter notebook
```

Path B: Install Jupyter using pip (experienced Python users only)

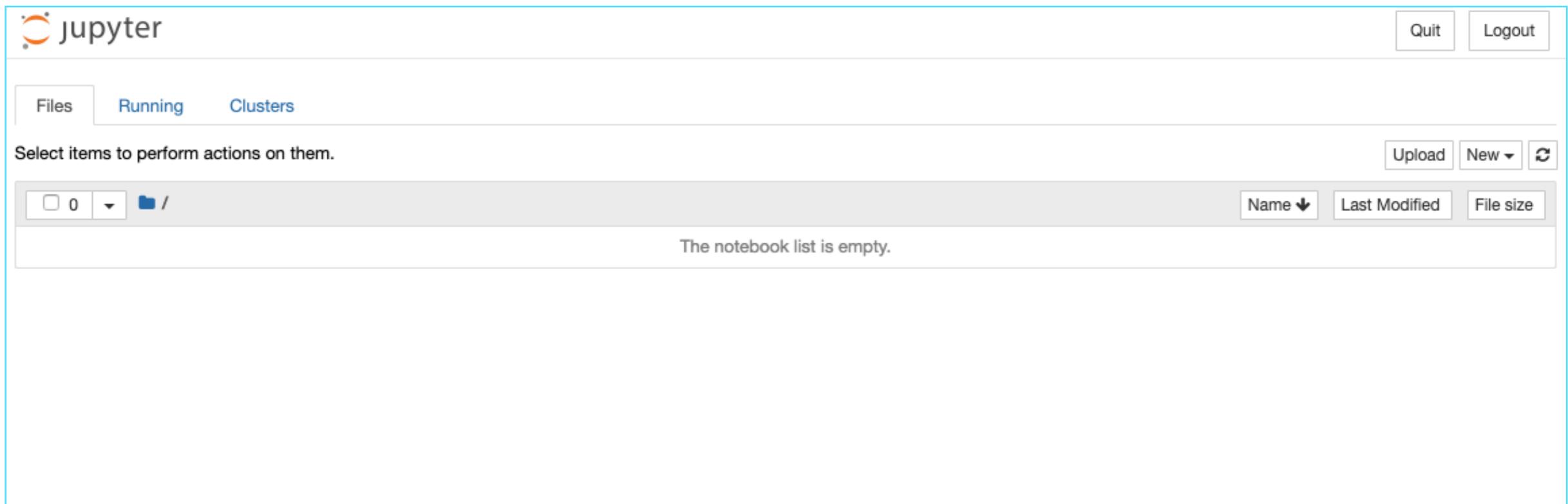
```
pip3 install --upgrade pip
```

```
pip3 install jupyter
```

## How to use Jupyter Notebook for data analysis

### Creating Notebooks

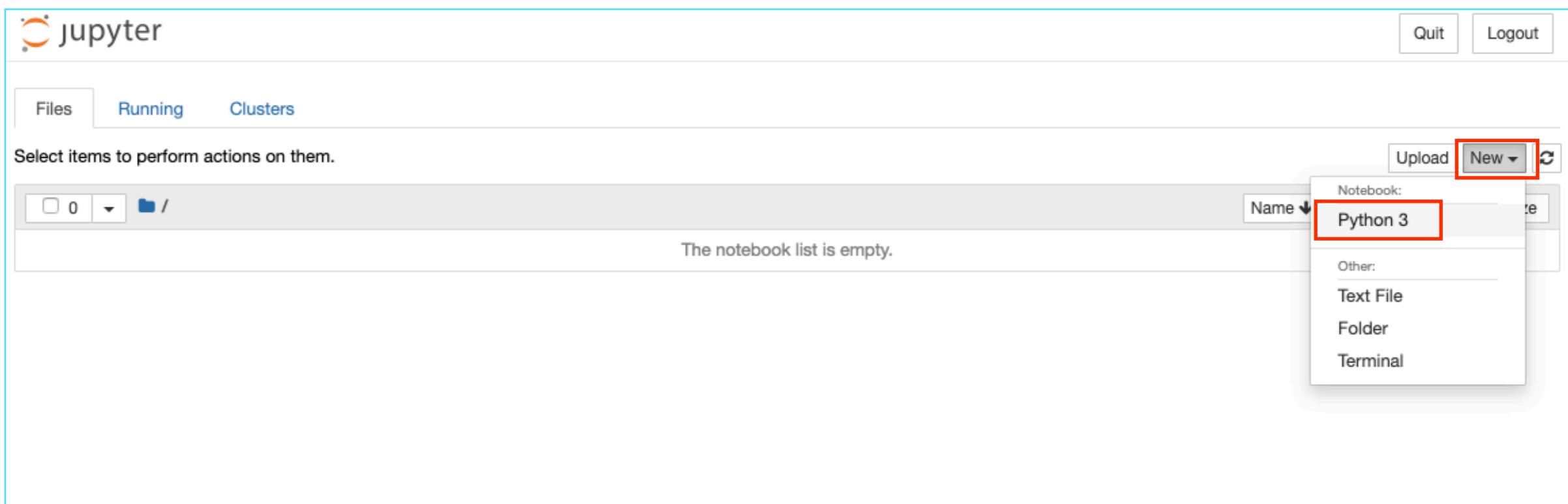
After following the installation steps and running command **jupyter notebook** you should be presented with a web page that looks **similar** to this one:



## How to use Jupyter Notebook for data analysis

Creating Notebooks

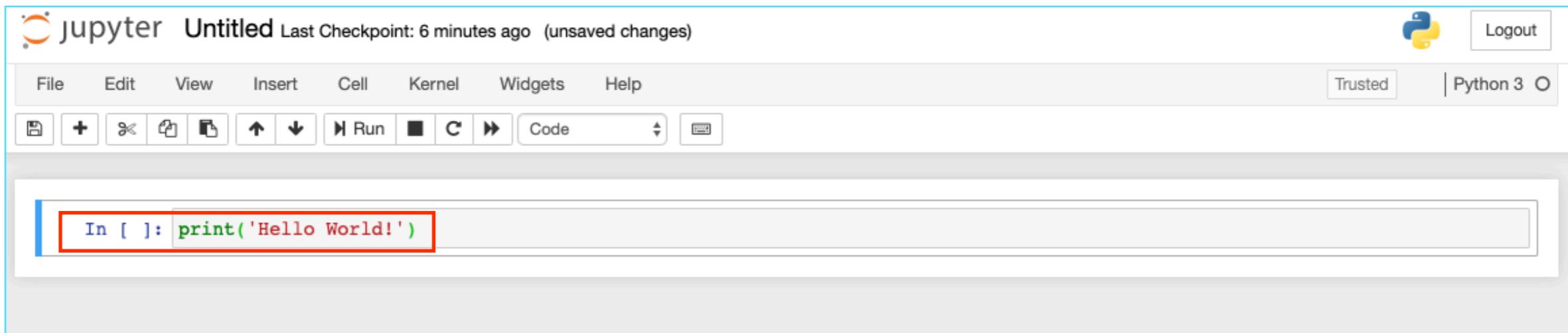
To open a **new notebook** click on **New** then **Python 3**:



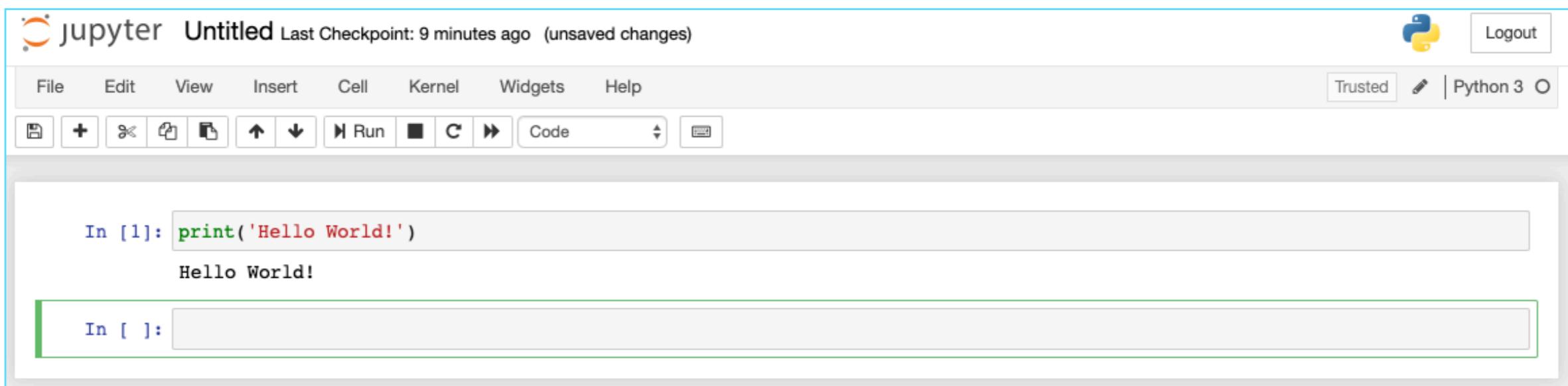
## How to use Jupyter Notebook for data analysis

### Running Cells

To run a cell, first type some code (see example below) then press the key combination **shift** and **enter** to run that code



A screenshot of the Jupyter Notebook interface. The title bar says "jupyter Untitled Last Checkpoint: 6 minutes ago (unsaved changes)". The toolbar includes File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, and Python 3. Below the toolbar is a toolbar with icons for file operations like new, open, save, and run. A code cell is shown with the input "In [ ]: print('Hello World!')". The code is highlighted with a red border.

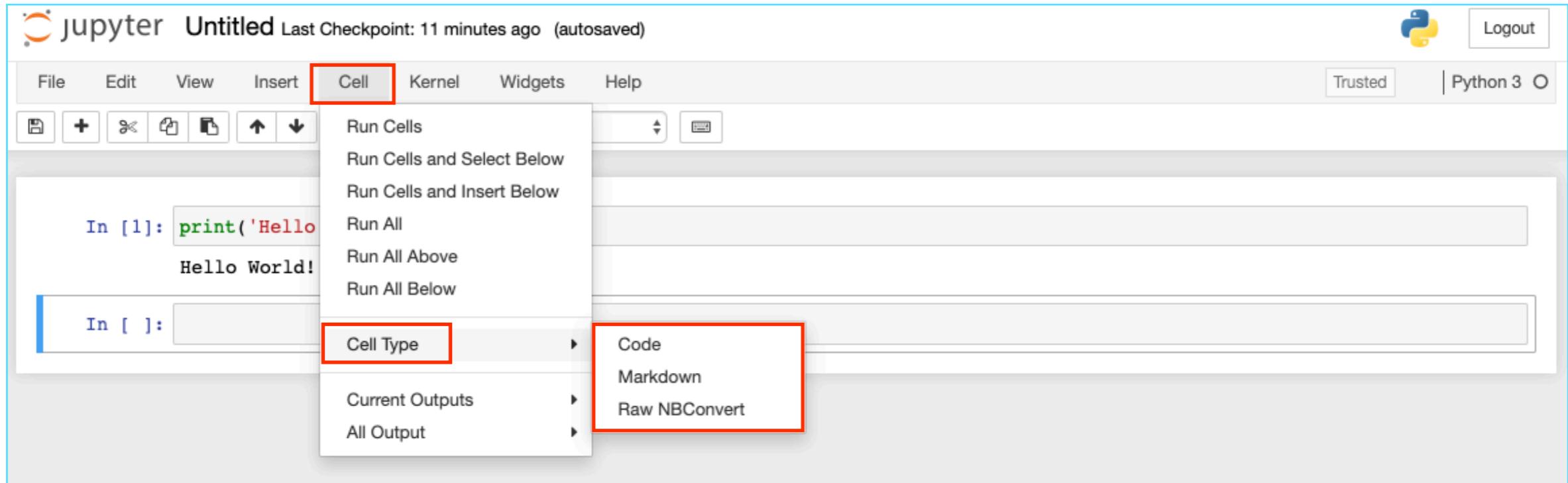


A screenshot of the Jupyter Notebook interface. The title bar says "jupyter Untitled Last Checkpoint: 9 minutes ago (unsaved changes)". The toolbar and code cell are identical to the one above. Below the code cell, the output "Hello World!" is displayed. A new, empty code cell "In [ ]:" is shown at the bottom, indicated by a green border.

## How to use Jupyter Notebook for data analysis

### Cell Types

Cells in a notebook can be of a different type. During the course we will be using both the **code** cell type and the **markdown** cell type.



More about **cell types**:

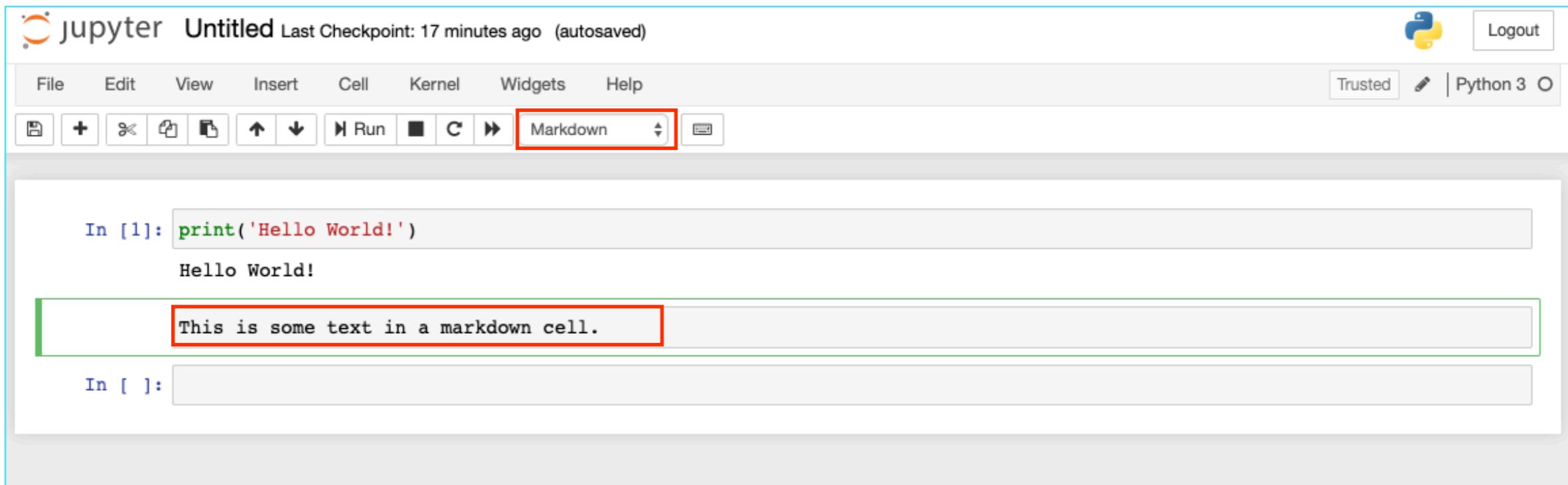
The **code** cell type is used when you want to execute/run some Python or pandas code in the notebook

The **markdown** cell type is used when you want to enter some plain text into the notebook (see example on the next page)

## How to use Jupyter Notebook for data analysis

### Markdown cells

Use cells of type **markdown** to enter explanatory text into your notebook:



More about **markdown**:

Markdown is a markup language (think a light version of HTML) often used to write web documents.

For some guidance on markdown syntax go here:

<https://www.markdownguide.org/basic-syntax>

# HOW TO WRITE PYTHON CODE

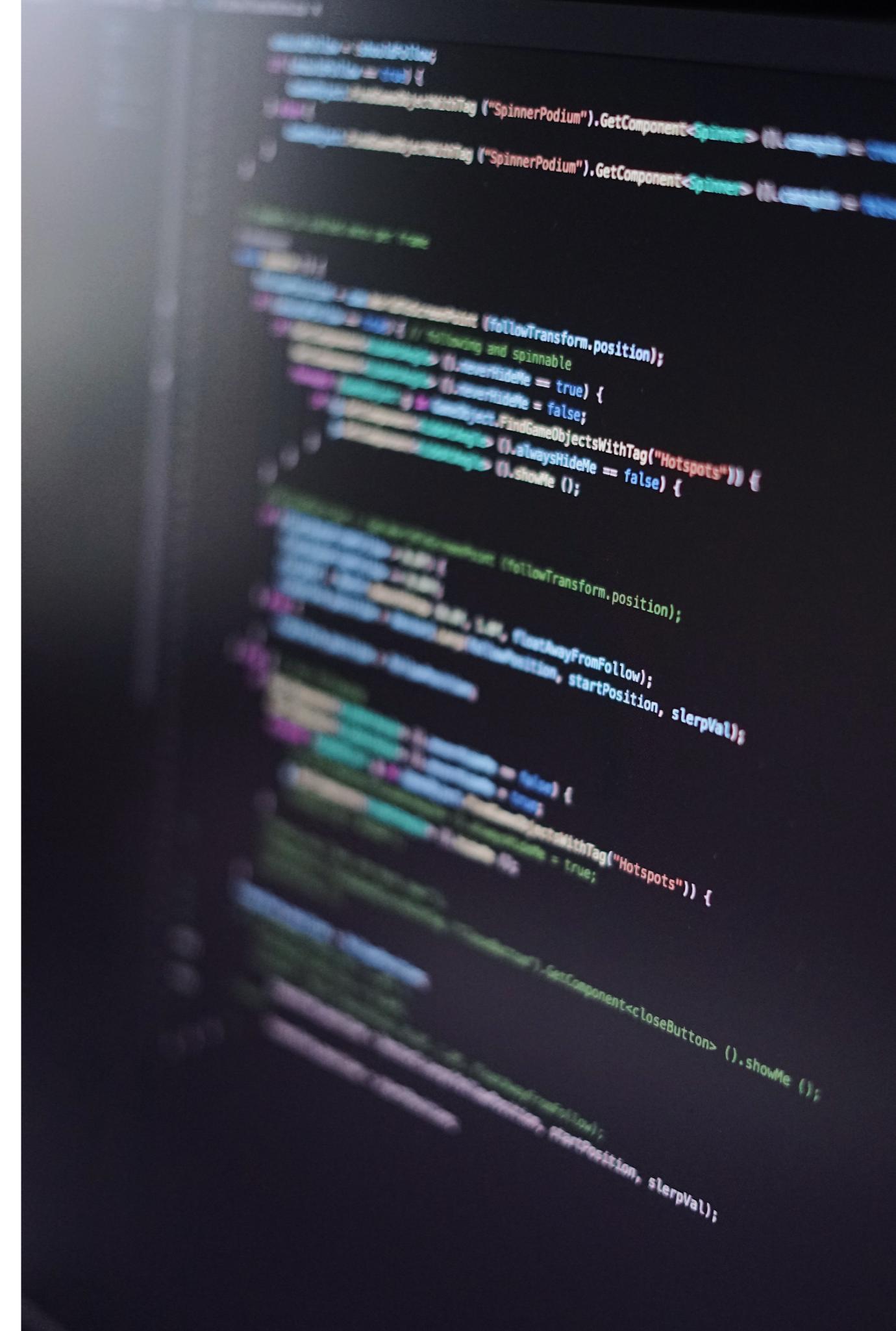
## What is Python?

Python is a high-level programming language.

As far as programming languages go it is very human readable and therefore easier to learn than other languages.

Python is commonly used in the **Data Analysis/Data Science** fields and as such has access to many external libraries that support these activities including **NumPy**, **Matplotlib** and **Pandas**.

Here, we will explore some of Python's basic syntax, data structures and how to work with files.



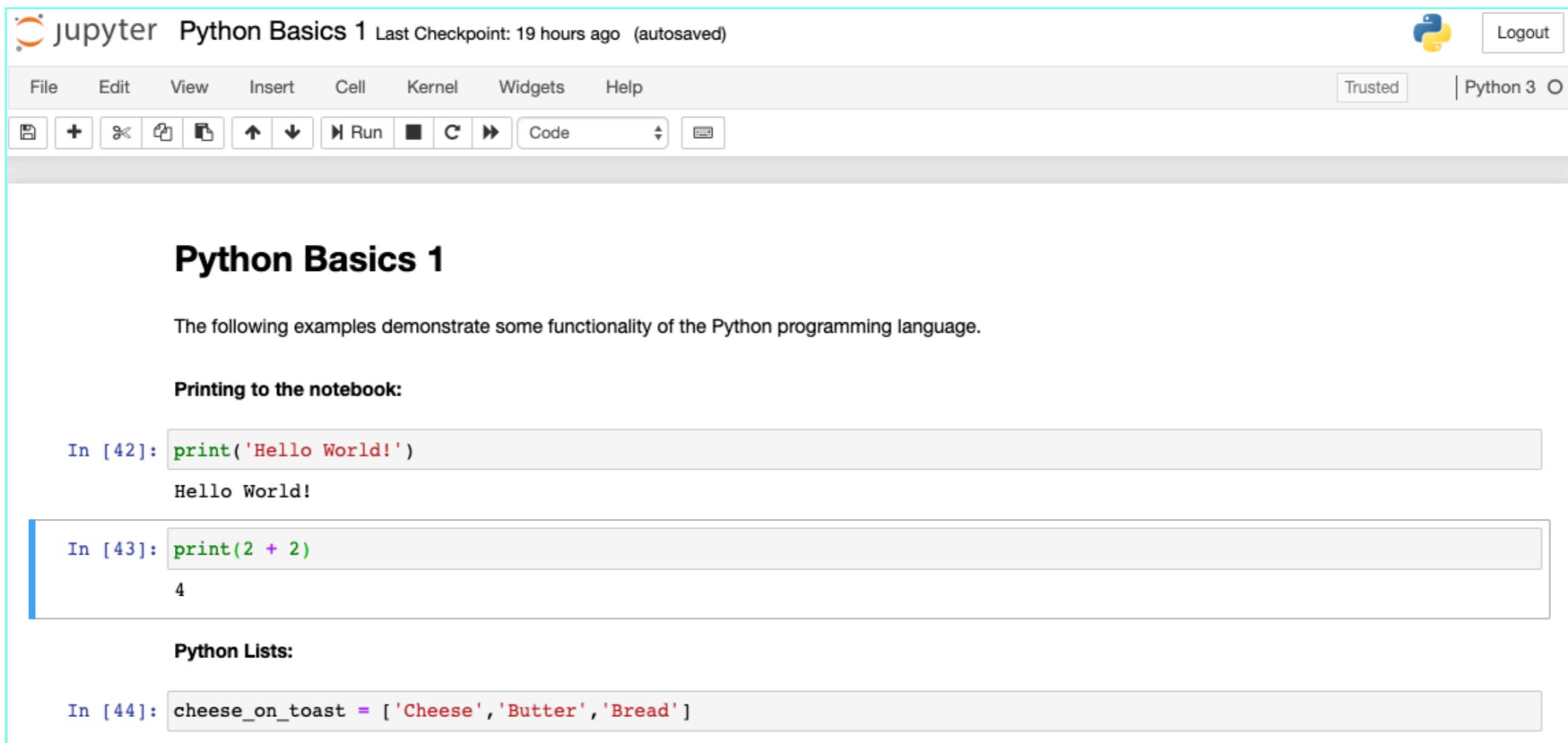
## How to write Python code

Using Jupyter Notebook to run Python commands

Python Basics 1

Open the file **Python Basics 1.ipynb** by issuing the command:

```
jupyter notebook Python Basics 1.ipynb
```



The screenshot shows a Jupyter Notebook interface with the title "jupyter Python Basics 1 Last Checkpoint: 19 hours ago (autosaved)". The toolbar includes File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, and Python 3. Below the toolbar are standard notebook controls for cell selection, running, and kernel management.

The main content area displays the following text:

## Python Basics 1

The following examples demonstrate some functionality of the Python programming language.

**Printing to the notebook:**

```
In [42]: print('Hello World!')  
Hello World!
```

```
In [43]: print(2 + 2)  
4
```

**Python Lists:**

```
In [44]: cheese_on_toast = ['Cheese', 'Butter', 'Bread']
```

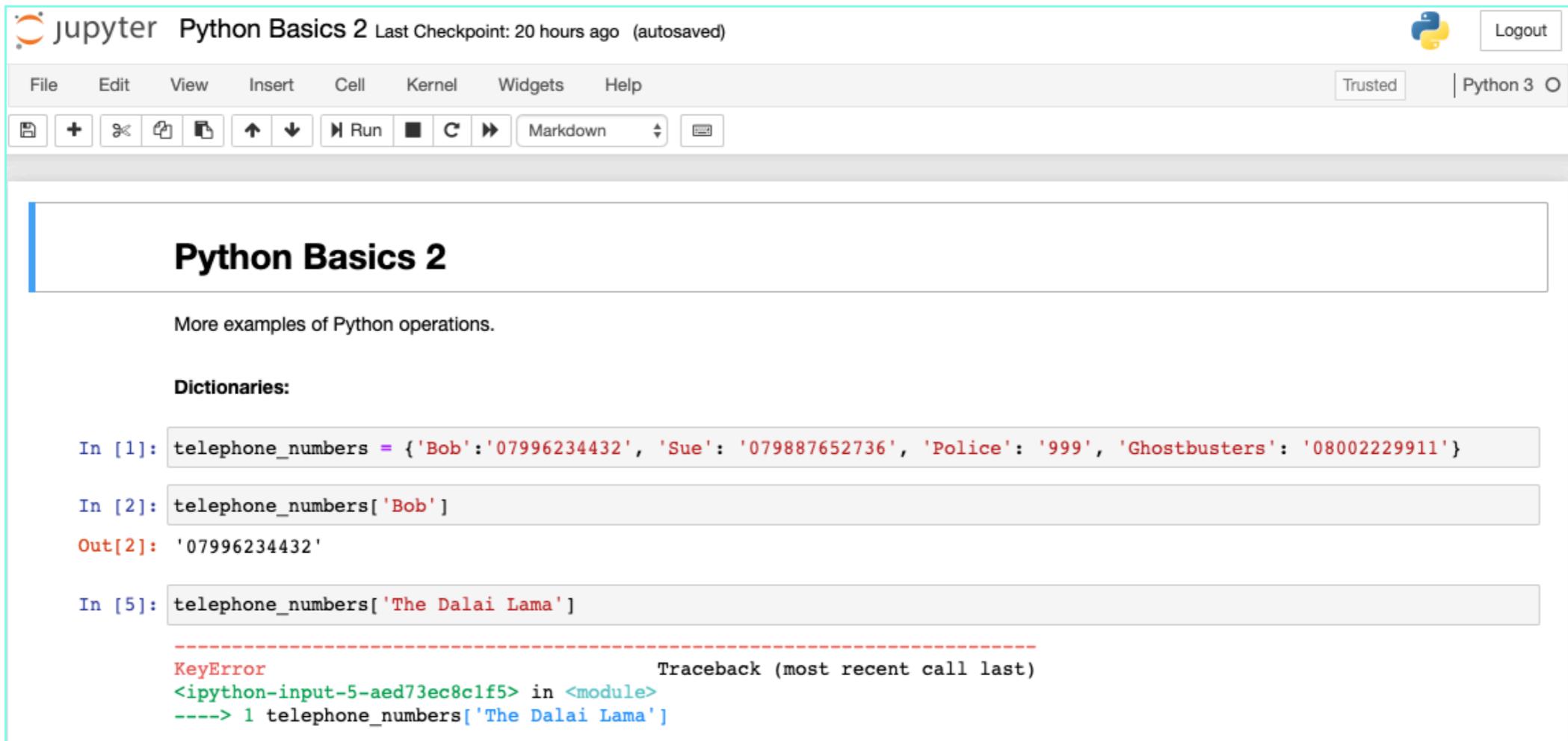
## How to write Python code

Using Jupyter Notebook to run Python commands

Python Basics 2

Open the file **Python Basics 2.ipynb** by issuing the command:

```
jupyter notebook Python Basics 2.ipynb
```



The screenshot shows a Jupyter Notebook interface with the following details:

- Title Bar:** jupyter Python Basics 2 Last Checkpoint: 20 hours ago (autosaved)
- Toolbar:** File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, Python 3
- Cell Buttons:** New, Run, Cell, Cell, Cell, Cell, Markdown, Cell
- Section Header:** Python Basics 2
- Text:** More examples of Python operations.
- Code Cells:**
  - In [1]: `telephone_numbers = {'Bob': '07996234432', 'Sue': '079887652736', 'Police': '999', 'Ghostbusters': '08002229911'}`
  - In [2]: `telephone_numbers['Bob']`
  - Out[2]: `'07996234432'`
  - In [5]: `telephone_numbers['The Dalai Lama']`
- Traceback:**

```
-----  
KeyError                                 Traceback (most recent call last)  
<ipython-input-5-aed73ec8c1f5> in <module>  
----> 1 telephone_numbers['The Dalai Lama']
```

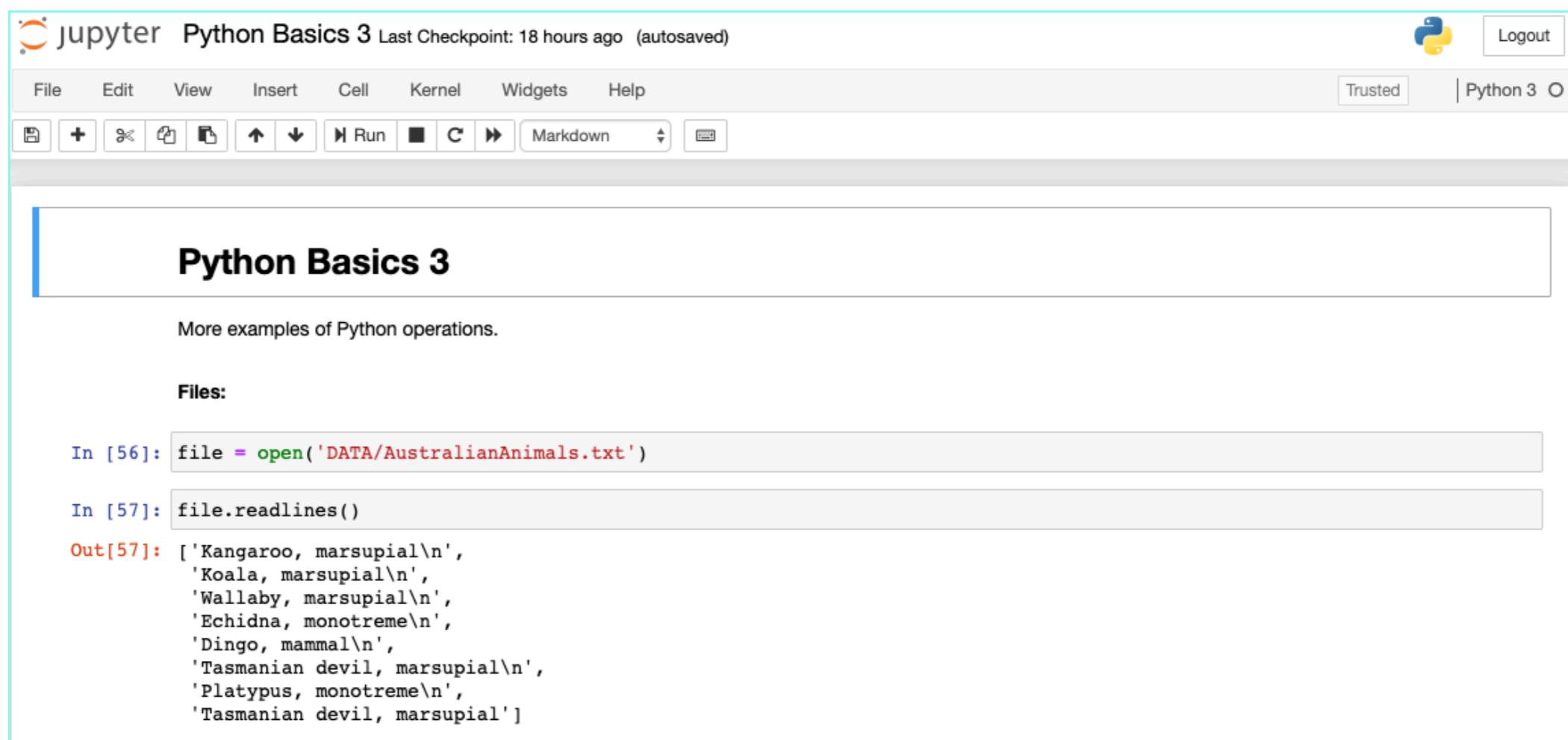
## How to write Python code

Using Jupyter Notebook to run Python commands

Python Basics 3

Open the file **Python Basics 3.ipynb** by issuing the command:

```
jupyter notebook Python Basics 3.ipynb
```



The screenshot shows a Jupyter Notebook interface with the title "jupyter Python Basics 3 Last Checkpoint: 18 hours ago (autosaved)". The toolbar includes File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, and Python 3. Below the toolbar is a toolbar with icons for file operations like Open, Save, and Run, along with a dropdown for Markdown.

The main area contains a code cell titled "Python Basics 3". The cell displays the following Python code and output:

```
In [56]: file = open('DATA/AustralianAnimals.txt')
In [57]: file.readlines()
Out[57]: ['Kangaroo, marsupial\n',
          'Koala, marsupial\n',
          'Wallaby, marsupial\n',
          'Echidna, monotreme\n',
          'Dingo, mammal\n',
          'Tasmanian devil, marsupial\n',
          'Platypus, monotreme\n',
          'Tasmanian devil, marsupial']
```

# USING PANDAS LIBRARY TO EXPLORE AND VISUALISE DATA

**pandas** is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language (<https://pandas.pydata.org/about.html>).



## Pandas

Examples using Jupyter Notebook to run pandas commands to do EDA (Exploratory Data Analysis) and Visualisation

Use file **Descriptive Statistics.ipynb** for this portion of course.

Open file by using command:

```
jupyter notebook Descriptive Statistics.ipynb
```

### Descriptive Statistics with Pandas

Import pandas (Data Analysis Library) and matplotlib (Plotting Library)

Renaming as 'pd' and 'plt' is not strictly necessary but can help reduce typing.

```
In [343]: import pandas as pd  
import matplotlib.pyplot as plt
```

### Create sample data

If required we can create sample data directly using pandas. Below is a list of column names:

```
In [344]: columnNames = ['Student Name',  
                     'January Calories',  
                     'February Calories',  
                     'March Calories',  
                     'Total Calories'  
]
```

Below is a list of lists that represents the individual rows in a table:

```
In [345]: calories_per_month = [['Bob', 56000, 61000, 55000, 172000],  
                           ['Sue', 49000, 51000, 48000, 148000],  
                           ['Barack', 50000, 51000, 52000, 153000],  
                           ['Boris', 70000, 69000, 75000, 214000],  
                           ['Nancy', 41000, 47000, 43000, 131000]]
```

## Next Steps

Check out the **useful\_links.md** file provided in the folder you downloaded at the beginning of the course.

For reference I have included them below:

### Jupyter Notebook

<https://realpython.com/jupyter-notebook-introduction/>

### Python

<https://realpython.com/learning-paths/python3-introduction/>

### Pandas

<https://realpython.com/pandas-python-explore-dataset/>

<https://realpython.com/python-data-cleaning-numpy-pandas/>