

Python: How To Get Started

Dr Tamara Celford

To download a copy of the slides go to
<https://tamaracleford.co.uk/iop.html>



INTRODUCTION TO PYTHON

- What is Python
- Examples from my work
- Python fundamentals
- How can I learn more?



WHAT IS PYTHON?

- Created in 1991 by Guido Van Rossum
- Interpreted
- High-level, focusing on readability
- General purpose
- Object-orientated and functional

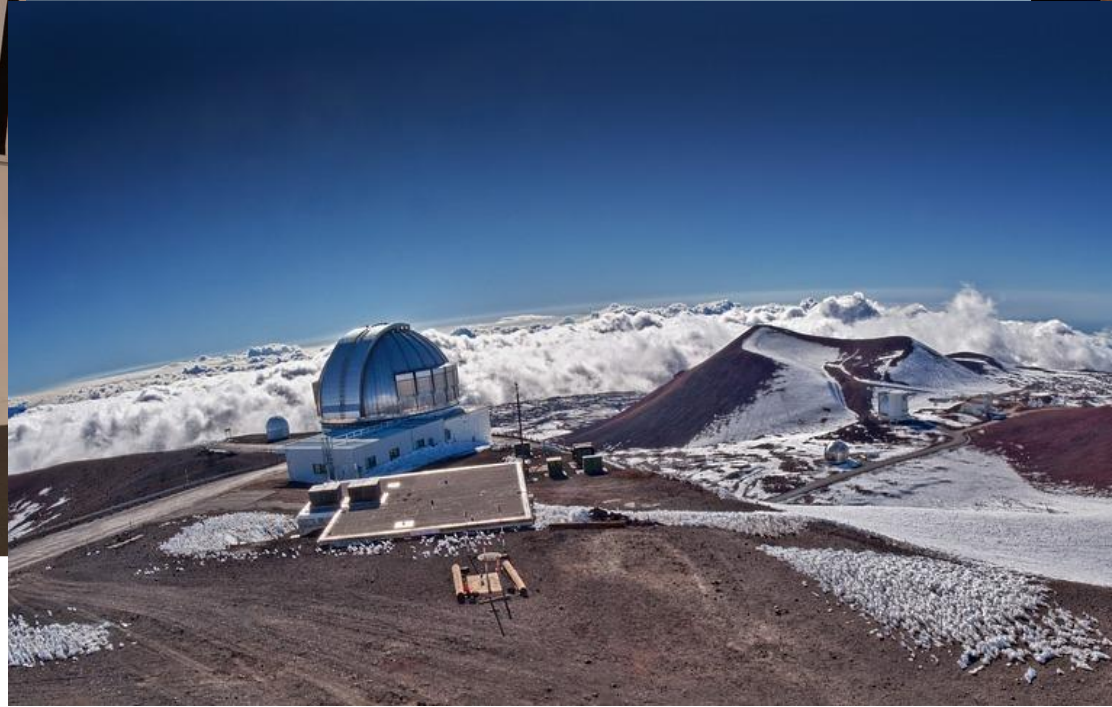
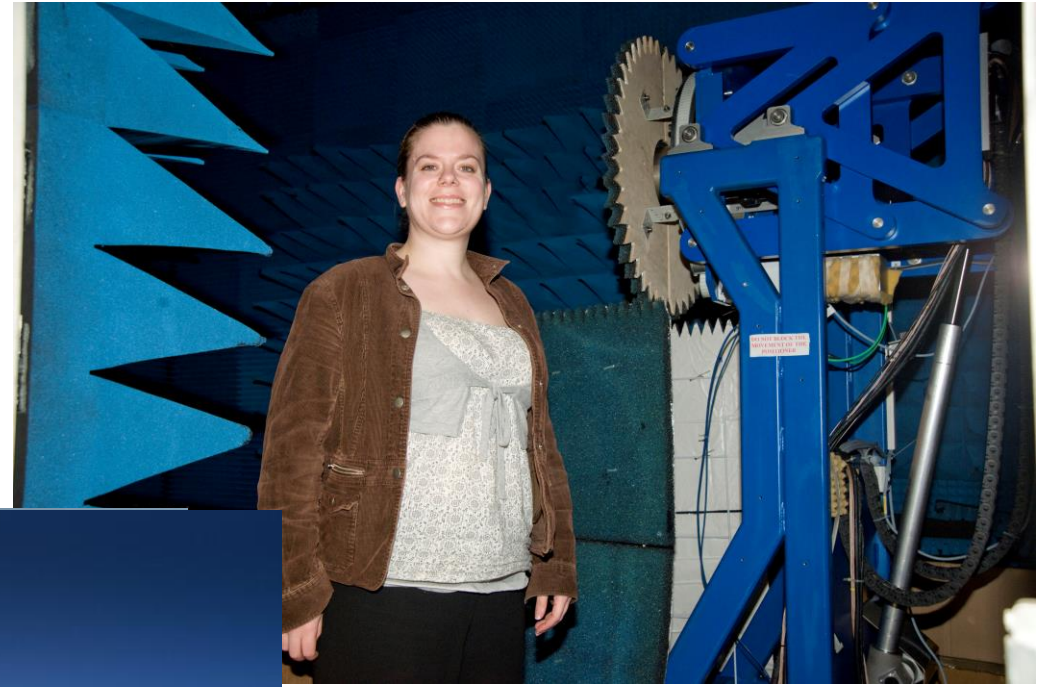
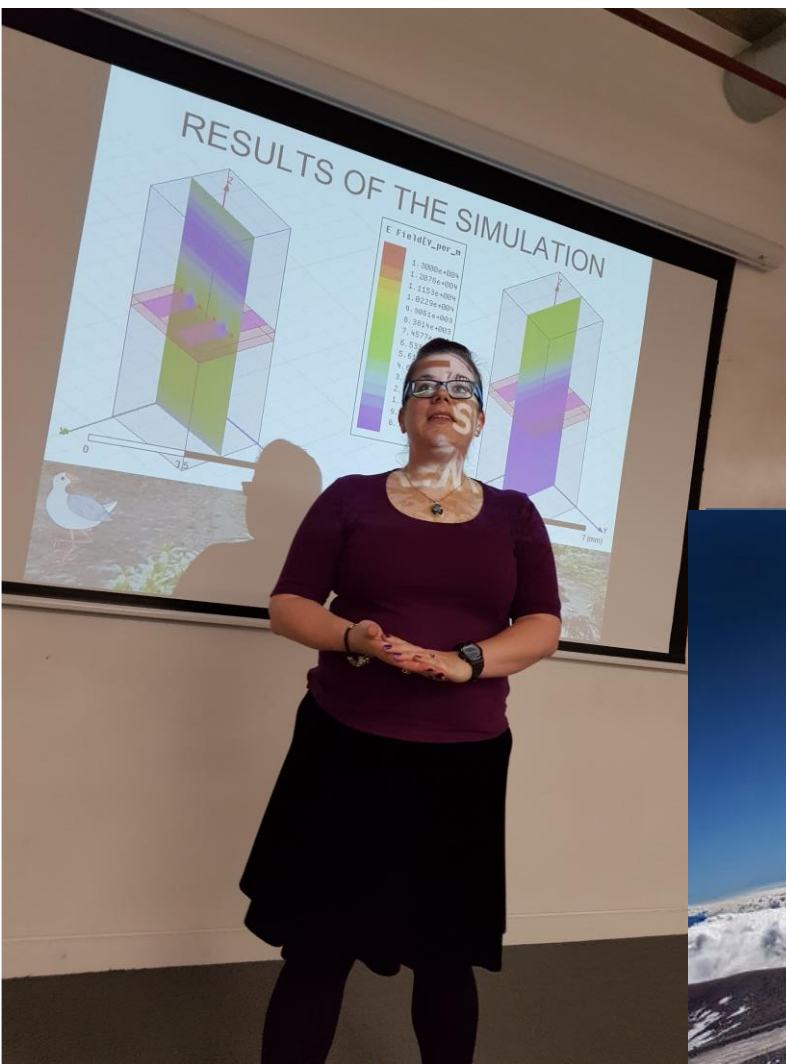


WHY USE PYTHON?

- Free
- Open source
- Used by a lot of different industries
- Modules, Packages and Libraries
 - <https://wiki.python.org/moin/UsefulModules>
 - <https://docs.python.org/3/library/>



ABOUT ME


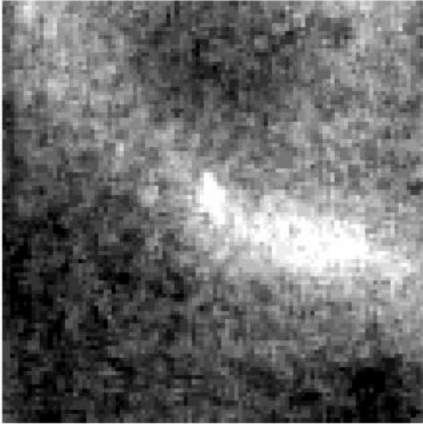

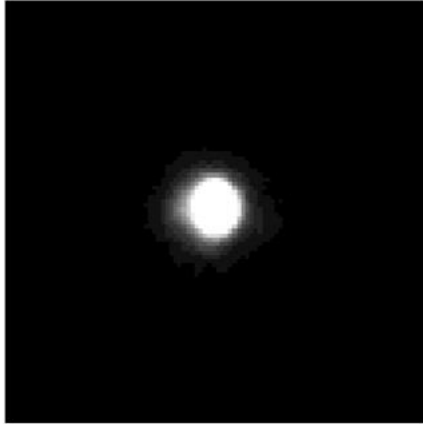
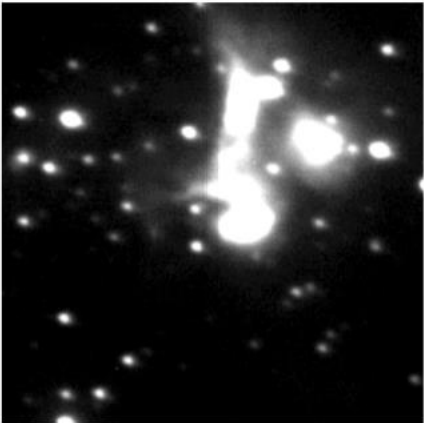
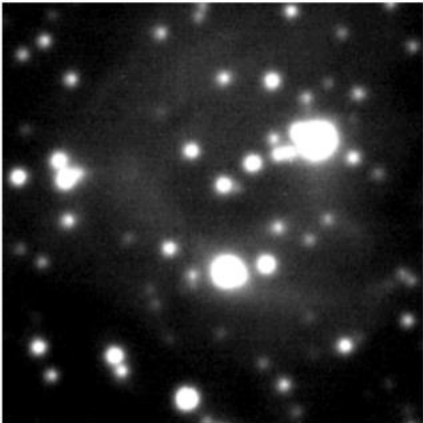
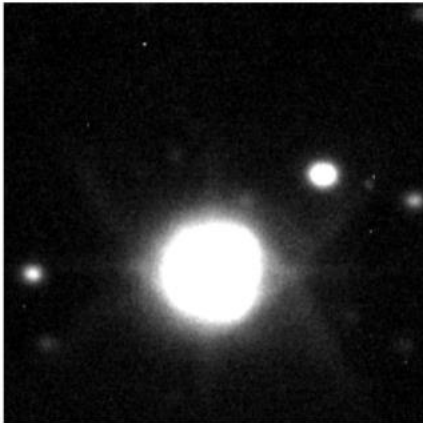
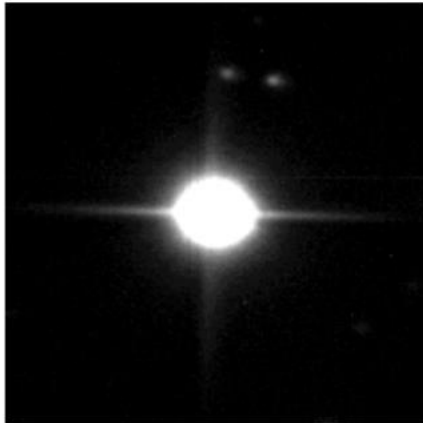


SWAMPHEN ENTERPRISES

<https://www.needpix.com/photo/920796/mountain-telescope-hawaii-summit-astronomy-astrophysics-mauna-kea-keck-telescope-subaru-telescope>

Dr TAMARA CLELFORD

STELLAR CLASSIFICATIONS

	MYSO	HII	PN	Star
MICHELLE				
UFTI				

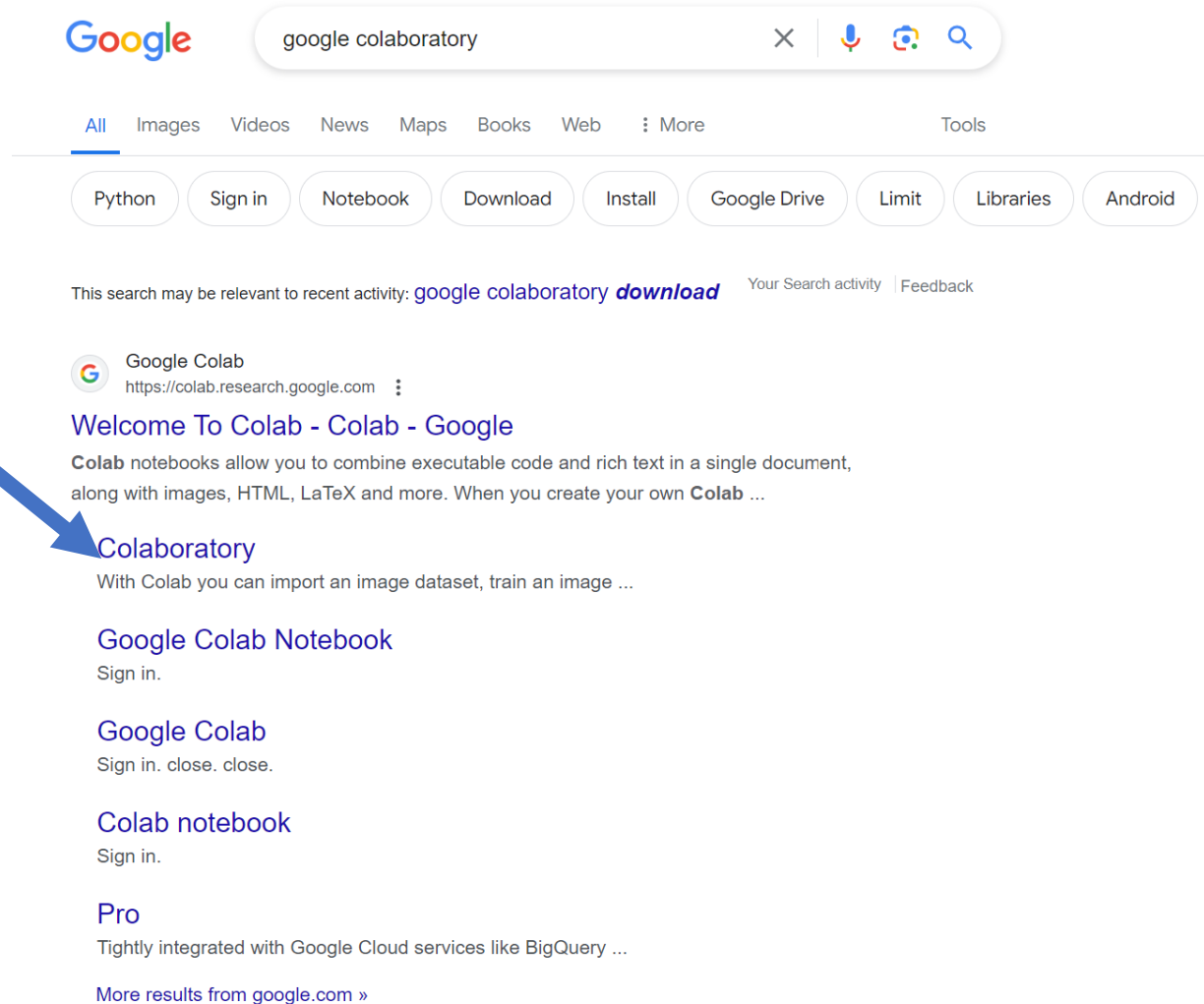


SET UP PYTHON TOOLS

- Integrated Development Environment (IDE)
 - Jupyter notebook style
 - Google Colaboratory



GOOGLE 'GOOGLE COLABORATORY'



The screenshot shows a Google search interface. The search bar contains 'google colaboratory'. Below the search bar, there are tabs for 'All', 'Images', 'Videos', 'News', 'Maps', 'Books', 'Web', 'More', and 'Tools'. Under the 'All' tab, there are buttons for 'Python', 'Sign in', 'Notebook', 'Download', 'Install', 'Google Drive', 'Limit', 'Libraries', and 'Android'. Below these buttons, a message states: 'This search may be relevant to recent activity: [google colaboratory download](#)'. Below this, the search results are listed. The first result is 'Google Colab' with the URL 'https://colab.research.google.com'. Below this, there is a blue arrow pointing to the 'Colaboratory' result. The 'Colaboratory' result has the title 'Welcome To Colab - Colab - Google' and the description 'Colab notebooks allow you to combine executable code and rich text in a single document, along with images, HTML, LaTeX and more. When you create your own Colab ...'. Below this, there are several other results: 'Colaboratory' with the description 'With Colab you can import an image dataset, train an image ...', 'Google Colab Notebook' with the description 'Sign in.', 'Google Colab' with the description 'Sign in. close. close.', 'Colab notebook' with the description 'Sign in.', and 'Pro' with the description 'Tightly integrated with Google Cloud services like BigQuery ...'. At the bottom, there is a link 'More results from google.com »'.

Google

google colaboratory

All Images Videos News Maps Books Web More Tools

Python Sign in Notebook Download Install Google Drive Limit Libraries Android

This search may be relevant to recent activity: [google colaboratory download](#) Your Search activity | Feedback

Google Colab
<https://colab.research.google.com>

Welcome To Colab - Colab - Google

Colab notebooks allow you to combine executable code and rich text in a single document, along with images, HTML, LaTeX and more. When you create your own Colab ...

Colaboratory

With Colab you can import an image dataset, train an image ...

Google Colab Notebook

Sign in.

Google Colab

Sign in. close. close.

Colab notebook

Sign in.

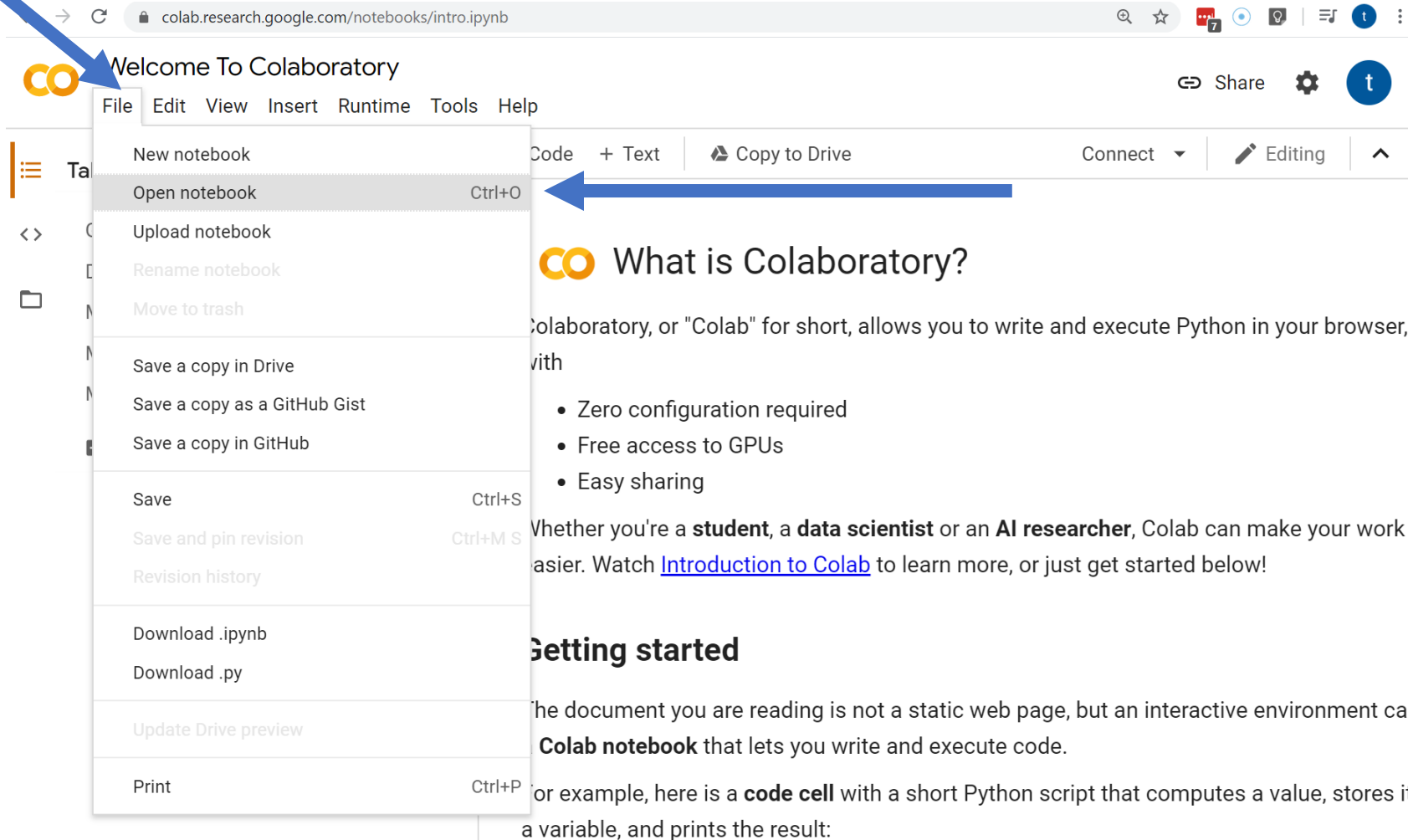
Pro

Tightly integrated with Google Cloud services like BigQuery ...

[More results from google.com »](#)



OPEN A NEW PYTHON 3 NOTEBOOK



The screenshot shows the Google Colaboratory web interface. The browser address bar displays `colab.research.google.com/notebooks/intro.ipynb`. The main header includes the Colab logo, the text "Welcome To Colaboratory", and a "Share" button. Below the header is a menu bar with "File", "Edit", "View", "Insert", "Runtime", "Tools", and "Help". The "File" menu is open, showing options: "New notebook", "Open notebook" (highlighted with a blue arrow and the keyboard shortcut "Ctrl+O"), "Upload notebook", "Rename notebook", "Move to trash", "Save a copy in Drive", "Save a copy as a GitHub Gist", "Save a copy in GitHub", "Save" (with "Ctrl+S"), "Save and pin revision" (with "Ctrl+M S"), "Revision history", "Download .ipynb", "Download .py", "Update Drive preview", and "Print" (with "Ctrl+P"). The main content area displays the "What is Colaboratory?" introduction, which states that Colab allows writing and executing Python in the browser with zero configuration, free GPU access, and easy sharing. It also includes a "Getting started" section.

colab.research.google.com/notebooks/intro.ipynb

Welcome To Colaboratory

File Edit View Insert Runtime Tools Help

New notebook

Open notebook **Ctrl+O**

Upload notebook

Rename notebook

Move to trash

Save a copy in Drive

Save a copy as a GitHub Gist

Save a copy in GitHub

Save **Ctrl+S**

Save and pin revision **Ctrl+M S**

Revision history

Download .ipynb

Download .py

Update Drive preview

Print **Ctrl+P**

What is Colaboratory?

Colaboratory, or "Colab" for short, allows you to write and execute Python in your browser, with

- Zero configuration required
- Free access to GPUs
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch [Introduction to Colab](#) to learn more, or just get started below!

Getting started

The document you are reading is not a static web page, but an interactive environment called a **Colab notebook** that lets you write and execute code.

For example, here is a **code cell** with a short Python script that computes a value, stores it in a variable, and prints the result:



GET TO GRIPS WITH YOUR GOOGLE COLAB

- Print a string:

```
print('hello, world!')
```
- Work out how to execute the code block
- Work out how to get a new code block
- Try printing some other strings
- Try some maths:

```
print(4+6)  
print(9<3)
```





data types, variable names

```
[1] 1 type(1.1)
```

```
[2] 1 name = 'hello, world!'
```

```
[3] 1 print(name)
```

```
[4] 1 type(name)
```

operators

```
[5] 1 a = 9  
    2 b = 5.3  
    3 c = 22.4
```

```
[6] 1 print(a*b)
```

```
[8] 1 print(c>1)
```

lists

```
[9] 1 my_list = [1,2,3,4,5,6,7,8,9]
```

```
[10] 1 print(my_list)
```

```
[11] 1 my_list[1:3]
```

```
[12] 1 my_list.append(10)  
    2 print(my_list)
```

```
[13] 1 my_list.remove(10)  
    2 print(my_list)
```

```
[14] 1 new_list = []  
    2 print(new_list)  
    3 type(new_list)
```

```
[15] 1 new_list.append('first')  
    2 print(new_list)  
    3 new_list.append('second')  
    4 print(new_list)
```

FOR LOOPS

```
1 print(0)
2 print(1)
3 print(2)
4 print(3)
5 print(4)
```

```
1 for item in [0,1,2,3,4]:
2     print(item)
```

```
1 print(range(5))
```

```
1 for item in range(5):
2     print(item)
```



OUR DATA SET

- possum_mf = [0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1]
- possum_total = [89, 91, 95, 92, 85, 90, 89, 91, 91, 89, 89, 92, 89, 91, 85, 86, 89, 90, 90, 89, 96, 91, 89, 84, 91, 90, 85, 87, 88, 84, 93, 94, 89, 85, 85, 88, 82, 80, 75, 84, 77, 81, 81, 89, 85, 85, 88, 85, 93, 91, 91, 92, 93, 93, 91, 96, 88, 86, 90, 88, 89, 88, 86, 85, 88, 88, 87, 90, 80, 82, 83, 89, 89, 84, 81, 81, 84, 85, 82, 81, 80, 92, 86, 93, 87, 84, 85, 89, 85, 82, 84, 88, 83, 86, 84, 86, 81, 82, 89, 82, 89]



READ IN DATA

```
1 possum_mf = [0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0]
2 possum_total = [89, 91, 95, 92, 85, 90, 89, 91, 91, 89, 89, 92, 89, 91, 85, 8
```

```
1 print(possum_mf)
2 print(possum_total)
```

```
1 len(possum_mf)
```

```
1 for i in range(len(possum_mf)):
2     print(i, possum_mf[i])
```



IF STATEMENTS

```
1 for i in range(len(possum_mf)):
2     if i == 1:
3         print(i)
```

```
1 # can skip and do both together
2 female_total = []
3
4 for i in range(len(possum_mf)):
5     if possum_mf[i] == 1:
6         female_total.append(possum_total[i])
7
8 print(female_total)
9 print(sum(female_total)/len(female_total))
```

```
1 # can skip and do both together
2 male_total = []
3
4 for i in range(len(possum_mf)):
5     if possum_mf[i] == 0:
6         #print(possum_num[data])
7         male_total.append(possum_total[i])
8
9 print(male_total)
10 print(sum(male_total)/len(male_total))
```

```
1 # can skip if done these separately
2 male_total = []
3 female_total = []
4
5 for i in range(len(possum_mf)):
6     if possum_mf[i] == 0:
7         male_total.append(possum_total[i])
8     else:
9         female_total.append(possum_total[i])
10
11 print(sum(male_total)/len(male_total))
12 print(sum(female_total)/len(female_total))
```



PLOTTING

```
1 plt.plot(male_total, label = 'male', color = 'g')
2 plt.plot(female_total, label = 'female', color = 'k')
3 plt.title('possum total length')
4 plt.xlabel('number')
5 plt.ylabel('possum total length (cm)')
6 plt.legend()
```

```
1 plt.hist(female_total, alpha = 0.5, label = 'female')
2 plt.hist(male_total, alpha = 0.5, label = 'male')
3 plt.title('possum total length')
4 plt.xlabel('possum total length (cm)')
5 plt.ylabel('number')
6 plt.legend()
```



HOW DO I LEARN MORE?

- Practice programming in Python
- Use the resources on the web
- Practice!!
- Get to grips with the libraries such as:
 - NumPy, pandas and scikit-learn





https://en.wikipedia.org/wiki/Rubber_duck_debugging



QUESTIONS!

Dr Tamara Celford 

https://tamaracleford.co.uk/online_courses.html

