



# Chapter 01. Introduction & Environments

Python Programming for Bioinformatics

Robert C. Chi

# Agenda

- **About This Course**
- **Python Introduction**
- **Development Environment**





# **ABOUT THIS COURSE**

# Robert C. Chi (紀俊男)

- **Education**

- Ph.D. Candidate / Bioinformatics  
*Taiwan International Graduate Program (TIGP), 2003-2007*
- Master / Computer Sciences  
*Queens College, CUNY, 1994-1996*
- Bachelor / Computer Sciences  
*Fu-Jen Catholic University*

- **Experience**

- Training Director / AMI (2014-2020)
- Founder / Hatch Information Co., Ltd. (2007-2013)
- Research Assistant / Academia Sinica (2000-2007)
- Manager of Tech Support / Trend Micro Co., Ltd. (1998-2000)
- Game Developer / CG Animation Co., Ltd. (1997-1998)

- **Expertise**

- Artificial Intelligence (AI), Embedded System, Computer Security, Game Programming.





# Syllabus

- **Part I. Python (10 Hr)**

- Python & Environments
- Literals & Variables
- Input & Output
- Branch & Loop
- String Manipulation
- Compound Data Types
  - Tuple, List, Dictionary, Set
- Functions
- Data Science Packages
  - NumPy, Pandas, Matplotlib

- **Part II. BioPython (14 Hr)**

- Introduction
- Read/Write Bioinformatic Files
  - FASTA, GenBank, SwissProt, ExPASy, KEGG...
- Sequence Manipulation
  - Transcription, Translation, Alignment
- Databases Handling
  - BLAST, NCBI Entrez...
- Working with 3D Structures
- Machine Learning
  - Data Pre-Processing
  - Classification
  - Clustering



# Schedule, Environment, & Scoring

- **Schedule**
  - Part I
    - 2021/08/06 ~ 2021/09/03
    - Fri. 15:00 ~ 17:00
  - Part II
    - To Be Discussed
- **Location: Online**
  - <https://www.gotomeet.me/TeacherChi/BioPython>
- **Environments**
  - Google Colab
- **Lecturing in**
  - English
- **Teaching Style**
  - Part I: Lecturing
  - Part II: Lecturing + Practicing

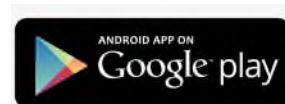
# Live Broadcasting

- URL: <https://www.gotomeet.me/TeacherChi/BioPython>



## App Download

**GoToMeeting**

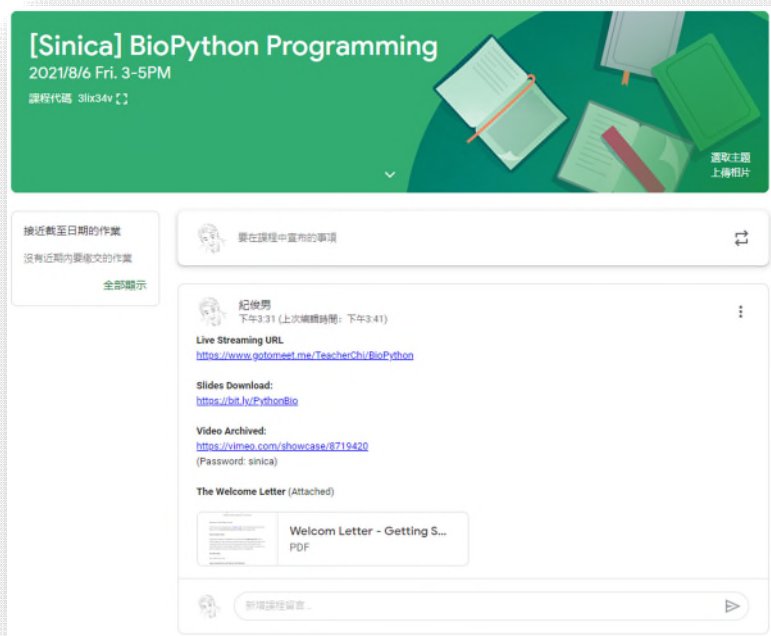


Meeting ID : 117-684-245



# Resources & Courses Make Up

- **Google Classroom** : <https://bit.ly/BioPy-202108>



## App Download



**Google  
Classroom**



Join ID: **3lix34v**





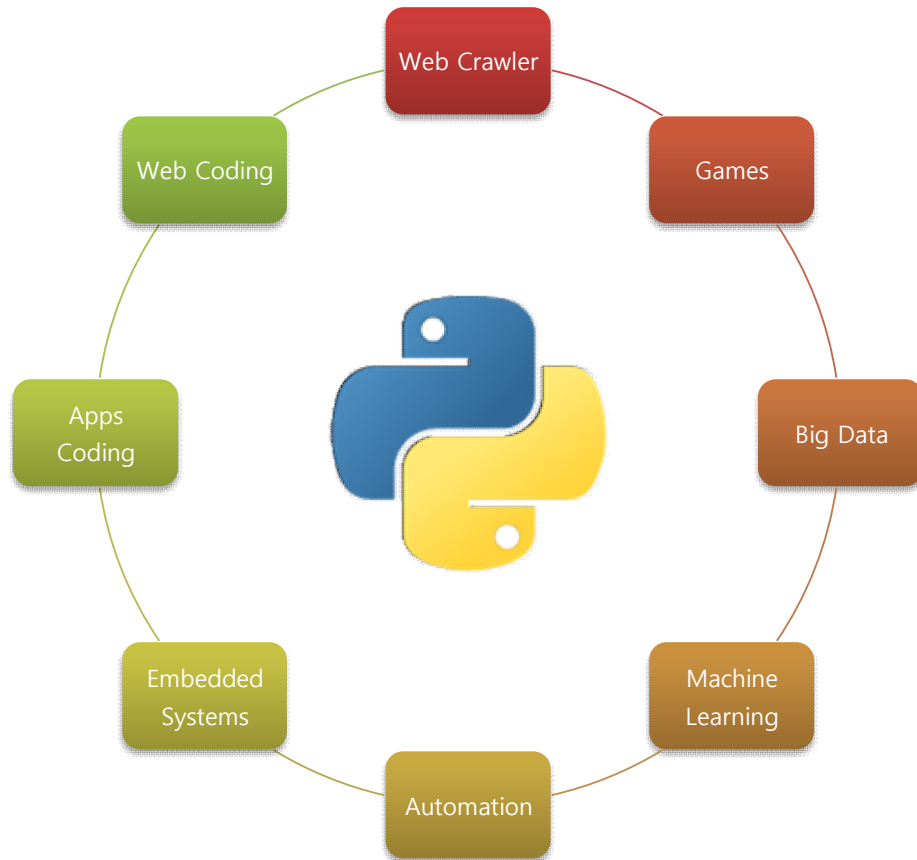
# PYTHON INTRODUCTION

# Inventor and Reason



- **Inventor**
  - Guido van Rossum
  - Netherlanders
- **Reason**
  - December 1989
  - Boring & looking for something to kill the time.
  - Wanna make a simple, easy to learn, yet powerful programming language.

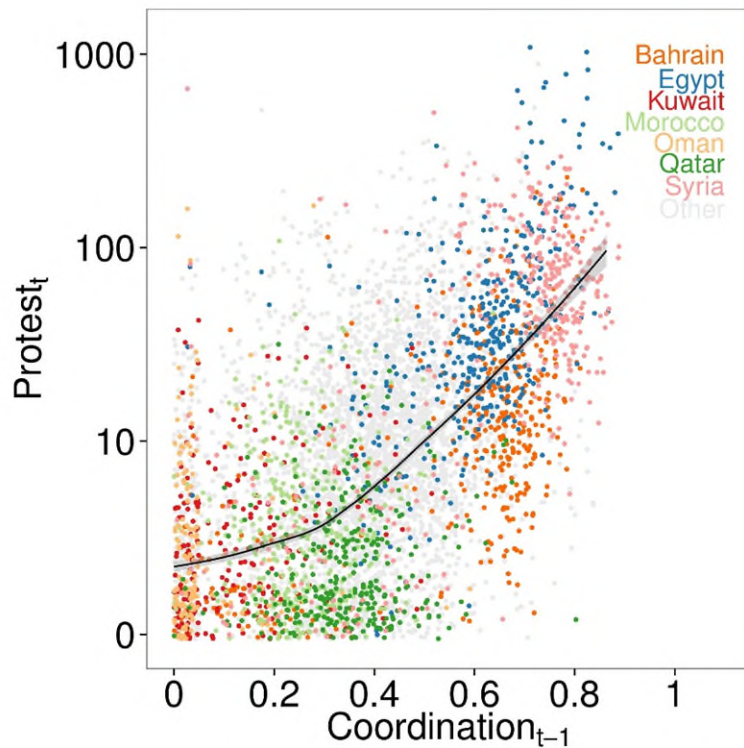
# Applications



# Big Data



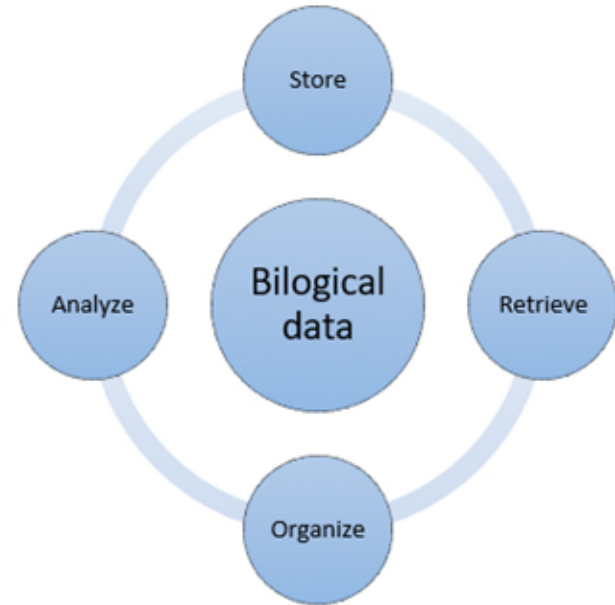
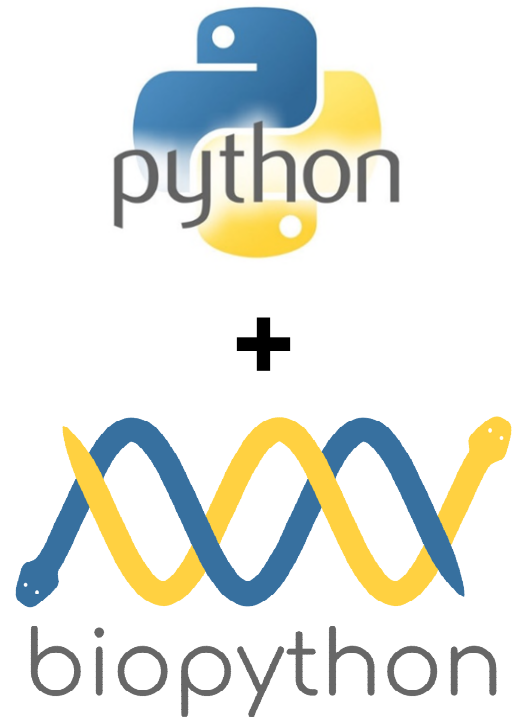
+



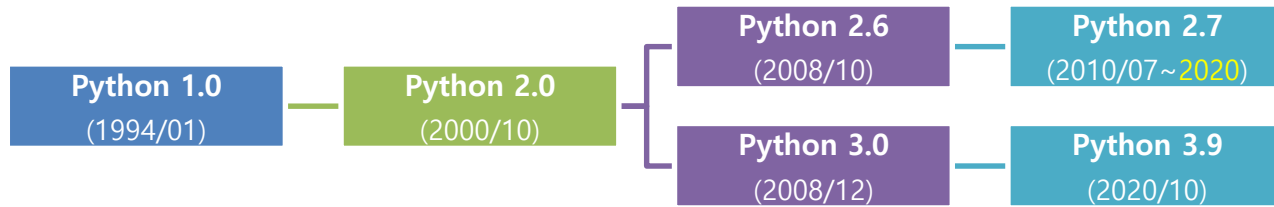
# Machine Learning



# Bioinformatics



# The Versions of Python



- Python 1.0 (1994/01)
- Python 1.5 (1997/12/31)
- Python 1.6 (2000/09/05)
- Python 2.0 (2000/10/16)
- Python 2.1 (2001/04/17)
- Python 2.2 (2001/12/21)
- Python 2.3 (2003/07/29)
- Python 2.4 (2004/11/30)
- Python 2.5 (2006/09/19)
- Python 2.6 (2008/10/01)
- Python 2.7 (2010/07/03)
- Python 3.0 (2008/12/03)
- Python 3.1 (2009/06/27)
- Python 3.2 (2011/02/20)
- Python 3.3 (2012/09/29)
- Python 3.4 (2014/03/16)
- Python 3.5 (2015/09/13)
- Python 3.6 (2016/12/23)
- Python 3.7 (2018/06/27)
- Python 3.8 (2019/10/14)
- Python 3.9 (2020/10/05)



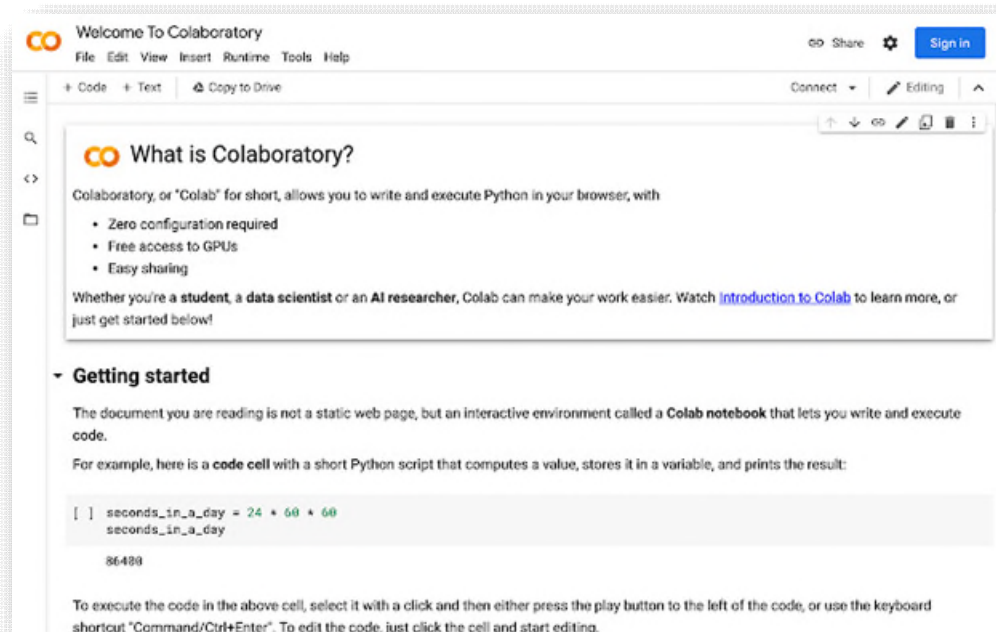


Google Colab

# DEVELOPMENT ENVIRONMENT

# What is “Colab”?

- **Free**, **cloud-based Jupyter Notebook** provided by **Google**.



# How to Get into Colab?

- <https://colab.research.google.com>

## Colab Welcome Page



Google Account



The screenshot shows the 'Welcome To Colaboratory' interface. At the top, there is a navigation bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. Below this is a 'Table of contents' sidebar on the left with categories like 'Getting started', 'Data science', 'Machine learning', 'More Resources', and 'Machine Learning Examples'. The main content area is titled 'What is Colaboratory?' and contains the following text:

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:

```
[ ] seconds_in_a_day = 24 * 60 * 60
seconds_in_a_day
```

86400

# Create a New File

Change File Name

Auto-Saved

All changes saved

New File

The screenshot shows the Google Colab interface. The top bar includes the Colab logo, a notebook name 'Test.ipynb', and a star icon. The menu bar contains 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. The 'File' menu is open, showing options like 'New notebook', 'Open notebook', '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', 'Download .ipynb', 'Download .py', 'Update Drive preview', and 'Print'. A file explorer overlay is visible, showing a list of folders under '我的雲端硬碟'. The 'Colab Notebooks' folder is highlighted with a red box and labeled 'Default Location'.

名稱	擁有者
影片	我
Classroom	我
Colab Notebooks	我
Google 相簿	我
Google 筆記本存檔	我

# Introduction of UIs

Outline  
Snippets  
Files

The screenshot shows the JupyterLab interface for a file named 'Test.ipynb'. The interface is annotated with red text and arrows pointing to specific UI elements:

- Outline Snippets Files:** A vertical list on the left side of the interface.
- Upload, Refresh, Mount:** Three vertical red labels pointing to icons in the top-left file management area.
- Add Comments:** A red label pointing to the 'Comment' button in the top-right toolbar.
- Share Page:** A red label pointing to the 'Share' button in the top-right toolbar.
- Virtual Machine:** A red label pointing to the 'RAM' and 'Disk' sliders in the top-right toolbar.
- Text Input:** A red label pointing to the text '測試檔' (Test File) in the file browser.
- Code Input:** A red label pointing to the code cell containing `print("Hello!")`.

The interface also shows a menu bar (File, Edit, View, Insert, Runtime, Tools, Help), a file browser on the left with a 'sample\_data' folder, and a status bar at the bottom indicating '76.49 GB available'.

# Edit a “Text Area”

New Text Area

The screenshot shows a text editor interface with a top bar containing '+ Code' and '+ Text' buttons. The '+ Text' button is highlighted with a red box. To the right of the top bar, there are indicators for RAM and Disk usage, an 'Editing' mode indicator, and a 'Delete Text Area' button (trash icon) highlighted with a red box. Below the top bar is a rich text editor toolbar with icons for bold, italic, link, image, list, and other formatting options. The main editing area is split into two columns. The left column contains the text '# Quadratic Equation' (highlighted with a red box) and 'Markdown for Text', followed by 'Formula:' and '\$\$f(x)=3x^2+2x+1\$\$' (highlighted with a red box), and 'LaTeX for Formula' and '\$\$ LaTeX Syntax \$\$'. The right column is labeled 'Preview' and shows the rendered output: 'Quadratic Equation' and 'Formula:  $f(x) = 3x^2 + 2x + 1$ '.

- Markdown Reference : <https://bit.ly/3ivPeBj>
- LaTeX Reference : <https://bit.ly/3iscPCR>

# Edit & Run a “Code Area”

Add New Code Area

The screenshot shows a code editor window with a toolbar at the top containing '+ Code' and '+ Text' buttons. Below the toolbar is a section titled 'Quadratic Equation' with a 'Formula:' label and the equation  $f(x) = 3x^2 + 2x + 1$ . Below the formula is another code editor area with a '+ Code' button and a play button. The code in this area is:

```
1 # Import Required Libraries
2
3 import matplotlib.pyplot as plt
4 import numpy as np

[ ] 1 x = np.linspace(-1, 1, 50)
     2 y = 3*x**2 + 2*x + 1
     3
     4 plt.plot(x, y)
     5 plt.show()
```

Only Execute this Area

Run All Codes

Runtime	
Run all	Ctrl+F9
Run before	Ctrl+B
Run the focused cell	Ctrl+Enter
Run selection	Ctrl+Shift+Enter
Run after	Ctrl+F10

Enter Python Source Code

Full Source: <https://bit.ly/2VC7Lms>



# Use “Scratch Code Cell”

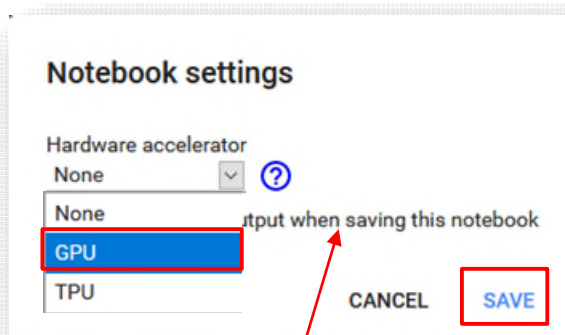
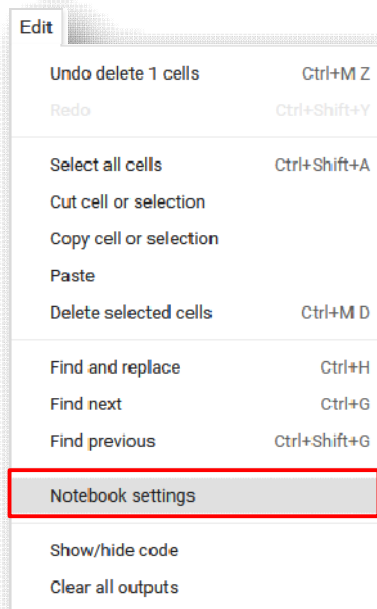
The screenshot shows a Jupyter Notebook interface for a file named 'Test.ipynb'. The top menu bar includes 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. A 'Create New' dropdown menu is open, listing options: 'Code cell' (Ctrl+M B), 'Text cell', 'Section header cell', 'Scratch code cell' (Ctrl+Alt+N), and 'Code snippets' (Ctrl+Alt+P). The 'Scratch code cell' option is highlighted with a red box. A red arrow points from this option to a 'Scratch cell' window. The scratch cell contains a play button, a line of code '1 print("Hello!")', and the output 'Hello!'. The main notebook area shows a 'Quadratic' section with two code cells. The first code cell contains imports for matplotlib and numpy. The second code cell contains a plot of a quadratic function.

**Create New**

**Test any Python Code Temporarily**

# Obtain GPU/TPU Acceleration

- **Edit > Notebook settings**



**Only Save Source Code  
without Running Results**

**Notice:**

- GPU/TPU may not be available for free users.
- GPU/TPU resources are prioritized for Colab Pro users.
- Colab Pro monthly fee = US\$ 9.99 / month.

# Install Python Packages

**!pip list List all installed packages**

```
tensorboard 2.2.2
tensorboard-plugin-wit 1.6.0.post3
tensorboardcolab 0.0.22
tensorflow 2.2.0
tensorflow-addons 0.8.3
tensorflow-
tensorflow-
tensorflow-
tensorflow-privacy 0.2
tensorflow-probability 0.1
termcolor 1.1
terminado 0.8
```

**!pip list | grep keras Check for Specific Package**

```
keras-vis 0.4.1
```

**!pip install keras Install a New Package**

```
Requirement already satisfied: keras in /usr/local/lib/python3.6/dist-packages (2.3.1)
Requirement already satisfied: scipy>=0.14 in /usr/local/lib/python3.6/dist-packages (from keras) (3.
Requirement already satisfied: pyyaml in /usr/local/lib/python3.6/dist-packages (from keras) (3.
Requirement
Requirement
Requirement
Requirement
```

**import keras**

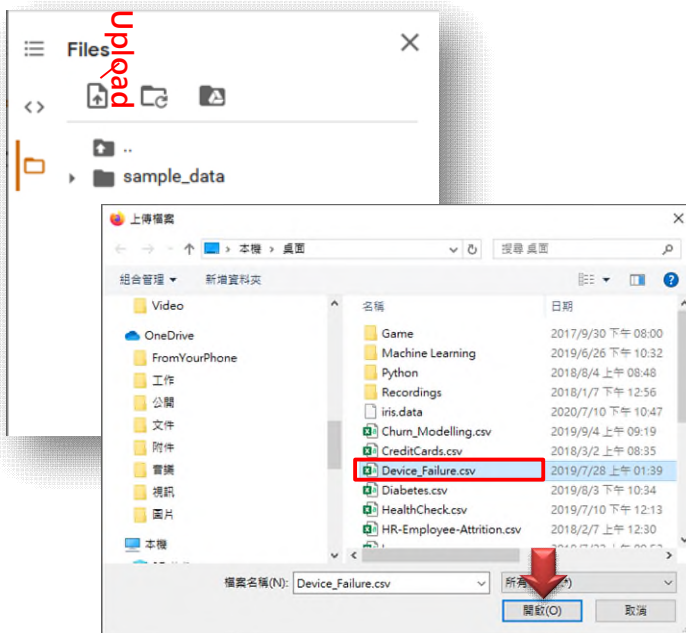
```
print(keras.__version__)
```

Using TensorFlow backend.  
2.3.1

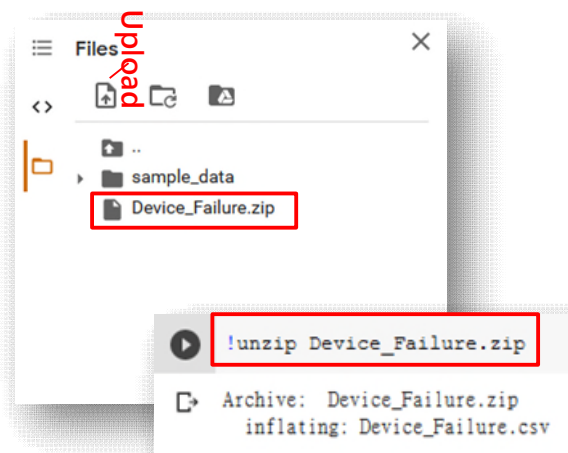
**Verify the Package has been installed**

# Upload Files onto Colab

- Upload Regular Files



- Upload ZIP file & UnZIP



**Notice:**

- All the files are stored in a temporary "virtual machine".
- Once the VM is unloaded (e.g., no action for 30 minutes), files will be deleted.
- If you want to save files permanently, use Google Drive alternatively.

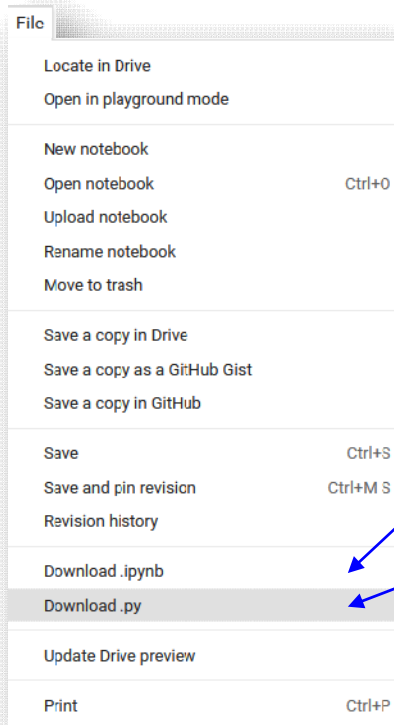
# Version Control of Colab

The image displays the Google Colab interface with several key elements highlighted:

- File Menu:** The 'Revision history' option is highlighted with a red box.
- Comparison View:** Two versions of a notebook are shown side-by-side. The left version (red border) is from 'Fri Jul 10 2020 22:55:44 GMT+0800' and contains a text cell with a markdown header and a code cell with a formula. The right version (green border) is from 'Fri Jul 10 2020 23:17:28 GMT+0800' and includes an additional code cell for unzipping a file.
- Version List:** A dropdown menu on the right shows a list of revisions. The 'Latest' version (Jul 10, 2020 11:17 PM) is selected. The 'Older' version (Jul 10, 2020 10:55 PM) is highlighted with a red box. A blue circle highlights the 'Rename', 'Open New', and 'Reverse to' options for the selected version.



# Export Your Source Codes



Download as a .ipynb file  
(Able to open by Jupyter Notebook)

Download as a .py file  
(Able to open by all Python Environments)

# Share Your Source Codes

The screenshot shows the Google Colab interface for a notebook named 'Test.ipynb'. The top navigation bar includes 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help', with a status indicator 'All changes saved'. A 'Share' button is highlighted with a red box. A 'Get link' dialog box is open, displaying a shareable URL: <https://colab.research.google.com/drive/1HWV5nxL7m8Ykm465RgijUbhIJD...>. The 'Copy link' button is also highlighted with a red box. The dialog shows the current permission is 'Anyone with the link' and a dropdown menu is open, showing 'Viewer' as the selected permission. The 'Done' button is at the bottom of the dialog. The background code cell contains the line `plt.plot(x, y)`.

Table of contents

一元二次方程式

Section

Get link

<https://colab.research.google.com/drive/1HWV5nxL7m8Ykm465RgijUbhIJD...>

Copy link

Anyone with the link

Anyone on the internet with this link can view

Viewers of this file can see comments and suggestions.

Feedback?

Viewer

Viewer

Commenter

Editor

Done

```
plt.plot(x, y)
```