



# Introduction to Python

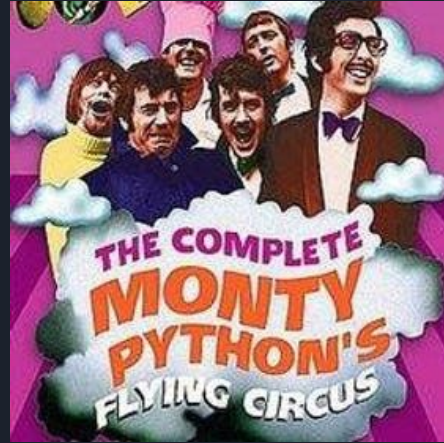
Tc Liu, LeCosPA, NTU

# Origin of Python



Python的創始人為吉多·范羅蘇姆。1989年的聖誕節期間，吉多·范羅蘇姆為了在阿姆斯特丹打發時間，決心開發一個新的腳本解釋程式，作為ABC語言的一種繼承。之所以選中Python作為程式的名字，是因為他是BBC電視劇——蒙提·派森的飛行馬戲團的愛好者。ABC是由吉多參加設計的一種教學語言。就吉多本人看來，ABC這種語言非常優美和強大，是專門為非專業程式設計師設計的。但是ABC語言並沒有成功，究其原因，吉多認為是非開放造成的。吉多決心在Python中避免這一錯誤，並取得了非常好的效果，完美結合了C和其他一些語言。

# Origin of Python



# Origin of Python





# What is Python

- Open source language
- Easy to couple with other languages  
C/Obj-C/Java/Fortran
- Great interactive environment
- Great supporting of the society

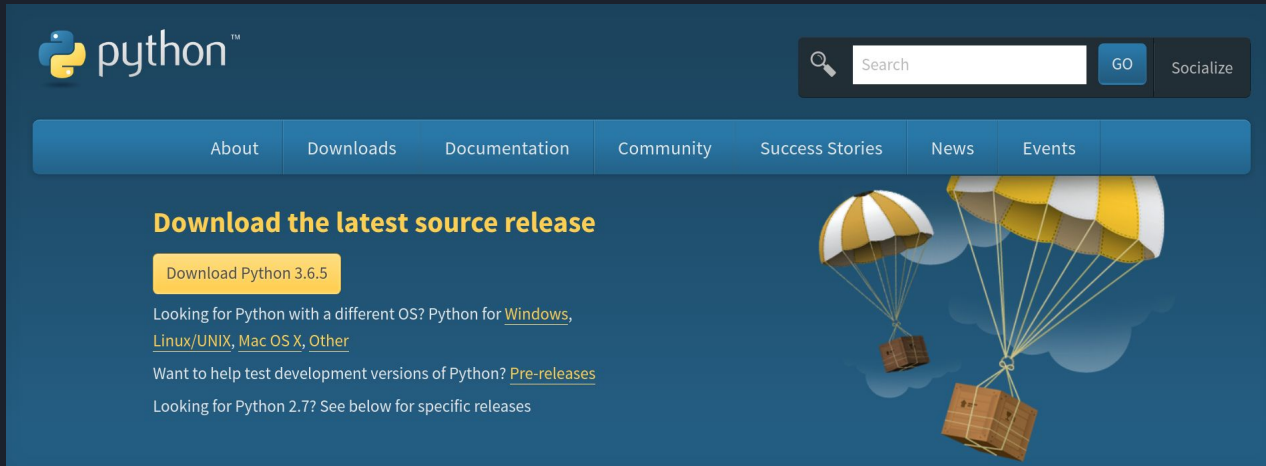


# Why Python?

- Cross platform
- Python is similar to the English language.
- Python runs on an interpreter system, This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-orientated way or a functional way.

# Where to get Python

- Python builtin in Mac and UNIX-like system
- Windows: <https://www.python.org/>



The screenshot shows the Python.org website homepage. At the top left is the Python logo and the word "python" with a trademark symbol. To the right is a search bar with a magnifying glass icon, a "GO" button, and a "Socialize" button. Below the search bar is a navigation menu with links for "About", "Downloads", "Documentation", "Community", "Success Stories", "News", and "Events". The main content area features a yellow button labeled "Download Python 3.6.5" under the heading "Download the latest source release". Below the button are links for "Looking for Python with a different OS? Python for [Windows](#), [Linux/UNIX](#), [Mac OS X](#), [Other](#)" and "Want to help test development versions of Python? [Pre-releases](#)". At the bottom, it says "Looking for Python 2.7? See below for specific releases". On the right side of the page, there is an illustration of two yellow and white striped parachutes with cardboard boxes hanging from them, set against a blue sky with white clouds.



# Python is used for:

- **Web and Internet Development**
- **Scientific and Numeric**
- **Education**
- **Desktop GUIs**
- **Desktop Software Development**
- **Business Applications**

[reference links](#)

[some demo of small projects](#)





# What version of Python should we install?

- Python 2.? or Python 3.?
- supporting list of Python 3

<https://python3wos.appspot.com/>



# What version of Python should we install?

- Python 2.? or Python 3.?
- supporting list of Python 3

<https://python3wos.appspot.com/>

Package	Downloads
pip	16768050
six	9890562
botocore	9586781
python-dateutil	9450877
s3transfer	8722433
futures (py3k)	7958917
docutils	7492012
pyasn1	7443069
pyyaml	7354000



# Books on Python Web

<https://wiki.python.org/moin/PythonBooks>

## Free Books

<https://codeburst.io/15-free-ebooks-to-learn-python-c299943f9f2c>



# II. Technical Issues

## Install the Python

Tc Liu, LeCosPA, NTU



# Where to get Python

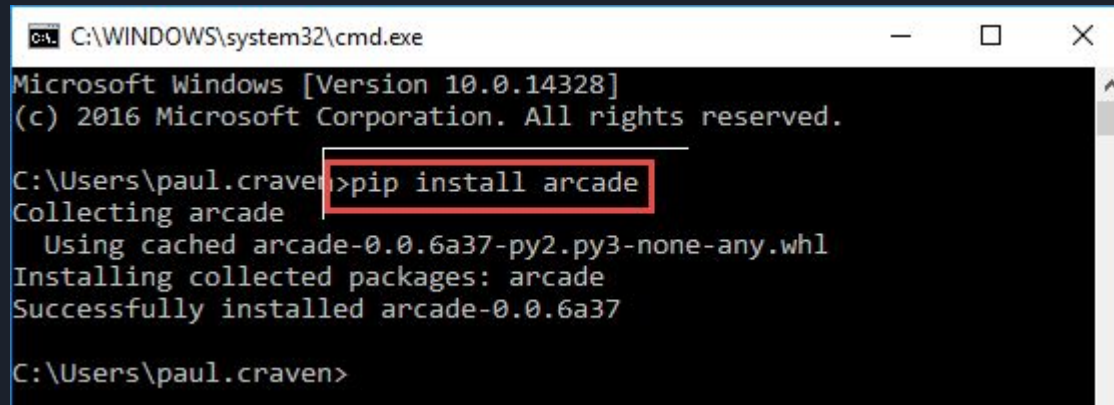
- Python builtin in Mac and UNIX-like system
  - Windows: <https://www.python.org/>

**or**

- ANAConda: Powerful collaboration and package managementor
  - Jupyter: Interactive computing developer

# install the Python and Packages

- Windows: <https://goo.gl/n5s3kz>
- ANAConda: <https://goo.gl/mp1P61>
- Jupyter: <http://www.largitdata.com/course/29/>
- package command: **pip install** *<package>*



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.14328]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\paul.craven>pip install arcade
Collecting arcade
  Using cached arcade-0.0.6a37-py2.py3-none-any.whl
Installing collected packages: arcade
Successfully installed arcade-0.0.6a37

C:\Users\paul.craven>
```



# Technical Issues

- Python 2.? or Python 3.?
- Supporting list of Python 3  
<https://python3wos.appspot.com/>



# Interactive Interface of Python on UNIX-like System

- Using `python` or `python3` to enter the interface

```
tcliu@localhost:~/Code/py_code/pygame/bullet
File Edit View Search Terminal Help
[tcliu@localhost bullet]$ python3
Python 3.5.4 (default, Aug 23 2017, 18:32:05)
[GCC 6.4.1 20170727 (Red Hat 6.4.1-1)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 2*(3+2)
10
>>> exit()
[tcliu@localhost bullet]$
```

- Using `exit()` or `CRTL+D` to quit the interface



# Running Code with Unix-like System

- python3 code.py

```
Applications  Places  Text Editor
Bullet.py
~/Code/py_code/pygame/bullet
Save

TkinterSoundRead.html  pythonindex.html  electricline2.py  electricline.py  TkinterMouse.html  Vpython.py  game01.py

# -*- coding: utf-8 -*-
import pygame
import math
import time
from sys import exit

def DrawLines():
    pygame.draw.line(screen, (255, 255, 255), (originX, originY), (originX+1200, originY));
    for i in range(0,12):
        screen.blit( font.render(str(i*10)+'m', True, (120, 120, 120)), (i*100+10, 410) );
        pygame.draw.line(screen, (120, 100, 100), (i*100+30, 400), (i*100+30, 150));

def Status():
    Vx2=Vx/ScaleFactor;
    Vy2=Vy/ScaleFactor;
    Angle = math.atan(float(Vy2)/Vx2)+degrees;
    print('Vx = %.1f m/s' % Vx);
    print('Vy = %.1f m/s' % Vy);
    print('Velocity = %.1f m/s' % Velocity);
    print('Angle = %.1f degrees' % Angle);

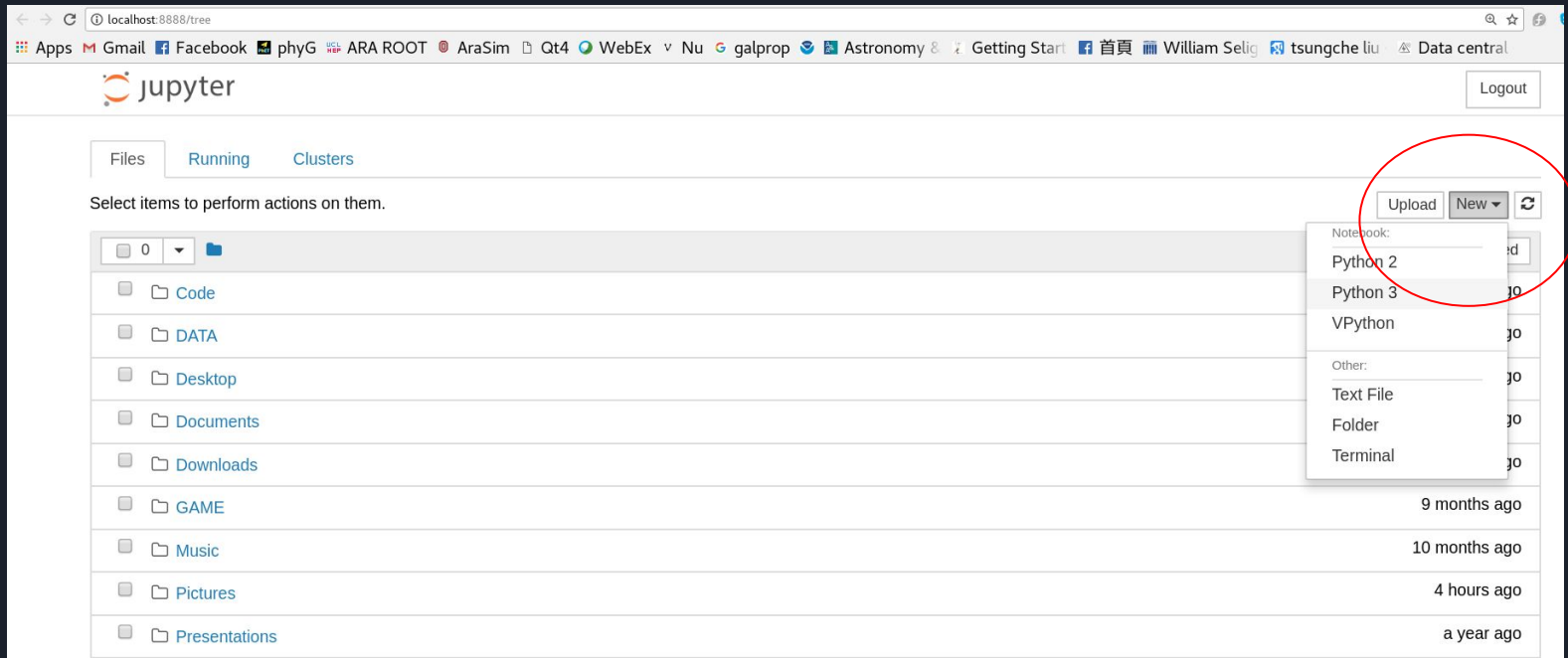
    (xList[i], yList[i]), 1);

    32);
caption of window

Python  Tab Width: 4  Ln 1, Col 1
```



# Running Code with Jupyter on Windows



The screenshot shows the Jupyter web interface in a browser window. The address bar displays 'localhost:8888/tree'. The top navigation bar includes the Jupyter logo and a 'Logout' button. Below the navigation bar, there are tabs for 'Files', 'Running', and 'Clusters'. The main content area shows a file tree with folders like 'Code', 'DATA', 'Desktop', 'Documents', 'Downloads', 'GAME', 'Music', 'Pictures', and 'Presentations'. A red circle highlights the 'New' dropdown menu, which is open and shows options for creating a notebook (Python 2, Python 3, VPython) and other files (Text File, Folder, Terminal). The 'Python 3' option is highlighted.

localhost:8888/tree

Apps Gmail Facebook phyG ARA ROOT AraSim Qt4 WebEx Nu galprop Astronomy & Getting Start 首頁 William Selig tsungche liu Data central

jupyter Logout

Files Running Clusters

Select items to perform actions on them.

0

- Code
- DATA
- Desktop
- Documents
- Downloads
- GAME
- Music
- Pictures
- Presentations

Upload New

Notebook:

- Python 2
- Python 3
- VPython

Other:

- Text File
- Folder
- Terminal

9 months ago

10 months ago


4 hours ago


a year ago













# Running Code with Jupyter on Windows



The screenshot displays the Jupyter Notebook interface. At the top, the title bar shows "jupyter" followed by "Untitled33 Last Checkpoint: a few seconds ago (unsaved changes)". On the right side of the title bar, there is a Python logo and a "Logout" button. Below the title bar is a menu bar with options: "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar, there is a "Trusted" status indicator, a pencil icon, and "Python 3" with a dropdown arrow. Below the menu bar is a toolbar with icons for saving, adding, deleting, copying, pasting, undo, redo, and running code. The main area of the notebook shows a code cell with the text "In [ ]:" followed by "1 5\*(2+3)|". A red circle highlights the code "5\*(2+3)|".

jupyter Untitled33 Last Checkpoint: a few seconds ago (unsaved changes)  Logout

File Edit View Insert Cell Kernel Help Trusted  Python 3 

        Run   Code  

In [ ]: 1 5\*(2+3)|

# Running Code with Jupyter on Windows

jupyter Untitled33 Last Checkpoint: a few seconds ago (unsaved changes)  Logout

File Edit View Insert Cell Kernel Help

Trusted  Python 3 

 Run **press Run**

```
In [ ]: 1 5*(2+3)
```

File Edit View Insert Cell Kernel Help

Trusted  Python 3 

 Run   Code 

```
In [1]: 1 5*(2+3)
```

```
Out[1]: 25
```

```
In [ ]: 1
```

# Running Code with Jupyter on Windows

jupyter Untitled33 Last Checkpoint: a few seconds ago (unsaved changes)  Logout

File Edit View Insert Cell Kernel Help

Trusted  Python 3 

 Run **press Run**

In [ ]: 1 5\*(2+3)

File Edit View Insert Cell Kernel Help

Trusted  Python 3 

**Shift+Enter**

In [1]: 5\*(2+3)

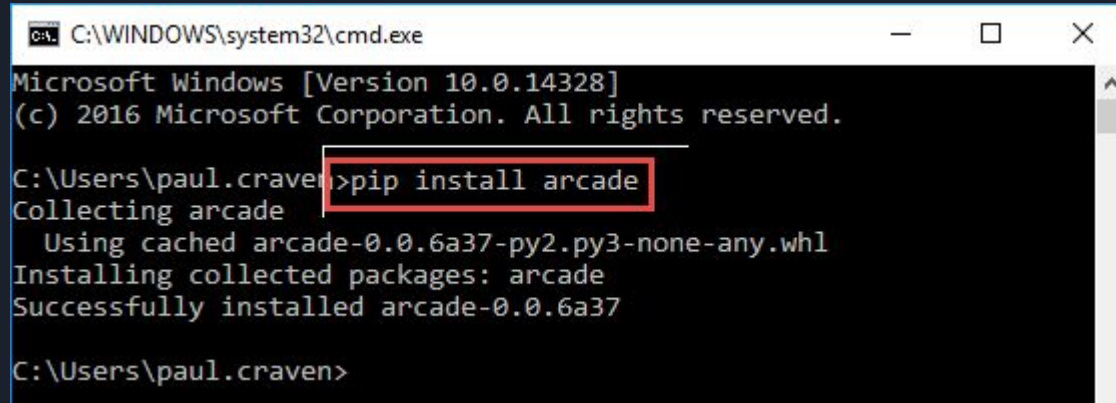
Out [1]: 25

In [ ]: 1

# Extend the functions to Python: install Package

- Running the CMD on Windows or Running the terminal on Unix-like system

command: **pip install** *<package>*



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.14328]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\paul.craven>pip install arcade
Collecting arcade
  Using cached arcade-0.0.6a37-py2.py3-none-any.whl
Installing collected packages: arcade
Successfully installed arcade-0.0.6a37

C:\Users\paul.craven>
```



# III. Library (Packages)

Tc Liu, LeCosPA, NTU



# What is Packages

The additional functions of the Python





# What is Packages

The additional functions of the Python

*ex: **Os**: The os module provides dozens of functions for interacting with the operating system*



# What is Packages

The additional functions of the Python

*ex: **Os**: The os module provides dozens of functions for interacting with the operating system*

```
try:  
import os  
os.getcwd()  
dir(os)
```

```
try:  
import os  
os.system('dir')
```



# What is Packages

The additional functions of the Python

*ex: **Math**: The math module gives access to the underlying C library functions for floating point math:*

```
try:  
import math  
dir(math)
```

```
try:  
    math.cos(1)  
try:  
    math.hypot(1, math.sqrt(3))
```



# What is Packages

The additional functions of the Python

*ex: **random**: The random module provides tools for making random selections:*

```
try:  
import random  
random.random()  
random.sample(range(47), 6)  
random.choice(['apple', 'pear', 'banana'])
```



# What is Packages

The additional functions of the Python

*ex: **statistics**: The statistics module calculates basic statistical properties (the mean, median, variance, etc.) of numeric data:*

```
try:  
import statistics  
dataS = [1, 3, 5, 7, 9, 11, 130]  
statistics.mean(dataS)  
statistics.median(dataS)  
statistics.variance(dataS)
```



# What is Packages

The additional functions of the Python

*ex: **datetime**: The date and time libs of Python*

try:

```
from datetime import date
```

```
now = date.today()
```

```
now
```

```
now.strftime("%m-%d-%y. %d %b %Y is a %A on the %d  
day of %B.")
```

```
birthday = date(1964, 7, 31)
```

```
age = now - birthday
```

```
print("age is", age.days/365)
```



# What is Packages

The additional functions of the Python

*ex: **datetime**: The date and time libs of Python*

try:

```
from datetime import date
```

```
now = date.today()
```

```
now
```

```
now.strftime("%m-%d-%y. %d %b %Y is a %A")
```

```
birthday = date(1964, 7, 31)
```

```
age = now - birthday
```

```
print("age is", age.days/365)
```



# What is Packages

The additional functions of the Python

## More Standard Packages:

<https://docs.python.org/3/tutorial/stdlib.html>

## Give Me **MORE** Packages:

<https://python3wos.appspot.com/>





# SciPy; NumPy;

Offers Matlab-ish capabilities within Python ·  
Fast array operations · 2D arrays, multi-D arrays,  
linear algebra etc.

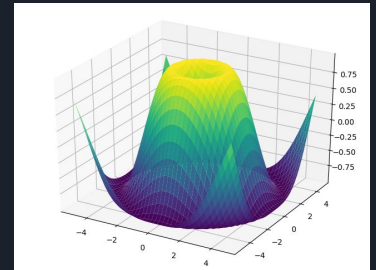
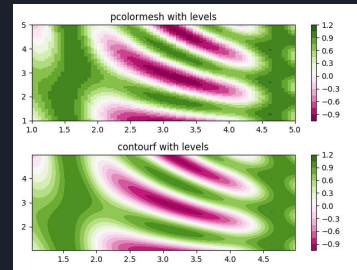
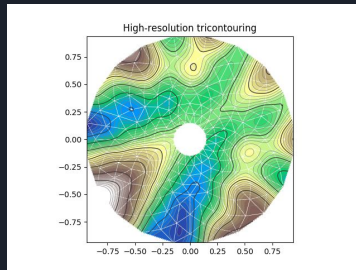
- Downloads: <https://www.scipy.org/>
- User guide: <https://docs.scipy.org/doc/>

# Matplotlib;

## High quality plotting library

- Downloads: <https://www.scipy.org/>
- User guide:

<https://matplotlib.org/tutorials/index.html>





# PyFITS

**The PyFITS module is a Python library providing access to FITS files**

- User guide: <https://pythonhosted.org/pyfits/>
- download some sample FITS file  
<https://data.nasa.gov/>

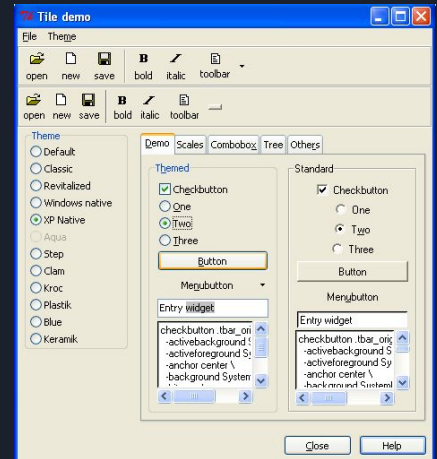
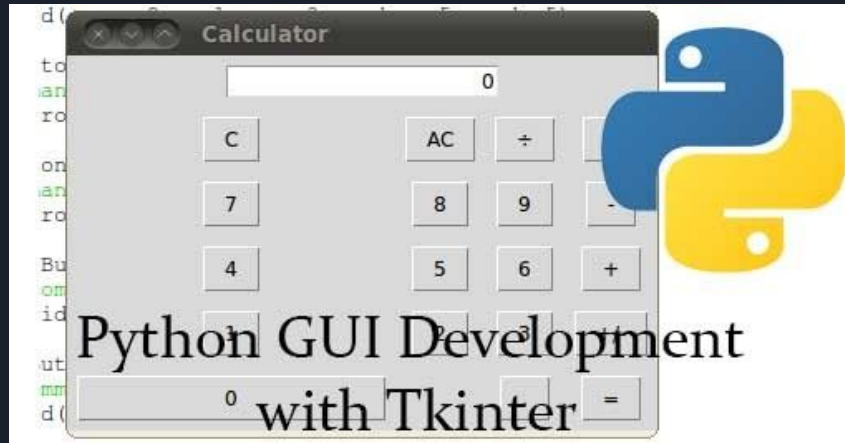
```
try:  
    import pyfits  
    hdulist = pyfits.open('input.fits')  
    hdulist.info()
```

# GUI: Tkinter, wxPython...

## The graphic user interface

User guide:

<https://docs.python.org/2/library/tkinter.html>





# Some Astronomy links for Python:

Python Astronomy Modules: <http://astlib.sourceforge.net/>

Python Astronomer Wiki: <http://macsingularity.org/astrowiki/tiki-index.php?page=python>

AstroPy: <http://www.astro.washington.edu/users/rowen/AstroPy.html>

Python for Astronomers: <http://www.iac.es/sieinvens/siepedia/pmwiki.php?n=HOWTOs.EmpezandoPython>



# Sympy

SymPy is a Python library for symbolic mathematics. It aims to become a full-featured computer algebra system (CAS) while keeping the code as simple as possible in order to be comprehensible and easily extensible. SymPy is written entirely in Python.

**web:**

<http://www.sympy.org/en/index.html>

**Tutorial:**

<http://docs.sympy.org/latest/tutorial/index.html>

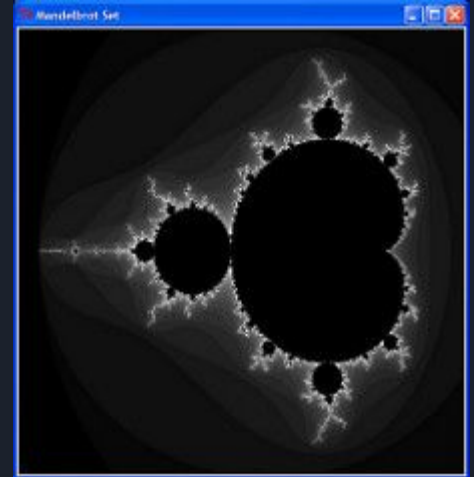
# GPU: pyCUDA

**PyCUDA:**

<https://documen.tician.de/pycuda/>

**Tutorial:**

<https://documen.tician.de/pycuda/tutorial.html>





# Machine Learning:

## SciKit-Learn:

<https://machine-learning-python.kspax.io/Introduction/intro.html>

# Deep Learning:

## Keras:

<https://fgc.stpi.narl.org.tw/activity/videoDetail/4b1141305d9cd231015d9d03cfd10027>

## TensorFlow:

<https://www.tensorflow.org/>