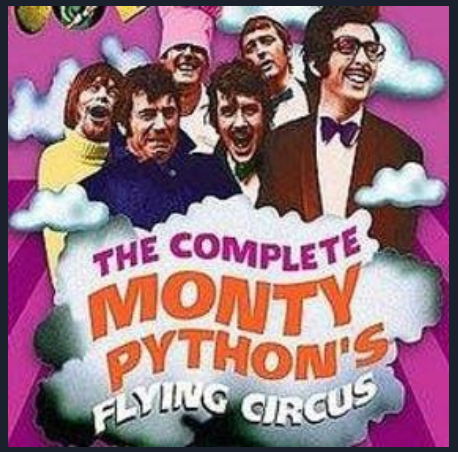
Introduction to Python

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

python™

What is Python

- Open source language
- Easy to couple with other languages

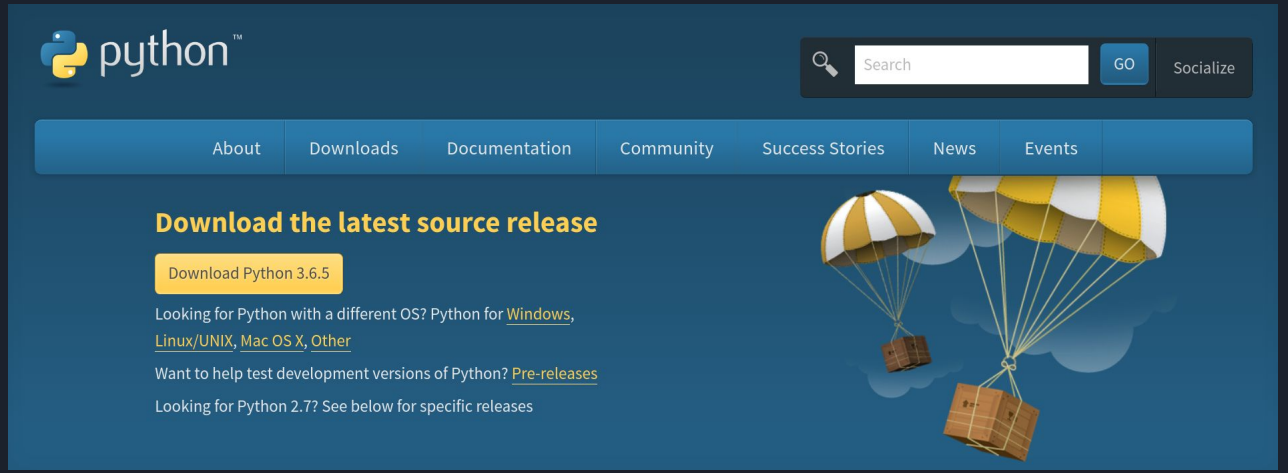
  C/Obj-C/Java/Fortran
- Great interactive environment
- Great supporting of the society

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/



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

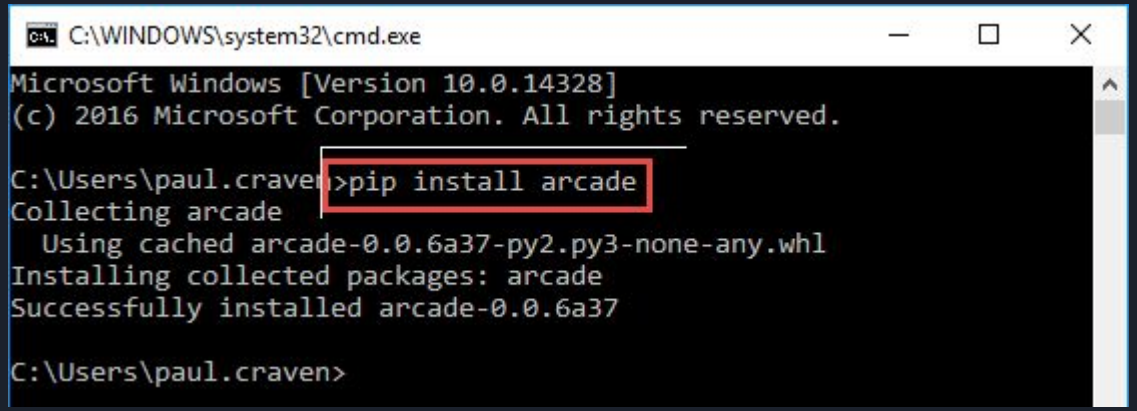
- Python builtin in Mac and UNIX-like system
  - Windows: https://www.python.org/

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>*

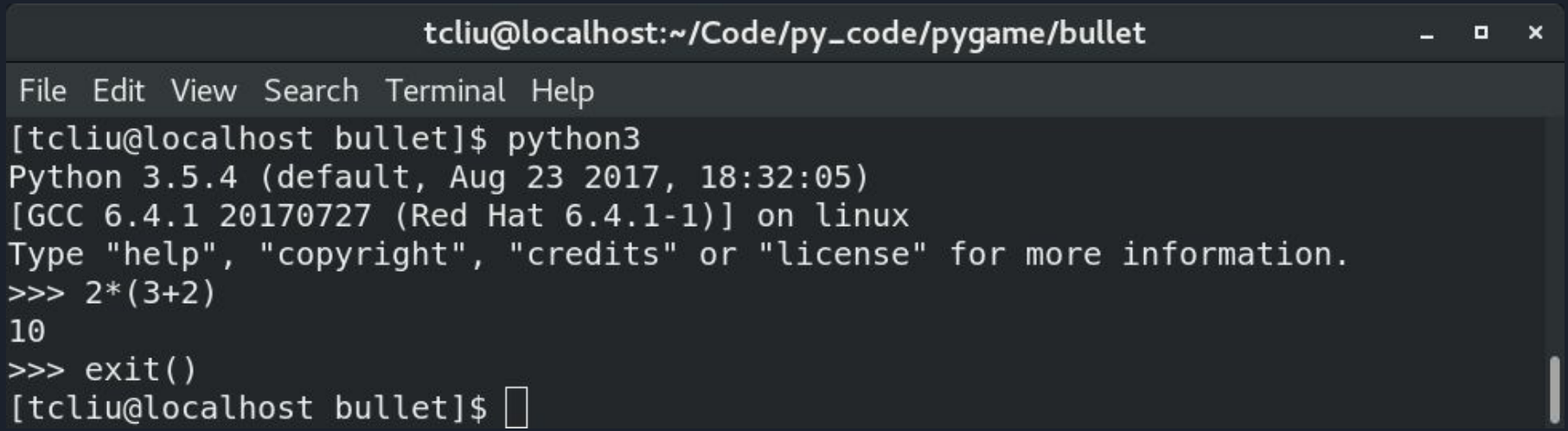


Technical Issues

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

Interactive Interface of Python on UNIX-like System

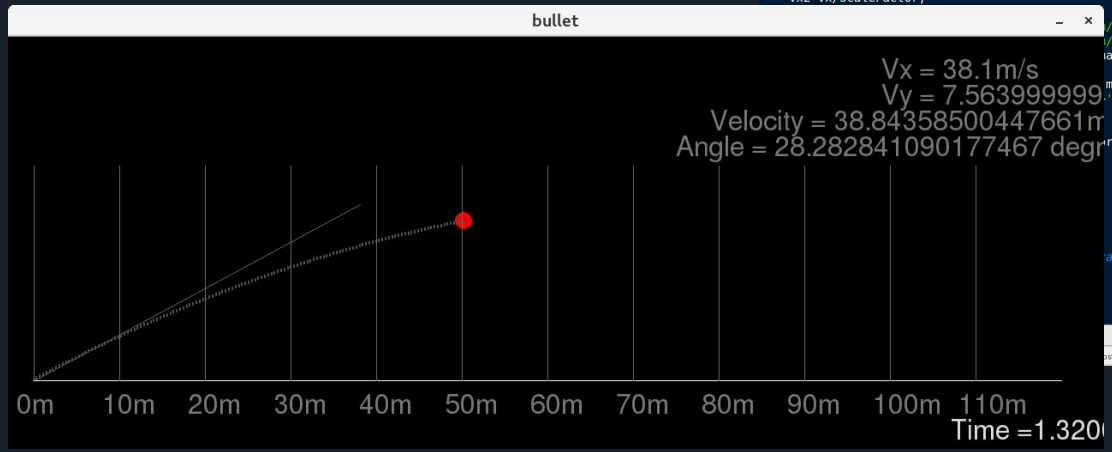
- Using python or python3 to enter the interface

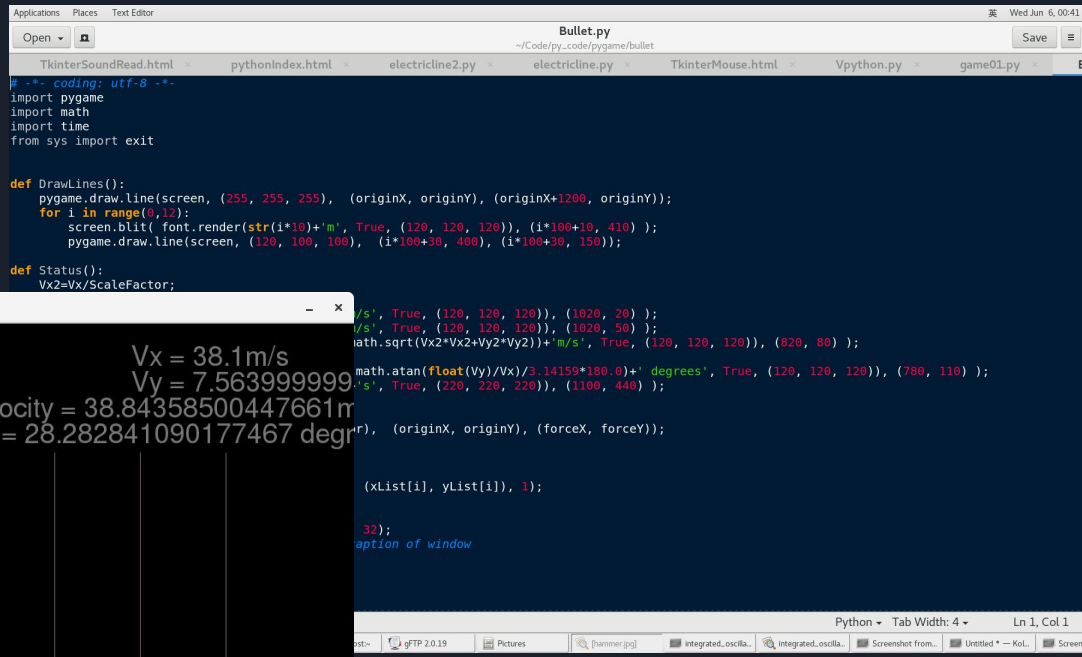


- Using exit() or CRTL+D to quit the interface

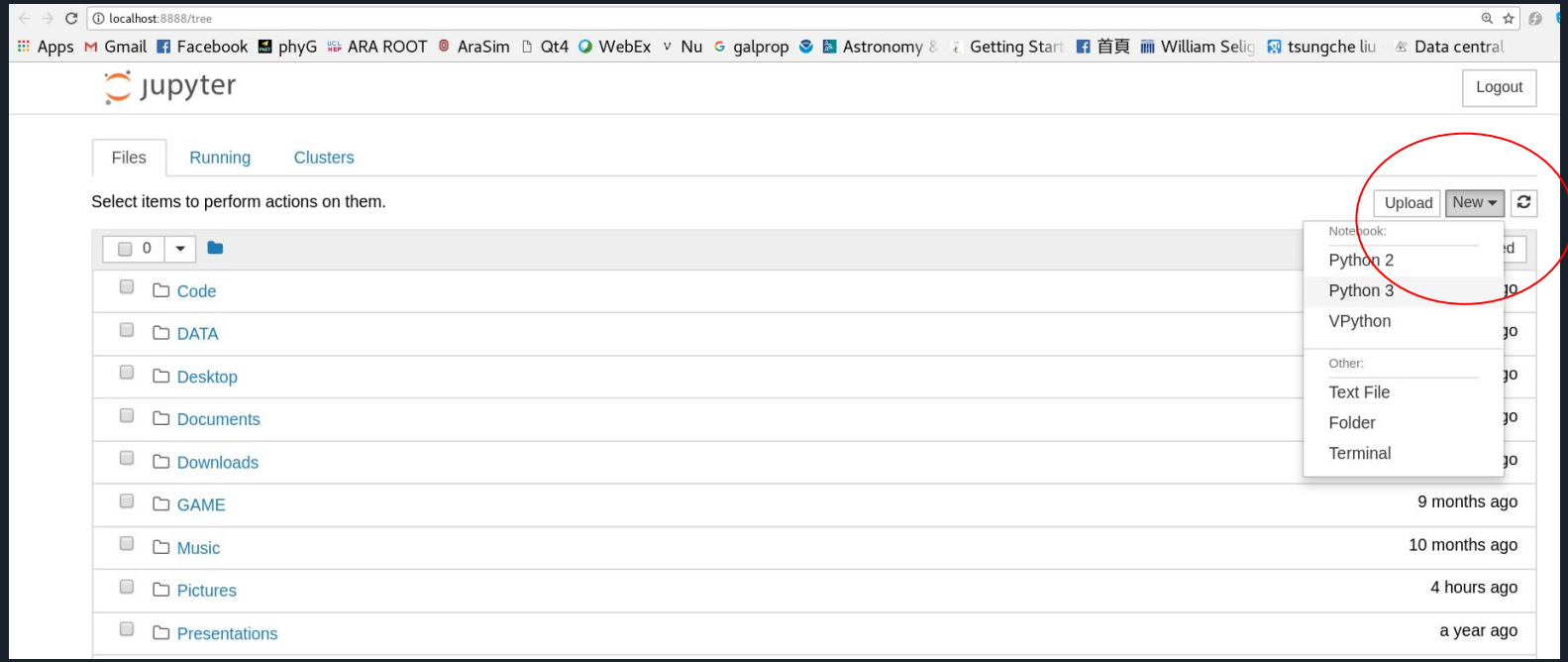
Running Code with Unix-like System

- python3 *code.py*



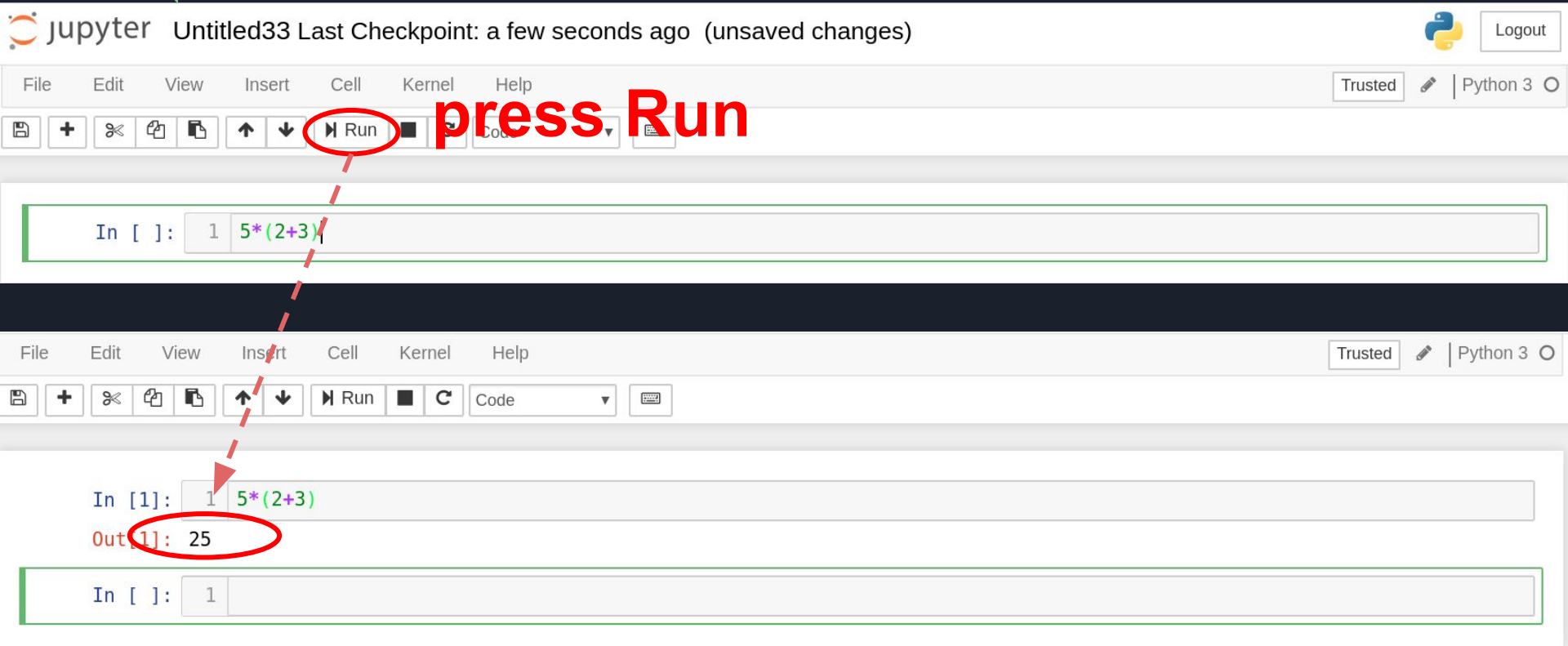


Running Code with Jupyter on Windows

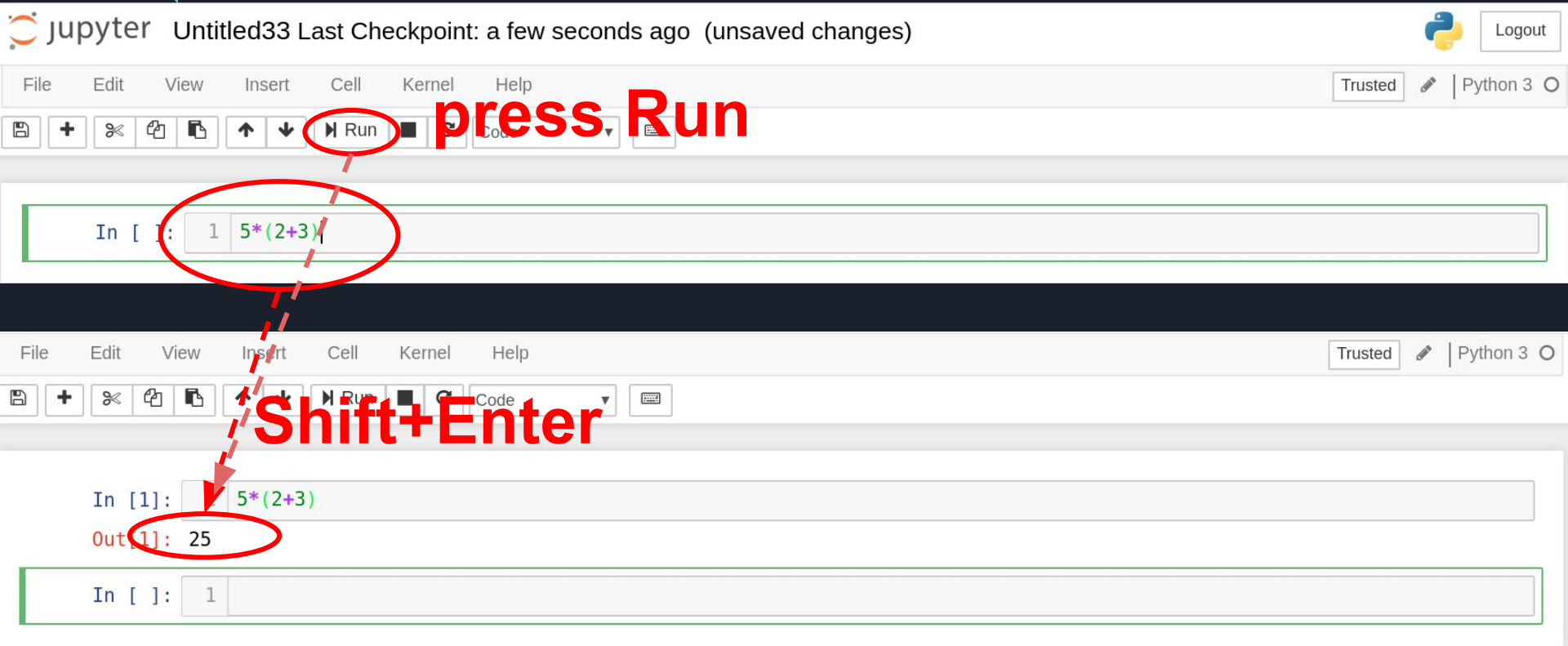


Running Code with Jupyter on Windows

Running Code with Jupyter on Windows



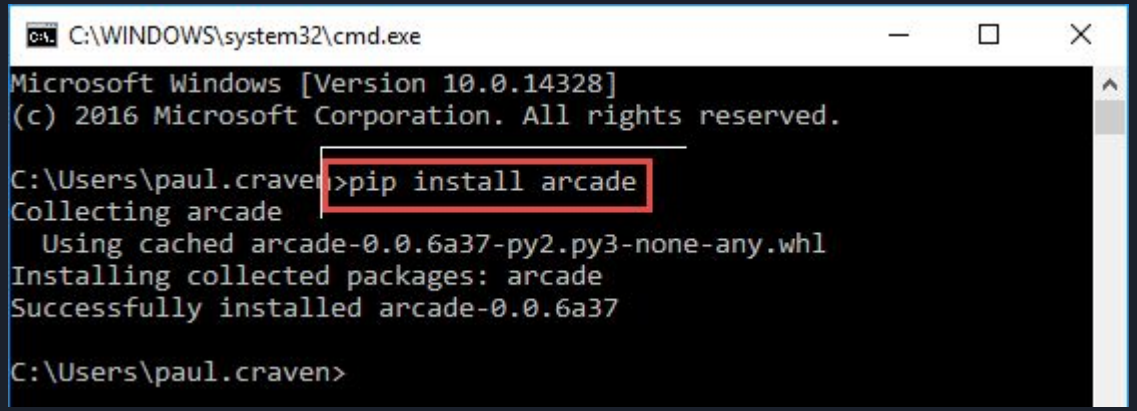
Running Code with Jupyter on Windows



Extend the functions to Python: install Package

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

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



III. Library (Packages)

Tc Liu, LeCosPA, NTU

What is Packages

The addiational functions of the Python

What is Packages

The addiational functions of the Python

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

What is Packages

The addiational 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 addiational 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 addiational 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 addiational 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 addiational 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 addiational 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 addiational 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.

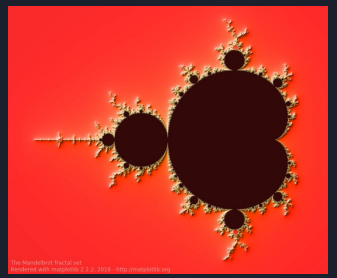
- Downloads: https://www.scipy.org/
- User guide: https://docs.scipy.org/doc/

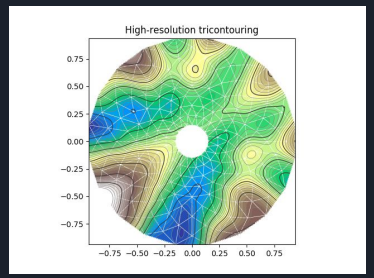
Matplotlib;

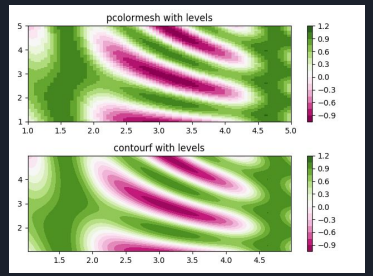
High quality plotting library

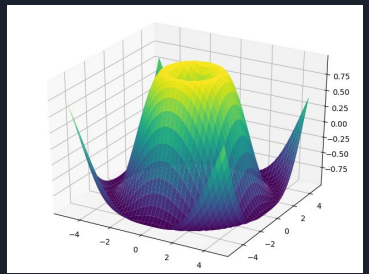
- Downloads: https://www.scipy.org/
- User guide:

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









PyFITS

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

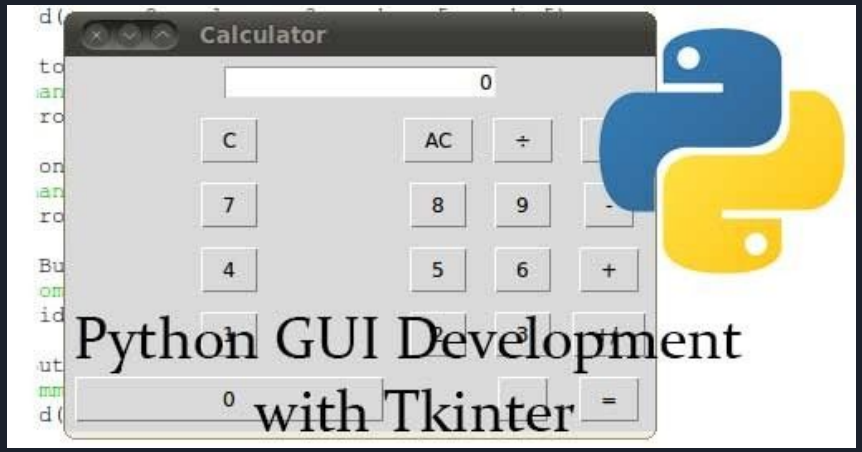
- User guide: https://pythonhosted.org/pyfits/
- download some sample FITS file https://data.nasa.gov/

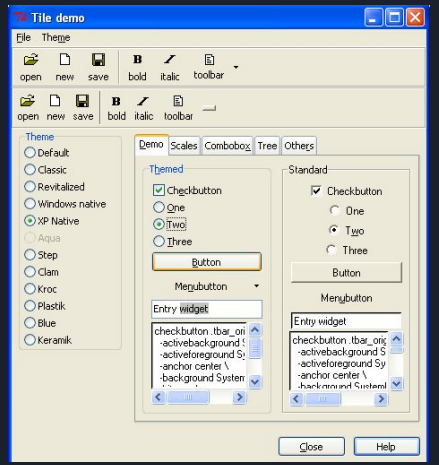
```
try:
import pyfits
hdulist = pyfits.open('input.fits')
hdulist.info()
```

GUI: Tkinter, wxPython....

The graphic user interface

User guide:
https://docs.python.org/2/library/tkinter.html





Some Astromony 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.

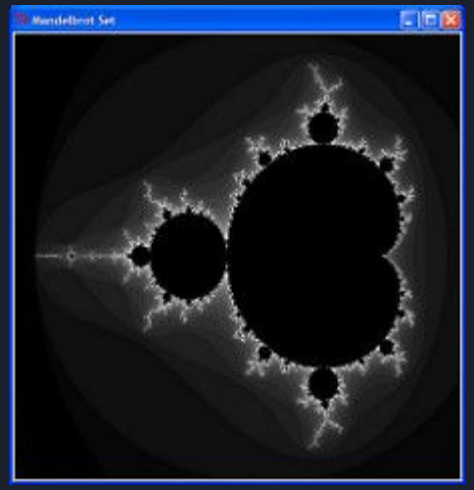
web:

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/