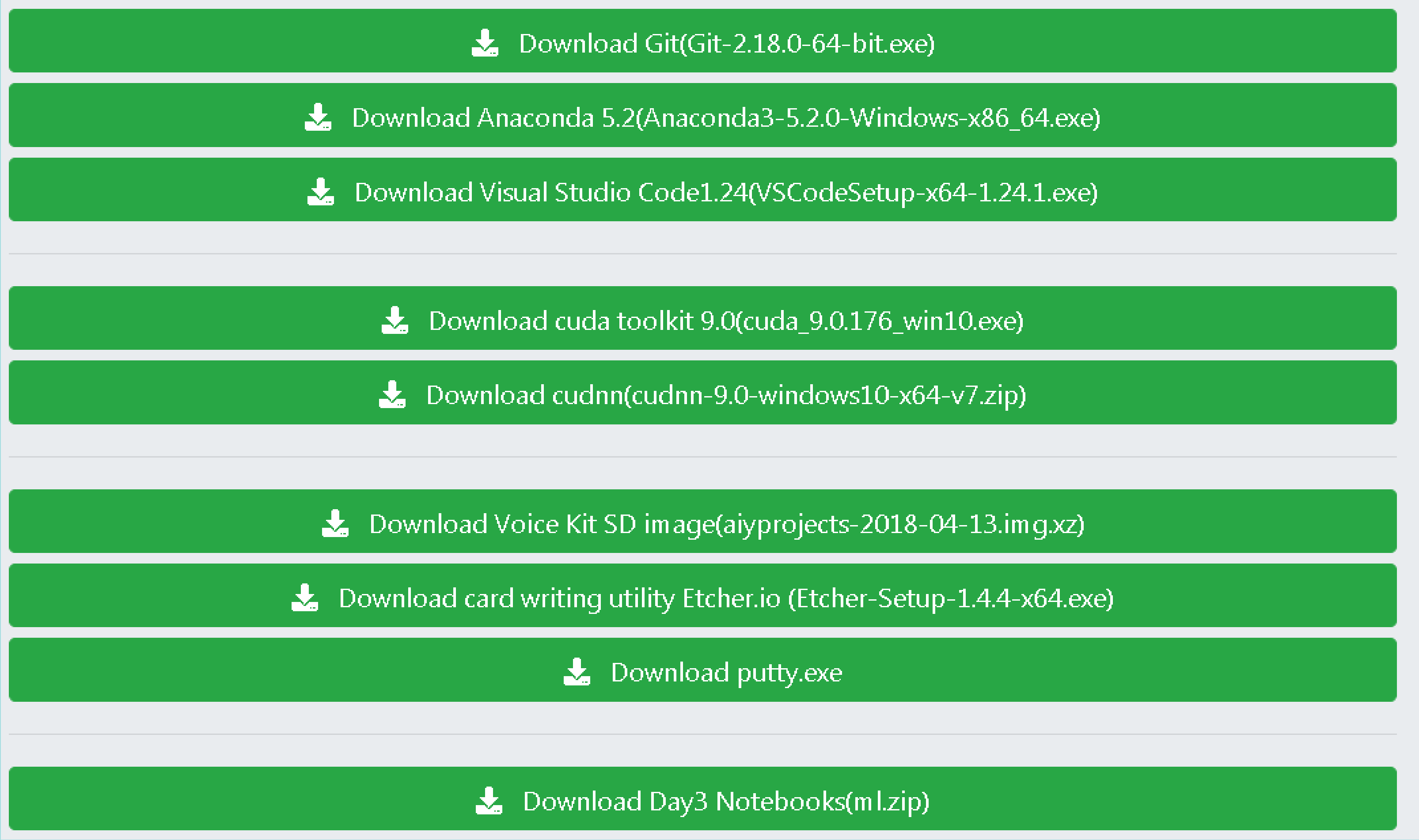
# 2018 Asia University AI Summer Program (Day3-July5) A: Python Basics

Frank Hsueh-Ting Chu

# Content

- Anaconda
- Jupyter notebook
- Markdown
- Python Basics: variables, statements
- The Python Standard Library
- Python ML Libraries: numpy scipy matplotlib

# Anaconda



Download Git(Git-2.18.0-64-bit.exe)

Download Anaconda 5.2(Anaconda3-5.2.0-Windows-x86_64.exe)

Download Visual Studio Code1.24(VSCodeSetup-x64-1.24.1.exe)

Download cuda toolkit 9.0(cuda_9.0.176_win10.exe)

Download cudnn(cudnn-9.0-windows10-x64-v7.zip)

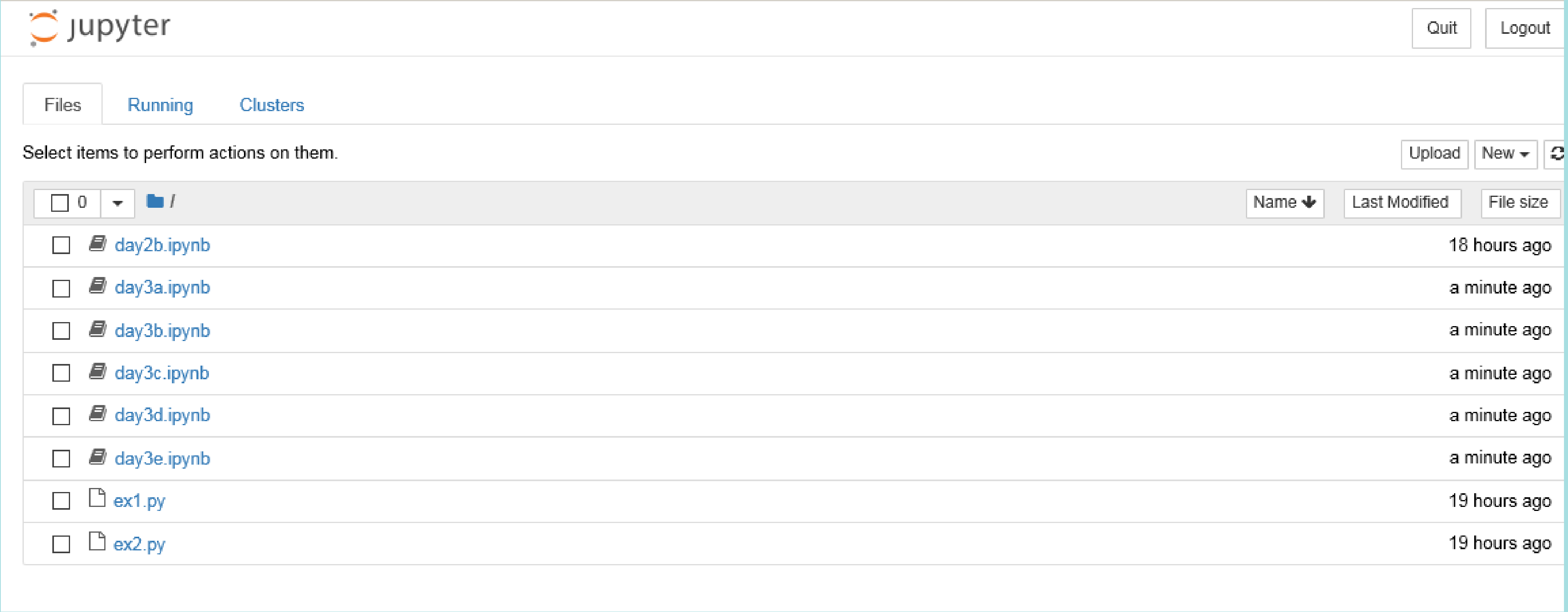
Download Voice Kit SD image(aiyprojects-2018-04-13.img.xz)

Download card writing utility Etcher.io (Etcher-Setup-1.4.4-x64.exe)

Download putty.exe

Download Day3 Notebooks(ml.zip)

# Jupyter notebook



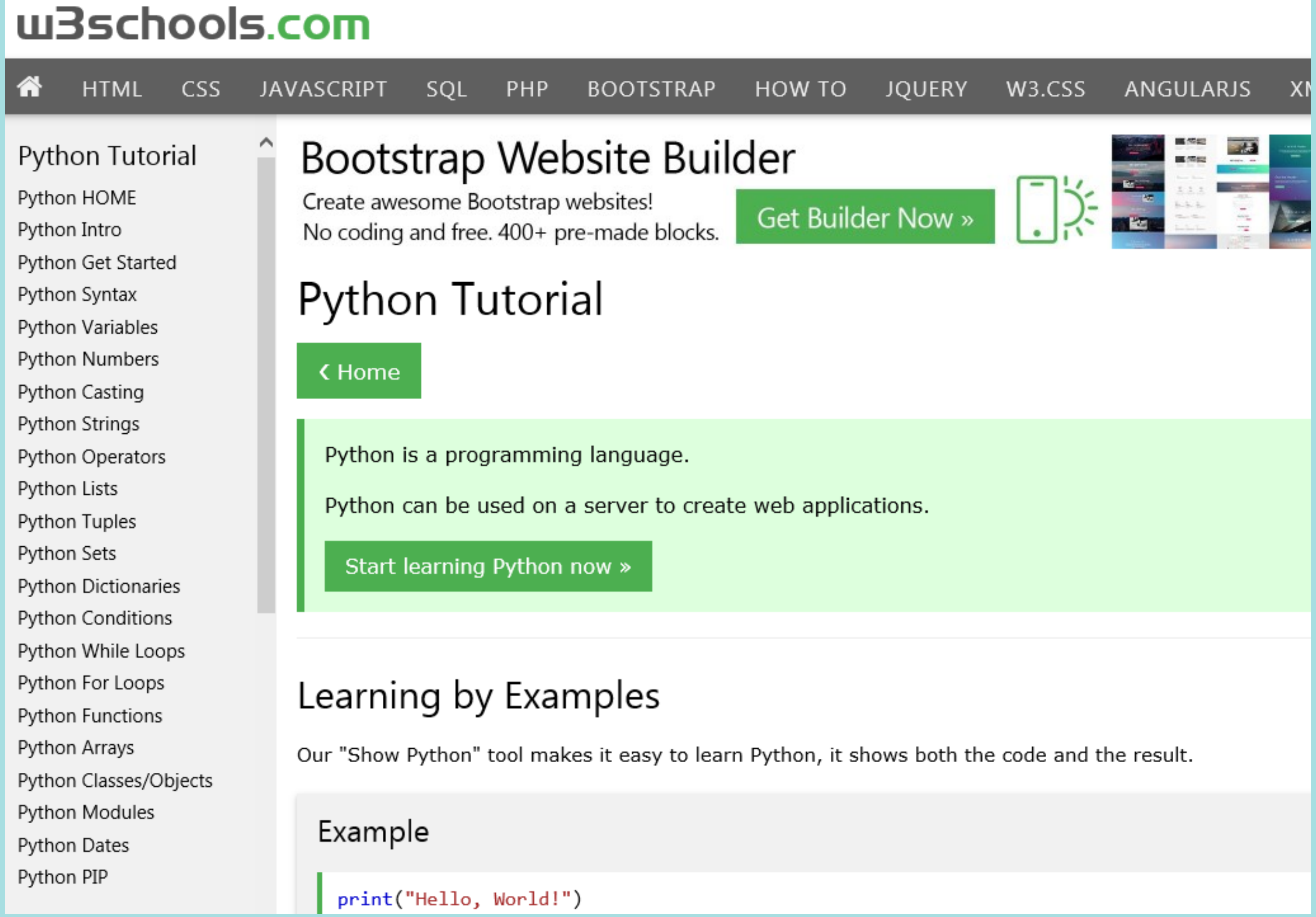
Select items to perform actions on them.

| □ 0 / | Name | Last Modified | File size |
|---|---|---|---|
| □ day2b.ipynb | | 18 hours ago | |
| □ day3a.ipynb | | a minute ago | |
| □ day3b.ipynb | | a minute ago | |
| □ day3c.ipynb | | a minute ago | |
| □ day3d.ipynb | | a minute ago | |
| □ day3e.ipynb | | a minute ago | |
| □ ex1.py | | 19 hours ago | |
| □ ex2.py | | 19 hours ago | |

# Markdown

- Headings
  - #
  - ##
- Links
  - http://xxx.yyy.zzz
- Lists:
  - 1. aaa
  - * bbb

# Python Basics: variables, statements


w3schools.com
HTML CSS JAVASCRIPT SQL PHP BOOTSTRAP HOW TO JQUERY W3.CSS ANGULARJS XI

Python Tutorial

- Python HOME
- Python Intro
- Python Get Started
- Python Syntax
- Python Variables
- Python Numbers
- Python Casting
- Python Strings
- Python Operators
- Python Lists
- Python Tuples
- Python Sets
- Python Dictionaries
- Python Conditions
- Python While Loops
- Python For Loops
- Python Functions
- Python Arrays
- Python Classes/Objects
- Python Modules
- Python Dates
- Python PIP

## Python Tutorial

‹ Home

Python is a programming language.

Python can be used on a server to create web applications.

Start learning Python now »

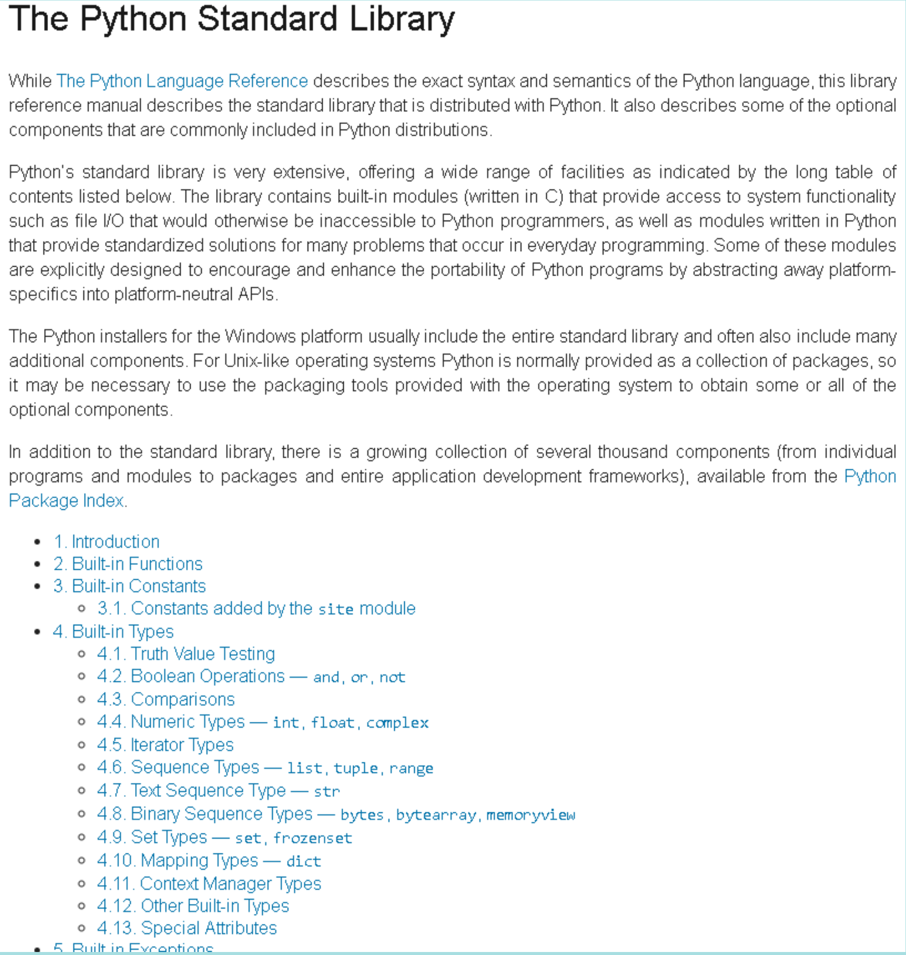
## Learning by Examples

Our "Show Python" tool makes it easy to learn Python, it shows both the code and the result.

### Example

```
print("Hello, World!")
```

# The Python Standard Library



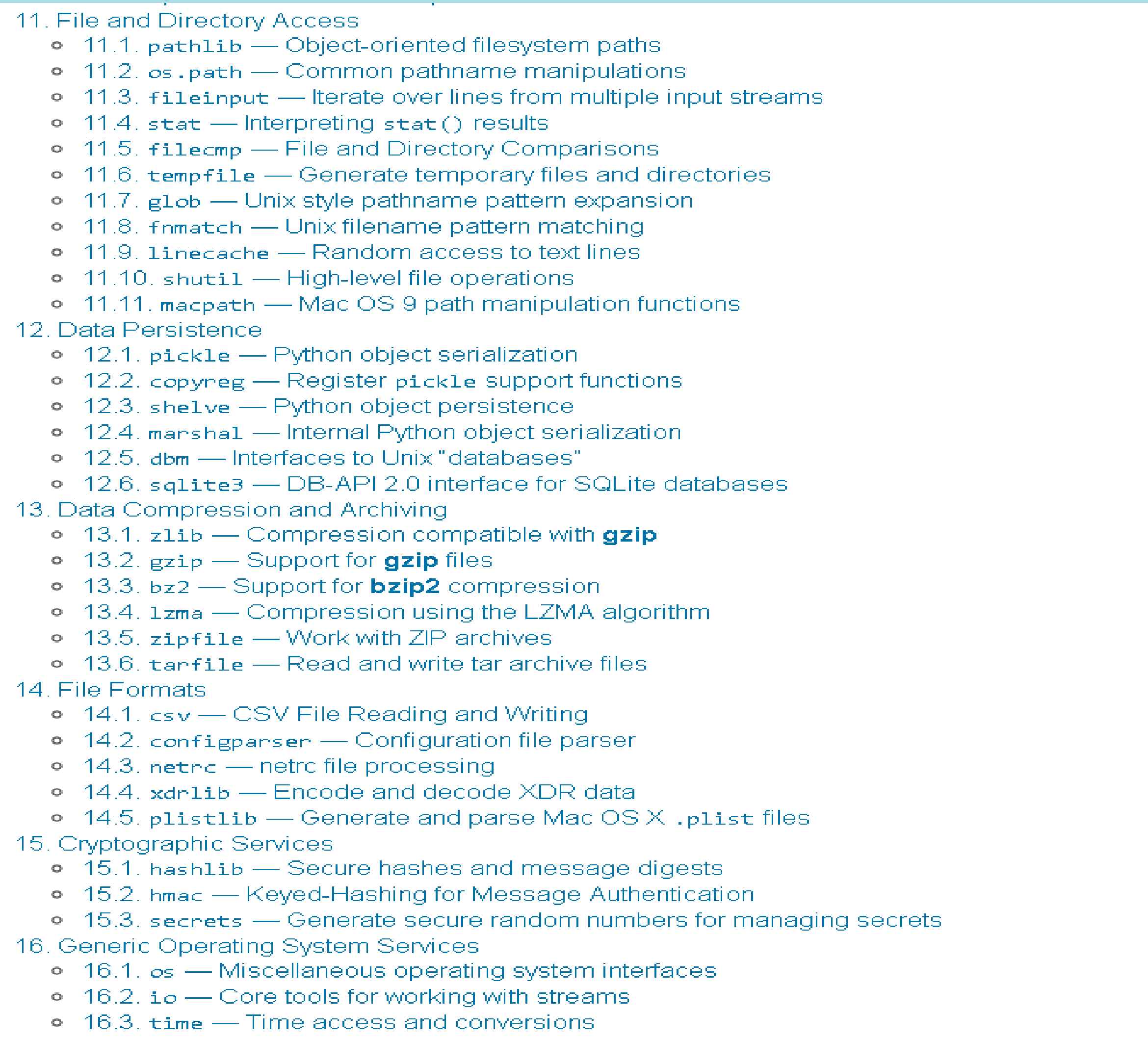
## The Python Standard Library

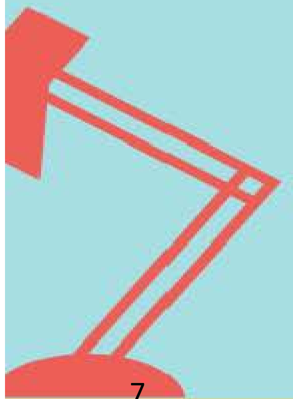
While The Python Language Reference describes the exact syntax and semantics of the Python language, this library reference manual describes the standard library that is distributed with Python. It also describes some of the optional components that are commonly included in Python distributions.

Python's standard library is very extensive, offering a wide range of facilities as indicated by the long table of contents listed below. The library contains built-in modules (written in C) that provide access to system functionality such as file I/O that would otherwise be inaccessible to Python programmers, as well as modules written in Python that provide standardized solutions for many problems that occur in everyday programming. Some of these modules are explicitly designed to encourage and enhance the portability of Python programs by abstracting away platform-specifics into platform-neutral APIs.

The Python installers for the Windows platform usually include the entire standard library and often also include many additional components. For Unix-like operating systems Python is normally provided as a collection of packages, so it may be necessary to use the packaging tools provided with the operating system to obtain some or all of the optional components.

In addition to the standard library, there is a growing collection of several thousand components (from individual programs and modules to packages and entire application development frameworks), available from the Python Package Index.





# A simple crawler

```
import urllib.request
import re

with urllib.request.urlopen('http://www.asia.edu.tw/news1.php') as response:
  html = response.read().decode('utf-8')
  pattern = '<font color="#446666" face="微軟正黑體" size="2">'
  for pos in re.finditer(pattern,html):
    pos2 = html.find('</font>', pos.end())
    sub = html[pos.end():pos2]
    print(sub)
```

# Thanks! Q&A