

# Useful\_libraries

January 10, 2021

```
[1]: # The following is to know when this notebook has been run and with which ↵  
      ↪python version.  
import time, sys  
print(time.ctime())  
print(sys.version.split('|')[0])
```

```
Mon Nov  9 20:34:48 2020  
3.7.6 (default, Jan  8 2020, 13:42:34)  
[Clang 4.0.1 (tags/RELEASE_401/final)]
```

```
[2]: import numpy as np
```

## 1 F Some useful libraries that make life easier

This is part of the Python lecture given by Christophe Morisset at IA-UNAM.

This lecture will give some insights to the most useful python libraries. It is NOT exhaustive, you have to read the corresponding manual pages to find the best use you can have of them. The list of all python-included libraries is here: <https://docs.python.org/3/library/>

### 1.0.1 time, datetime and timeit

- <https://docs.python.org/3/library/time.html>
- <https://docs.python.org/3/library/datetime.html>

```
[3]: import time  
import datetime
```

```
[7]: print(time.ctime()) # current time, in a string format
```

```
Mon Nov  9 20:40:01 2020
```

```
[6]: lt = time.localtime()  
print(lt)
```

```
time.struct_time(tm_year=2020, tm_mon=11, tm_mday=9, tm_hour=20, tm_min=39,  
tm_sec=59, tm_wday=0, tm_yday=314, tm_isdst=0)
```

```
[12]: time.strftime("%a, %d %b %Y %H:%M:%S",lt)
```

```
[12]: 'Mon, 09 Nov 2020 20:39:59'
```

```
[13]: print(datetime.datetime.today())
```

```
2020-11-09 20:42:32.543704
```

```
[14]: today = datetime.date.today()
      philae_landing = datetime.date(2014, 11, 11)
```

```
[15]: time_to_landing = philae_landing - today
```

```
[16]: if philae_landing > today:
      print('Philae landing in {} days.'.format(time_to_landing.days))
      else:
      print('Philae landed since {} days.'.format(-time_to_landing.days))
```

```
Philae landed since 2190 days.
```

```
[19]: start = time.time()
      for i in np.arange(1000):
          t = np.math.factorial(i)
      end = time.time()
      print('1000 factorials done in {0:.3f} secs.'.format(end-start))
```

```
1000 factorials done in 0.018 secs.
```

```
[20]: from timeit import Timer
      command = """\
      for i in np.arange(1000):
          t = np.math.factorial(i)
      """
      t = Timer(command, setup='import numpy as np')
      print(t.timeit(number=10))
```

```
0.16812326000035682
```

in ipython, one can use the magic timeit function:

```
[21]: %timeit np.math.factorial(50)
```

```
525 ns ± 18.3 ns per loop (mean ± std. dev. of 7 runs, 1000000 loops each)
```

or for the whole cell:

```
[22]: %%timeit
      for i in np.arange(1000):
          t = np.math.factorial(i)
```

13.5 ms  $\pm$  481  $\mu$ s per loop (mean  $\pm$  std. dev. of 7 runs, 100 loops each)

### 1.0.2 os

<https://docs.python.org/3/library/os.html> This module provides a portable way of using operating system dependent functionality. If you just want to read or write a file see `open()`, if you want to manipulate paths, see the `os.path` module, and if you want to read all the lines in all the files on the command line see the `fileinput` module. For creating temporary files and directories see the `tempfile` module, and for high-level file and directory handling see the `shutil` module.

```
[23]: import os
```

```
[24]: # os.environ is a dictionary containing the environment variables and values
if 'HOME' in os.environ:
    print(os.environ['HOME'])
else:
    print('This OS is too limited to understand HOME')
```

/Users/christophemorisset

```
[26]: # if variable doesn't exists, getenv returns None
print(os.getenv('HOMMMMMME'))
```

None

```
[27]: # get current working directory
cwd = os.getcwd()
print(cwd)
print(cwd.split('/')[ -1])
```

/Users/christophemorisset/Google Drive/Pro/Python-MySQL/Notebooks/Notebooks  
Notebooks

```
[29]: new_dir = '/tmp/test3456'
if not os.path.exists(new_dir):
    os.mkdir(new_dir)
    print('Dir {} created'.format(new_dir))
else:
    print('Dir {} exists'.format(new_dir))
```

Dir /tmp/test3456 exists

```
[30]: # list all the files from a directory
os.listdir(new_dir)
```

```
[30]: []
```

```
[33]: os.chdir(new_dir)
os.getcwd()
```

```
[33]: '/private/tmp/test3456'
```

```
[37]: # Renaming files in a directory
cwd = os.getcwd()
os.chdir('/tmp')
for filename in os.listdir('.'):
    #print(filename)
    base_file, ext = os.path.splitext(filename)
    #print(base_file, ext)
    if ext == '.txt':
        newname = base_file + '.exe'
        try:
            os.rename(filename, newname)
            print('file {0} renamed to {1}'.format(filename, newname))
        except:
            print('file {0} NOT renamed'.format(filename))
os.chdir(os.environ['HOME'])
```

### 1.0.3 glob

<https://docs.python.org/3/library/glob.html>

No tilde expansion is done, but \*, ?, and character ranges expressed with [] will be correctly matched.

```
[38]: from glob import glob
```

```
[44]: os.chdir('/Users/christophemorisset/Google Drive/Pro/Python-MySQL/Notebooks/
↪Notebooks')
```

```
[45]: ipynb_files = glob('*.ipynb')
print(ipynb_files)
```

```
['Calling Fortran.ipynb', 'Test-Install.ipynb', 'intro_Scipy.ipynb',
'Ex1_with_res.ipynb', 'Parallel.ipynb', 'OOP.ipynb', 'Using_astropy.ipynb',
'Tarea_1.ipynb', 'Interact with files.ipynb', 'Tarea2.ipynb',
'Useful_libraries.ipynb', 'Tarea2_done.ipynb', 'intro_numpy.ipynb',
'intro_Python.ipynb', 'Ex1_done.ipynb', 'Ex1.ipynb', 'intro_Matplotlib.ipynb',
'Optimization.ipynb', 'Tarea2_helper.ipynb', 'Using_PyMySQL.ipynb']
```

### 1.0.4 pathlib

```
[46]: from pathlib import Path
```

```
[47]: new_dir = Path('/tmp/test1234')
print(new_dir)
print(new_dir.name)
print(new_dir.parent)
```

```
/tmp/test1234
test1234
/tmp
```

```
[48]: new_dir = Path('/tmp') / Path('test1234')
      print(new_dir)
      print(new_dir.name)
      print(new_dir.parent)
```

```
/tmp/test1234
test1234
/tmp
```

```
[50]: if not new_dir.exists():
      new_dir.mkdir() # may use exist_ok=True
      print('Dir {} created'.format(new_dir))
      else:
      print('Dir {} exists'.format(new_dir))
```

Dir /tmp/test1234 exists

```
[51]: current = Path('.')
      for f in current.glob('*.ipynb'):
          print(f, f.stem, f.suffix)
```

```
Calling Fortran.ipynb Calling Fortran .ipynb
Test-Install.ipynb Test-Install .ipynb
intro_Scipy.ipynb intro_Scipy .ipynb
Ex1_with_res.ipynb Ex1_with_res .ipynb
Parallel.ipynb Parallel .ipynb
OOP.ipynb OOP .ipynb
Using_astropy.ipynb Using_astropy .ipynb
Tarea_1.ipynb Tarea_1 .ipynb
Interact with files.ipynb Interact with files .ipynb
Tarea2.ipynb Tarea2 .ipynb
Useful_libraries.ipynb Useful_libraries .ipynb
Tarea2_done.ipynb Tarea2_done .ipynb
intro_numpy.ipynb intro_numpy .ipynb
intro_Python.ipynb intro_Python .ipynb
Ex1_done.ipynb Ex1_done .ipynb
Ex1.ipynb Ex1 .ipynb
intro_Matplotlib.ipynb intro_Matplotlib .ipynb
Optimization.ipynb Optimization .ipynb
Tarea2_helper.ipynb Tarea2_helper .ipynb
Using_PyMySQL.ipynb Using_PyMySQL .ipynb
```

```
[52]: new_dir = Path('/tmp') / Path('test1234')
      for f in new_dir.glob('test*.txt*'):
```

```

f.unlink()

f = new_dir / Path('test1234.txt')
f.touch()
f = new_dir / Path('test5678.txt')
f.touch()
print(list(new_dir.glob('test*.txt*')))

for f in new_dir.glob('test*.txt'):
    new_f = f.with_suffix('.txt2')
    if not new_f.exists():
        f.rename(new_f)
    else:
        print('{} exists'.format(new_f))

print(list(new_dir.glob('test*.txt*')))

```

```

[PosixPath('/tmp/test1234/test5678.txt'),
PosixPath('/tmp/test1234/test1234.txt')]
[PosixPath('/tmp/test1234/test1234.txt2'),
PosixPath('/tmp/test1234/test5678.txt2')]

```

### 1.0.5 sys

<https://docs.python.org/3/library/sys.html> This module provides access to some variables used or maintained by the interpreter and to functions that interact strongly with the interpreter. It is always available.

The `sys.argv` list contains the arguments passed to the script, when the interpreter was started. The first item contains the name of the script itself.

```

[53]: %%writefile sys_text.py
import sys
for arg in sys.argv:
    print(arg)

```

Overwriting `sys_text.py`

```

[54]: !cat sys_text.py

```

```

import sys
for arg in sys.argv:
    print(arg)

```

```

[58]: ! python sys_text.py tralala trololo

```

```

sys_text.py
tralala
trololo

```

```
[64]: %%writefile fact.py
import sys
import numpy as np
print(np.math.factorial(int(sys.argv[1])))
```

Overwriting fact.py

```
[69]: ! python fact.py 60
```

```
83209871127413901442763411832233643807541726063612459524492776964096000000000000
00
```

A more complete management of the argument (especially those like `-v`) is available using the `argparse` library. <https://docs.python.org/3/library/argparse.html>

### 1.0.6 Talking to the OS subprocess

The `os.popen` method is deprecated, `subprocess` is preferred. A good tutorial is here: <http://pymotw.com/3/subprocess/>

```
[70]: import subprocess
```

```
[71]: ls -l
```

```
total 63880
-rw----- 1 christophemorriset staff 4229587 May 31 2016 CALIFA_ah7.dat.gz
-rw----- 1 christophemorriset staff 20137 Nov 1 2018 Calling
Fortran.ipynb
-rw----- 1 christophemorriset staff 86815 Jun 2 2016 Calling
Fortran.pdf
-rw-r--r-- 1 christophemorriset staff 226 Oct 12 22:49 Demo.pickle
-rw-r--r-- 1 christophemorriset staff 198 Oct 12 22:52 Demo.pklz
-rw----- 1 christophemorriset staff 8265 Sep 21 18:13 Ex1.ipynb
-rw----- 1 christophemorriset staff 381624 Sep 21 18:13 Ex1_done.ipynb
-rw----- 1 christophemorriset staff 315455 Sep 21 18:13 Ex1_with_res.ipynb
-rw-r--r-- 1 christophemorriset staff 93801 Oct 22 10:47 Fig1.jpg
-rw-r--r-- 1 christophemorriset staff 64091 Oct 22 10:47 Fig1.pdf
-rw-r--r-- 1 christophemorriset staff 123182 Oct 22 10:47 Fig1.png
-rw-r--r-- 1 christophemorriset staff 111863 Oct 22 10:31 HII.dat
-rw-r--r-- 1 christophemorriset staff 46901 Oct 22 10:18 HII.dat.gz
-rw-----@ 1 christophemorriset staff 860517 Sep 21 16:38 InstallPython.key
-rw----- 1 christophemorriset staff 554465 Oct 13 08:20 Interact with
files.ipynb
-rw----- 1 christophemorriset staff 388118 Jun 30 2017 Interact with
files.pdf
-rw----- 1 christophemorriset staff 131497 May 31 2016 Intro_1.odp
-rw----- 1 christophemorriset staff 174202 May 31 2016 Intro_1.pdf
-rw----- 1 christophemorriset staff 130234 May 31 2016 Intro_2014.odp
-rw----- 1 christophemorriset staff 110524 Nov 22 2017 MySQL.pdf
```

```

-rw----- 1 christophemorriset staff 1270918 May 31 2016 MySQL.pdf.gz
-rw----- 1 christophemorriset staff 688451 Oct 3 2019 OOP.ipynb
-rw----- 1 christophemorriset staff 568413 Jun 30 2017 OOP.pdf
-rw----- 1 christophemorriset staff 870171 Nov 7 2019 Optimization.ipynb
-rw----- 1 christophemorriset staff 481656 Jun 2 2016 Optimization.pdf
-rw----- 1 christophemorriset staff 2537 Nov 9 2016 Parallel.ipynb
-rw-r--r-- 1 christophemorriset staff 3679 Oct 22 11:00 Tarea2.ipynb
-rw-r--r-- 1 christophemorriset staff 339696 Oct 22 10:55 Tarea2_done.ipynb
-rw-r--r-- 1 christophemorriset staff 257277 Oct 22 10:32
Tarea2_helper.ipynb
-rw-r--r-- 1 christophemorriset staff 6154 Oct 7 14:03 Tarea_1.ipynb
-rw-r--r-- 1 christophemorriset staff 1434 Sep 22 19:01 Test-Install.ipynb
-rw-r--r-- 1 christophemorriset staff 35552 Nov 9 21:31
Useful_libraries.ipynb
-rw-r--r--@ 1 christophemorriset staff 77840 Jul 1 23:05
Useful_libraries.pdf
-rw----- 1 christophemorriset staff 282484 Nov 9 09:23
Using_PyMySQL.ipynb
-rw----- 1 christophemorriset staff 221936 Nov 17 2016 Using_PyMySQL.pdf
-rw----- 1 christophemorriset staff 35209 Nov 9 19:51
Using_astropy.ipynb
-rw----- 1 christophemorriset staff 206058 Jun 30 2017 Using_astropy.pdf
drwxr-xr-x 4 christophemorriset staff 128 Oct 7 12:31
__pycache__/_
-rw-r--r-- 1 christophemorriset staff 65 Oct 12 21:36 data1.dat
-rw-r--r-- 1 christophemorriset staff 14 Oct 12 22:41 data10.dat
-rw-r--r-- 1 christophemorriset staff 32 Oct 12 22:43 data11.dat
-rw-r--r-- 1 christophemorriset staff 104 Oct 12 22:47 data12.dat
-rw-r--r-- 1 christophemorriset staff 148 Oct 12 21:40 data2.dat
-rw-r--r-- 1 christophemorriset staff 170 Oct 12 22:04 data3.dat
-rw-r--r-- 1 christophemorriset staff 1190 Oct 12 22:21 data4.dat
-rw-r--r-- 1 christophemorriset staff 1186 Oct 12 22:32 data5.dat
-rw----- 1 christophemorriset staff 298 May 31 2016 dcumsum.f
-rw----- 1 christophemorriset staff 30716 Nov 10 2016 dcumsum.so
-rw----- 1 christophemorriset staff 1447 Oct 7 12:31 ex1.html
-rw-r--r-- 1 christophemorriset staff 155 Oct 7 12:31 ex1.py
-rw----- 1 christophemorriset staff 288 Jun 29 2017 ex1.pyc
-rw----- 1 christophemorriset staff 8 May 31 2016 ex1.txt
-rw----- 1 christophemorriset staff 74 Nov 9 21:29 fact.py
-rw----- 1 christophemorriset staff 4214 May 31 2016 grep.py
-rw----- 1 christophemorriset staff 4995 May 31 2016 grep2.py
-rw----- 1 christophemorriset staff 6319 May 31 2016 grep3.py
-rw----- 1 christophemorriset staff 5575 Dec 9 2016 grep3.pyc
-rw----- 1 christophemorriset staff 49 May 31 2016 hello.py
-rw----- 1 christophemorriset staff 1090 May 31 2016 hellofortran.c
-rw----- 1 christophemorriset staff 178 May 31 2016 hellofortran.f
-rw----- 1 christophemorriset staff 33976 May 31 2016 hellofortran.so
-rw----- 1 christophemorriset staff 150 May 31 2016 index.html

```



```

-rw----- 1 christophemorriset staff 2368116 Oct 19 23:10
intro_Matplotlib.ipynb
-rw----- 1 christophemorriset staff 1643810 Jun 30 2017
intro_Matplotlib.pdf
-rw----- 1 christophemorriset staff 326924 May 31 2016 intro_Python.html
-rw----- 1 christophemorriset staff 97772 Oct 7 12:31 intro_Python.ipynb
-rw----- 1 christophemorriset staff 143102 Jun 30 2017 intro_Python.pdf
-rw----- 1 christophemorriset staff 179449 May 31 2016 intro_Python_2.pdf
-rw----- 1 christophemorriset staff 103229 May 31 2016 intro_Python_3.pdf
-rw----- 1 christophemorriset staff 813305 Nov 8 15:37 intro_Scipy.ipynb
-rw----- 1 christophemorriset staff 570062 Jun 30 2017 intro_Scipy.pdf
-rw----- 1 christophemorriset staff 110260 Oct 13 08:20 intro_numpy.ipynb
-rw----- 1 christophemorriset staff 148857 Jun 30 2017 intro_numpy.pdf
-rw----- 1 christophemorriset staff 8830080 May 31 2016 n10017o.fits
-rw-r--r-- 1 christophemorriset staff 5760 Oct 12 23:05 new.fits
-rw----- 1 christophemorriset staff 435801 May 31 2016 query_res.dat
-rw----- 1 christophemorriset staff 1059840 May 31 2016 swp04345.mxhi
-rw----- 1 christophemorriset staff 47 Nov 9 21:25 sys_text.py
-rw----- 1 christophemorriset staff 354 Oct 31 2019 tab1.dat
-rw----- 1 christophemorriset staff 668 Oct 31 2019 tab_cds1.ascii
-rw----- 1 christophemorriset staff 43200 May 31 2016 tab_cds1.fits
-rw----- 1 christophemorriset staff 1334 Oct 31 2019 tab_cds1.tex
-rw----- 1 christophemorriset staff 1617 Oct 31 2019 tab_cds2.ascii
drwxr-xr-x 4 christophemorriset staff 128 Sep 21 16:30 tt/

```

```

[72]: subprocess.call(['ls -l'], shell=True)
# The results is not visible in Notebook. In command line, the files are listed

```

[72]: 0

```

[75]: output = subprocess.check_output(['ls', '-l'])
print(type(output))
print(output.decode())

```

```

<class 'bytes'>
total 63896
-rw----- 1 christophemorriset staff 4229587 May 31 2016 CALIFA_ah7.dat.gz
-rw----- 1 christophemorriset staff 20137 Nov 1 2018 Calling
Fortran.ipynb
-rw----- 1 christophemorriset staff 86815 Jun 2 2016 Calling
Fortran.pdf
-rw-r--r-- 1 christophemorriset staff 226 Oct 12 22:49 Demo.pickle
-rw-r--r-- 1 christophemorriset staff 198 Oct 12 22:52 Demo.pklz
-rw----- 1 christophemorriset staff 8265 Sep 21 18:13 Ex1.ipynb
-rw----- 1 christophemorriset staff 381624 Sep 21 18:13 Ex1_done.ipynb
-rw----- 1 christophemorriset staff 315455 Sep 21 18:13 Ex1_with_res.ipynb
-rw-r--r-- 1 christophemorriset staff 93801 Oct 22 10:47 Fig1.jpg
-rw-r--r-- 1 christophemorriset staff 64091 Oct 22 10:47 Fig1.pdf

```

```

-rw-r--r-- 1 christophemorriset staff 123182 Oct 22 10:47 Fig1.png
-rw-r--r-- 1 christophemorriset staff 111863 Oct 22 10:31 HII.dat
-rw-r--r-- 1 christophemorriset staff 46901 Oct 22 10:18 HII.dat.gz
-rw-----@ 1 christophemorriset staff 860517 Sep 21 16:38 InstallPython.key
-rw----- 1 christophemorriset staff 554465 Oct 13 08:20 Interact with
files.ipynb
-rw----- 1 christophemorriset staff 388118 Jun 30 2017 Interact with
files.pdf
-rw----- 1 christophemorriset staff 131497 May 31 2016 Intro_1.odp
-rw----- 1 christophemorriset staff 174202 May 31 2016 Intro_1.pdf
-rw----- 1 christophemorriset staff 130234 May 31 2016 Intro_2014.odp
-rw----- 1 christophemorriset staff 110524 Nov 22 2017 MySQL.pdf
-rw----- 1 christophemorriset staff 1270918 May 31 2016 MySQL.pdf.gz
-rw----- 1 christophemorriset staff 688451 Oct 3 2019 OOP.ipynb
-rw----- 1 christophemorriset staff 568413 Jun 30 2017 OOP.pdf
-rw----- 1 christophemorriset staff 870171 Nov 7 2019 Optimization.ipynb
-rw----- 1 christophemorriset staff 481656 Jun 2 2016 Optimization.pdf
-rw----- 1 christophemorriset staff 2537 Nov 9 2016 Parallel.ipynb
-rw-r--r-- 1 christophemorriset staff 3679 Oct 22 11:00 Tarea2.ipynb
-rw-r--r-- 1 christophemorriset staff 339696 Oct 22 10:55 Tarea2_done.ipynb
-rw-r--r-- 1 christophemorriset staff 257277 Oct 22 10:32
Tarea2_helper.ipynb
-rw-r--r-- 1 christophemorriset staff 6154 Oct 7 14:03 Tarea_1.ipynb
-rw-r--r-- 1 christophemorriset staff 1434 Sep 22 19:01 Test-Install.ipynb
-rw-r--r-- 1 christophemorriset staff 43179 Nov 9 21:33
Useful_libraries.ipynb
-rw-r--r--@ 1 christophemorriset staff 77840 Jul 1 23:05
Useful_libraries.pdf
-rw----- 1 christophemorriset staff 282484 Nov 9 09:23
Using_PyMySQL.ipynb
-rw----- 1 christophemorriset staff 221936 Nov 17 2016 Using_PyMySQL.pdf
-rw----- 1 christophemorriset staff 35209 Nov 9 19:51
Using_astropy.ipynb
-rw----- 1 christophemorriset staff 206058 Jun 30 2017 Using_astropy.pdf
drwxr-xr-x 4 christophemorriset staff 128 Oct 7 12:31 __pycache__
-rw-r--r-- 1 christophemorriset staff 65 Oct 12 21:36 data1.dat
-rw-r--r-- 1 christophemorriset staff 14 Oct 12 22:41 data10.dat
-rw-r--r-- 1 christophemorriset staff 32 Oct 12 22:43 data11.dat
-rw-r--r-- 1 christophemorriset staff 104 Oct 12 22:47 data12.dat
-rw-r--r-- 1 christophemorriset staff 148 Oct 12 21:40 data2.dat
-rw-r--r-- 1 christophemorriset staff 170 Oct 12 22:04 data3.dat
-rw-r--r-- 1 christophemorriset staff 1190 Oct 12 22:21 data4.dat
-rw-r--r-- 1 christophemorriset staff 1186 Oct 12 22:32 data5.dat
-rw----- 1 christophemorriset staff 298 May 31 2016 dcumsum.f
-rw----- 1 christophemorriset staff 30716 Nov 10 2016 dcumsum.so
-rw----- 1 christophemorriset staff 1447 Oct 7 12:31 ex1.html
-rw-r--r-- 1 christophemorriset staff 155 Oct 7 12:31 ex1.py
-rw----- 1 christophemorriset staff 288 Jun 29 2017 ex1.pyc

```

```

-rw----- 1 christophemorriset staff      8 May 31 2016 ex1.txt
-rw----- 1 christophemorriset staff      74 Nov  9 21:29 fact.py
-rw----- 1 christophemorriset staff    4214 May 31 2016 grep.py
-rw----- 1 christophemorriset staff    4995 May 31 2016 grep2.py
-rw----- 1 christophemorriset staff    6319 May 31 2016 grep3.py
-rw----- 1 christophemorriset staff    5575 Dec  9 2016 grep3.pyc
-rw----- 1 christophemorriset staff      49 May 31 2016 hello.py
-rw----- 1 christophemorriset staff    1090 May 31 2016 hellofortran.c
-rw----- 1 christophemorriset staff     178 May 31 2016 hellofortran.f
-rw----- 1 christophemorriset staff   33976 May 31 2016 hellofortran.so
-rw----- 1 christophemorriset staff     150 May 31 2016 index.html
-rw----- 1 christophemorriset staff 2368116 Oct 19 23:10
intro_Matplotlib.ipynb
-rw----- 1 christophemorriset staff 1643810 Jun 30 2017
intro_Matplotlib.pdf
-rw----- 1 christophemorriset staff   326924 May 31 2016 intro_Python.html
-rw----- 1 christophemorriset staff   97772 Oct  7 12:31 intro_Python.ipynb
-rw----- 1 christophemorriset staff  143102 Jun 30 2017 intro_Python.pdf
-rw----- 1 christophemorriset staff  179449 May 31 2016 intro_Python_2.pdf
-rw----- 1 christophemorriset staff  103229 May 31 2016 intro_Python_3.pdf
-rw----- 1 christophemorriset staff   813305 Nov  8 15:37 intro_Scipy.ipynb
-rw----- 1 christophemorriset staff   570062 Jun 30 2017 intro_Scipy.pdf
-rw----- 1 christophemorriset staff   110260 Oct 13 08:20 intro_numpy.ipynb
-rw----- 1 christophemorriset staff   148857 Jun 30 2017 intro_numpy.pdf
-rw----- 1 christophemorriset staff  8830080 May 31 2016 n10017o.fits
-rw-r--r-- 1 christophemorriset staff    5760 Oct 12 23:05 new.fits
-rw----- 1 christophemorriset staff   435801 May 31 2016 query_res.dat
-rw----- 1 christophemorriset staff  1059840 May 31 2016 swp04345.mxhi
-rw----- 1 christophemorriset staff      47 Nov  9 21:25 sys_text.py
-rw----- 1 christophemorriset staff     354 Oct 31 2019 tab1.dat
-rw----- 1 christophemorriset staff     668 Oct 31 2019 tab_cds1.ascii
-rw----- 1 christophemorriset staff   43200 May 31 2016 tab_cds1.fits
-rw----- 1 christophemorriset staff    1334 Oct 31 2019 tab_cds1.tex
-rw----- 1 christophemorriset staff    1617 Oct 31 2019 tab_cds2.ascii
drwxr-xr-x 4 christophemorriset staff    128 Sep 21 16:30 tt

```

```

[76]: command = 'ls -l'
      popen = subprocess.Popen(command, shell=True)
      # The same apply here, no output in Notebook
      print(popen)

```

<subprocess.Popen object at 0x7f96d07c4550>

```

[77]: popen = subprocess.Popen(command, stdout=subprocess.PIPE, shell=True)
      res = popen.communicate()[0]
      print(type(res))

```

```
print(res.decode())
```

```
<class 'bytes'>
total 63896
-rw----- 1 christophemorriset staff 4229587 May 31 2016 CALIFA_ah7.dat.gz
-rw----- 1 christophemorriset staff 20137 Nov 1 2018 Calling
Fortran.ipynb
-rw----- 1 christophemorriset staff 86815 Jun 2 2016 Calling
Fortran.pdf
-rw-r--r-- 1 christophemorriset staff 226 Oct 12 22:49 Demo.pickle
-rw-r--r-- 1 christophemorriset staff 198 Oct 12 22:52 Demo.pklz
-rw----- 1 christophemorriset staff 8265 Sep 21 18:13 Ex1.ipynb
-rw----- 1 christophemorriset staff 381624 Sep 21 18:13 Ex1_done.ipynb
-rw----- 1 christophemorriset staff 315455 Sep 21 18:13 Ex1_with_res.ipynb
-rw-r--r-- 1 christophemorriset staff 93801 Oct 22 10:47 Fig1.jpg
-rw-r--r-- 1 christophemorriset staff 64091 Oct 22 10:47 Fig1.pdf
-rw-r--r-- 1 christophemorriset staff 123182 Oct 22 10:47 Fig1.png
-rw-r--r-- 1 christophemorriset staff 111863 Oct 22 10:31 HII.dat
-rw-r--r-- 1 christophemorriset staff 46901 Oct 22 10:18 HII.dat.gz
-rw-----@ 1 christophemorriset staff 860517 Sep 21 16:38 InstallPython.key
-rw----- 1 christophemorriset staff 554465 Oct 13 08:20 Interact with
files.ipynb
-rw----- 1 christophemorriset staff 388118 Jun 30 2017 Interact with
files.pdf
-rw----- 1 christophemorriset staff 131497 May 31 2016 Intro_1.odp
-rw----- 1 christophemorriset staff 174202 May 31 2016 Intro_1.pdf
-rw----- 1 christophemorriset staff 130234 May 31 2016 Intro_2014.odp
-rw----- 1 christophemorriset staff 110524 Nov 22 2017 MySQL.pdf
-rw----- 1 christophemorriset staff 1270918 May 31 2016 MySQL.pdf.gz
-rw----- 1 christophemorriset staff 688451 Oct 3 2019 OOP.ipynb
-rw----- 1 christophemorriset staff 568413 Jun 30 2017 OOP.pdf
-rw----- 1 christophemorriset staff 870171 Nov 7 2019 Optimization.ipynb
-rw----- 1 christophemorriset staff 481656 Jun 2 2016 Optimization.pdf
-rw----- 1 christophemorriset staff 2537 Nov 9 2016 Parallel.ipynb
-rw-r--r-- 1 christophemorriset staff 3679 Oct 22 11:00 Tarea2.ipynb
-rw-r--r-- 1 christophemorriset staff 339696 Oct 22 10:55 Tarea2_done.ipynb
-rw-r--r-- 1 christophemorriset staff 257277 Oct 22 10:32
Tarea2_helper.ipynb
-rw-r--r-- 1 christophemorriset staff 6154 Oct 7 14:03 Tarea_1.ipynb
-rw-r--r-- 1 christophemorriset staff 1434 Sep 22 19:01 Test-Install.ipynb
-rw-r--r-- 1 christophemorriset staff 43179 Nov 9 21:33
Useful_libraries.ipynb
-rw-r--r--@ 1 christophemorriset staff 77840 Jul 1 23:05
Useful_libraries.pdf
-rw----- 1 christophemorriset staff 282484 Nov 9 09:23
Using_PyMySQL.ipynb
-rw----- 1 christophemorriset staff 221936 Nov 17 2016 Using_PyMySQL.pdf
```

```

-rw----- 1 christophemorriset staff 35209 Nov 9 19:51
Using_astropy.ipynb
-rw----- 1 christophemorriset staff 206058 Jun 30 2017 Using_astropy.pdf
drwxr-xr-x 4 christophemorriset staff 128 Oct 7 12:31 __pycache__
-rw-r--r-- 1 christophemorriset staff 65 Oct 12 21:36 data1.dat
-rw-r--r-- 1 christophemorriset staff 14 Oct 12 22:41 data10.dat
-rw-r--r-- 1 christophemorriset staff 32 Oct 12 22:43 data11.dat
-rw-r--r-- 1 christophemorriset staff 104 Oct 12 22:47 data12.dat
-rw-r--r-- 1 christophemorriset staff 148 Oct 12 21:40 data2.dat
-rw-r--r-- 1 christophemorriset staff 170 Oct 12 22:04 data3.dat
-rw-r--r-- 1 christophemorriset staff 1190 Oct 12 22:21 data4.dat
-rw-r--r-- 1 christophemorriset staff 1186 Oct 12 22:32 data5.dat
-rw----- 1 christophemorriset staff 298 May 31 2016 dcumsum.f
-rw----- 1 christophemorriset staff 30716 Nov 10 2016 dcumsum.so
-rw----- 1 christophemorriset staff 1447 Oct 7 12:31 ex1.html
-rw-r--r-- 1 christophemorriset staff 155 Oct 7 12:31 ex1.py
-rw----- 1 christophemorriset staff 288 Jun 29 2017 ex1.pyc
-rw----- 1 christophemorriset staff 8 May 31 2016 ex1.txt
-rw----- 1 christophemorriset staff 74 Nov 9 21:29 fact.py
-rw----- 1 christophemorriset staff 4214 May 31 2016 grep.py
-rw----- 1 christophemorriset staff 4995 May 31 2016 grep2.py
-rw----- 1 christophemorriset staff 6319 May 31 2016 grep3.py
-rw----- 1 christophemorriset staff 5575 Dec 9 2016 grep3.pyc
-rw----- 1 christophemorriset staff 49 May 31 2016 hello.py
-rw----- 1 christophemorriset staff 1090 May 31 2016 hellofortran.c
-rw----- 1 christophemorriset staff 178 May 31 2016 hellofortran.f
-rw----- 1 christophemorriset staff 33976 May 31 2016 hellofortran.so
-rw----- 1 christophemorriset staff 150 May 31 2016 index.html
-rw----- 1 christophemorriset staff 2368116 Oct 19 23:10
intro_Matplotlib.ipynb
-rw----- 1 christophemorriset staff 1643810 Jun 30 2017
intro_Matplotlib.pdf
-rw----- 1 christophemorriset staff 326924 May 31 2016 intro_Python.html
-rw----- 1 christophemorriset staff 97772 Oct 7 12:31 intro_Python.ipynb
-rw----- 1 christophemorriset staff 143102 Jun 30 2017 intro_Python.pdf
-rw----- 1 christophemorriset staff 179449 May 31 2016 intro_Python_2.pdf
-rw----- 1 christophemorriset staff 103229 May 31 2016 intro_Python_3.pdf
-rw----- 1 christophemorriset staff 813305 Nov 8 15:37 intro_Scipy.ipynb
-rw----- 1 christophemorriset staff 570062 Jun 30 2017 intro_Scipy.pdf
-rw----- 1 christophemorriset staff 110260 Oct 13 08:20 intro_numpy.ipynb
-rw----- 1 christophemorriset staff 148857 Jun 30 2017 intro_numpy.pdf
-rw----- 1 christophemorriset staff 8830080 May 31 2016 n10017o.fits
-rw-r--r-- 1 christophemorriset staff 5760 Oct 12 23:05 new.fits
-rw----- 1 christophemorriset staff 435801 May 31 2016 query_res.dat
-rw----- 1 christophemorriset staff 1059840 May 31 2016 swp04345.mxhi
-rw----- 1 christophemorriset staff 47 Nov 9 21:25 sys_text.py
-rw----- 1 christophemorriset staff 354 Oct 31 2019 tab1.dat
-rw----- 1 christophemorriset staff 668 Oct 31 2019 tab_cds1.ascii

```

```
-rw----- 1 christophemorisset staff 43200 May 31 2016 tab_cds1.fits
-rw----- 1 christophemorisset staff 1334 Oct 31 2019 tab_cds1.tex
-rw----- 1 christophemorisset staff 1617 Oct 31 2019 tab_cds2.ascii
drwxr-xr-x 4 christophemorisset staff 128 Sep 21 16:30 tt
```

```
[80]: popen = subprocess.Popen(command, stdout=subprocess.PIPE, shell=True)
res = popen.communicate()[0]
for line in res.decode().split('\n'):
    lsp1 = line.split()
    if len(lsp1) > 4:
        if int(lsp1[4]) > 1000000:
            print(lsp1[8])
```

```
CALIFA_ah7.dat.gz
MySQL.pdf.gz
intro_Matplotlib.ipynb
intro_Matplotlib.pdf
n10017o.fits
swp04345.mxhi
```

### 1.0.7 re

The re module provides regular expression tools for advanced string processing. For complex matching and manipulation, regular expressions offer succinct, optimized solutions:

```
[85]: import re
# Search the words starting with "f"
re.findall(r'\bf[a-z]*', 'which foOt or hand fell fastest')
```

```
[85]: ['fo', 'fell', 'fastest']
```

```
[90]: # Remove duplicate words
re.sub(r'(\b[a-z]+) \1', r'\1', 'cat in the the hat')
```

```
[90]: 'cat in the hat'
```

More in <https://docs.python.org/3/library/re.html>

### 1.0.8 urllib

```
[91]: # from urllib2 import urlopen # python 2
from urllib.request import urlopen
```

```
[ ]: for line in urlopen('http://dev.on-rev.com/myip.irev'):
    print(line)
    if 'IP' in str(line):
        IP = str(line).split(':')[1][:7]
```

```
print('-----')
print(IP)
```

### 1.0.9 Tables with power: pandas at <https://pandas.pydata.org/>

```
[92]: import pandas as pd
```

```
[94]: %%writefile data2.dat
# The following data are for test purpose
N    f    x    y type
1    2.3  6    8 star
2    3.5  7    9 galaxy
3   -4.2  5    7 cluster
#4  -10.5  5    7 test
```

Overwriting data2.dat

```
[95]: df = pd.read_csv('data2.dat', comment='#', delim_whitespace=True)
```

```
[96]: df
```

```
[96]:
```

	N	f	x	y	type
0	1	2.3	6	8	star
1	2	3.5	7	9	galaxy
2	3	-4.2	5	7	cluster

```
[97]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 5 columns):
#   Column  Non-Null Count  Dtype
---  -
0    N           3 non-null    int64
1    f           3 non-null    float64
2    x           3 non-null    int64
3    y           3 non-null    int64
4    type        3 non-null    object
dtypes: float64(1), int64(3), object(1)
memory usage: 248.0+ bytes
```

```
[98]: df.describe()
```

```
[98]:
```

	N	f	x	y
count	3.0	3.000000	3.0	3.0
mean	2.0	0.533333	6.0	8.0
std	1.0	4.142865	1.0	1.0

```

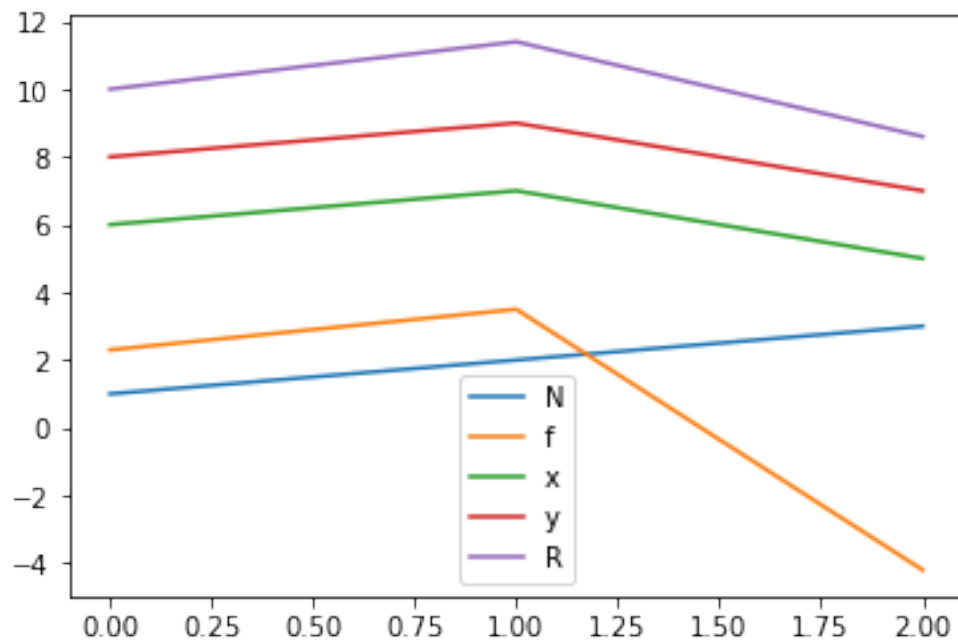
min    1.0 -4.200000  5.0  7.0
25%    1.5 -0.950000  5.5  7.5
50%    2.0  2.300000  6.0  8.0
75%    2.5  2.900000  6.5  8.5
max    3.0  3.500000  7.0  9.0

```

```
[100]: df['R'] = np.sqrt(df['x']**2 + df.y**2)
```

```
[101]: df.plot()
```

```
[101]: <matplotlib.axes._subplots.AxesSubplot at 0x7f96d0aa8550>
```



#### 1.0.10 Plotting with alternative style: seaborn at <https://seaborn.pydata.org/>

```
[102]: import seaborn as sns
```

```
[106]: fmri = sns.load_dataset("fmri") # medical data from https://github.com/mwaskom/
      ↪ Waskom_CerebCortex_2017
fmri
```

```
[106]:
```

	subject	timepoint	event	region	signal
0	s13	18	stim	parietal	-0.017552
1	s5	14	stim	parietal	-0.080883
2	s12	18	stim	parietal	-0.081033
3	s11	18	stim	parietal	-0.046134



4	s10	18	stim	parietal	-0.037970
...	...	...	...	...	...
1059	s0	8	cue	frontal	0.018165
1060	s13	7	cue	frontal	-0.029130
1061	s12	7	cue	frontal	-0.004939
1062	s11	7	cue	frontal	-0.025367
1063	s0	0	cue	parietal	-0.006899

[1064 rows x 5 columns]

```
[107]: sns.relplot(data=fmri, kind="line",
                  x="timepoint", y="signal", col="region",
                  hue="event", style="event");
```

