

CS2021 Python Programming

Week 3

Systems Programming

PP-Part II

Python Documentation

```
$open /Library/Frameworks/  
Python.framework/Versions/3.4/  
Resources/English.lproj/  
Documentation/index.html
```

Searching and Finding Documents

If you cannot locate docs on local machine
use web address

<http://www.python.org> or
<https://docs.python.org/3/>

Search for docs on

```
sys.getrecursionlimit()
```

Exploring Sys Module

Run the following

```
>>> import sys
```

```
>>> dir(sys)
>>> help(sys)
>>> help(sys.getrecursionlimit)
    Experiment with recursion limit
>>> sys.setrecursionlimit(50)
>>> sys.getrecursionlimit()
=>50
>>> def countdownrec(n):
    if n >0:
        print(n)
        countdownrec(n-1)
>>> countdown(52)
```

More.py

Load PP4E/System/more.py

```
def more(text, numlines=15):
    lines = text.splitlines()           # like
split('\n') but no " at end
    while lines:
        chunk = lines[:numlines]
```

```
    lines = lines[numlines:]
    for line in chunk: print(line)
    if lines and input('More?') not in ['y',
'Y']: break
if __name__ == '__main__':
    import sys                # when run,
not imported
    more(open(sys.argv[1]).read(), 10) #
page contents of file on cmdline
```

2 ways for running more.py

```
$cd PP4E/System
$python
>>>from more import more
>>>import sys
>>>more(sys.__doc__)
```

```
$cd PP4E/System
$python more.py more.py
                sys.argv
```

```

#file arguments.py
import sys
    # it's easy to print this list of course:
print sys.argv
    # or it can be iterated via a for loop:
for i in range(len(sys.argv)):
    if i == 0:
        print ("Function name: %s" %
sys.argv[0])
    else:
        print ("%d. argument: %s" %
(i,sys.argv[i]))
        Experiment with sys.argv
        PP4E/System/testargv2.py
"collect command-line options in a
dictionary"
def getoptv(argv):
    opts = {}
    while argv:
        if argv[0][0] == '-':           # find "-"
name value" pairs

```

```
        opts[argv[0]] = argv[1]           # dict
key is "-name" arg
        argv = argv[2:]
    else:
        argv = argv[1:]
    return opts
```

```
if __name__ == '__main__':
    from sys import argv                 #
example client code
    myargs = getopt(argv)
    if '-i' in myargs:
        print(myargs['-i'])
    print(myargs)
```

```
$C:\...\PP2E\System>python testargv2.py
{}
```

```
$C:\...\PP2E\System>python testargv2.py -i
data.txt -o results.txt
{'-o': 'results.txt', '-i': 'data.txt'}
```

CWD and Import Paths

```
#whereami.py
import os, sys
print('my os.getcwd =>',
      os.getcwd())
print('my sys.path =>',
      sys.path[:6])
      # shows first 6 import paths
input()
```

Testing whereami.py

```
$cd PP4E/System
```

```
$python whereami.py
```

```
$cd ..
```

```
$python System/whereami.py
```

```
$cd System/temp
```

```
$python ../whereami.py
```

Directory Tools (PP pg 163)

A common system task is to process a set of files in a directory (a folder)

Three ways to scan a directory

```
os.popen('ls')
os.listdir('.')
glob.glob('*')
```

```
>>> os.chdir('/Users/fredannexstein/Google
Drive/PP4E')
```

```
>>> os.popen('ls').readlines()
['Ai\n', 'Dbase\n', 'Dstruct\n', 'Gui\n',
'Integrate\n', 'Internet\n', 'Lang\n',
'LaunchBrowser.pyw\n',
'Launch_PyDemos.pyw\n',
'Launch_PyGadgets_bar.pyw\n',
'Launcher.py\n', 'Preview\n', 'PyDemos-pil-
note.txt\n',.....
```

```
>>> os.listdir('.')
['.DS_Store', '__init__.py', '__init__.pyc',
'__pycache__', '_pymailerrlog.txt', 'Ai',
'Dbase', 'Dstruct', 'echo.py', 'Gui',
'Integrate', 'Internet', 'Lang',
'Launch_PyDemos.pyw',
```

```
'Launch_PyGadgets_bar.pyw',  
'LaunchBrowser.pyw', 'Launcher.py',  
'launchmodes.py', 'Preview', 'PyDemos-pil-  
note.txt', 'PyDemos.pyw', 'PyGadgets.py',  
'PyGadgets_bar.pyw', 'README-PP4E.txt',  
'System', 'TempParts', 'textConfig.py',  
'Tools']
```

```
>>> import glob  
>>> glob.glob('*')  
['__init__.py', '__init__.pyc', '__pycache__',  
'_pymailerrlog.txt', 'Ai', 'Dbase', 'Dstruct',  
'echo.py', 'Gui', 'Integrate', 'Internet', 'Lang',  
'Launch_PyDemos.pyw',  
'Launch_PyGadgets_bar.pyw',  
'LaunchBrowser.pyw', 'Launcher.py',  
'launchmodes.py', 'Preview', 'PyDemos-pil-  
note.txt', 'PyDemos.pyw', 'PyGadgets.py',  
'PyGadgets_bar.pyw', 'README-PP4E.txt',  
'System', 'TempParts', 'textConfig.py',  
'Tools']
```

glob

Glob uses usual filename syntax of many shells for matching:

? -single char, * -any pattern , []- set of chars

```
>>> for path in
glob.glob('*.py'):
    ...    print(path)
__init__.py
echo.py
Launcher.py
launchmodes.py
PyGadgets.py
textConfig.py
```

```
>>> for path in glob.glob('./*/
*.py'):
    ...    print(path)
./Ai/__init__.py
./Dbase/__init__.py
./Dbase/castdump.py
./Dbase/castinit.py
./Dbase/filepickle.py
```

```
./Dbase/person-start.py
./Dbase/person.py
./Dbase/testdata.py
./Dstruct/__init__.py
./Gui/__init__.py
./Integrate/__init__.py
```

Walking Directory Trees

Application :Find all files with a global variable name

os.walk yields a sequence of 3-tuples

- Current directory

- All subdirectories

- All files within current directory

- Recursively walks subdirectories

```
import os
for (dirname,subdirs,files) in
os.walk
('/Users/fredannexstein/Google
Drive/PP4E'):
```

```
print ('[' + dirname + ']')
for f in files:
```

```
print(os.path.join(dirname, f))
```

Exercise

Using the `str.endswith()` function, write a python script that counts and prints all python files (ending with `.py`) in PP4E that reference name 'tkinter'.

You may run into the error:

UnicodeDecodeError: 'ascii' codec can't decode byte 0xef in position 0: ordinal not in range(128)

We will see later how to process such files, for now you will pass over errors by adding try-except handler

```
try:
```

```
        if 'tkinter' in
open(pname).read():
            print(pname)
            count = count
```

```
+1
```

except:

pass