0. Pre-Lab Questions

What is the difference between subprocess.Popen and os.system?

If you check out the subprocess section of the Python docs, you'll notice there is an example of how to replace os.system() with subprocess.Popen():

```
sts = os.system("mycmd" + " myarg")
```

...does the same thing as...

```
sts = Popen("mycmd" + " myarg", shell=True).wait()
```

The "improved" code looks more complicated, but it's better because once you know subprocess.Popen(), you don't need anything else. subprocess.Popen() replaces several other tools (os.system() is just one of those) that were scattered throughout three other Python modules.

If it helps, think of subprocess.Popen() as a very flexible os.system().

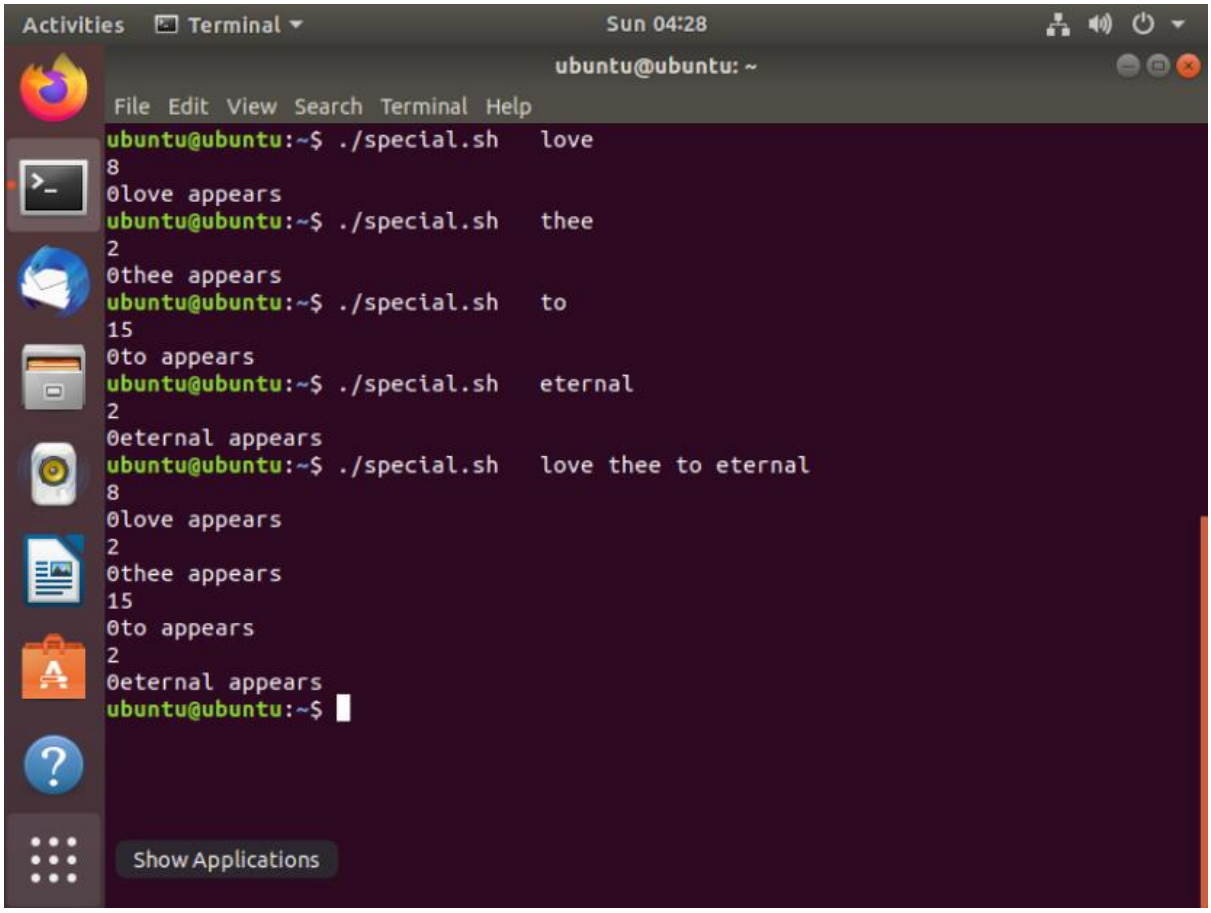
How do I access command line arguments in Python?

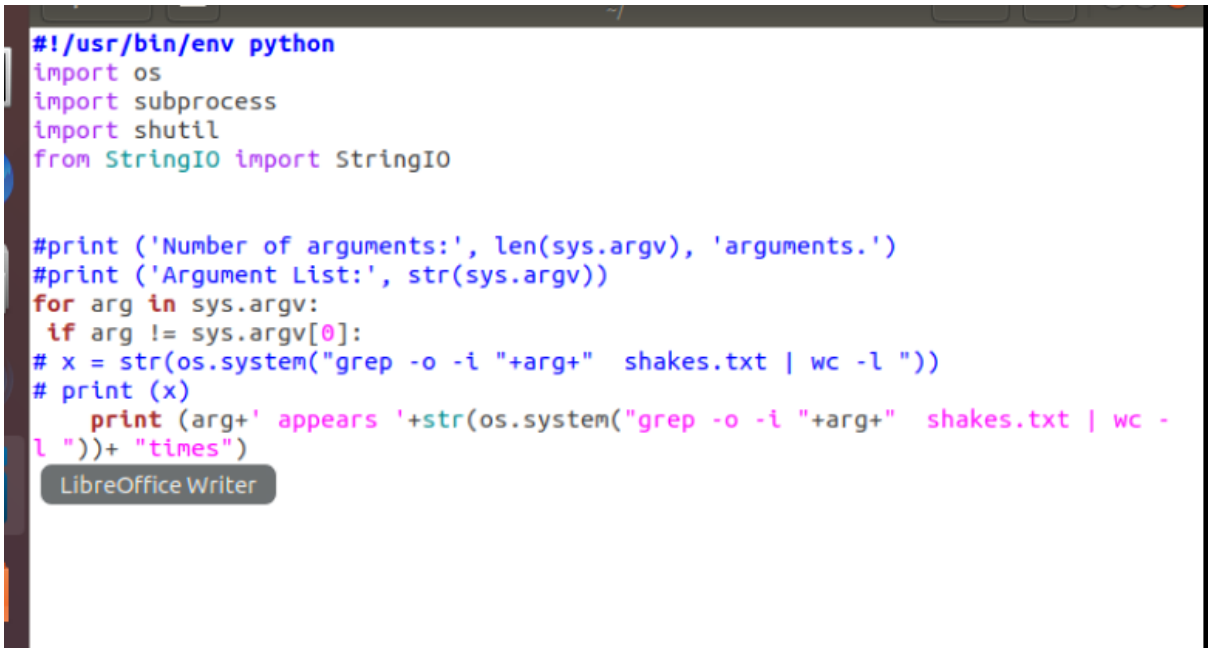
```
import sys

print(sys.argv)
```

More specifically, if you run python example.py one two three:

```
>>> import sys
>>> print(sys.argv)
['example.py', 'one', 'two', 'three']
```


Activities
Terminal
Sun 04:28
ubuntu@ubuntu: ~
File Edit View Search Terminal Help
ubuntu@ubuntu:~$ ./special.sh love
8
0love appears
ubuntu@ubuntu:~$ ./special.sh thee
2
0thee appears
ubuntu@ubuntu:~$ ./special.sh to
15
0to appears
ubuntu@ubuntu:~$ ./special.sh eternal
2
0eternal appears
ubuntu@ubuntu:~$ ./special.sh love thee to eternal
8
0love appears
2
0thee appears
15
0to appears
2
0eternal appears
ubuntu@ubuntu:~$
Show Applications


#!/usr/bin/env python
import os
import subprocess
import shutil
from StringIO import StringIO
#print ('Number of arguments:', len(sys.argv), 'arguments.')
#print ('Argument List:', str(sys.argv))
for arg in sys.argv:
if arg != sys.argv[0]:
# x = str(os.system("grep -o -i "+arg+" shakes.txt | wc -l "))
# print (x)
print (arg+' appears '+str(os.system("grep -o -i "+arg+" shakes.txt | wc -
l "))+ "times")
LibreOffice Writer