W. Alec Akin

IT244 – Python Programming

Purdue University Global

Dr. Ed Lavieri

March 30th, 2022

Text

Description automatically generated

First, we start the Python3 interpreter by executing the ‘python3’ command from a command line terminal

Text

Description automatically generated

Next, we import the modules we will be using in our assignment. This is effectively the same as starting out a Python program in an IDE with “import os.” It gives us the ability to use these packages and libraries and the functions and methods they contain.

Text

Description automatically generated

The first command/function we execute in the interpreter is ‘os.getlogin()’ which displays the currently running account. I was doing reverse engineering today and forgot to close the root account so that’s what we are seeing here.

Graphical user interface, text

Description automatically generated

In this screenshot we’ve used ‘os.get\_exec\_path()’ which is showing us all of the locations that different binaries and executables “live” on my MacBook.

Graphical user interface, text

Description automatically generated

In this screenshot, we execute ‘sys.path’ which shows us many of the Python frameworks, packages, and libraries I have installed. Not every package is listed here due to my use of ‘pipx’ for automated venv environments or ones that I’ve manually added in virtual environments, but the core ones installed with pip are listed.

Graphical user interface, text

Description automatically generated

Next, we leverage the ‘sys.byteorder’ function to display the byte order of my MacBook Pro. This goes back to the concept of byte order and ‘little’ tells us this is on ‘little-endian’ or “least significant byte first.”

Text

Description automatically generated

Next, we use ‘os.listdir(path=”/”)’ to view the contents of the main drive on this system or commonly referred to as “drive C” in the Windows world. Since MacOS is a Unix/Free BSD variant, we tell python to do this by listing the directories in “/” or “root” filesystem. As you can see, I forgot I was in the Python Shell momentarily and tried to “cd” to root filesystem which obviously isn’t going to work and did not. Muscle-memory can getcha’ sometimes!

Graphical user interface

Description automatically generated

This one took way more time and a couple hours of reading docs (there’s limited documentation), forum posts, Reddit posts, Stack Overflow, etc. and I couldn’t get it to work with just the regular syntax. I did try using the shlex.split function to split the commands properly, but subprocess.Popen really didn’t like the Unix redirect, or pipe – even when I used stdout=PIPE. I hope it’s ok, but I wrote a tiny little shell script that would encapsulate what I needed, called the script, set shell=True, and executed with ‘/bin/sh’ to get the output. In this screenshot you will see the Python3 shell, the output of the script in pythonOut.txt, and you’ll also see the tiny shell script I wrote.

A screenshot of a computer

Description automatically generated with medium confidence

This is quite different than the Windows command from the guide. Essentially, you need to call osascript, give it the -e argument, and pass the string “tell application \”Calculator\” to activate” which will spawn the Calculator app (also note that quotes need to be escaped). It seems like Windows is actually a bit more straightforward with subprocess.Popen, and even mainline Linux. My guess is that with all of the protections Apple institutes on MacOS that you can’t just call the Calculator.app directly – you need to do so through an intermediary.

# References

BogoToBogo. (n.d.). *Python Tutorial: subprocesses module - 2020*. Retrieved March 30, 2022, from https://www.bogotobogo.com/python/python\_subprocess\_module.php

Brent, M. (2022, February 19). *Shell Scripting Tutorial: How to Create Shell Script in Linux/Unix*. Guru99. Retrieved March 30, 2022, from https://www.guru99.com/introduction-to-shell-scripting.html

Jarred, C. (2021, March 13). *Finding path to “reminder.app” on Mac terminal (non-English localization)*. Stack Overflow. Retrieved March 30, 2022, from https://stackoverflow.com/questions/66605034/finding-path-to-reminder-app-on-mac-terminal-non-english-localization

Kinder, K. (2011, January 25). *python subprocess calling bash script - need to print the quotes out too*. Stack Overflow. Retrieved March 30, 2022, from https://stackoverflow.com/questions/4798331/python-subprocess-calling-bash-script-need-to-print-the-quotes-out-too

Python Software Foundation. (n.d.-a). *os — Miscellaneous operating system interfaces — Python 3.10.4 documentation*. Python. Retrieved March 30, 2022, from https://docs.python.org/3/library/os.html#module-os

Python Software Foundation. (n.d.-b). *subprocess — Subprocess management — Python 3.10.4 documentation*. Python. Retrieved March 30, 2022, from https://docs.python.org/3/library/subprocess.html

Python Software Foundation. (n.d.-c). *sys — System-specific parameters and functions — Python 3.10.4 documentation*. Python. Retrieved March 30, 2022, from https://docs.python.org/3/library/sys.html#module-sys

Shaik, H. K. (2021, June 4). *How to Run bash scripts Using Python?* Geekflare. Retrieved March 30, 2022, from https://geekflare.com/python-run-bash/