Python - Custom Functions and OS Module

OS Command

This script allows you to send commands to your OS through the input() function in Python.

Note: We use 'clear' to clear the screen. This is for MacOS and Linux. For Windows use 'cls'.

lab-command.py

```
import os
os.system('clear')

while True:
    command = input('Command: ')
    os.system('clear')

    try:
        response = os.popen(command).read()
    except:
        response = 'ERROR'
    finally:
        print(response)
```

Up/ Down Script

This script creates a basic Up/Down tool for a site or a server. We create a function to ping a host and then simply print the results to the screen.

Note: This works on MacOS and Linux. The Ping command works differently on Windows.
For Windows have command = f'ping {site}

lab-up-down.py

```
from time import sleep

site = 'cnn.com'

def status(site):
    command = f'ping -c 1 {site}'
    response = os.popen(command).read()

    return response

while True:
    os.system('clear')
    result = status(site)

    print(site)
    print(result)

    sleep(1)
```

Up/ Down HTML Dashboard

This script allows you to ping multiple hosts and then create an HTML web page based on the results. The script loops every 2 seconds, and the HTML page will auto update every 5. If a site responds we will change the backgound-color CSS attribute to green, and if it doesn't it will be red.

Note: Windows users... EXTRA CREDIT... Ping responds differently in Windows so you'll have to find a string that is in the response that you can trigger off of.

Linux users... instead of '1 packets received' you may need to change to '1 packet received'.

lab-dashboard.py

```
import os
from time import sleep

site = ['cnn.com', 'fox.com', 'tacobell.com', 'notarealsite.tv']
header = '<meta http-equiv="refresh" content="5">'

def status(site):
    color = 'red'
    command = f'ping -c 1 {site}'
    response = os.popen(command).read()

    if '1 packets received' in response:
        color = 'green'

    return response, color

while True:
    os.system('clear')
    page = header
    for item in site:
        result = status(item)
        page = f'''
                    {page}
                    <h2 style="background-color:{result[1]};">{item}</h2>
                    <pre>{result[0]}</pre>
                '''

    try:
        with open('dashboard.html', 'w') as file:
            file.write(page)
    except:
        print('ERROR - Writing to File')

    sleep(1)
```