## L07 Homework Assignment – Remote Deployment

### **Background**

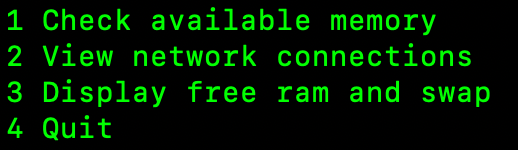
According to the Linux Foundation, 96.3% of the world's top one million servers run on Linux and 95% of all cloud infrastructure operates on it. According to DevOps.com, 83.1% of developers surveyed said Linux is the platform they prefer to work on.

Developing applications in Python and other languages often end up deployed on Linux severs. In this assignment, you will develop a Python application which implements a class whose objects can carry out a set of specified actions. These actions will consist of Linux utilities that will allow a Linux user to easily run 3 different utilities from a simple menu system. The Linux utilities will be provided for you.

You will need a module with the menu class and another module with your main function and deploy them both to the ember lab Linux environment. After deploying to a Linux server, you will also need to test your application via ssh terminal at the command line.

### **Procedure**

Create a Python program that implements a class with a menu to start the specified Linux utilities.

1. Create two Python modules, main.py and menu.py
2. In the menu.py module, create a “Menu” class. An object of this class will display:
   1. Then the menu waits for the user to supply a value. If the user does not supply a valid value (1-4), the user is notified of the proper usage and menu is redisplayed. The user should be allowed to try again.
3. In the Menu class, you will need to
   1. Create an empty list for menu options
   2. Method to display the menu
   3. Method to Collect Input

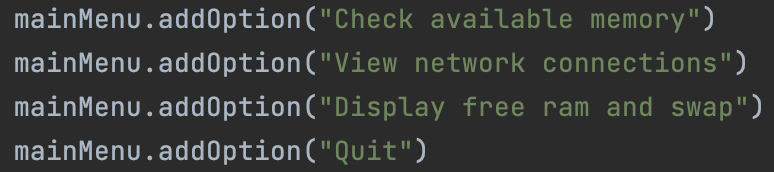
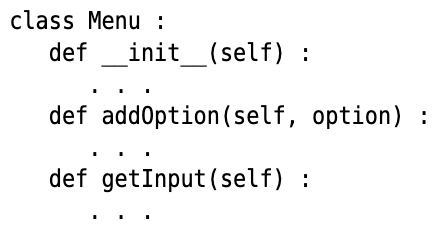
A screenshot of a computer

Description automatically generated

1. Create an empty menu object then add options to a list to hold the options. An example of an addOption method could be:

A screen shot of a computer program

Description automatically generated

1. Specify the public interface in the main module as a main() function.
   1. Create an instance of the Menu() class
   2. Establish the menu options by using an addOption method such as: 
   3. Call a getUserInput method to collect user input: 
2. To complete the public interface, you need to specify the constructor. Create a constructor that requires no arguments for the public interface (basically): 
3. Add documentation to the public interface class. This can be # comments or doc strings. An examples could be: A screenshot of a computer program

   Description automatically generated
4. Determine the instance variables. The object needs to be able to process every method using just its instance variables and the method arguments. Start with the addOption method. Store the options in a collection that can be efficiently updated by appending or using index values.
5. To verify valid input for getInput method, you should read the input as an integer. Then compare the input known quantity of menu options. The len() function is useful for this if you store them in a list.
6. Implement the constructor of the class which defines and initializes any instance variables such setting \_options as an empty list. A black text with black letters

   Description automatically generated with medium confidence
7. Implement the class methods. Implement the methods in your class, one at a time. For example, here is the implementation of the addOption method: A close-up of a word

   Description automatically generated
8. Create a getInput method to collect user inputs and validate against the options 1 - 4. If the user inputs an incorrect value, notify the user via proper usage message and provide menu options again.
9. If the user does not enter an invalid entry or Q to quit, call the run\_bash\_cmd function to perform the Linux utilities. This will require importing the [“os” library.](https://docs.python.org/3/library/os.html)
10. The run\_bash\_cmd function is provided and will need to be copied into your project.
11. Overall, your class should have at least 2 methods.
12. Deploy your Python project to your home directory on ember.hpc.lab Linux server.
    1. Host: ember.hpc.lab - use 10.200.208.103
    2. User: Canvas user name, ie cnavarro1234
    3. Pass: Canvas password
    4. Root path: /home/cnavarro1234
    5. Deployment path: menu\_linux
13. To start the Python program on ember lab, either connect via Pycharm terminal or ssh. From CMD prompt: **ssh** [**cnavarro1234@10.200.208.103**](mailto:cnavarro1234@10.200.208.103) **(**

Once connected to Linux shell, then type: **python3 menu\_linux/main.py**

1. Review the included video that provides an example of how the interface should respond.
2. **Submit two python modules**, main.py and menu.py.
3. **Submit word/pdf document** with screenshot(s) of all (1-4) input and output. You will need to test from the Pycharm terminal connected to ember.hpc.lab ( 10.200.208.103) or from a ssh utility in the cmd prompt. Your screenshots should resemble the video provided, but can be 1 or more stills.

**Paste Screenshot**

Rubric

|  |  |
| --- | --- |
| **Requirement** | **Points** |
| Two python modules submitted with proper use of import | 20 |
| Doc/Pdf shows program output in screenshot(s) running successfully deployed on ember lab with all 4 user inputs tested. | 20 |
| Class (one or more) defined which contains a menu system that can be used with the run\_bash\_cmd function. Class should also have a getInput, addOption method for adding menu choices. | 20 |
| Establish the menu options by using an addOption method against an instance of the Menu class. | 10 |
| Python program runs without error displaying a menu and allowing a user to select Linux utilities by entering 1-4. | 10 |
| Allow user to exit the program with q or Q to quit. | 10 |
| Handle invalid input. | 10 |
|  |  |