Classes for Storing Network Information

In this lab you will define a class, use that class definition to create objects to store information about networking devices, and print the information for those network device objects.

Note: For this exercise you will hard-code your device information as you create your network device objects.

Step 1

Define a class called NetworkDevice. Define a method within the class that takes device information as parameters: device name, OS-type, IP address, username, and password. Allow the username and password to be omitted, providing default values of 'cisco' and 'cisco'.

Answer

```
class NetworkDevice():

    def set_info(self, name, os, ip, user='cisco', pw='cisco'):
        self.name = name
        self.ip_address = ip
        self.os_type    = os
        self.username = user
        self.password = pw
```

Step 2

Define a function to print a table of device information (name, OS-type, IP, username, password) for every device. Pass in a list of devices, where each device is an object of type NetworkDevice.

Answer

```
#---- Function to go through devices printing them to table -----------
def print_device_info(devices_list):

    print ''
    print 'Name        OS-type  IP address   Username  Password'
    print '---------   -------  ----------   --------  --------'

    # Go through the list of devices, printing out values in nice format
    for device in devices_list:

        print '{0:11} {1:8} {2:12} {3:9} {4:9}'.format(device.name,
                                                      device.os_type,
                                                      device.ip_address,
                                                      device.username,
                                                      device.password)

    print ''
```

Step 3

Your 'main' code should create two or more NetworkDevice objects. For each object, call your method to set the device information.

Note: since you are hard-coding these devices, you are not reading from a file, or using a loop. Create the first object and set its info, then create the second object and set its info.

Answer

After creating your device objects, add them to a Python list of devices. Call your print function, passing in the devices list.

```
#---- Main: read device info, then print ------------------------------

dev1 = NetworkDevice()
dev1.set_info('dev1','IOS-NX','9.9.9.9')

dev2 = NetworkDevice()
dev2.set_info('dev2','IOS-XE','8.8.8.8','chuck','secret')

print_device_info([dev1,dev2])
```

Defining a Class with Information Set at Initialization

In this lab, you will define a network device class, with an initialization method for setting attributes for each created object.

You will read device information from two files - the PRNE/section12/devices and PRNE/section12/real-devices.

Step 4

Define a class called NetworkDevice. Define an initialization method (called __init__) within the class that takes device information as parameters (device name, OS-type, IP address, username, and password.

Answer

Remember that 'self' must be the first parameter for every method.

```
#---- Class to hold information about a network device ----------------
class NetworkDevice():

    def __init__(self, name, ip, os, user='cisco', pw='cisco'):
        self.name = name
        self.ip_address = ip
        self.os_type    = os
        self.username = user
        self.password = pw
```

Step 5

Create a function that takes the name of the devices file as input, reads the device information from the file, and creates network device objects, adding them to a list of devices. The result will be a list of network device objects, based on the information read from the file. The function should return the list of devices to the caller.

Answer

```
#---- Function to read device information from file -------------------
def read_device_info(devices_file):

    devices = [] # Create a list for all devices

    # Read in the devices from the file
    file = open(devices_file,'r')
    for line in file:

        device_info = line.strip().split(',') # Get device info into list

        # Create a device object with this data
        device = NetworkDevice(device_info[0],device_info[2],
                               device_info[1],device_info[3],device_info[4])

        devices.append(device) # add this device object to list

    file.close() # Close the file since we are done with it

    return devices # return a reference to the list we created
```

Step 6

Create a print function that takes as input a list of network device objects, and prints a table of the devices from the list.

Answer

```
#---- Function to go through devices printing them to table -----------
def print_device_info(devices_list):

    print ''
    print 'Name        OS-type  IP address       Username  Password'
    print '---------   -------  --------------   --------  --------'

    # Go through the list of devices, printing out values in nice format
    for device in devices_list:

        print '{0:11} {1:8} {2:16} {3:9} {4:9}'.format(device.name,
                                                      device.os_type,
                                                      device.ip_address,
                                                      device.username,
                                                      device.password)

    print ''
```

Step 7

Your main code will (a) call the function to read device information from the file, and (b) print the device information. It will do this twice; once for the 'devices' file, and once for the 'real-devices' file.

Answer

```
#---- Main: read device info, then print ------------------------------

devices_list = read_device_info('devices')
print_device_info(devices_list)

devices_list = read_device_info('real-devices')
print_device_info(devices_list)
```