

Lab week 6 objective:

- Working with files in memory
 - Reading and writing to file in memory
- Converting from bin to decimal numbers
- Practicing working with lists from lesson 2 from week 1
- Practice splitting strings from lesson 2 from week 1

Lab Time:

Start: February 8, 2023 2:00pm

Due: February 14, 2023 11:59pm

General comments:

- Make sure that you use proper spaces needed in the blocks:
 - with open () as name
 - while loops
 - if statements
 - try- except-finally statements, functions

MARKS

Step	Comments	Marks
Start #2	Download the 2 .txt file	2
Setup #2	Name .py file	3
Setup #3	3 variables to hold paths, filenames	3
Setup #4	Variable to hold IP address info	2
Step #1	Create a file object for file opening TXT format	1
	Create a file object for file opening BIN format	1
Step #1	File object: Shuffled List (use method in 1a)	2
Step #2	File object: Binary file (use method as above) Complete step 2	4
Step #4	Convert form BIN to DEC	2
Step #5	Creation file open() and follow steps a,b,c	3
Step #6	Use .sort method	2
Step #7	Creation file open() for sorted file Complete Step 7	2
	TOTAL MARKS	27

Start of Lab

1. You are going to work with 3 files, 2 are going to be downloaded from FOL and the third you are going to create.
2. Download `ipaddress_shuffled_list.txt` and `ipaddress_shuffled_list_bin.txt`
3. In the lab section and save them in a place you have access
4. The two files hold 2 lists of private IP address in the format of `10.x.x.x`.
 - a. The second octet tells us if the IP address is a switch(1), server(2), or host(3).
 - b. The third octet says for which floor the device is located on.
 - c. The 4th octet holds is the host.

Your goal in the Labs is to write a python script that would go through the IP address in `ipaddress_shuffled_list.txt` and `ipaddress_shuffled_list_bin.txt` and sort them as follow

- IP Address that have 1 in the second octet (10.1.x.x) will be put in `switch_ipaddress.txt`
- IP Address that have 2 in the second octet (10.2.x.x) will be put in `server_ipaddress.txt`
- IP Address that have 3 in the second octet (10.3.x.x) will be put in `host_ipaddress.txt`

The catch is that the

- `ipaddress_shuffled_list` file has address in decimal
- `ipaddress_shuffled_list_bin` has the address in binary, 1010.1111.111.1111 format.

Your goal is to create a single list that combines both list that is in decimal format, ex 10.2.25.125. and it would be saved in a third file that you will create.

Setup steps

1. If you have not done so already, download text files: `ipaddress_shuffled_list.txt` and `ipaddress_shuffled_list_bin.txt` and save them in a known location(use for later).
2. Create a python file with *you_initials_ip_sort.py* and save it in the same folder that you saved the download text files.
3. If your name is John Smith then it should be `JS_ip_sort.py`

4. Create 3 variables that would hold the path and file names for each of the files. Don't forget to add the extension of the file
 - a. One for `ipaddress_shuffled_list.txt`;
 - b. One for `ipaddress_shuffled_list_bin.txt`; and
 - c. One for `[your initials]_ipaddress_sorted.txt` .
 - i. **If your name is John Smith then it should be `JS_ipaddress_sorted.txt`**
5. Create a list that would hold the IP address from `ipaddress_shuffled_list.txt` and `ipaddress_shuffled_list_bin.txt`, in a decimal format.

Steps

1. Create a file object using the `open()` function with the option of reading from shuffled list.
 - a. **With `open(shuffled_list, "r")` as `f`:**
 - b. Read and add each line of the file without add the newline command `"\n"` to the list that holds the IP address.
 - i. **Example can be seen in `connect_to_files.py` found in the lecture of week 6.**
2. Create another file object using the `open()` function with the option of reading from the Binary Shuffled list.
 - a. Same as above but before adding to collective list change the bin IP address to Decimal
 - i. Ex: `111.111.111.111` to `5.5.5.5`
 - ii. To do this, you will need to split the IP address to 4 octets

Example:

```
with open(shuffled_list, "r") as f:
    for line in f:
        #removing the \n at the end of the line
        line_wo_nline = line[:-1]

        b_ip_list = line_wo_nline.split(".")
```

This creates a list named `b_ip_list` that has for items. One for each octet. Take each octet and change the bin value to its decimal value using the `int()` function. In lecture is gives an example how to convert a bin to decimal.

- iii. Turn each octet from binary to decimal. Use the `int()` option using the of using a base. Review `Int(string,base)` in the last lecture
- iv. Once having the decimal values for each of the octets, reassemble the octets to a correct IP address string, ex "10.3.7.25"

3. You can use the following as a way to format a string

4. `Ip = "{0}.{1}.{2}.{3}\n".format(10,4,11,22)`

`{#}` holds a spot for a variable that associated to the a location of variables' location in the `format()` function location. In our case `{0}` will be 10, `{1}` is 4, and so one. Use the variables DEC values gotten from the IP octets . this case you will not have any space around the "."s.

Add the formed IP address it to the IP address collection list.

- 5. Create a file object using the `open()` function with option to write to file you have created with the name of `[your initials]_ipaddress_sorted.txt`
 - a. Use the write function of the file object and `[your initials]_ipaddress_sorted.txt`. An example can be seen in seen in the `connect_to_files.py` in the form. Located in FOL weeks 6 lecture folder.
 - b. Firstly, you should add a line saying "file created by and your name", put your name so I would know it is yours.
 - c. Remember, using the writing mode, data will delete if the file exists.
- 6. Sort the list using the list method called `sort()`. For example `list.sort()`
 - a. Note if you have IP address 9.10.125.25 and 9.10.25.25 the order would be 9.10.125.25 and 9.10.25.25. This is because we are using strings.

7. Create file object `open()` this time use the append the file `[your_initials]_ipaddress_sorted.txt`, this will allow you to add information to the end of the file. We want to add the address after the line with your name.

- a. in this step you will add the IP_address data
 - i. could be done either using `writelines()` function.

`with open(file,mode) as f:`

`f.writelines(data)`

or a for loop and use the `f.write(data)`

The file should have the list of the IP address in decimal format, 10.2.12.34. Each IP address should be on its own line.

10.2.12.34

10.3.9.12

Lab Scoring:

- Lab is graded out of 27 points for 2.5% of your final grade.
- Submitting files in zip format will result in 0.

Submission:

Submit your .py file

`[your_initials]_ip_sort.py`

The python .py file that has the script that you wrote and the text file that has the output.