

LAB – USING TUPLES AND SETS**OBJECTIVE**

In this lab, you will be working with tuples and sets.

PART 1

Open a terminal and switch to the lab directory

STEP 1: OPEN A TERMINAL WINDOW

Double-click the Terminal icon on the desktop to open the terminal window for use in this lab.

STEP 2: CHANGE DIRECTORY

Change to the directory **labs/prne/** in the user home directory, which holds the files for the course labs.

```
~$ cd labs/prne/
```

PART 2

Open **Visual Studio Code**, create a new file and save it with a filename of **using-tuples-and-sets-part-2.py**. Ensuring to save the file in the **~/labs/prne/** directory, as otherwise the code will require modification to find the associated files that are used. This python application will read the device information from the file, and display the items from the file into a tuple.

STEP 1: IMPORT PPRINT

Import the pprint function to enable displaying the list nicely formatted.

```
# Import required modules/packages/library
from pprint import pprint
```

STEP 2: OPEN FILE FOR READING

Open the file **devices-05.txt** for reading.

```
# Open the file and read in the single line of device info
file = open('devices-05.txt', 'r')
file_line = file.readline().strip()
```

STEP 3: DISPLAY CONTENTS READ IN

Display the contents of the file that was read in.

```
# Display the line I just read
print('Read line: ', file_line)
```

STEP 4: CREATE TUPLE

Convert the contents of the file into a tuple, splitting the string using the comma.

```
# Convert to a tuple, using 'split()' to provide a python list
device_info = tuple(file_line.split(','))
```

STEP 5: DISPLAY THE TUPLE

Display in the terminal the tuple nicely formatted.

```
# Display a blank line to make easier to read
print('')

# Display a title
print('Input converted to a tuple:')

# Display the dictionary with nice formatting
pprint(device_info)
```

STEP 6: CLOSE FILE

Close the file.

```
# Close the file
file.close()
```

STEP 7: SAVE, RUN AND VERIFY APPLICATION

Save you application and then run it from the terminal rather than from within visual studio code.

```
~/labs/prne$ python3 using-tuples-and-sets-part-2.py
```

The output from your application will be displayed in your terminal window, verify that it is comparable to below.

```
devasc@labvm:~/labs/prne$ python3 using-tuples-and-sets-part-2.py
Read line:  device1,ios,10.3.21.5,user1,pass1

Input converted to a tuple:
('device1', 'ios', '10.3.21.5', 'user1', 'pass1')
```

PART 3

Open **Visual Studio Code**, create a new file and save it with a filename of **using-tuples-and-sets-part-3.py**. Ensuring to save the file in the **~/labs/prne/** directory, as otherwise the code will require modification to find the associated files that are used. This python application will read multiple device information from the file, and display the items from the file into a list of tuples.

STEP 1: IMPORT PPRINT

Import the pprint function to enable displaying the list nicely formatted.

```
# Import required modules/packages/library
from pprint import pprint
```

STEP 2: CREATE EMPTY LIST

Create an empty outer list called devices.

```
# Create the outer list for all devices
devices = []
```

STEP 3: OPEN FILE FOR READING

Open the file **devices-04.txt** for reading.

```
# Open the file and read in the device info
file = open('devices-04.txt', 'r')
for line in file:
```

STEP 4: ADD TO TUPLE

Put the device information into a tuple, splitting the string using the comma.

```
# Add device info into tuple
device_info = tuple(line.strip().split(','))
```

STEP 5: DISPLAY CONTENTS READ IN

Display the contents of the file that was read in and what has been built so far.

```
# Display what we have read and built so far
print('Read line: ', device_info)
```

STEP 6: APPEND TUPLE TO LIST

For each device, take the device information tuple you have created and put the tuple into a list.

```
# Append our device and its info onto our 'devices' list
devices.append(device_info)
```

STEP 7: DISPLAY THE TUPLE

Display in the terminal the tuple nicely formatted.

```
# Display a blank line to make easier to read
print('')

# Display a title
print('Input converted to a list tuples:')

# Display the tuple with nice formatting
pprint(devices)
```

STEP 8: CLOSE FILE

Close the file.

```
# Close the file
file.close()
```

STEP 9: SAVE, RUN AND VERIFY APPLICATION

Save you application and then run it from the terminal rather than from within visual studio code.

```
~/labs/prne$ python3 using-tuples-and-sets-part-3.py
```

The output from your application will be displayed in your terminal window, verify that it is comparable to below.

```
devasc@labvm:~/labs/prne$ python3 using-tuples-and-sets-part-3.py
Read line: ('device1', 'ios', '10.3.21.5', 'user1', 'pass1')
Read line: ('device2', 'ios', '10.3.21.6', 'user2', 'pass2')
Read line: ('device3', 'nx-os', '10.3.21.7', 'user3', 'pass3')
Read line: ('device4', 'nx-os', '10.3.21.8', 'user4', 'pass4')
Read line: ('device5', 'ios-xr', '10.3.21.9', 'user5', 'pass5')
Read line: ('device6', 'ios-xr', '10.3.21.10', 'user6', 'pass6')

Input converted to a list of tuples:
[('device1', 'ios', '10.3.21.5', 'user1', 'pass1'),
 ('device2', 'ios', '10.3.21.6', 'user2', 'pass2'),
 ('device3', 'nx-os', '10.3.21.7', 'user3', 'pass3'),
 ('device4', 'nx-os', '10.3.21.8', 'user4', 'pass4'),
 ('device5', 'ios-xr', '10.3.21.9', 'user5', 'pass5'),
 ('device6', 'ios-xr', '10.3.21.10', 'user6', 'pass6')]
```

PART 4

Open **Visual Studio Code**, create a new file and save it with a filename of **using-tuples-and-sets-part-4.py**. Ensuring to save the file in the **~/labs/prne/** directory, as otherwise the code will require modification to find the associated files that are used. This python application will read multiple device information from the file, and create a dictionary of named tuples.

STEP 1: IMPORT PPRINT AND NAMEDTUPLE

Import the pprint function to enable displaying the list nicely formatted and the namedtuple function from the collections module.

```
# Import required modules/packages/library
from pprint import pprint
from collections import namedtuple
```

STEP 2: CREATE NAMED TUPLE

Create a named tuple for storing information about devices.

```
# Create named tuple to store information
Dev_info = namedtuple('Dev_info', ['name', 'os', 'ip', 'user',
                                   'password'])
```

STEP 3: CREATE EMPTY DICTIONARY

Create an empty dictionary called devices to hold all devices.

```
# Create the dictionary
devices = {}
```

STEP 4: OPEN FILE FOR READING

Open the file **devices-04.txt** for reading the information one line at a time.

```
# Open the file and read in the device info
file = open('devices-04.txt', 'r')
for line in file:
```

STEP 5: ADD EACH DEVICE INTO NAMED TUPLE

For each device, take the device information and put it into a named tuple.

```
# Add device info into a named tuple
device_info = Dev_info(*(line.strip().split(',')))
```

STEP 6: DISPLAY CONTENTS

Display the contents of the file that was read in and what has been built so far.

```
# Display what we have read and built so far
print('Device Information: ', device_info)
```

STEP 7: ADD NAMED TUPLE TO DICTIONARY

For each device, add the named tuple to a dictionary holding all device information that is indexed by device name.

```
# Add the named tuple to a dictionary
devices[device_info.name] = device_info
```

STEP 8: DISPLAY THE TUPLE

Display in the terminal the tuple nicely formatted.

```
# Display a blank line to make easier to read
print('')

# Display a title
print('Input converted to a dictionary of named tuples:')

# Display the tuple with nice formatting
pprint(devices)
```

STEP 9: CLOSE FILE

Close the file.

```
# Close the file
file.close()
```

STEP 10: SAVE, RUN AND VERIFY APPLICATION

Save your application and then run it from the terminal rather than from within Visual Studio Code.

```
~/labs/prne$ python3 using-tuples-and-sets-part-4.py
```

The output from your application will be displayed in your terminal window, verify that it is comparable to below.

```
devasc@labvm:~/labs/prne$ python3 using-tuples-and-sets-part-4.py
Device Information: Dev_info(name='device1', os='ios', ip='10.3.21.5', user='user1', password='pass1')
Device Information: Dev_info(name='device2', os='ios', ip='10.3.21.6', user='user2', password='pass2')
Device Information: Dev_info(name='device3', os='nx-os', ip='10.3.21.7', user='user3', password='pass3')
Device Information: Dev_info(name='device4', os='nx-os', ip='10.3.21.8', user='user4', password='pass4')
Device Information: Dev_info(name='device5', os='ios-xr', ip='10.3.21.9', user='user5', password='pass5')
Device Information: Dev_info(name='device6', os='ios-xr', ip='10.3.21.10', user='user6', password='pass6')

Input converted to a dictionary of named tuples:
{'device1': Dev_info(name='device1', os='ios', ip='10.3.21.5', user='user1', password='pass1'),
 'device2': Dev_info(name='device2', os='ios', ip='10.3.21.6', user='user2', password='pass2'),
 'device3': Dev_info(name='device3', os='nx-os', ip='10.3.21.7', user='user3', password='pass3'),
 'device4': Dev_info(name='device4', os='nx-os', ip='10.3.21.8', user='user4', password='pass4'),
 'device5': Dev_info(name='device5', os='ios-xr', ip='10.3.21.9', user='user5', password='pass5'),
 'device6': Dev_info(name='device6', os='ios-xr', ip='10.3.21.10', user='user6', password='pass6')}
```

PART 5

Open **Visual Studio Code**, create a new file and save it with a filename of **using-tuples-and-sets-part-5.py**.

Ensuring to save the file in the **~/labs/prne/** directory, as otherwise the code will require modification to find the associated files that are used. This python application will read multiple device information from the file, and create a set of all the OS types present for the list of devices.

STEP 1: IMPORT PPRINT AND NAMEDTUPLE

Import the pprint function to enable displaying the list nicely formatted and the namedtuple function from the collections module.

```
# Import required modules/packages/library
from pprint import pprint
from collections import namedtuple
```

STEP 2: CREATE NAMED TUPLE

Create a named tuple for storing information about devices.

```
# Create named tuple to store information
Dev_info = namedtuple('Dev_info', ['name', 'os_type', 'ip', 'user',
                                   'password'])
```

STEP 3: CREATE EMPTY SET

Create a set for holding all OS types.

```
# Create a set for holding OS types
os_types = set()
```

STEP 4: OPEN FILE FOR READING

Open the file **devices-04.txt** for reading the information one line at a time. Put the information for each device into a named tuple.

```
# Open the file and read in the device info
file = open('devices-04.txt', 'r')
for line in file:

    # Add device info into a named tuple
    device_info = Dev_info(*(line.strip().split(',')))
```

STEP 5: DISPLAY CONTENTS

Display the contents of the file that was read in and what has been built so far.

```
# Display what we have read and built so far
print('Device Information: ', device_info)
```

STEP 6: CREATE LIST

For each device, add the OS type to the set holding all OS types.

```
# Add the OS type to the set holding all OS types
if device_info.os_type not in os_types:
    os_types.add(device_info.os_type)
```

STEP 7: DISPLAY THE SET

Display in the terminal the set nicely formatted.

```
# Display a blank line to make easier to read
print('')

# Display a title
print('Input converted to a set of OS types present:')

# Display the tuple with nice formatting
pprint(os_types)
```

STEP 8: CLOSE FILE

Close the file.

```
# Close the file
file.close()
```


STEP 9: SAVE, RUN AND VERIFY APPLICATION

Save your application and then run it from the terminal rather than from within visual studio code.

```
~/labs/prne$ python3 using-tuples-and-sets-part-5.py
```

The output from your application will be displayed in your terminal window, verify that it is comparable to below.

```
devasc@labvm:~/labs/prne$ python3 using-tuples-and-sets-part-5.py
Device Information: Dev_info(name='device1', os_type='ios', ip='10.3.21.5', user='user1', password='pass1')
Device Information: Dev_info(name='device2', os_type='ios', ip='10.3.21.6', user='user2', password='pass2')
Device Information: Dev_info(name='device3', os_type='nx-os', ip='10.3.21.7', user='user3', password='pass3')
Device Information: Dev_info(name='device4', os_type='nx-os', ip='10.3.21.8', user='user4', password='pass4')
Device Information: Dev_info(name='device5', os_type='ios-xr', ip='10.3.21.9', user='user5', password='pass5')
Device Information: Dev_info(name='device6', os_type='ios-xr', ip='10.3.21.10', user='user6', password='pass6')

Input converted to a set of OS types present:
{'nx-os', 'ios-xr', 'ios'}
```

PART 6 (OPTIONAL BUT HIGHLY RECOMMENDED)

As this lab is completed in NETLAB+ and your code files will be erased when the reservation ends, it is advisable to save your files in GitHub under your repository for this course.