"Router Configuration using Python"

A Project Report

Submitted in part fulfilment of the requirements for the

Industrial Summer Training, 2019

Submitted by:

(SIDDHARTH VASHISHT, 01911503117)

(Bachelor of Technology in Information Technology)

Under the Supervision of

Mr. Nikesh Garg

Senior Manager

Nokia Corporation

Bharati Vidyapeeth's College of Engineering

New Delhi-110063

Session: 2017-2021

NOKIA

<u>CERTIFICATE</u>

Certified that **Siddharth Vashisht** has carried out the project work titled **"Router Configuration using Python"** from **4th June 2019** to **23rd July 2019** at **Nokia Corporation** under my supervision. The report includes original work and studies carried out by student himself and the contents of the report do not form the basis for the award of any other degree to the candidate or to anybody else.

Mr. Nikesh Garg

Designation: Senior Manager

Place: Gurugram, Haryana

Date: 23rd July 2019

CANDIDATES DECLARATION

I hereby certify that the work which is being presented in this project titled **"Device Configuration using Python"** submitted to **"Bharati Vidyapeeth's College of Engineering"** is an authentic record of my own work carried out under the supervision of **Mr. Nikesh Garg.** The work contained in this project has not been submitted to any other university or institute for the award of any degree or diploma.

Siddharth Vashisht

B.Tech. (Information Technology)

01911503117

<u>ACKNOWLEDGMENT</u>

I take this opportunity to express my heartfelt gratitude to all the people who have extended their assistance. I am greatly thankful to them for their guidance and support throughout the project and for sparing their valuable time for me.

I earnestly express my gratitude to **Mr. Nikesh Garg** and **Mr. Vipin Dixit** and all the staff of the office for their invaluable guidance, keen interest, cooperation, inspiration and moral support throughout my internship tenure.

I am grateful that the opportunity of working at Nokia Corporation was provided to me. I would like to thank my teachers and staff at my college for granting me the opportunity to do so.

Siddharth Vashisht

B.Tech. (Information Technology)

01911503117

Table of Contents

Introduction

1.1 About the Organisation: -

Nokia Corporation is a Finnish multinational telecommunication, information technology, and consumer electronics company, founded in 1865. Nokia's headquarters are in Espoo, in the greater Helsinki metropolitan area. In 2018, Nokia employed approximately 103,000 people across over 100 countries, did business in more than 130 countries, and reported annual revenues of around €23 billion. Nokia is a public limited company listed on the Helsinki Stock Exchange and New York Stock Exchange. It is the world's 415th-largest company measured by 2016 revenues according to the *Fortune Global 500,* having peaked at 85th place in 2009. It is a component of the Euro Stoxx 50 stock market index.

The company is viewed with national pride by Finns, as its mobile phone business made it by far the largest worldwide company and brand from Finland. At its peak in 2000, during the telecoms bubble, Nokia alone accounted for 4% of the country's GDP, 21% of total exports, and 70% of the Helsinki Stock Exchange market capital.

Nokia Networks is Nokia Corporation's largest division. It is a multinational data networking and telecommunications equipment company headquartered in Espoo, Finland, and is the world's third-largest telecoms equipment manufacturer, measured by 2017 revenues (after Huawei and Cisco). In USA it competes with Ericsson on building 5G networks for operators, while Huawei Technologies and ZTE Corporation were effectively banned.

It has operations in around 150 countries.

Nokia Networks provides wireless and fixed network infrastructure, communications and networks service platforms and professional services to operators and service providers. It focuses on GSM, EDGE, 3G/W-CDMA, LTE and WiMAX radio access networks, supporting core networks with increasing IP and multiaccess capabilities and services.

LEADERSHIP TEAM (Board of Directors)

Mr. Rajeev Suri (President and Chief Executive Officer)

Mr. Kristian Pullola (Chief Financial Officer)

Mr. Joerg Erlemeier (Chief Operating Officer)

Mr. Barry French (Chief Marketing Officer)

Mr. Bhaskar Gorti (President of Nokia Software)

1.2   Problem Definition: -

Device Configuration

Nokia largely deals in providing routers and telecommunications equipment, these equipment actions are defined by writing its configuration files. Configuration file define where the router operating system is present, Internet protocols and all the other paths. There is always a large demand for creation of these files and it takes a lot of time if it is manually made. Work load can be decreased by using Python to make these files and there is less chances of errors made if Python is used.

Nokia Product Line

A **router** is a networking device that forwards data packets between computer networks. Routers perform the traffic directing functions on the Internet. Data sent through the internet, such as a web page or email, is in the form of data packets. A packet is typically forwarded from one router to another router through the networks that constitute an internetwork (e.g. the Internet) until it reaches its destination node.
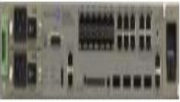
2.1 Alcatel-Lucent 7750 Service Router: -

7750 SR (Service Router) Portfolio Overview

	7750 SR-12e	7750 SR-12	7750 SR-7	7750 SR-a8	7750 SR-a4	7750 SR-c12/c4
Height (RU)	21 (½ Rack)	14 (1/3 Rack)	8	7	5	5/3
System Capacity (HD)	9.6 Tb/s	4 Tb/s	2 Tb/s	400G	200G	90 Gb/s
Slot Capacity (FD)	400Gb/s (redundant)	200Gb/s	200Gb/s	100G	100G	9.4Gb/s/4Gb/s
Total Slots	12	12	7	8	4	3 (6 MDA /12 CMA)/
Media Module Slots	9	10	5	8	4	4 CMA 2 Int. 10 GE

2.1 Alcatel-Lucent 7210 Service Access Switch: -

7210 SAS (Service Access Switch) Portfolio

	7210 SAS-D	7210 SAS-E	7210 SAS-T	7210 SAS-M	7210 SAS-X	7210 SAS-R
Speed	Wires peed 20 Gb/s	Wire speed 48 Gb/s	Wirespeed 62 Gb/s	Wire speed 128 Gb/s	Wire speed 88 Gb/s	Wire speed 120Gb/s
Power	Power redundancy	Power redundancy	Power redundancy	Power redundancy	Power redundancy	Power , control & switch fabric
Ports	6 x GigE (SFP) ports +4 x 10/100/1000 BASE-TX	12 x GigE (SFP) ports +12 x 10/100/1000 BASE-TX	4 x 10G +12 X GE SFP +10 x 10/100/1000 Incl 4 x PoE	Fixed configuration 24 x GigE (SFP) Up to 4 x 10GigE (XFP) +expansion slot	24 x GigE (SFP) 2 x 10GigE (XFP)	6 interface module slots For 10G and GE interfaces
Temp	-40 to +65°C variant	-40 to +65°C variant	-40 to +65°C variant	-40 to +65°C variant	-5to +45°C	-5to +45°C
Expansion				4 X E1 expansion card		

How configuration file was made earlier?

The clients sent a mail attaching a pdf file to the company employee which contained all the details regarding the new node that needs to be added in the network. The pdf contained details about: -

1)Host name

2)Location

3)System Ip

4)Loopback Ip

5)Ports

6)Interface Ip

7)ISIS instance

1.1.2. Activity [New Node - FBD_KRN_901_1AC_M_ASASMXPR020]

Basic Info	
Site OwnerCode	Mobility
NE Label	FBD_KRN_901_1AC_M_ASASMXPR020
NE Model	7210 SAS-Mxp
NetworkGroup	ANG
Is Existing Location	No
Vendor	ALCATEL
City Code	FBD
Location	FBD/KRN/901
Remarks	BCL -ID GIS Code M6 Code Address 5548 FK5548 FBD/KRN/901 (Khasra No.57/25/2/1), Plot No 5, Krishna Nagar Inds. Area, Sector 25, Faridabad
Descriprion	NO
Ring ID	R020
Ring Capacity	TenGigabitEthernet
Ring Tier Type	Access
ISIS ID	24
System IP	172.29.157.241
LoopBack IP	172.29.158.241

NE Label	Model	Vendor	Location	City Code	Holder Label	Model Equipment Type	SFP Type	Action
FBD_KRN_901_1AC_M_ASASMXPR020	7210 SAS-Mxp	ALCATEL	FBD/KRN/901	FBD	7210 SAS-Mxp-Shelf/Slot 1	card_iom_sas		FBD_KRN_901_1AC_M_ASASMXPR020
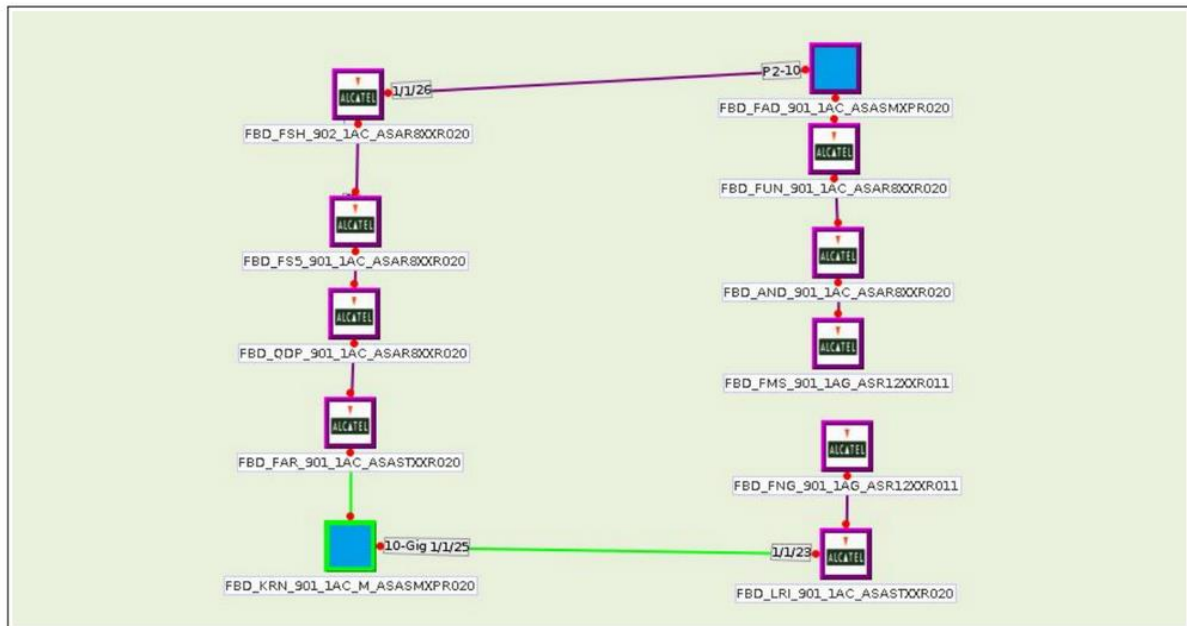
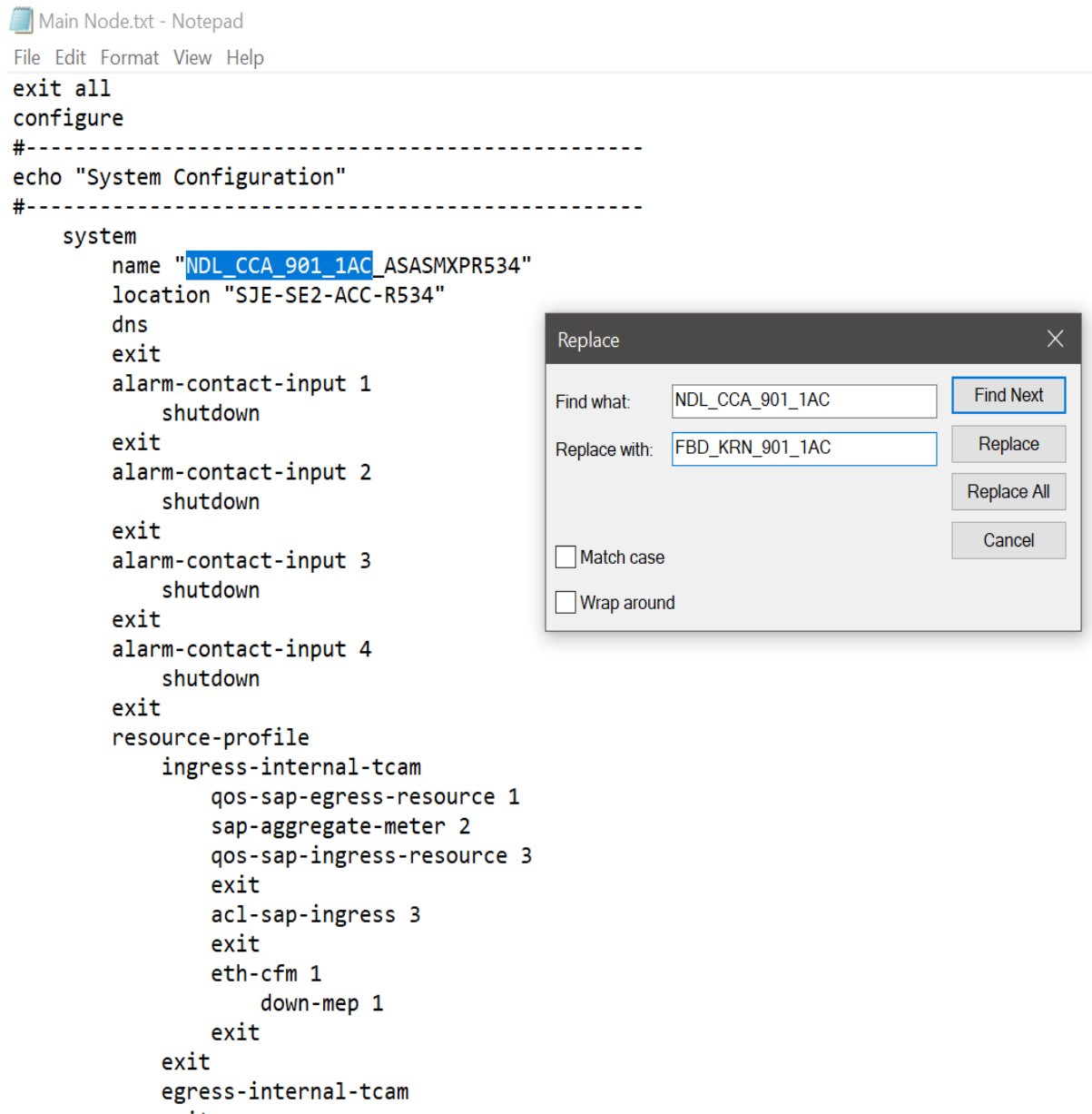**Existing Plan: -**
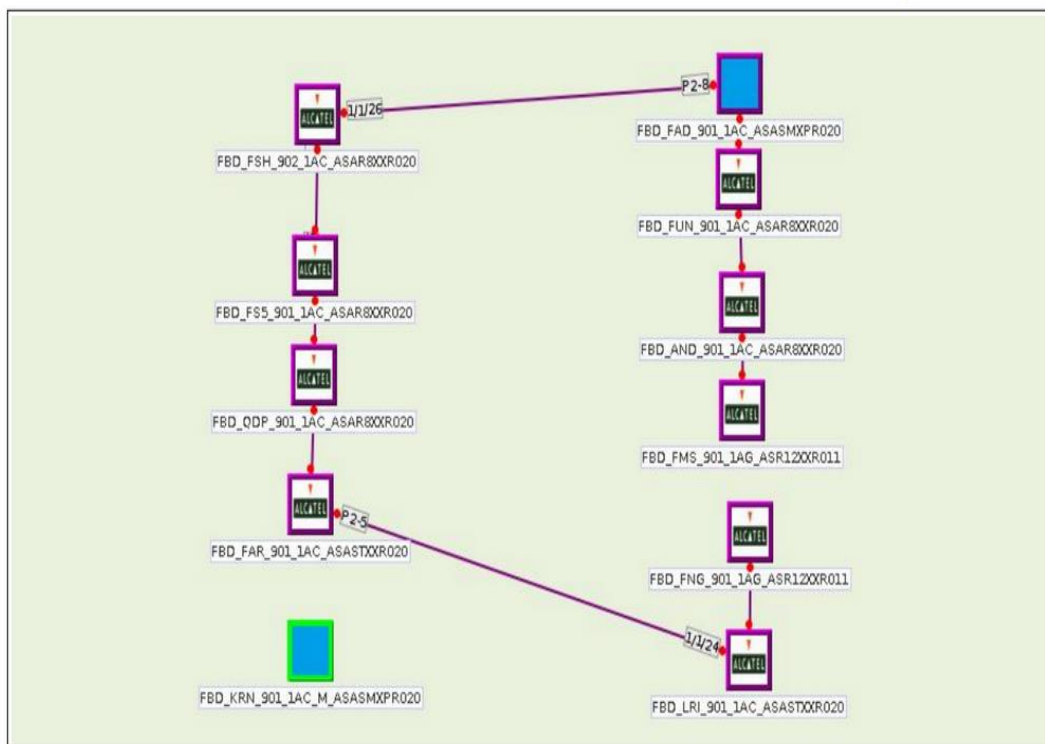
After the pdf file was received by the employee, he/she opens the template configuration file which was given to them by Nokia itself.

```
Main Node.txt - Notepad
File  Edit  Format  View  Help
exit all
configure
#-----------------------------------------------
echo "System Configuration"
#-----------------------------------------------
    system
        name "NDL_CCA_901_1AC_ASASMXPR534"
        location "SJE-SE2-ACC-R534"
        dns
        exit
        alarm-contact-input 1
            shutdown
        exit
        alarm-contact-input 2
            shutdown
        exit
        alarm-contact-input 3
            shutdown
        exit
        alarm-contact-input 4
            shutdown
        exit
        resource-profile
            ingress-internal-tcam
                qos-sap-egress-resource 1
                sap-aggregate-meter 2
                qos-sap-ingress-resource 3
                exit
                acl-sap-ingress 3
                exit
                eth-cfm 1
                    down-mep 1
                exit
            exit
            egress-internal-tcam
            ....
```

The method that was adopted by the employees to make changes into the pre-existing template as per the client needs was very time consuming.

Time taken to make one configuration file: - 10 minutes

Configuration after change is implemented manually: -

```
Main Node.txt - Notepad
File  Edit  Format  View  Help
exit all
configure
#-----------------------------------------------------
echo "System Configuration"
#-----------------------------------------------------
    system
        name "FBD_KRN_901_1AC_ASASMXPR534"
        location "SJE-SE2-ACC-R534"
        dns
        exit
        alarm-contact-input 1
            shutdown
        exit
        alarm-contact-input 2
            shutdown
        exit
        alarm-contact-input 3
            shutdown
        exit
        alarm-contact-input 4
            shutdown
        exit
        resource-profile
            ingress-internal-tcam
                qos-sap-egress-resource 1
                sap-aggregate-meter 2
                qos-sap-ingress-resource 3
                exit
                acl-sap-ingress 3
                exit
                eth-cfm 1
                    down-mep 1
                exit
            exit
            egress-internal-tcam
```

This method was very time consuming and only one change could be made at a time. During hectic office hours the work load on employees used to increase to make these files. Also chances of error is high.

Python Implementation

To solve all the above problems faced by employee python code was implemented. By using python all the changes can be made at the same time thus removing all the intermediate steps involved and significantly decreasing the processing time. Chances of errors is also decreased.

Python (Programming Language)

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

File Handling: -

Python too supports file handling and allows users to handle files i.e., to read and write files, along with many other file handling options, to operate on files. The concept of file handling has stretched over various other languages, but the implementation is either complicated or lengthy, but alike other concepts of Python, this concept here is also easy and short. Python treats file differently as text or binary and this is important. Each line of code includes a sequence of characters and they form text file. Each line of a file is terminated with a special character, called the EOL or End of Line characters like comma {,} or newline character. It ends the current line and tells the interpreter a new one has begun.

F1=open ("file name"," mode")

F1. close ()

Here modes can be r, w, a, r+, w+, a+ and many more.

Built-in functions used: -

1. List(): -
   An object that could be a sequence (string, tuples) or collection (set, dictionary) or iterator object. The list() constructor returns a mutable sequence list of elements.

   vowelString = 'aeiou'
   print(list(vowelString))

   >>['a','e','i','o','u']

2. Remove(): -
   The remove() method takes a single element as an argument and removes it from the list. If the **element**(argument) passed to the remove() method doesn't exist, **valueError** exception is thrown.

```
animal = ['cat', 'dog', 'rabbit', 'guinea pig']
animal.remove('rabbit')

 >>['cat', 'dog', 'guinea pig']
```

3. Join(): -

   The join() method is a string method and returns a string in which the elements of sequence have been joined by str separator.
   The join() method takes iterable – objects capable of returning its members one at a time. Some examples are **List, Tuple, String, Dictionary and Set.**

   ```
   list1 = ['1','2','3','4']
   "-".join(list1)

   >>1-2-3-4
   ```

Auto Py to Exe: -

This project allows you to convert python scripts to executables with a simple interface. The interface uses chromes app mode and lists all possible flags for pyinstaller. The whole idea seems automatic as it cleans up after itself.

This application displays a simple interface that allows you to convert py to exe easily. By just selecting the file, if you want it to convert to onefile and if you want a console, you will only need to press convert and wait for the script to do the rest.

Auto Py to Exe

Script Location

```
Path to file
```
Browse

Onefile

One Directory | One File

Console Window

Console Based | Window Based (hide the console)

☑ Icon

☑ Additional Files

☑ Advanced

Current Command

```
pyinstaller -y ""
```

CONVERT .PY TO .EXE

Router Configuration using Python

Code: -

```python
#-------------------------------------------------------------------------------
texttosearch1="[$host]"
print("enter new host name")
texttoreplace1=input('>')

texttosearch2="[$loc]"
print("enter new location")
texttoreplace2=input('>')

texttosearch3="[$ip1]"
print("enter ntp ip address 1")
texttoreplace3=input('>')

texttosearch4="[$ip2]"
print("enter ntp ip address 2")
texttoreplace4=input('>')

texttosearch5="[$host1]"
print("Enter side host-1 description")
texttoreplace5=input(">")

texttosearch6="[$host2]"
print("Enter side host-2 description")
texttoreplace6=input(">")


texttosearch7="[$ip3]"
print("Enter loopback-1 ip")
texttoreplace7=input(">")


texttosearch8="[$ip4]"
print("Enter system ip")
texttoreplace8=input(">")


texttosearch14="[$mpls1]"
print("Enter MPLS-1 path")
texttoreplace14=input(">")
```

```python
texttosearch15="[$mpls2]"
print("Enter MPLS-2 path")
texttoreplace15=input(">")



texttosearch9="[$hhost]"
texttosearch10="[$hhost1]"
texttosearch11="[$hhost2]"
texttosearch12="[$ip31]"
texttosearch13="[$ip41]"


#------------------------------------------------------------------------

A=list(texttoreplace1)
B=list(texttoreplace5)
C=list(texttoreplace6)
for i in range(0,2):
    A.remove('_')
    B.remove('_')
    C.remove('_')
A="".join(A)
B="".join(B)
C="".join(C)



#------------------------------------------------------------------------
M=list(texttoreplace7)
for i in range(0,len(M)-1):
    if M[i]=='/':
        q=i
b=[]
for d in range(0,q):
    b=b+[M[d]]
MM="".join(b)
#------------------------------------------------------------------------
```

```python
N=list(texttoreplace8)
for z in range(0,len(N)-1):
    if N[z]=='/':
        flag=z
V=[]
for d in range(0,flag):
    V=V+[N[d]]
NN="".join(V)


#-------------------------------------------------------------------------------



file=open("config1.txt","r+")
f1=open("newconfig.txt","w+")
for line in file.readlines():
    if ((texttosearch1 in line) or (texttosearch2 in line)or (texttosearch3 in line) or
        (texttosearch4 in line)or (texttosearch5 in line)or (texttosearch6 in line)or
        (texttosearch7 in line)or (texttosearch8 in line)or (texttosearch9 in line)or
        (texttosearch10 in line)or (texttosearch11 in line) or (texttosearch12 in line) or
        (texttosearch13 in line) or (texttosearch14 in line) or (texttosearch15 in line)):
        m=line.replace( texttosearch1, texttoreplace1 )
        m=m.replace( texttosearch5, texttoreplace5 )
        m=m.replace( texttosearch2, texttoreplace2 )
        m=m.replace( texttosearch3, texttoreplace3 )
        m=m.replace( texttosearch4, texttoreplace4 )
        m=m.replace( texttosearch6, texttoreplace6 )
        m=m.replace( texttosearch7, texttoreplace7 )
        m=m.replace( texttosearch8, texttoreplace8 )
        m=m.replace( texttosearch9, A )
        m=m.replace( texttosearch10, B )
        m=m.replace( texttosearch11, C )
        m=m.replace( texttosearch12, MM)
        m=m.replace( texttosearch13, NN)
        m=m.replace( texttosearch14, texttoreplace14)
        m=m.replace( texttosearch15, texttoreplace15)
        f1.write(m)
    else:
        f1.write(line)


#-------------------------------------------------------------------------------
input("press any key to exit")
file.close()
f1.close()
```

OUTPUT: -

```
C:\Users\Siddharth Vashisht\AppData\Local\Programs\Python\Python36\Scripts\output\nokia\nokia.exe

enter new host name
>FBD_KRN_901_1AC
enter new location
>FBD-KRN-901
enter ntp ip address 1
>192.168.0.5
enter ntp ip address 2
>192.168.8.200
Enter side host-1 description
>FBD_FAR_901_1AC
Enter side host-2 description
>FBD_LRI_901_1AC
Enter loopback-1 ip
>172.29.158.241/32
```

```
C:\Users\Siddharth Vashisht\AppData\Local\Programs\Python\Python36\Scripts\output\nokia\nokia.exe

enter new host name
>FBD_KRN_901_1AC
enter new location
>FBD-KRN-901
enter ntp ip address 1
>192.168.0.5
enter ntp ip address 2
>192.168.8.200
Enter side host-1 description
>FBD_FAR_901_1AC
Enter side host-2 description
>FBD_LRI_901_1AC
Enter loopback-1 ip
>172.29.158.241/32
Enter system ip
>172.29.157.241/32
Enter MPLS-1 path
>NCRKRN901ACC:NCRFBD901AGR
Enter MPLS-2 path
>NCRKRN901ACC:NCRKRN901AGR
press any key to exit
```

After implementation of above code new configuration file is created.

Time take to make one configuration file using python: -   3 minutes

Name	Date modified	Type	Size
_bz2.pyd	19-06-2019 14:52	Python Extension Mo...	93 KB
_hashlib.pyd	19-06-2019 14:52	Python Extension Mo...	1,417 KB
_lzma.pyd	19-06-2019 14:52	Python Extension Mo...	249 KB
_socket.pyd	19-06-2019 14:52	Python Extension Mo...	71 KB
_ssl.pyd	19-06-2019 14:52	Python Extension Mo...	1,704 KB
base_library.zip	15-07-2019 09:55	WinRAR ZIP archive	759 KB
config1.txt	03-07-2019 14:05	Text Document	32 KB
newconfig.txt	22-07-2019 22:07	Text Document	33 KB
nokia.exe	15-07-2019 09:55	Application	1,391 KB
nokia.exe.manifest	15-07-2019 09:55	MANIFEST File	2 KB
pyexpat.pyd	19-06-2019 14:52	Python Extension Mo...	184 KB
python36.dll	19-06-2019 14:52	Application extension	3,473 KB
select.pyd	19-06-2019 14:52	Python Extension Mo...	27 KB
unicodedata.pyd	19-06-2019 14:52	Python Extension Mo...	885 KB
VCRUNTIME140.dll	19-06-2019 14:52	Application extension	86 KB

New configuration file after python code is implemented: -

```
newconfig.txt - Notepad
File  Edit  Format  View  Help
exit all
configure
#-------------------------------------------------
echo "System Configuration"
#-------------------------------------------------
    system
        name "FBD_KRN_901_1AC_ASASMXPR534"
        location "FBD-KRN-901"
        dns
        exit
        alarm-contact-input 1
            shutdown
        exit
        alarm-contact-input 2
            shutdown
        exit
        alarm-contact-input 3
            shutdown
        exit
        alarm-contact-input 4
            shutdown
        exit
        resource-profile
            ingress-internal-tcam
                qos-sap-egress-resource 1
                sap-aggregate-meter 2
                qos-sap-ingress-resource 3
                exit
                acl-sap-ingress 3
                exit
                eth-cfm 1
                    down-mep 1
                exit
            exit
            egress-internal-tcam
```

CONCLUSION

This report contains details of my work at Nokia Corporation. Configuration file of SAS MXP was successfully made.

Configuration files require a lot of time to be made if done manually but with the help of python module the time required to do so significantly decreases.

Time taken to make one configuration file manually-   10 minutes

Time taken to make one configuration file using python-   3 minutes

The python module was made in such a way that in future the employees using it can make changes in the python script if the need arises.

Chances of errors is also high if manually changes are made thus py module was very helpful as errors were much lower as compared to before.

This project has given me great insight into the corporate world and the processes involved in it. It has created an interest in Software Development and Project Management for me. I hope to work on it in the near future as well.

<u>**References**</u>

1.Pdf's given by Nokia: -

  1.1 ScalableIP_NRS1.pdf

  1.2 Technical Workshop.pdf

2. https://www.nokia.com/networks/products/7750-service-router/

3. https://en.wikipedia.org/wiki/Nokia

4. https://docs.python.org/3/library/functions.html

5. https://pypi.org/project/auto-py-to-exe/

6. https://www.w3schools.com/python/python_file_handling.asp

7. https://en.wikipedia.org/wiki/Python_(programming_language)

8. https://www.nokia.com/about-us/what-we-do/group-leadership-team/