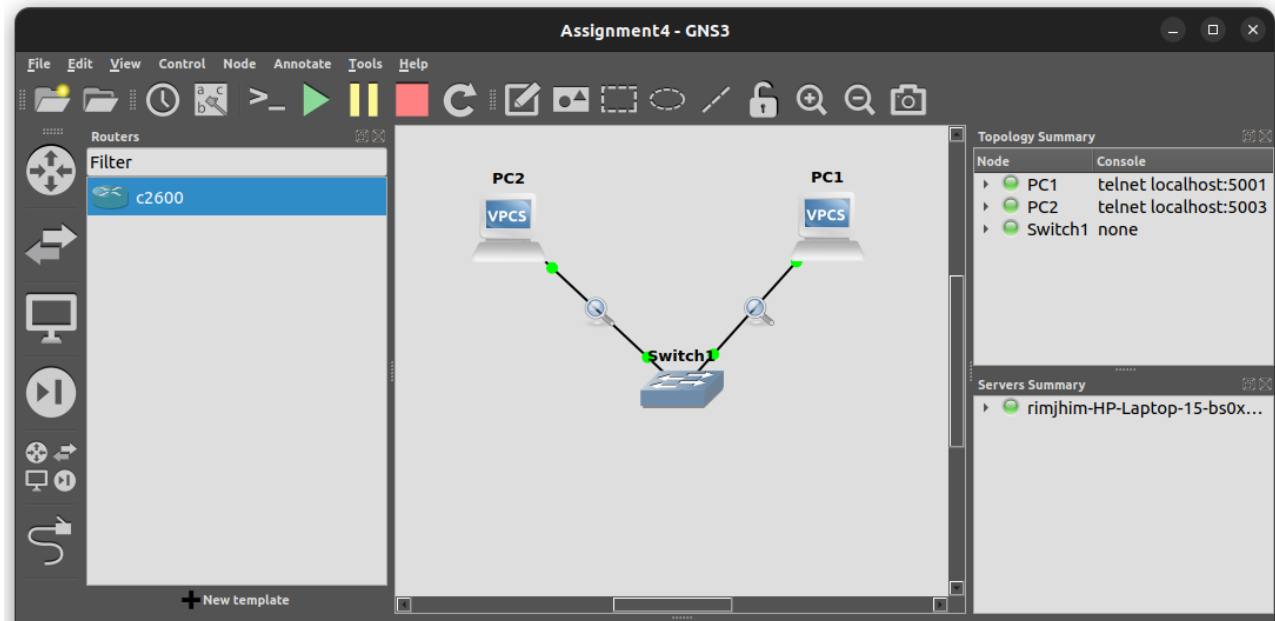


LAB: Assignment 5

Submitted by- Rimjhim Mittal (102103430- 2CO16)

Objective: To create simple **Network Topologies Using GNS**.

Q1. Demonstrate the working of **Address Resolution Protocol (ARP Protocol)** with using **Wireshark**.



```
PC1> ip 192.168.1.1 gateway 192.168.1.100
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.100
PC1>
```

```
PC2> ip 192.168.1.2 gateway 192.168.1.100
Checking for duplicate address...
PC2 : 192.168.1.2 255.255.255.0 gateway 192.168.1.100
PC2> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=1.042 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=0.977 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=0.472 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=0.425 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=0.277 ms
PC2>
```

Activities Terminal Feb 21 04:26 RingTopology - GNS3

File Edit View Control Node Annotate Tools Help

Filter: Capturing from - [PC2 Ethernet0 to Switch1 Ethernet0]

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl+>

No.	Time	Source	Destination	Protocol	Length	Info
4	0.001699	192.168.1.1	192.168.1.2	ICMP	98	Echo (ping) reply id=0
5	1.002295	192.168.1.2	192.168.1.1	ICMP	98	Echo (ping) request id=0
6	1.002810	192.168.1.1	192.168.1.2	ICMP	98	Echo (ping) reply id=0
7	2.003744	192.168.1.2	192.168.1.1	ICMP	98	Echo (ping) request id=0
8	2.004411	192.168.1.1	192.168.1.2	ICMP	98	Echo (ping) reply id=0
9	3.006124	192.168.1.2	192.168.1.1	ICMP	98	Echo (ping) request id=0
10	3.006915	192.168.1.1	192.168.1.2	ICMP	98	Echo (ping) reply id=0
11	4.008648	192.168.1.2	192.168.1.1	ICMP	98	Echo (ping) request id=0
12	4.009228	192.168.1.1	192.168.1.2	ICMP	98	Echo (ping) reply id=0

Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface -, id 0
Ethernet II, Src: Private 66:68:06 (00:50:79:66:68:06), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Address Resolution Protocol (request)

0000 ff ff ff ff ff 00 50 79 66 68 06 00 00 01 P yfh
0010 08 00 06 04 00 01 00 50 79 66 68 06 c0 a8 01 02 P yfh
0020 ff ff ff ff ff c0 a8 01 01 00 00 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Ready to load or capture
Copyright (c) 2006-2023 GNS3 Technologies.
Use Help -> GNS3 Doctor to detect common issues.
s>

PC2

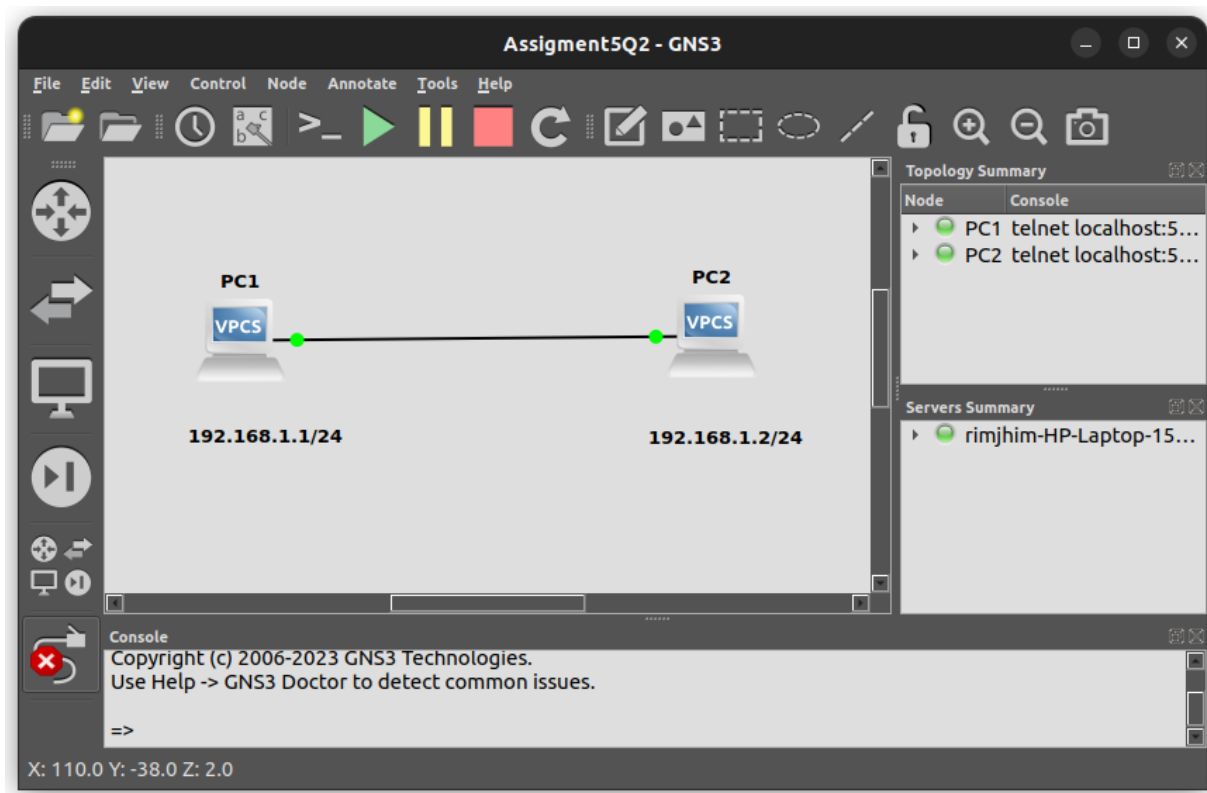
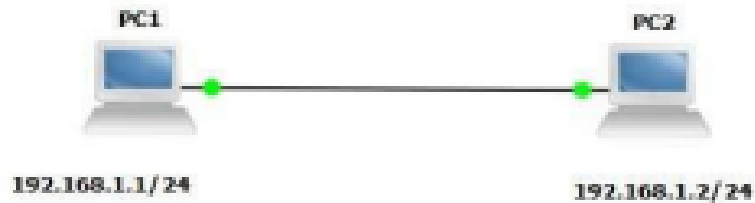
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=0.472 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=0.425 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=0.277 ms

PC2> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=1.107 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=0.908 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=1.312 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=1.296 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=1.129 ms

Packets: 12 - Displayed: 12 (100.0%) Pr

Q2. Connect one PC with another PC as per diagram below.

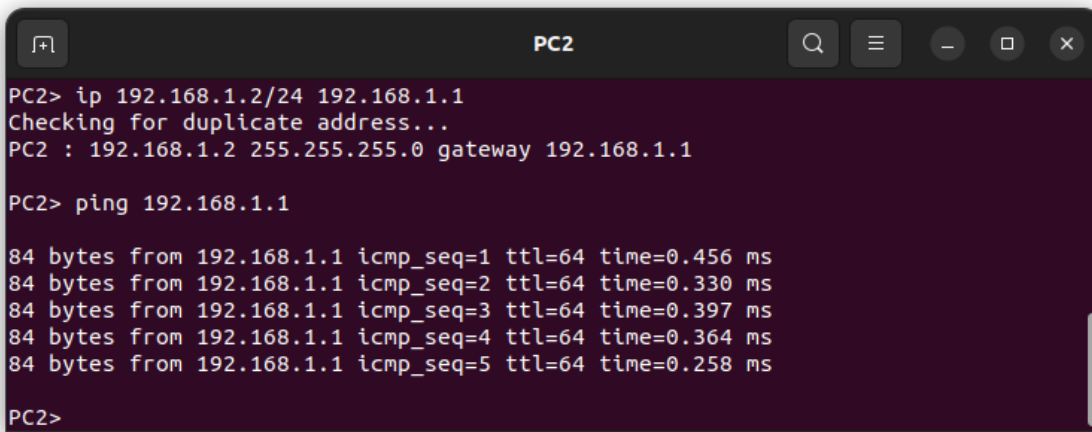


```
PC1> ip 192.168.1.1/24 192.168.1.2
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.2

PC1> ping 192.168.1.2

84 bytes from 192.168.1.2 icmp_seq=1 ttl=64 time=0.821 ms
84 bytes from 192.168.1.2 icmp_seq=2 ttl=64 time=0.648 ms
84 bytes from 192.168.1.2 icmp_seq=3 ttl=64 time=0.815 ms
84 bytes from 192.168.1.2 icmp_seq=4 ttl=64 time=0.271 ms
84 bytes from 192.168.1.2 icmp_seq=5 ttl=64 time=0.236 ms

PC1>
```



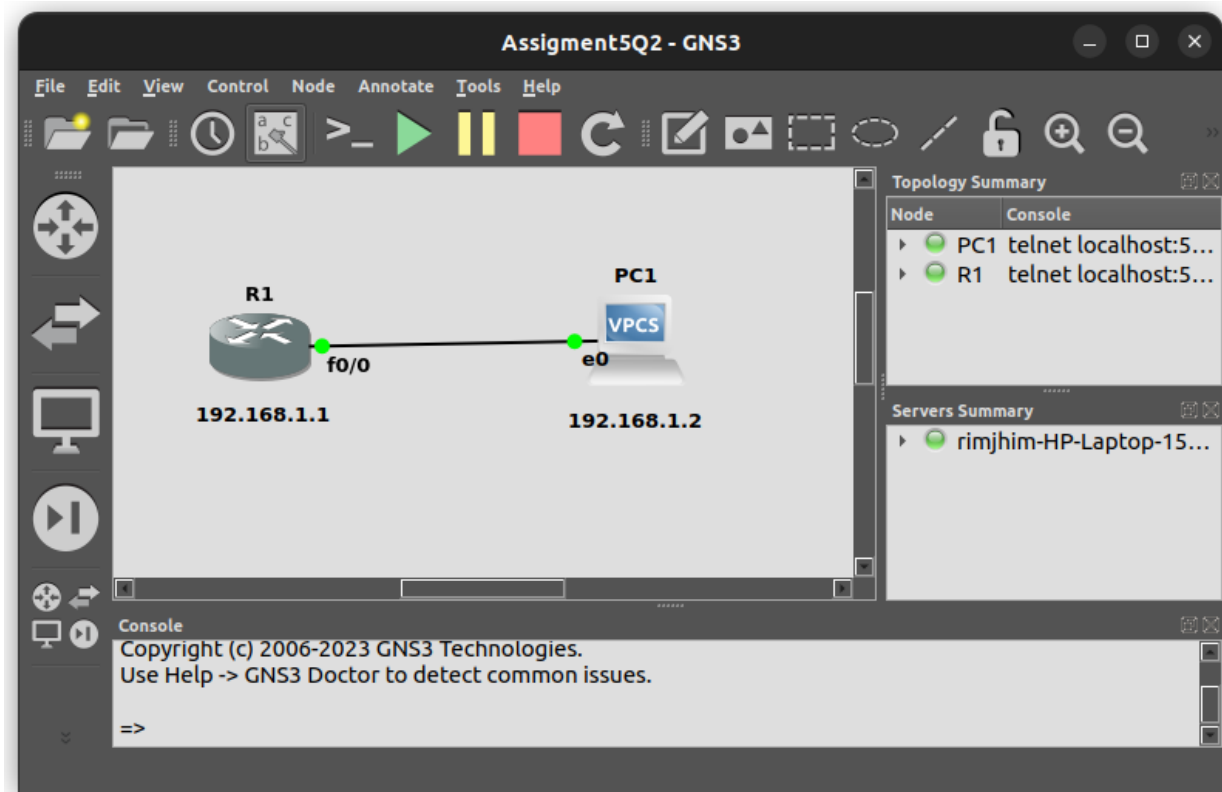
```
PC2> ip 192.168.1.2/24 192.168.1.1
Checking for duplicate address...
PC2 : 192.168.1.2 255.255.255.0 gateway 192.168.1.1

PC2> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=0.456 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=0.330 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=0.397 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=0.364 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=0.258 ms

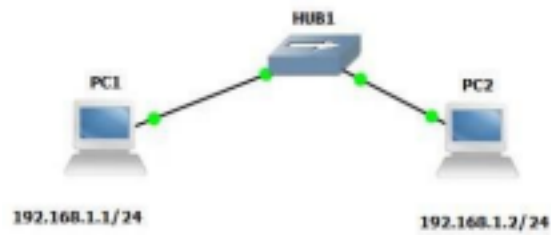
PC2>
```

Q3. Connect a PC with Router Ethernet port as shown below.



```
R1
R1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface f0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shut
R1(config-if)#
*Mar  1 00:01:43.667: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 00:01:44.667: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
R1(config-if)#^Z
R1#
*Mar  1 00:02:12.839: %SYS-5-CONFIG_I: Configured from console by console
R1#
```

Q4. Create a network of 2 PCs as below.



The screenshot displays a network simulation interface. At the top, a terminal window for PC1 shows the following commands and output:

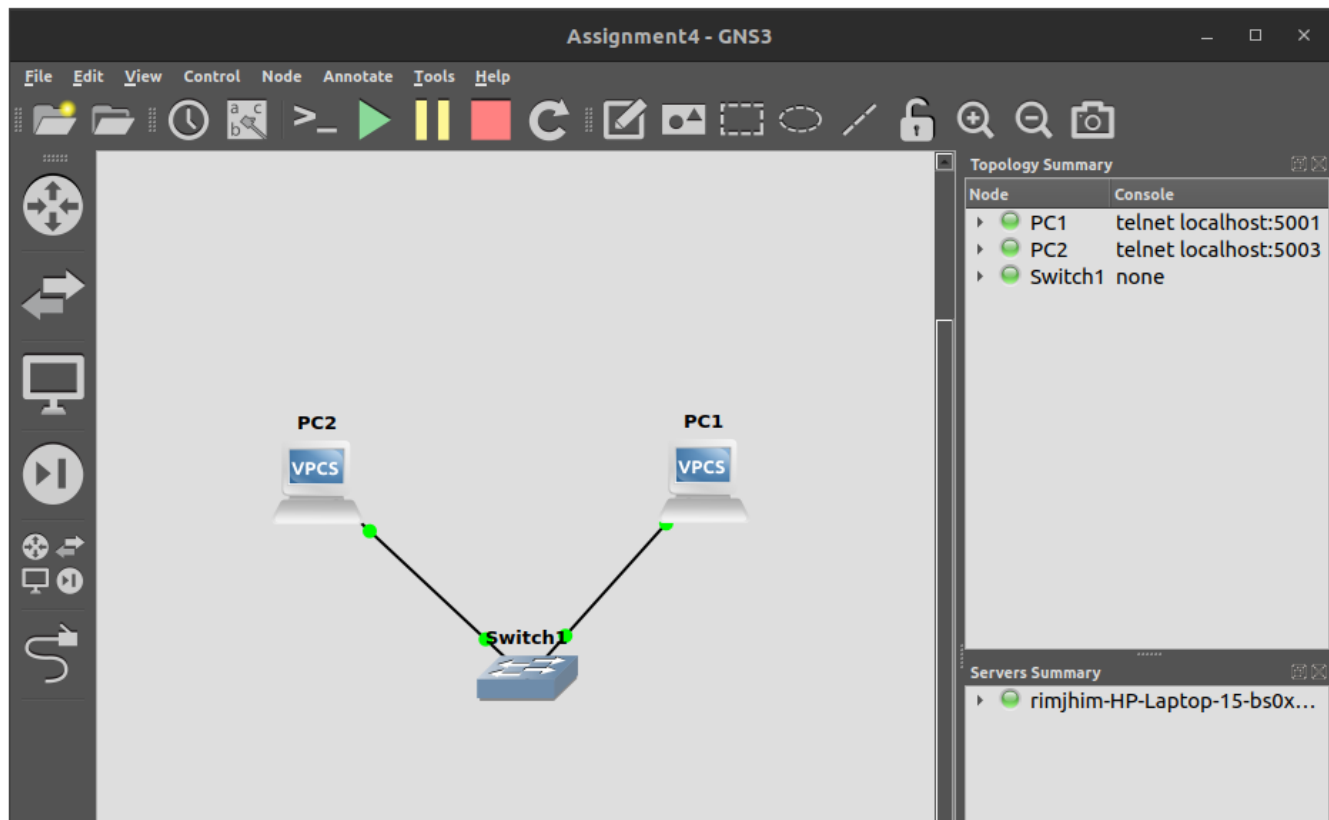
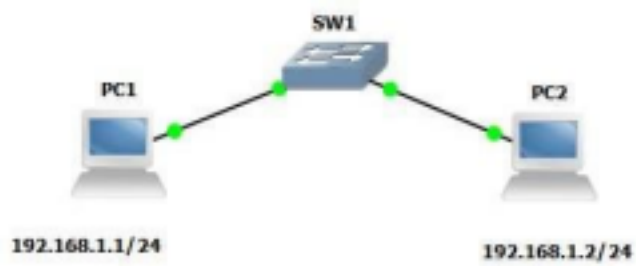
```
PC1> ip 192.168.1.1 gateway 192.168.1.100
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.100
PC1>
```

In the center, a network diagram shows a central Hub1 connected to two PCs. The PC on the left is labeled PC2 and VPCS, and the PC on the right is labeled PC1 and VPCS.

At the bottom, a terminal window for PC2 shows the following commands and output:

```
PC2> ip 192.168.1.2 gateway 192.168.1.100
Checking for duplicate address...
PC2 : 192.168.1.2 255.255.255.0 gateway 192.168.1.100
PC2>
```

Q5. Connect 2 PCs with a Switch below.




```
PC1

Press '?' to get help.

Executing the startup file

PC1>
PC1> ip 192.168.1.1/24 192.168.1.100
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.100

PC1> ip dns 192.168.1.100

PC1> show ip

NAME       : PC1[1]
IP/MASK    : 192.168.1.1/24
GATEWAY    : 192.168.1.100
DNS        : 192.168.1.100
MAC        : 00:50:79:66:68:00
LPORT     : 10004
RHOST:PORT : 127.0.0.1:10005
MTU        : 1500

PC1> 
```

```
PC2

Executing the startup file

PC2> ip 192.168.1.2/24 192.168.1.100
Checking for duplicate address...
PC2 : 192.168.1.2 255.255.255.0 gateway 192.168.1.100

PC2> ip dns 192.168.1.100

PC2> show ip

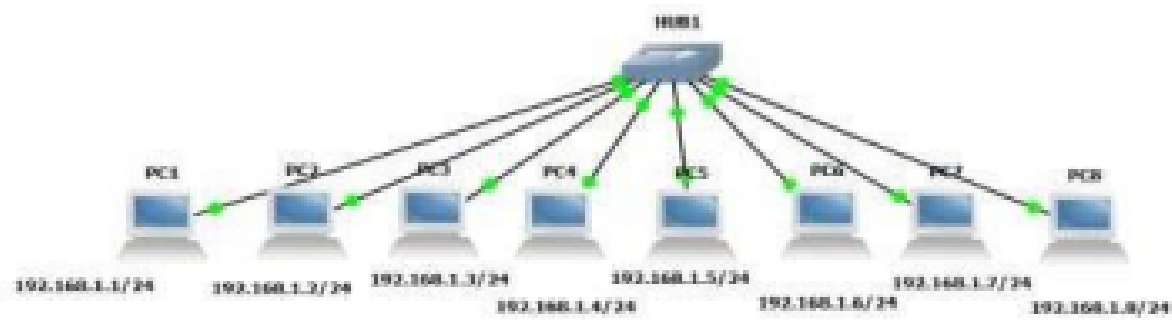
NAME       : PC2[1]
IP/MASK    : 192.168.1.2/24
GATEWAY    : 192.168.1.100
DNS        : 192.168.1.100
MAC        : 00:50:79:66:68:01
LPORT     : 10006
RHOST:PORT : 127.0.0.1:10007
MTU        : 1500

PC2> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=1.051 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=0.999 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=0.312 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=0.699 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=0.970 ms

PC2> 
```

Q6. Create BUS topologies as below.



(a)

The screenshot shows a GNS3 interface with a central Hub1 connected to eight PCs (PC1 to PC8). The PCs are labeled with their IP addresses: 192.168.1.1/24, 192.168.1.2/24, 192.168.1.3/24, 192.168.1.4/24, 192.168.1.5/24, 192.168.1.6/24, 192.168.1.7/24, and 192.168.1.8/24. The interface includes a menu bar, a toolbar, and a sidebar with various network components like ATM, c1700, c2600, c2691, Cloud, Ether..., Fram..., NAT, and VPCS. The main workspace displays the network diagram. On the right, there are terminal windows for each PC, showing the configuration of IP addresses and gateways. The console window at the bottom shows the error message: "Invalid remote server name".

```
PC1> ip 192.168.1.1 gateway 192.168.1.100
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.100
PC1> ip dns 192.168.1.100

PC2> ip 192.168.1.2 gateway 192.168.1.100
Checking for duplicate address...
PC2 : 192.168.1.2 255.255.255.0 gateway 192.168.1.100

PC3> ip 192.168.1.3 gateway 192.168.1.100
Checking for duplicate address...
PC3 : 192.168.1.3 255.255.255.0 gateway 192.168.1.100

PC4> ip 192.168.1.4 gateway 192.168.1.100
Checking for duplicate address...
PC4 : 192.168.1.4 255.255.255.0 gateway 192.168.1.100

PC5> ip 192.168.1.5 gateway 192.168.1.100
Checking for duplicate address...
PC5 : 192.168.1.5 255.255.255.0 gateway 192.168.1.100

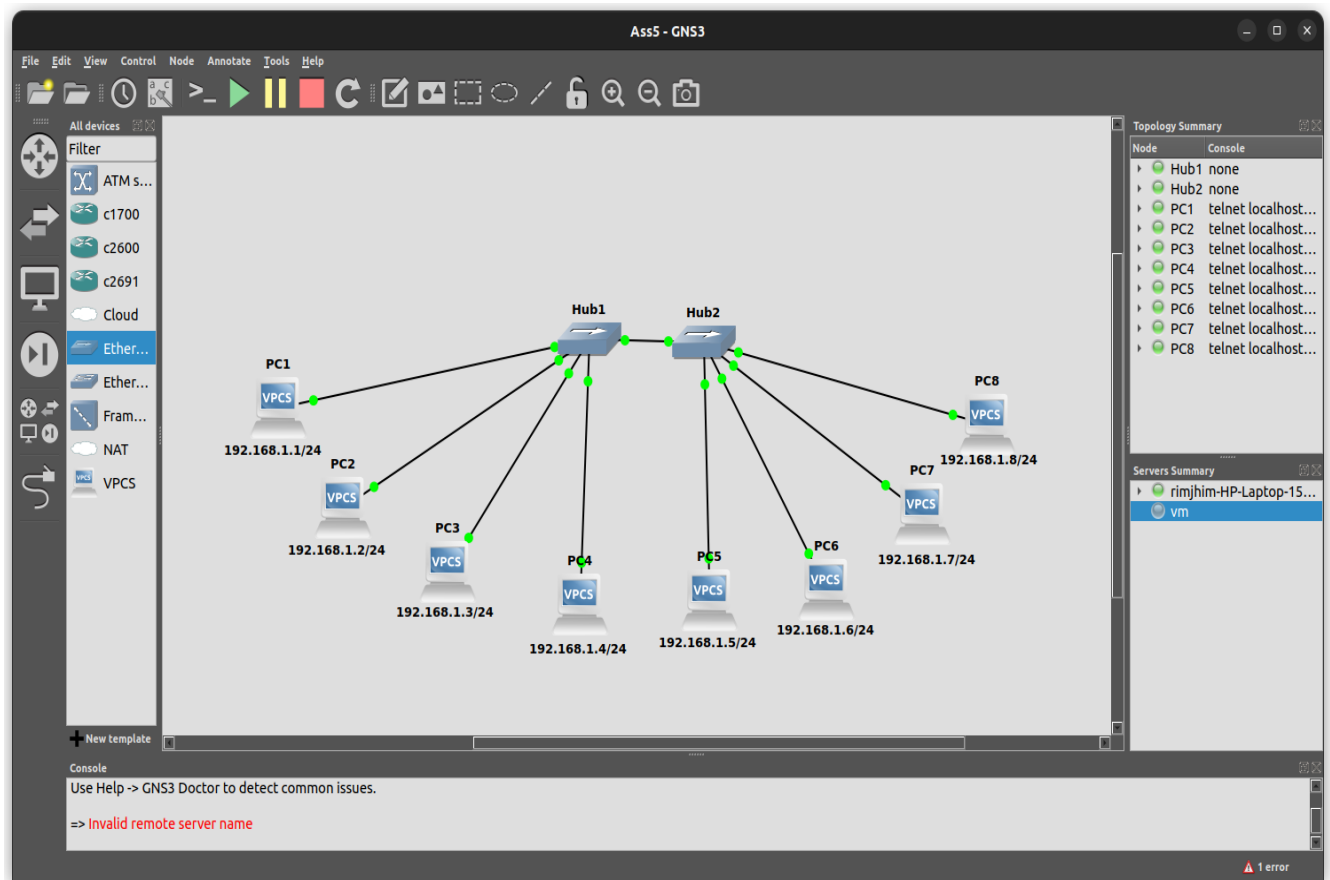
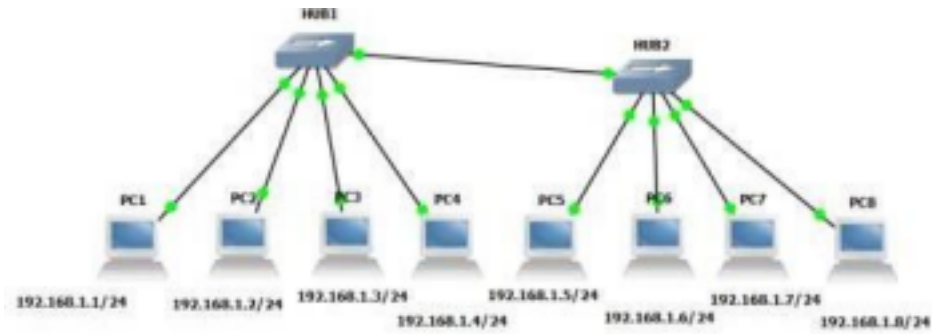
PC6> ip 192.168.1.6 gateway 192.168.1.100
Checking for duplicate address...
PC6 : 192.168.1.6 255.255.255.0 gateway 192.168.1.100

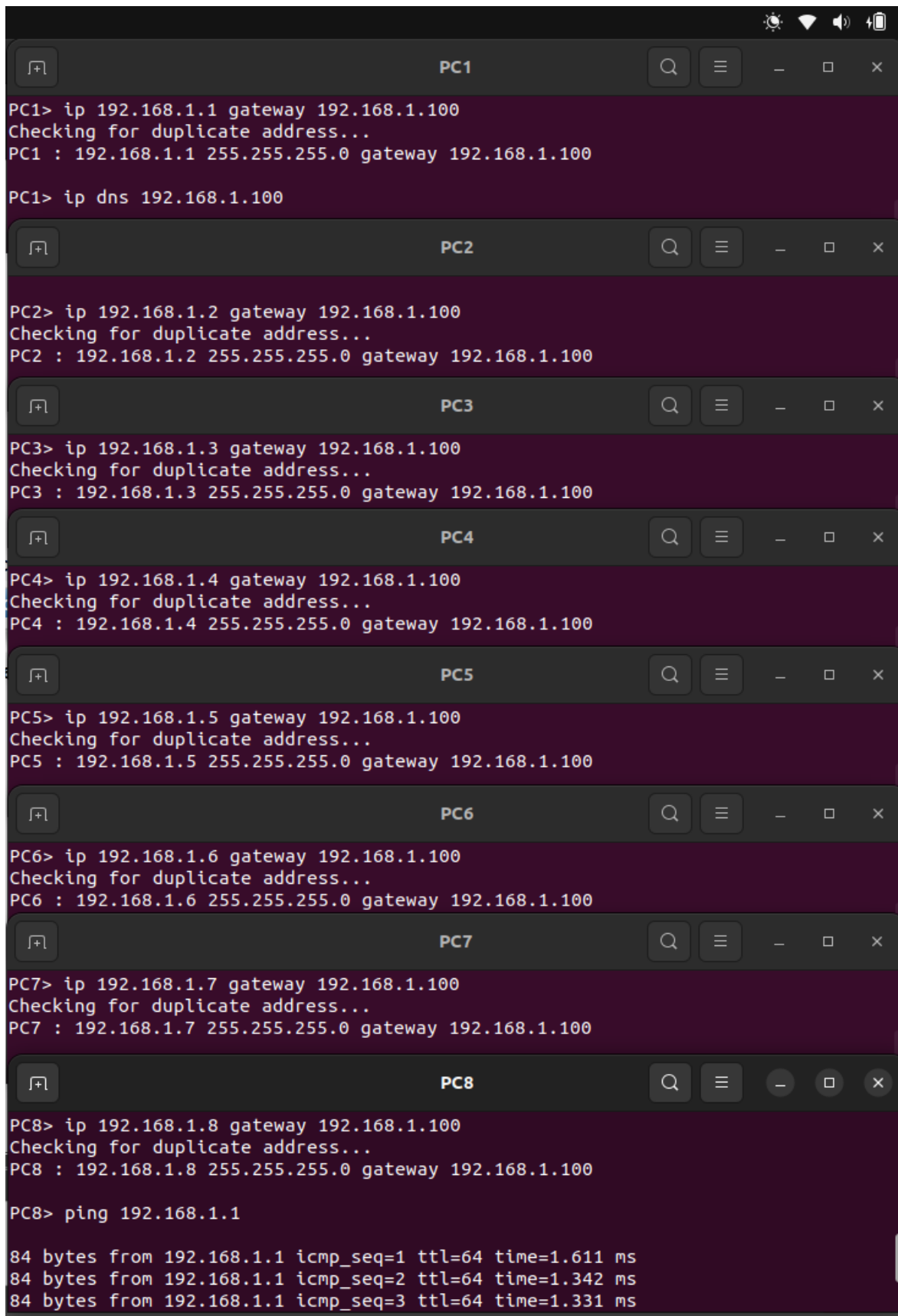
PC7> ip 192.168.1.7 gateway 192.168.1.100
Checking for duplicate address...
PC7 : 192.168.1.7 255.255.255.0 gateway 192.168.1.100

PC8> ip 192.168.1.8 gateway 192.168.1.100
Checking for duplicate address...
PC8 : 192.168.1.8 255.255.255.0 gateway 192.168.1.100

PC8> ping 192.168.1.1
84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=1.611 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=1.342 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=1.331 ms
```

(b)





The image displays a vertical stack of eight terminal windows, each representing a different PC in a network. Each window has a title bar with the PC number (PC1 to PC8) and standard window controls (search, menu, zoom, and close buttons). The terminal text shows the configuration of IP addresses and gateways for each PC, followed by a ping test from PC8 to PC1.

```
PC1> ip 192.168.1.1 gateway 192.168.1.100
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.100

PC1> ip dns 192.168.1.100

PC2> ip 192.168.1.2 gateway 192.168.1.100
Checking for duplicate address...
PC2 : 192.168.1.2 255.255.255.0 gateway 192.168.1.100

PC3> ip 192.168.1.3 gateway 192.168.1.100
Checking for duplicate address...
PC3 : 192.168.1.3 255.255.255.0 gateway 192.168.1.100

PC4> ip 192.168.1.4 gateway 192.168.1.100
Checking for duplicate address...
PC4 : 192.168.1.4 255.255.255.0 gateway 192.168.1.100

PC5> ip 192.168.1.5 gateway 192.168.1.100
Checking for duplicate address...
PC5 : 192.168.1.5 255.255.255.0 gateway 192.168.1.100

PC6> ip 192.168.1.6 gateway 192.168.1.100
Checking for duplicate address...
PC6 : 192.168.1.6 255.255.255.0 gateway 192.168.1.100

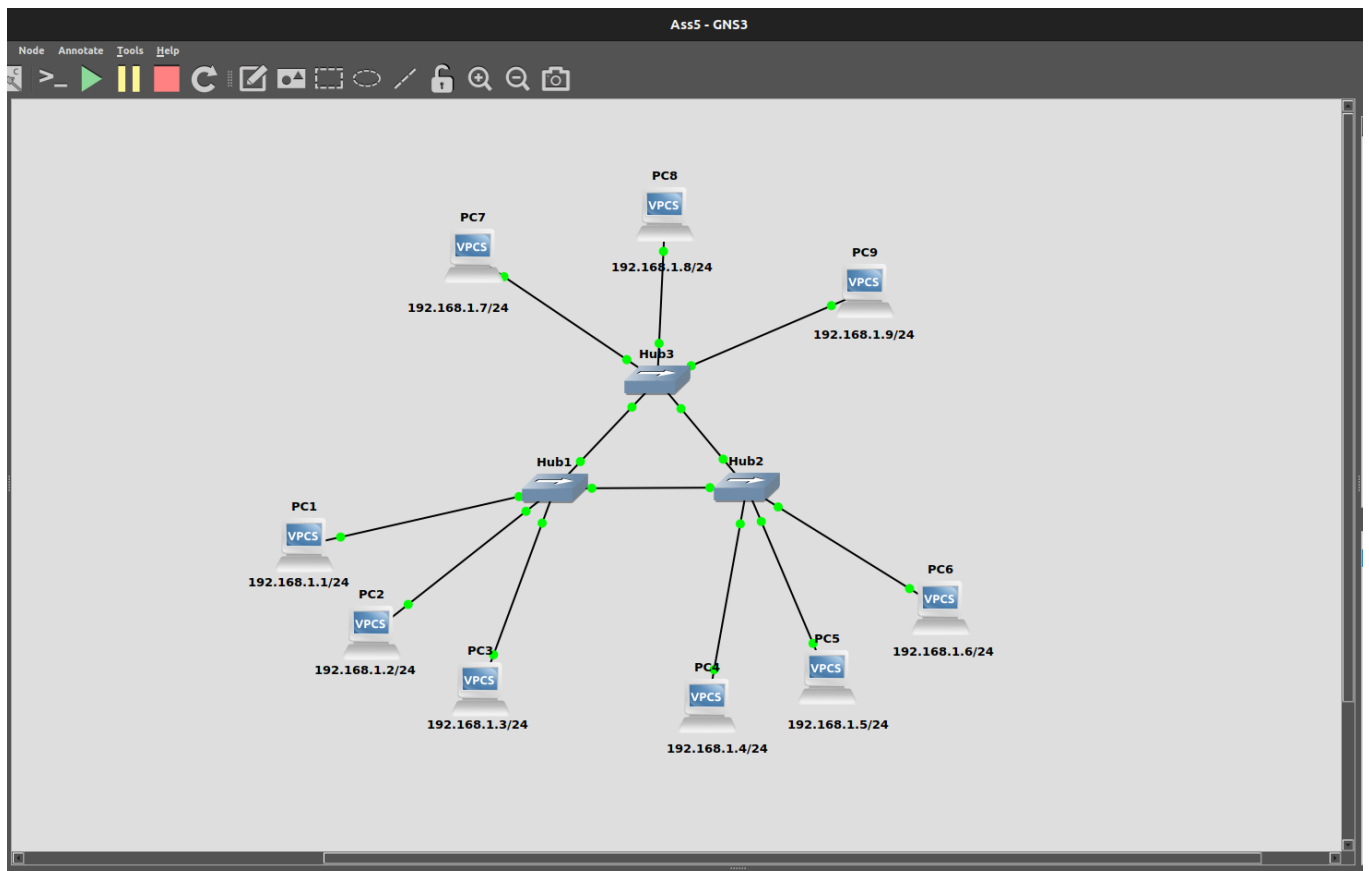
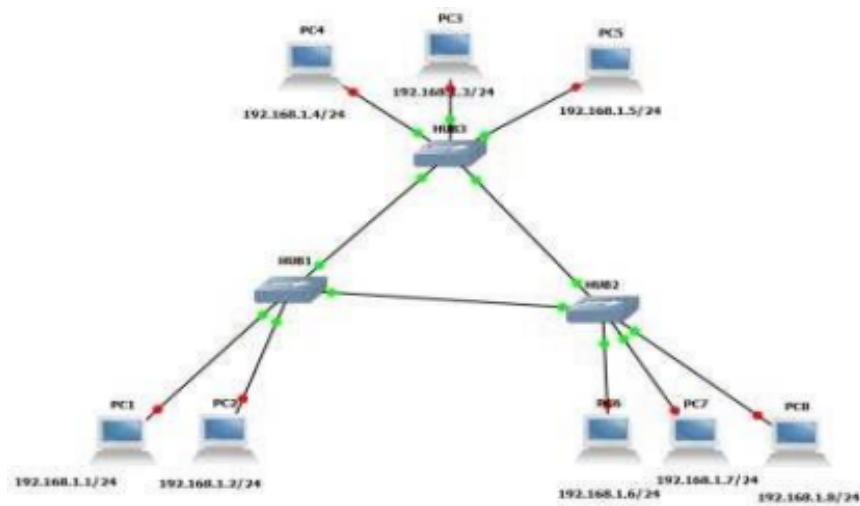
PC7> ip 192.168.1.7 gateway 192.168.1.100
Checking for duplicate address...
PC7 : 192.168.1.7 255.255.255.0 gateway 192.168.1.100

PC8> ip 192.168.1.8 gateway 192.168.1.100
Checking for duplicate address...
PC8 : 192.168.1.8 255.255.255.0 gateway 192.168.1.100

PC8> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=1.611 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=1.342 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=1.331 ms
```

Q7. Create the following Ring Topology.



```
PC1> ip 192.168.1.1 gateway 192.168.1.100
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.100

PC1> ip dns 192.168.1.100

PC2> ip 192.168.1.2 gateway 192.168.1.100
Checking for duplicate address...
PC2 : 192.168.1.2 255.255.255.0 gateway 192.168.1.100

PC3> ip 192.168.1.3 gateway 192.168.1.100
Checking for duplicate address...
PC3 : 192.168.1.3 255.255.255.0 gateway 192.168.1.100

PC4> ip 192.168.1.4 gateway 192.168.1.100
Checking for duplicate address...
PC4 : 192.168.1.4 255.255.255.0 gateway 192.168.1.100

PC5> ip 192.168.1.5 gateway 192.168.1.100
Checking for duplicate address...
PC5 : 192.168.1.5 255.255.255.0 gateway 192.168.1.100

PC6> ip 192.168.1.6 gateway 192.168.1.100
Checking for duplicate address...
PC6 : 192.168.1.6 255.255.255.0 gateway 192.168.1.100

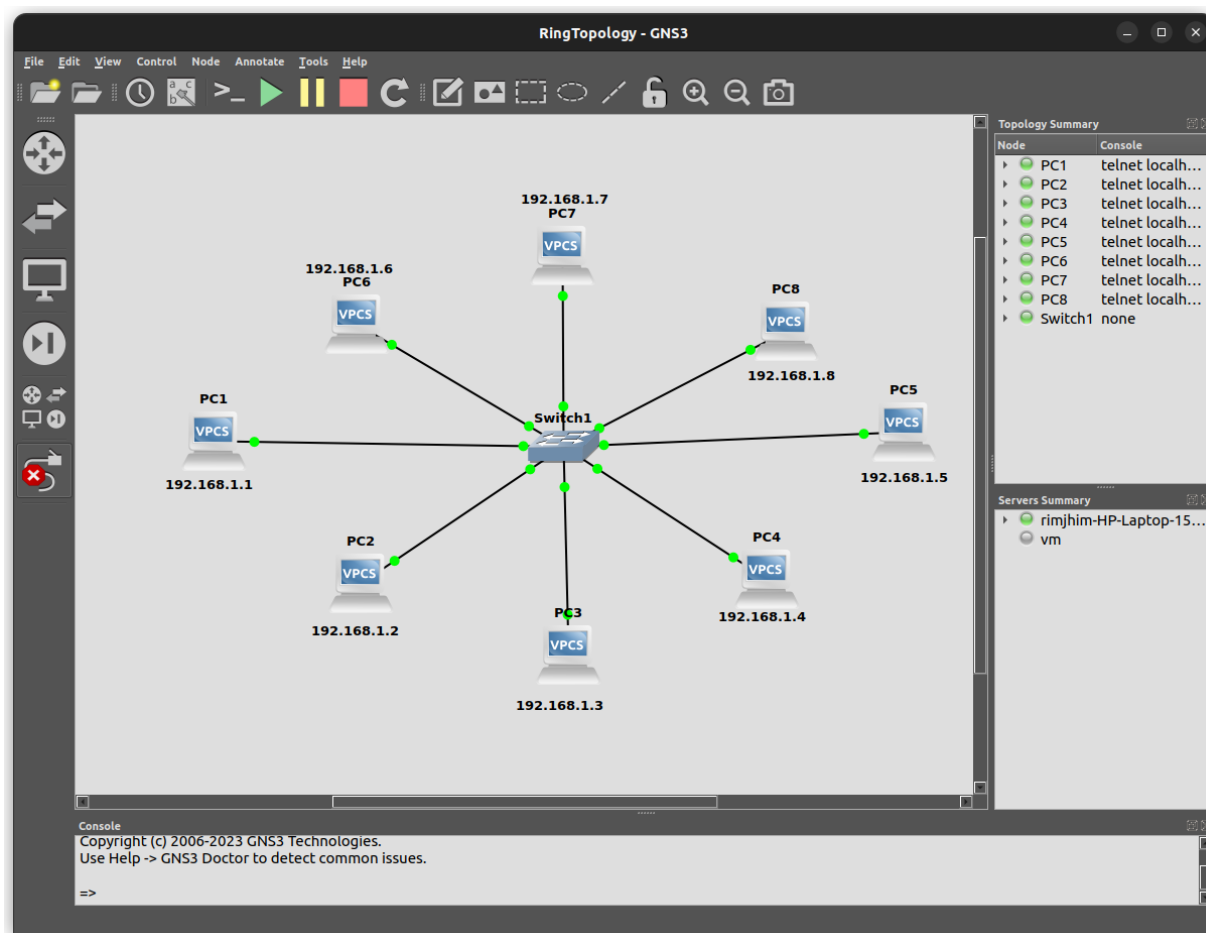
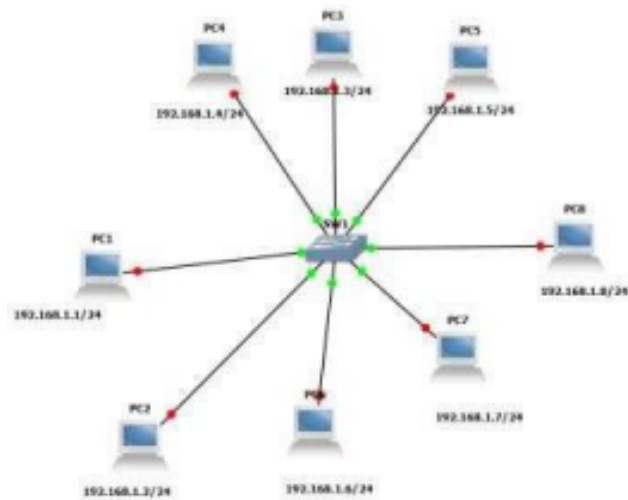
PC7> ip 192.168.1.7 gateway 192.168.1.100
Checking for duplicate address...
PC7 : 192.168.1.7 255.255.255.0 gateway 192.168.1.100

PC8> ip 192.168.1.8 gateway 192.168.1.100
Checking for duplicate address...
PC8 : 192.168.1.8 255.255.255.0 gateway 192.168.1.100

PC8> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=1.611 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=1.342 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=1.331 ms
```

Q8. Create the following Ring Topology.



```
PC1> ip 192.168.1.1 gateway 192.168.1.100
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.100

PC1> ip dns 192.168.1.100

PC2> ip 192.168.1.2 gateway 192.168.1.100
Checking for duplicate address...
PC2 : 192.168.1.2 255.255.255.0 gateway 192.168.1.100

PC3> ip 192.168.1.3 gateway 192.168.1.100
Checking for duplicate address...
PC3 : 192.168.1.3 255.255.255.0 gateway 192.168.1.100

PC4> ip 192.168.1.4 gateway 192.168.1.100
Checking for duplicate address...
PC4 : 192.168.1.4 255.255.255.0 gateway 192.168.1.100

PC5> ip 192.168.1.5 gateway 192.168.1.100
Checking for duplicate address...
PC5 : 192.168.1.5 255.255.255.0 gateway 192.168.1.100

PC6> ip 192.168.1.6 gateway 192.168.1.100
Checking for duplicate address...
PC6 : 192.168.1.6 255.255.255.0 gateway 192.168.1.100

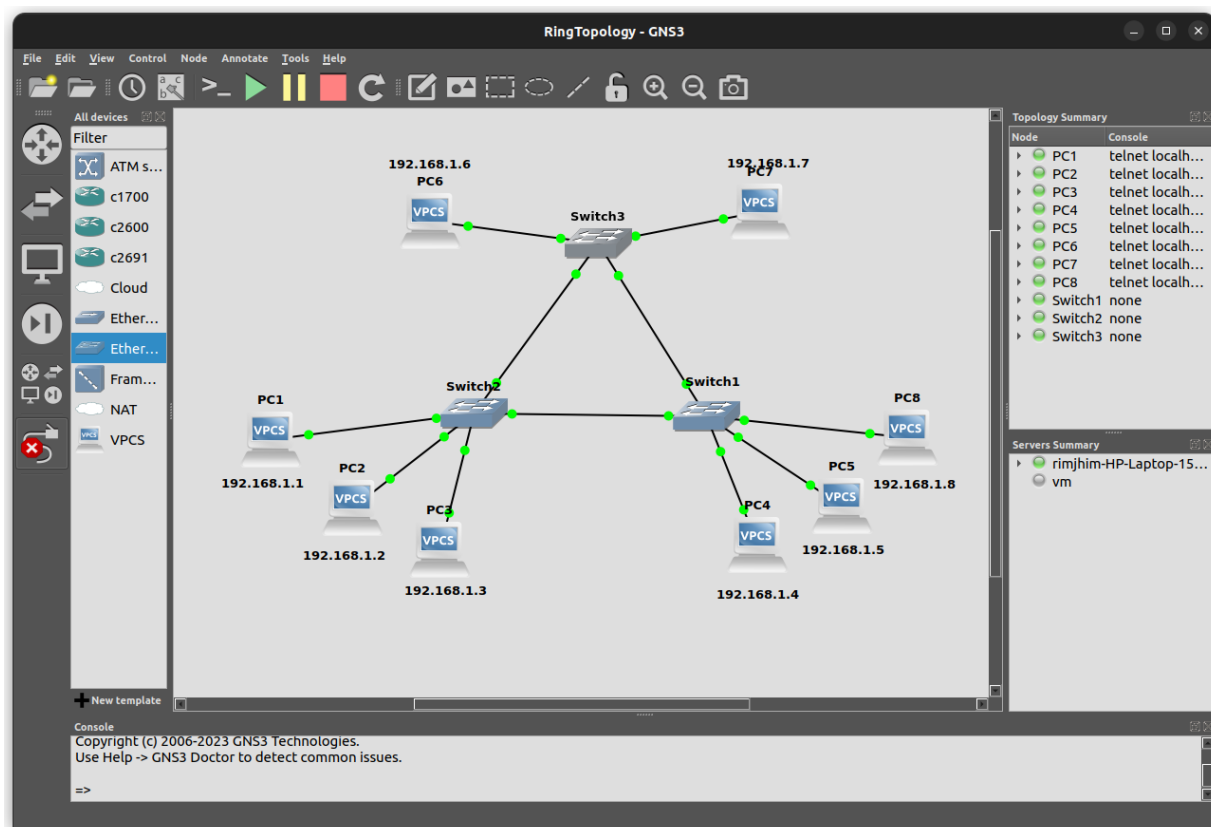
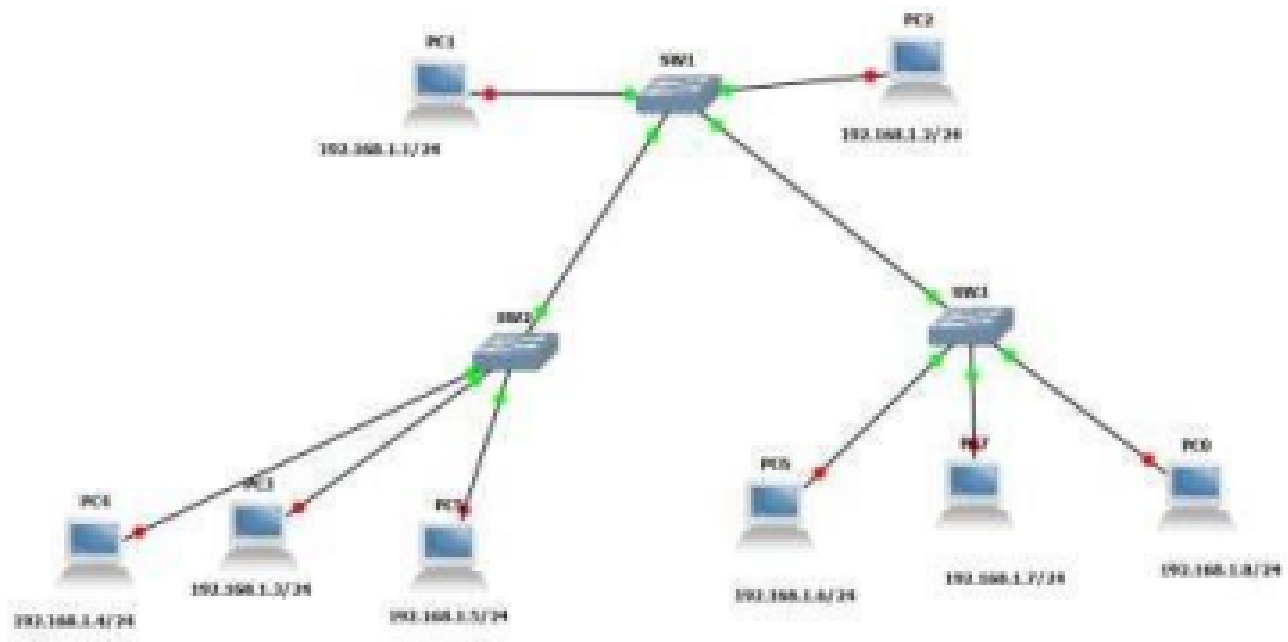
PC7> ip 192.168.1.7 gateway 192.168.1.100
Checking for duplicate address...
PC7 : 192.168.1.7 255.255.255.0 gateway 192.168.1.100

PC8> ip 192.168.1.8 gateway 192.168.1.100
Checking for duplicate address...
PC8 : 192.168.1.8 255.255.255.0 gateway 192.168.1.100

PC8> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=1.611 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=1.342 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=1.331 ms
```


Q9. Create the following Tree Topology.



```
PC1> ip 192.168.1.1 gateway 192.168.1.100
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0 gateway 192.168.1.100

PC1> ip dns 192.168.1.100

PC2> ip 192.168.1.2 gateway 192.168.1.100
Checking for duplicate address...
PC2 : 192.168.1.2 255.255.255.0 gateway 192.168.1.100

PC3> ip 192.168.1.3 gateway 192.168.1.100
Checking for duplicate address...
PC3 : 192.168.1.3 255.255.255.0 gateway 192.168.1.100

PC4> ip 192.168.1.4 gateway 192.168.1.100
Checking for duplicate address...
PC4 : 192.168.1.4 255.255.255.0 gateway 192.168.1.100

PC5> ip 192.168.1.5 gateway 192.168.1.100
Checking for duplicate address...
PC5 : 192.168.1.5 255.255.255.0 gateway 192.168.1.100

PC6> ip 192.168.1.6 gateway 192.168.1.100
Checking for duplicate address...
PC6 : 192.168.1.6 255.255.255.0 gateway 192.168.1.100

PC7> ip 192.168.1.7 gateway 192.168.1.100
Checking for duplicate address...
PC7 : 192.168.1.7 255.255.255.0 gateway 192.168.1.100

PC8> ip 192.168.1.8 gateway 192.168.1.100
Checking for duplicate address...
PC8 : 192.168.1.8 255.255.255.0 gateway 192.168.1.100

PC8> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=1.611 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=1.342 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=1.331 ms
```