WEEK 2 ASSIGNMENT 1: BUILDING A SWITCH AND A ROUTER NETWORK

RENE OLUOCH
CS-CNS09-25030

Contents

INTROUCTION

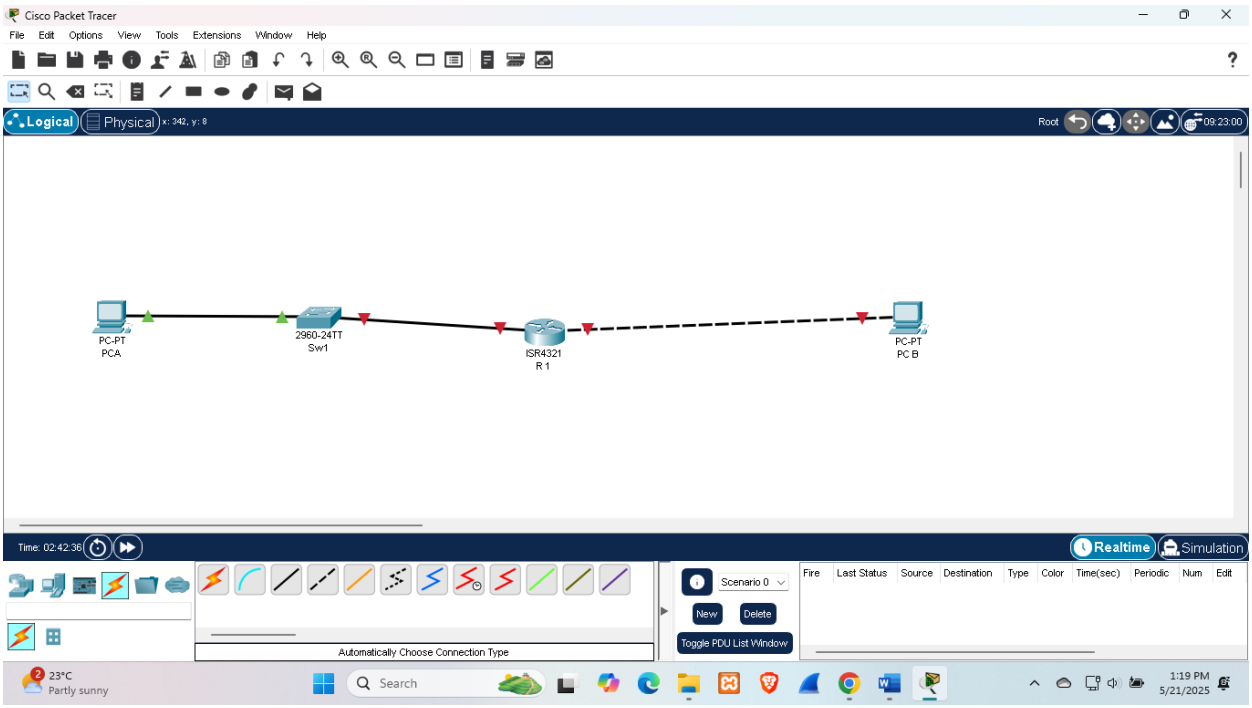
In this lab, I built a basic network using a router, a switch, and two PCs. The goal was to configure IP addresses, enable interfaces, and set up both IPv4 and IPv6 communication. I also practiced using Cisco IOS commands to apply security settings, verify device connectivity, and check routing and interface information. This hands-on activity helped me apply key networking concepts and test my understanding of device configuration without relying heavily on guided steps.

Set Up the Topology and Initialize Devices

Before setting up the topology I ensured I had the following devices,

- 1 Cisco Router (e.g., 4321)
- 1 Cisco Switch (e.g., Catalyst 2960)
- 2 PCs

To begin the lab, I set up the network topology by connecting the router, switch, and PCs using the correct Ethernet cables as shown in the diagram. The pings were not successful because the router and other interfaces were not yet configured.



Once everything was physically connected, I powered on all devices. I then initialized the router and switch by erasing any previous configurations and reloading them to ensure I was starting with a clean setup if at all there were previous set ups. Afterward, I configured static IP addresses, subnet masks, and

default gateways on both PC-A and PC-B based on the addressing table. I attempted to ping PC-B from PC-A to test initial connectivity, but the pings were not successful. This was because the Gigabit Ethernet interfaces on the router were still shut down by default and needed to be manually enabled before communication between devices could occur.

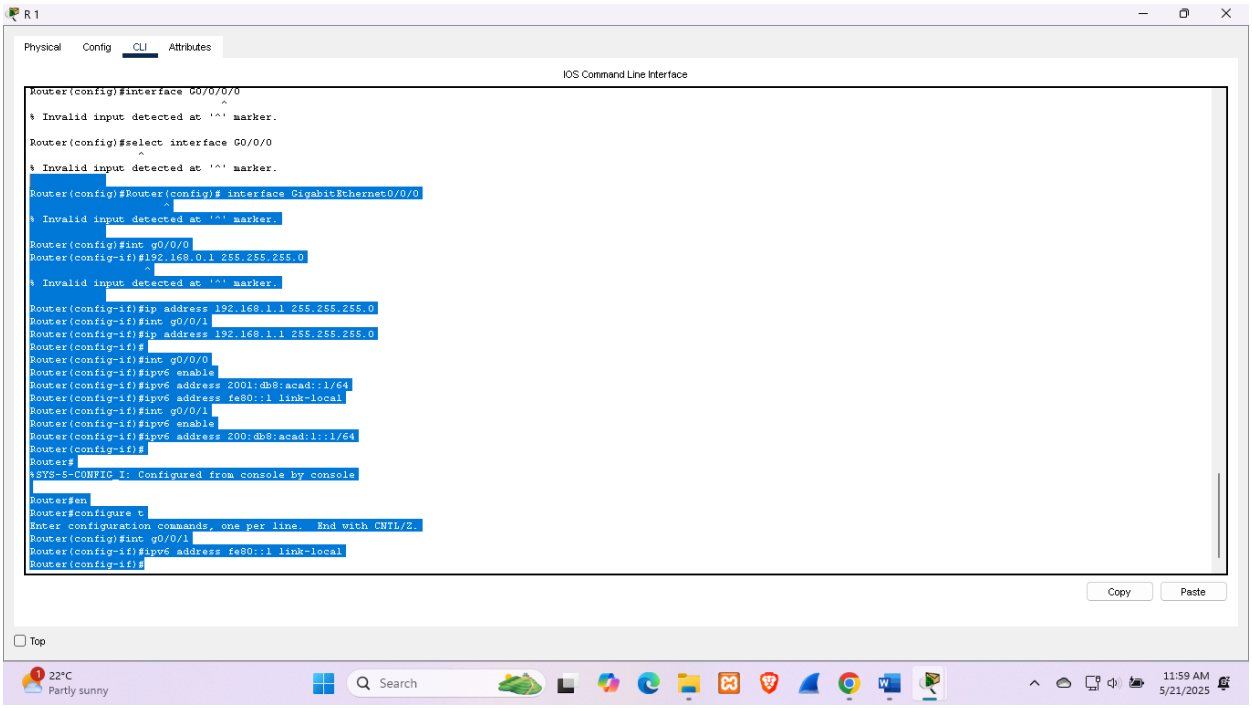


Fig 1.1Router configuration

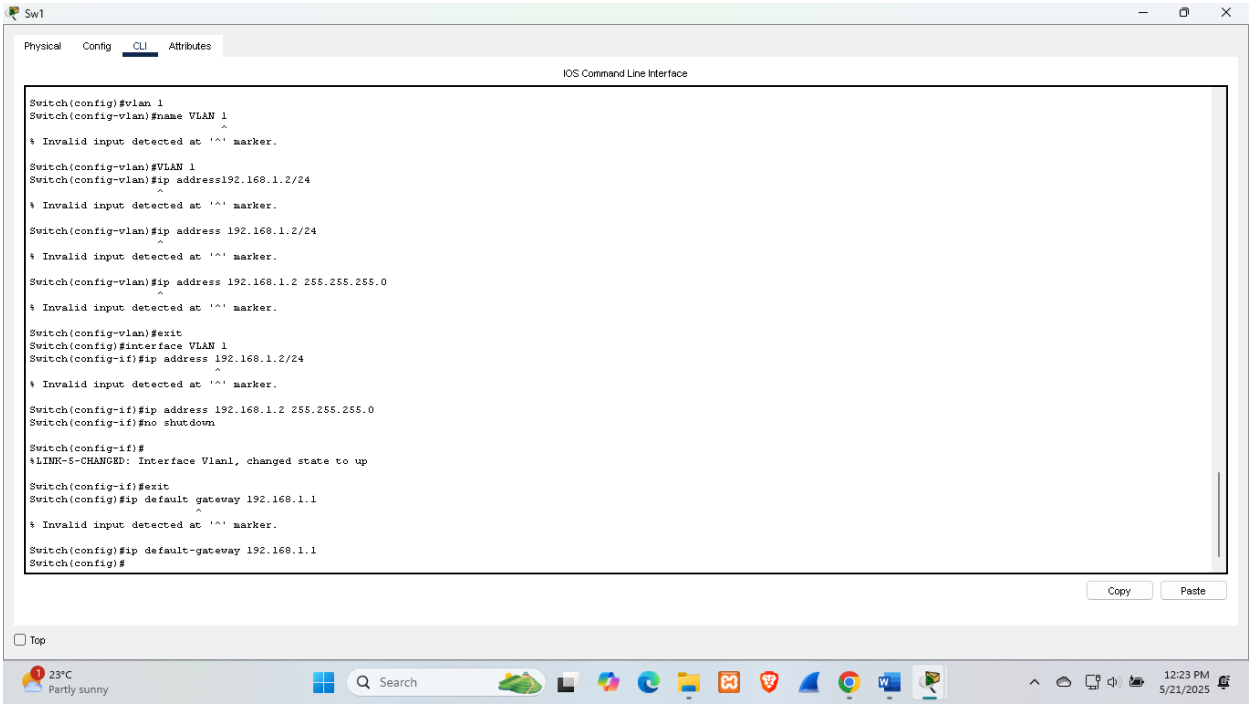


Fig 1.2 switch configuration

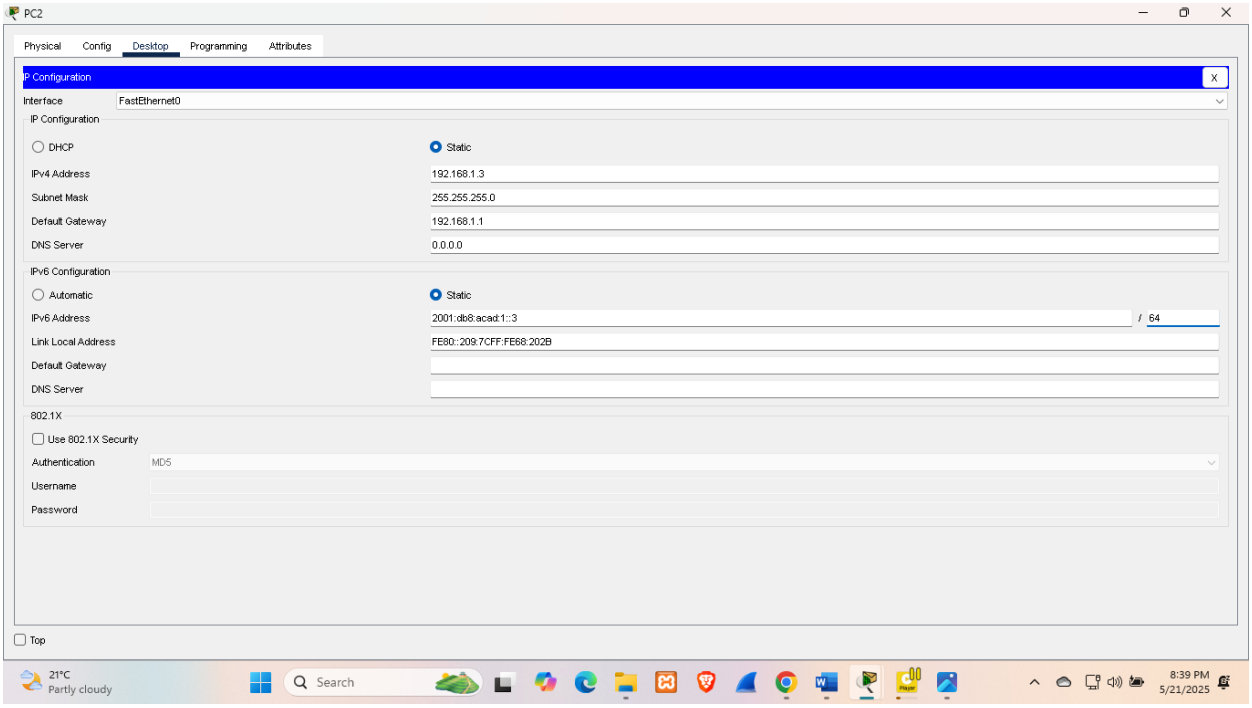


Fig 1.3PC A configuration

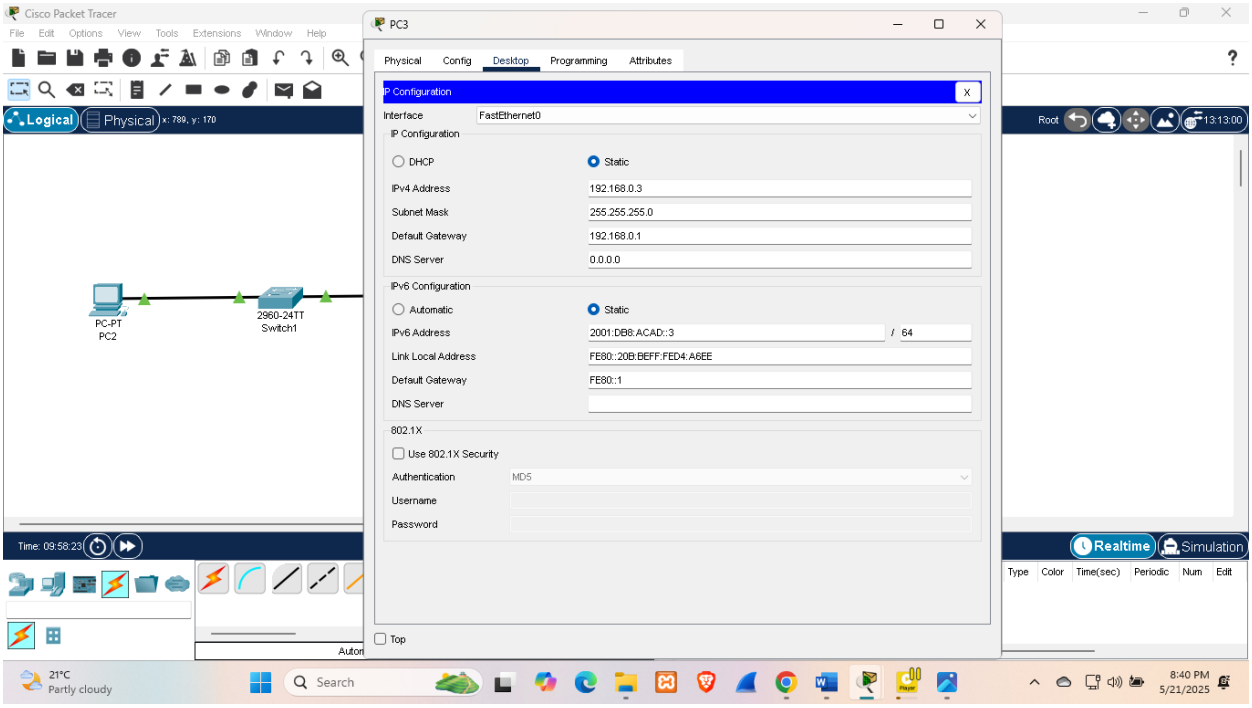


Fig 1.4 PC B configuration

The pings were successful after activating the links and configuring the interfaces correctly.

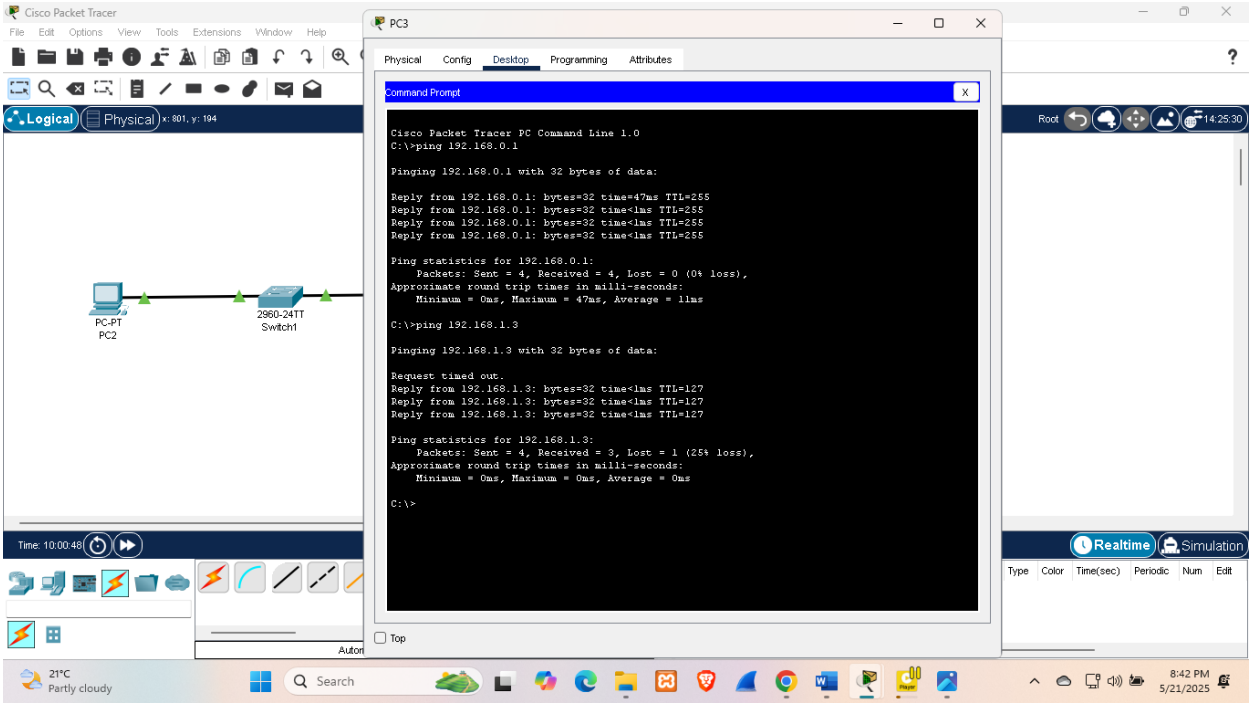


Fig 1.5 Successful ping

Configure the Router (R1)

H After setting up the physical connections, I accessed the router through the console port and entered privileged EXEC mode by typing enable. I then entered global configuration mode with the config terminal command. To personalize the device, I changed the hostname to R1, and then disabled DNS lookup to prevent the router from trying to resolve incorrect commands as hostnames using no ip domain-lookup. For security, I set an encrypted privileged EXEC password using enable secret class, configured a console password (cisco) and enabled login, then repeated similar steps for the VTY lines. I applied service password-encryption to encrypt all plaintext passwords. Next, I set up a login banner to warn against unauthorized access using banner motd $ Authorized Users Only! $.

I proceeded to configure the router interfaces. On **G0/0/0**, I assigned the IPv4 address 192.168.0.1 and the IPv6 address 2001:db8:acad::1/64, including the link-local address FE80::1. I used no shutdown to bring the interface up. I repeated the same steps for **G0/0/1** using the IP addresses 192.168.1.1 and 2001:db8:acad:1::1/64. I then added interface descriptions to clarify what each port was connected to. After that, I enabled IPv6 routing using ipv6 unicast-routing and saved the configuration with copy running-config startup-config. Lastly, I set the system clock using the clock set

command.

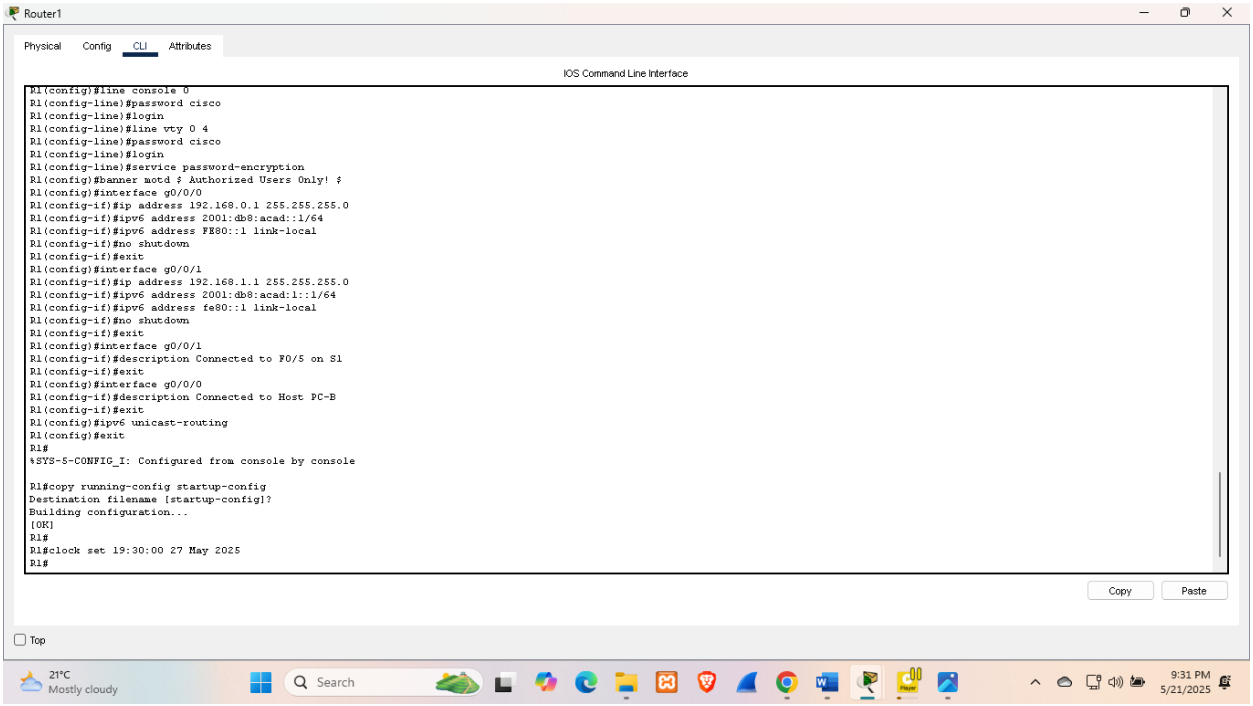


Fig 1.6 Configuring R1

Once configuration was complete, I pinged PC-B from PC-A. This time, the pings were successful because both router interfaces were up and correctly configured, allowing Layer 3

traffic to be routed between the two subnets.

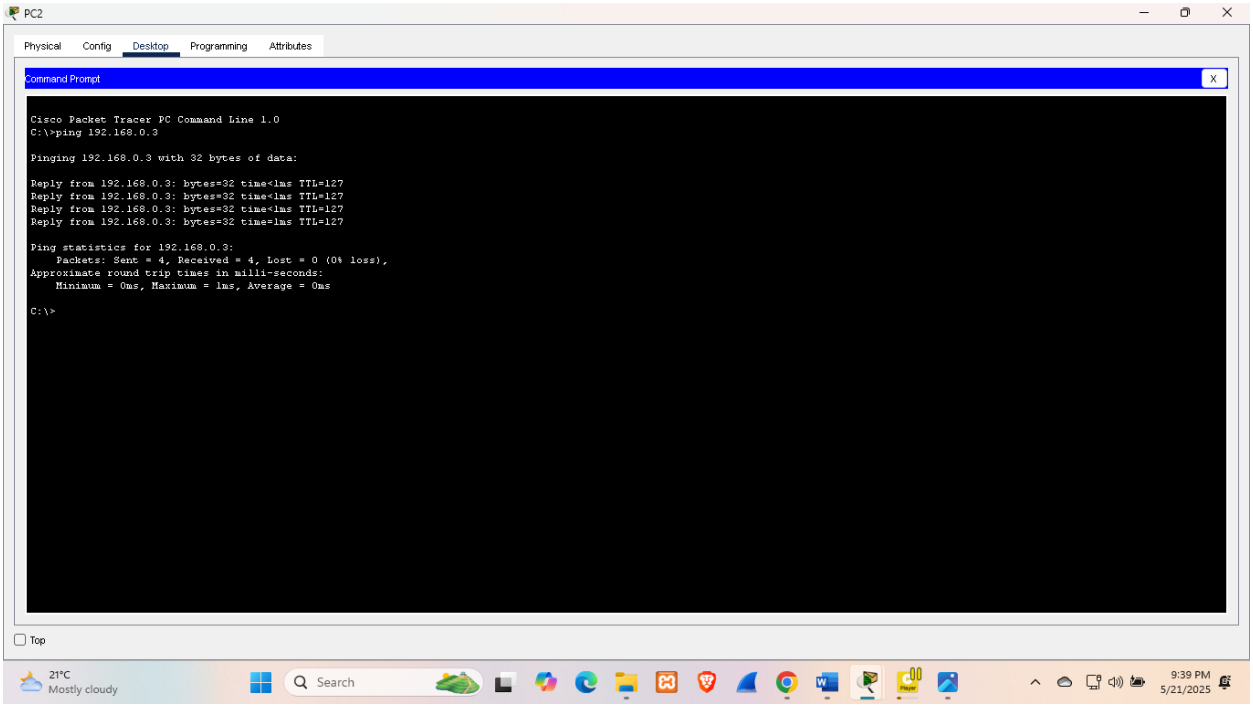


Fig 1.7 Successful ping

Configure the switch.

To configure the switch, I first consoled into it and entered privileged EXEC mode by typing enable, then entered global configuration mode with config terminal. I changed the hostname to S1 and disabled DNS lookup to prevent delays from invalid command entries using no ip domain-lookup. I then configured the management interface by entering interface vlan 1, assigning it the IP address 192.168.1.2 with a subnet mask of 255.255.255.0, and brought the interface up using no shutdown. After exiting interface configuration mode, I set the default gateway to 192.168.1.1 using the ip default-gateway command so the switch could communicate outside its local subnet. I saved the configuration with copy running-config startup-config.

To verify connectivity, I pinged PC-B from PC-A

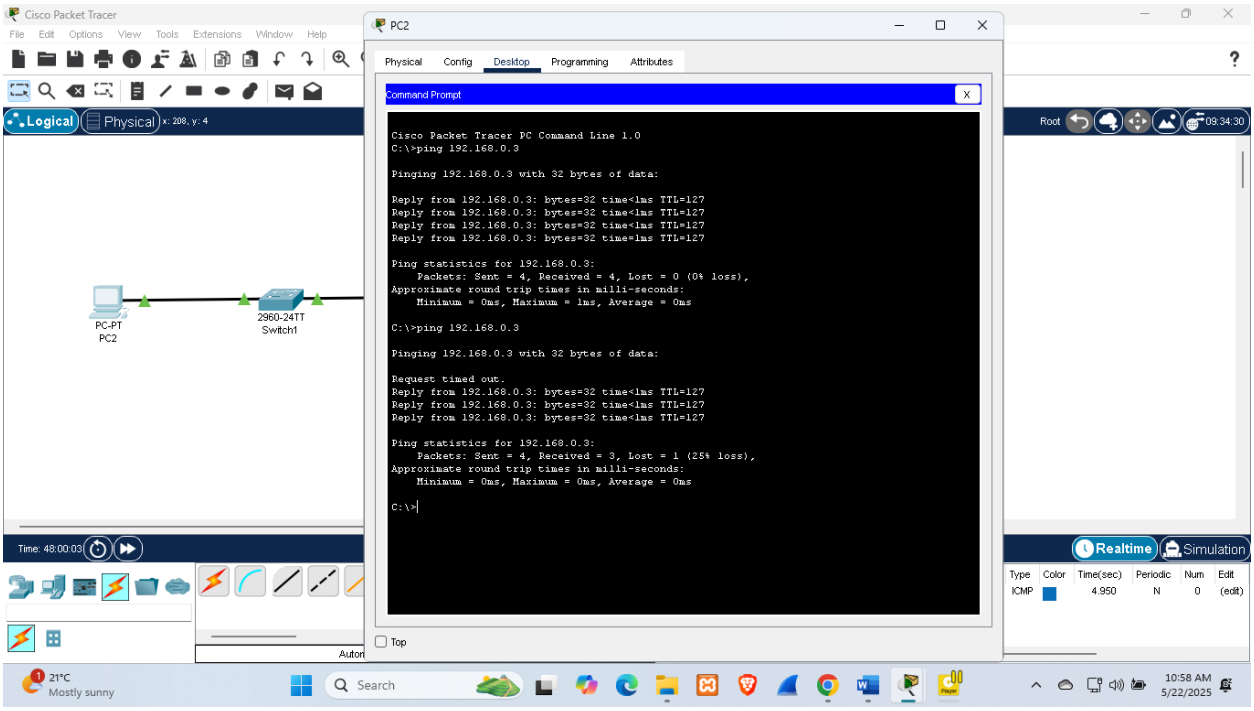


Fig 1.8 Successful ping(PC B from PC A)

and also from the switch (S1), and all pings were successful, confirming full end-to-end network communication.

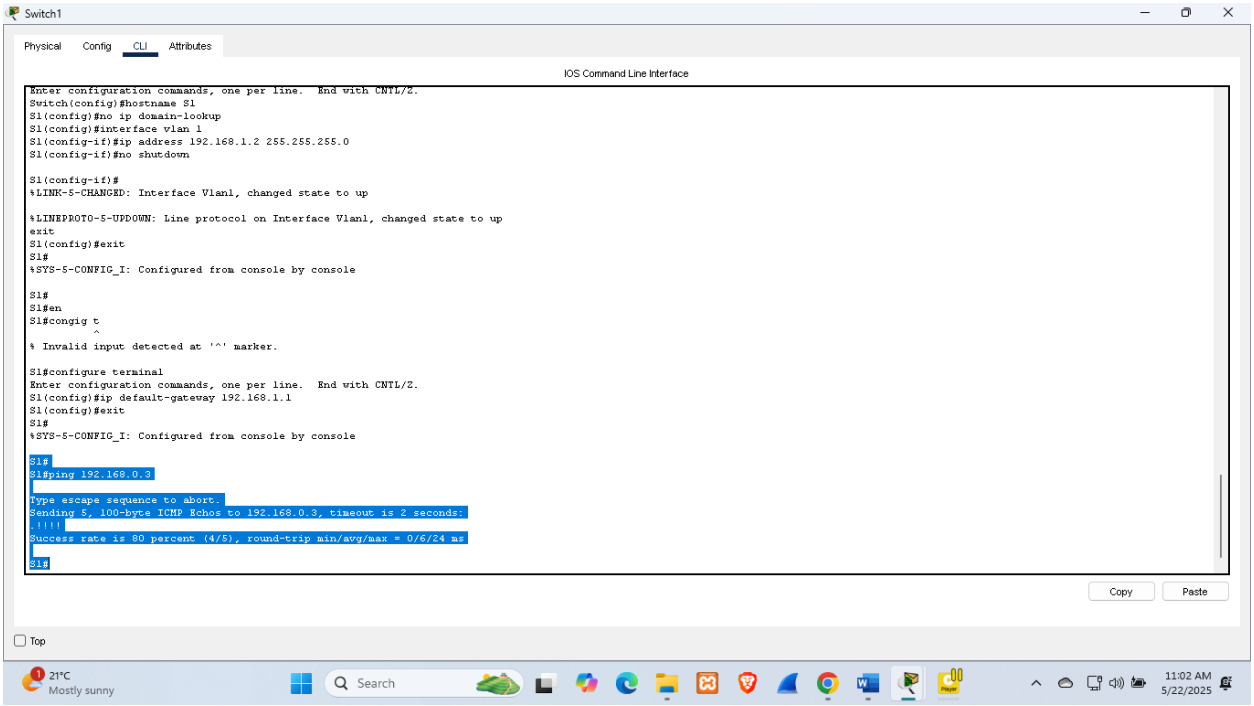


Fig 1.9 Successful ping(PC B from S1)

Display Device Information

Display the routing table on the router.

To examine the routing table on the router, I used the show ip route command in privileged EXEC mode. This command displayed a list of all known routes along with corresponding codes that indicate how each route was learned. From the output, I observed that the code **C** stands for **directly connected networks**, while **L** represents **local interfaces**—these are automatically added when an IP address is configured on an interface. There were **two C-coded entries** in the table: one for the 192.168.0.0/24 network connected to **GigabitEthernet0/0/0**, and one for the 192.168.1.0/24 network connected to **GigabitEthernet0/0/1**. This confirmed that both interfaces were properly configured and active, allowing the router to recognize and route traffic between the two directly connected subnets.

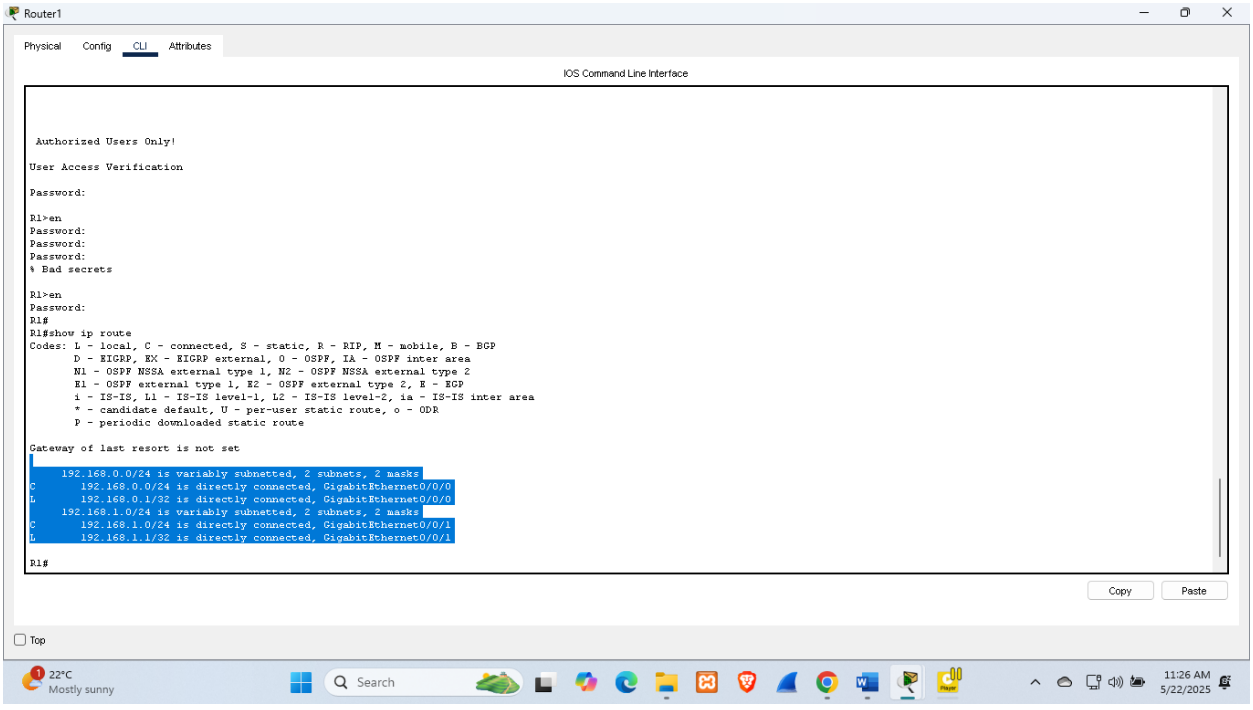


Fig 1.10 Routing Table

Then I used the command show ipv6 command on R1 to display the ipv6 routes

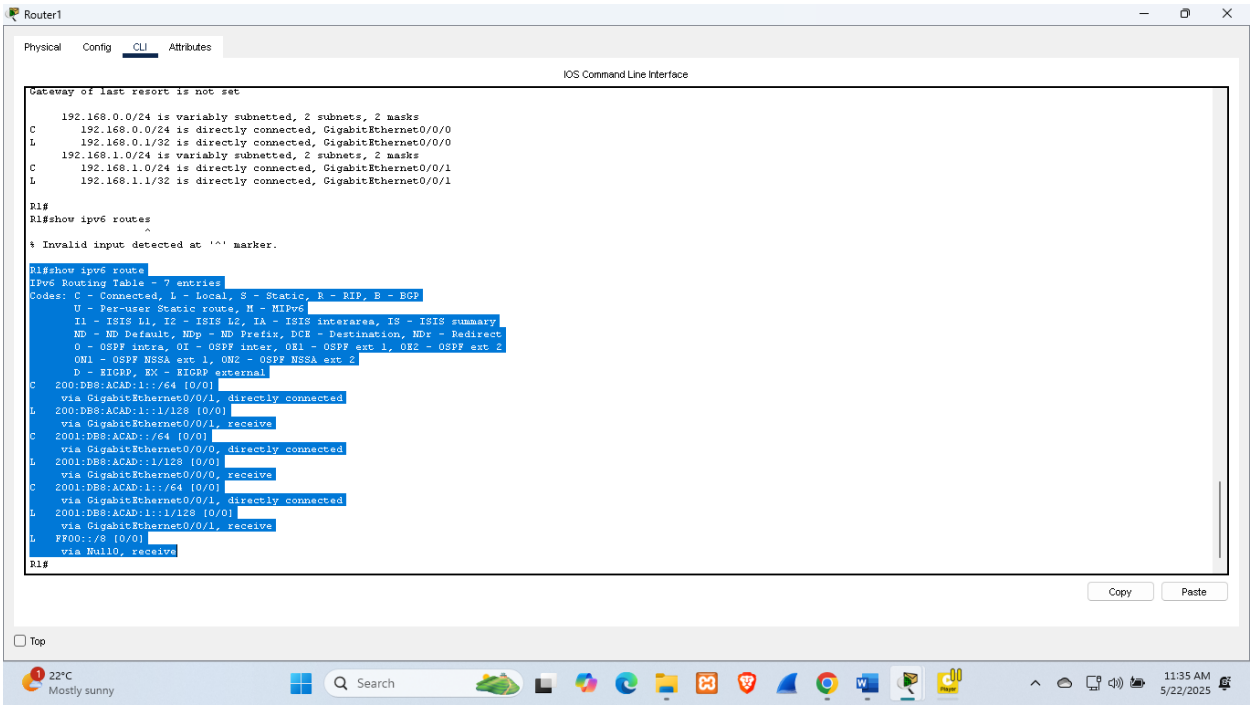


Fig 1.11 IPV6 Routes

Display interface information on the router R1.

To verify the configuration and status of the router's interface, I used the show ip interface g0/0/1 command on R1. This command provided detailed information about the interface's operational state, addressing, hardware, and traffic statistics. From the output, I confirmed that the interface **GigabitEthernet0/0/1 was up**, and the **line protocol was also up**, indicating it was fully operational and actively forwarding traffic. I also identified the **MAC address** of the interface, which appeared in the standard hexadecimal format (e.g., a0e0.af0d.e141). Additionally, I saw that the **IPv4 address was listed as 192.168.1.1/24**, confirming

that the interface was configured correctly with the expected IP settings as shown on the screenshot

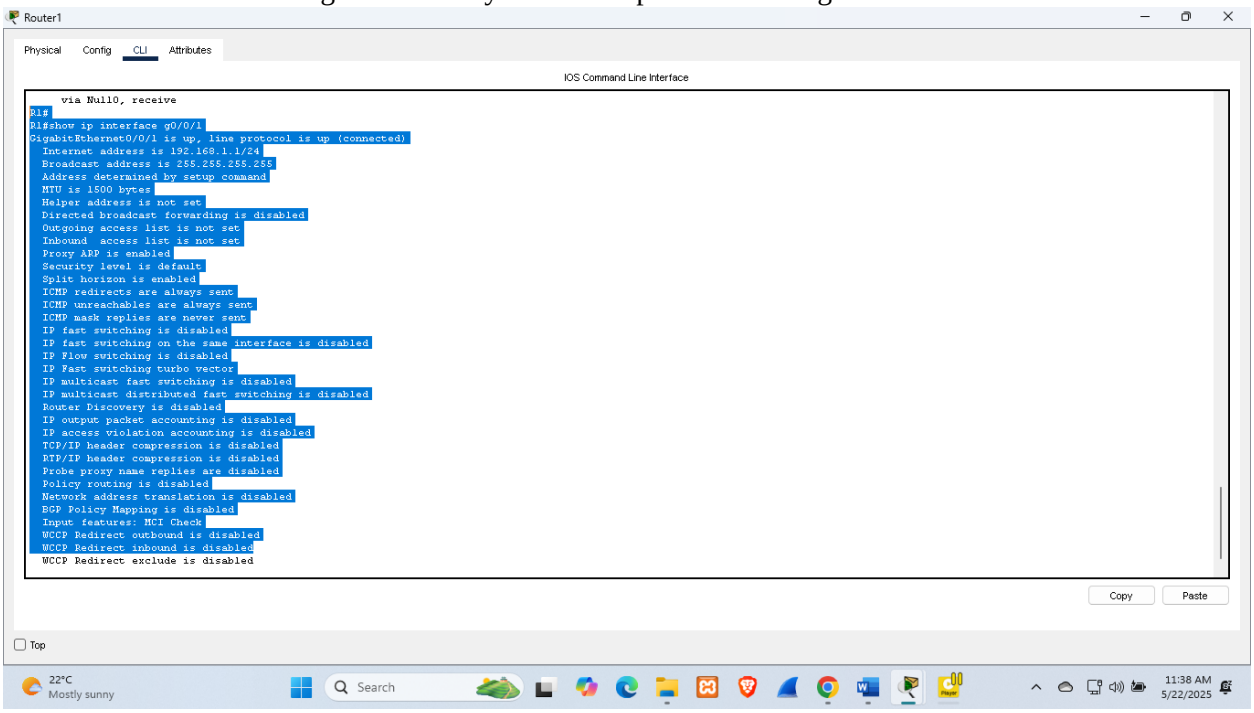


Fig 1.12 Info R1 interface

For the IPv6 information, I entered the **show ipv6 interface command**

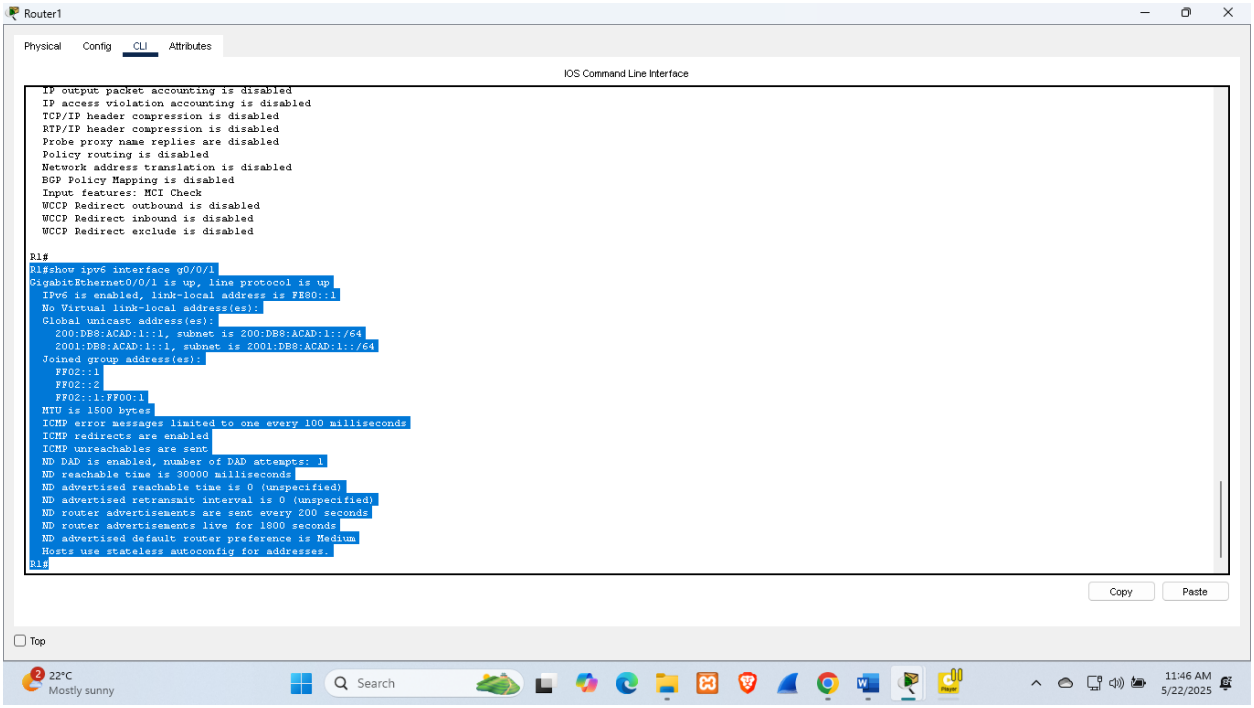


Fig 1.13 Info R1 interface(IPV6)

Display a summary list of the interfaces on the router and switch

Ip interface of R1

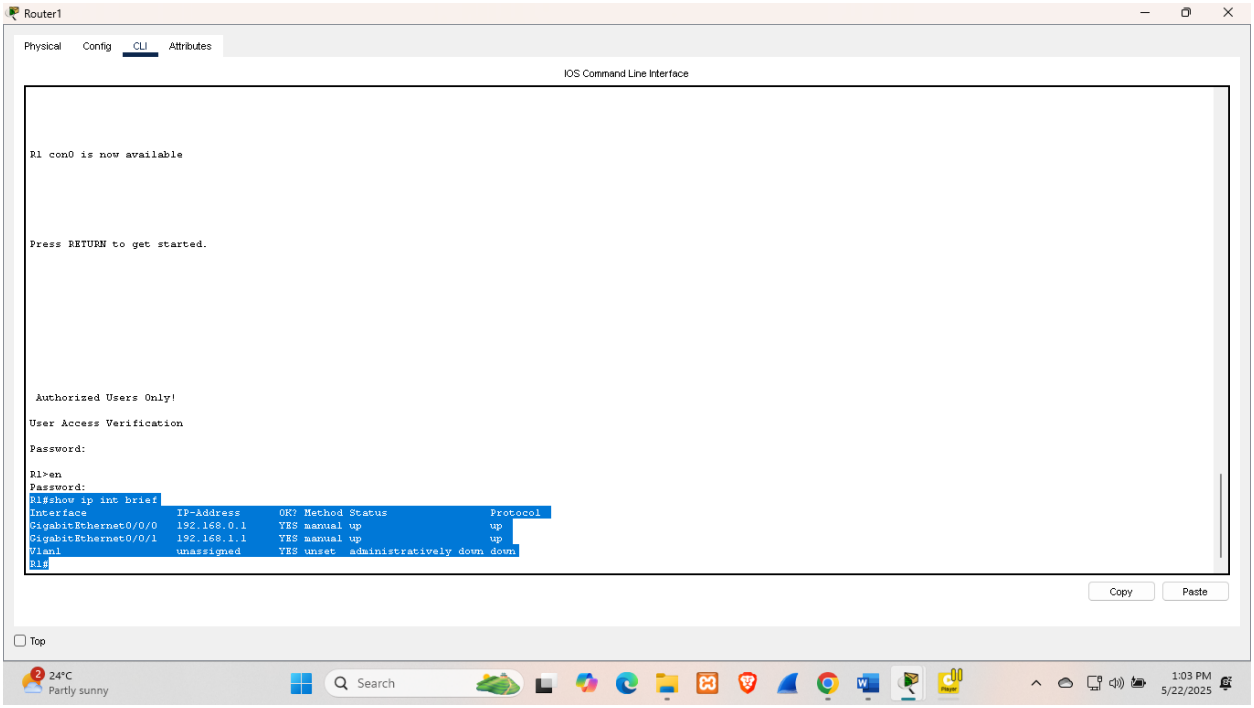


Fig 1.14 IPV4 interface(R1)

IPV6 interface of R1

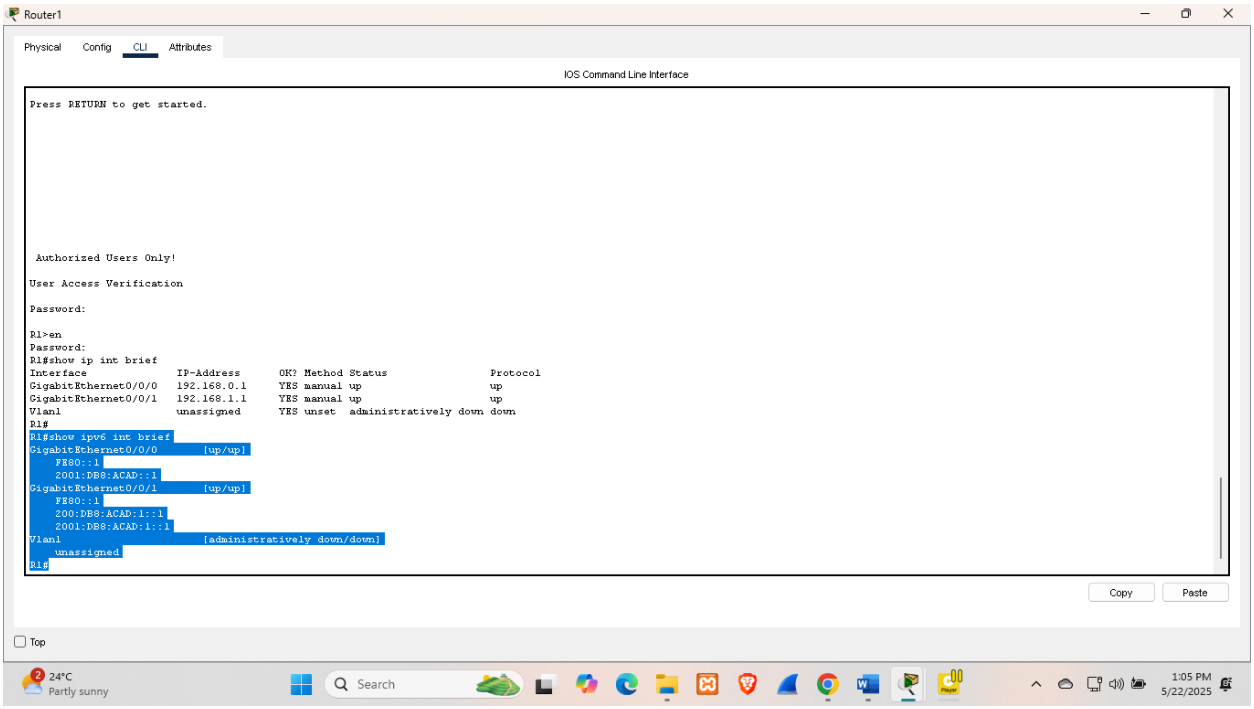


Fig 1.15 IPV6 interface(R1)

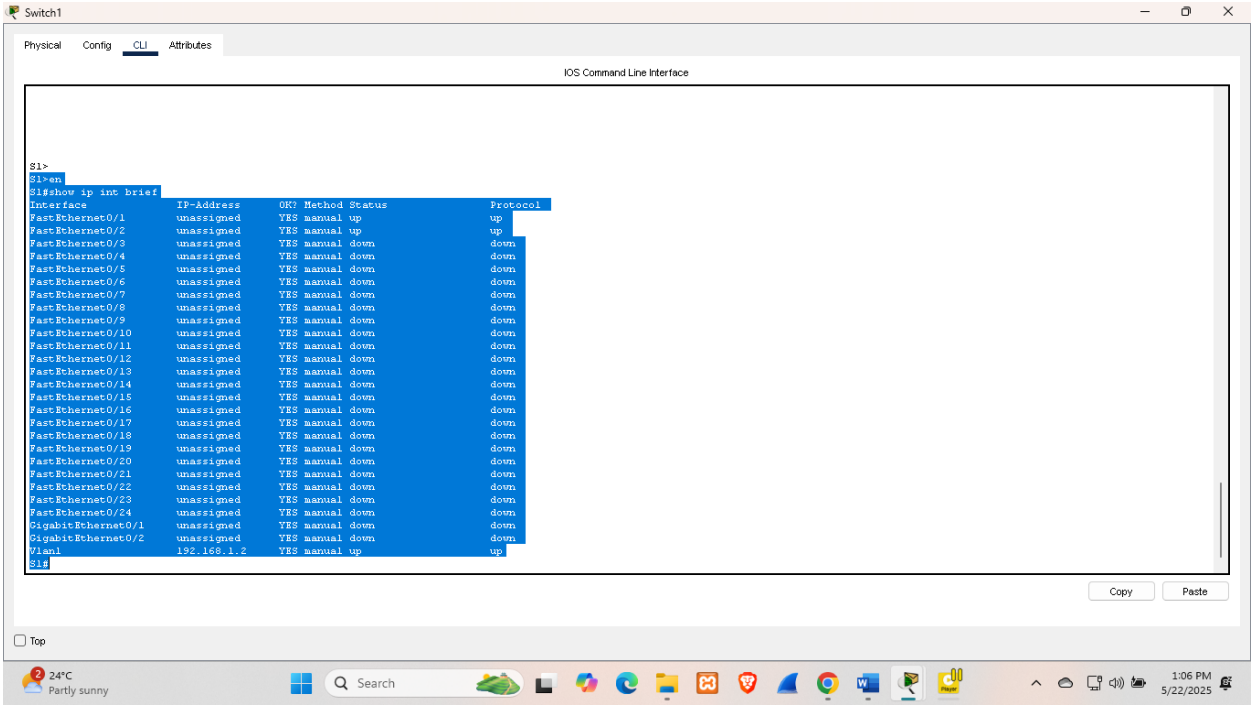


Fig 1.16 IPV4 interface(S1)

Reflective questions

If **G0/0/1 is administratively down**, I can bring it up by entering interface configuration mode and using the no shutdown command. This activates the interface so it can forward traffic.

If **G0/0/1 is mistakenly assigned 192.168.1.2**, PC-A won't be able to reach PC-B. That's because PC-A is set to use 192.168.1.1 as its default gateway, and if that address isn't configured on any device, packets won't be routed outside the local network. Proper IP configuration is essential for network communication.

CONCLUSION

This lab helped me gain confidence in setting up and managing network devices. I successfully configured the router, switch, and PCs, and verified communication between them using commands like **ping, show ip** route, and **show interface**. I also learned how simple misconfigurations, like assigning the wrong IP address or forgetting to enable an interface, can break connectivity. Overall, this was a valuable experience that strengthened my CLI skills and deepened my understanding of how routers and switches operate in a network.