COMPUTER NETWORKS LAB MANUAL

Created By

Daniyal Faheem 2024F-mulug-1002

Submitted to:

Shamin Batool

SCHOOL OF INFORMATION TECHNOLOGY
MINHAJ UNIVERSITY LAHORE

Contents

CHAPTER # 0

TOPIC: BASIC NETWORKING COMMANDS

Objective:

To familiarize myself with the Windows Command Line Interface (CLI) and execute basic networking commands to gather system and network information.

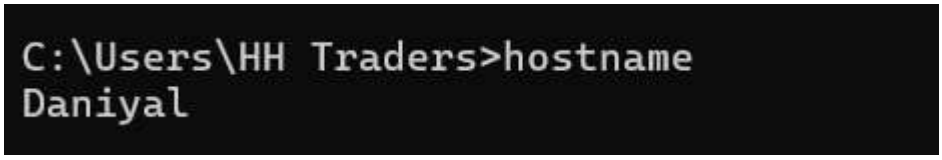
Procedure:

I opened the Command Prompt (cmd) on my Windows system and executed the following commands.

1. Hostname

This command is the simplest one. It just tells us the name of the computer we are currently working on.

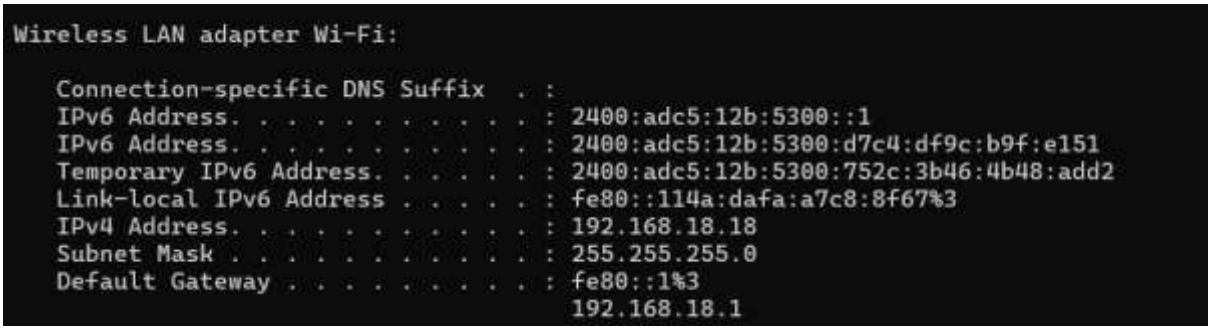
- **Command:** hostname
- **Observation:** The system returned the device name (e.g., Daniyal).



2. IPCONFIG

This is probably the most used command. It shows the current TCP/IP network configuration values. It helps me see my IP address, Subnet Mask, and Default Gateway.

- **Command:** ipconfig
- **Observation:** I could see the IPv4 address assigned to my Wi-Fi adapter.



3. IPCONFIG /ALL

This is a more detailed version of the previous command. It shows everything, including the DNS server and the physical address (MAC address).

- **Command:** ipconfig /all
- **Observation:** It listed all adapters, even the disconnected ones, and showed the physical MAC address for each.

```
C:\Users\HH Traders>ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : Daniyal
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No

Ethernet adapter Ethernet:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) Ethernet Connection (2) I219-LM
   Physical Address. . . . . . . . . : 54-E1-AD-48-6C-FC
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes

Ethernet adapter Ethernet 3:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : VirtualBox Host-Only Ethernet Adapter
   Physical Address. . . . . . . . . : 0A-00-27-00-00-08
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::5e4c:de24:91a7:4159%8(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.56.1(Preferred)
```

```
Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) Dual Band Wireless-AC 8260
   Physical Address. . . . . . . . . : E4-A4-71-E3-6B-AB
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   IPv6 Address. . . . . . . . . . . : 2400:adc5:12b:5300::1(Preferred)
   Lease Obtained. . . . . . . . . . : Sunday, November 30, 2025 1:56:43 PM
   Lease Expires . . . . . . . . . . : Monday, December 1, 2025 1:56:43 PM
   IPv6 Address. . . . . . . . . . . : 2400:adc5:12b:5300:d7c4:df9c:b9f:e151(Preferred)
   Temporary IPv6 Address. . . . . . : 2400:adc5:12b:5300:752c:3b46:4b48:add2(Preferred)
   Link-local IPv6 Address . . . . . : fe80::114a:dafa:a7c8:8f67%3(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.18.18(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Sunday, November 30, 2025 1:56:46 PM
   Lease Expires . . . . . . . . . . : Sunday, November 30, 2025 3:56:43 PM
   Default Gateway . . . . . . . . . : fe80::1%3
                                       192.168.18.1
   DHCP Server . . . . . . . . . . . : 192.168.18.1
   DHCPv6 IAID . . . . . . . . . . . : 65315953
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2C-95-39-B7-54-E1-AD-48-6C-FC
   DNS Servers . . . . . . . . . . . : fe80::1%3
                                       192.168.18.1
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

4. GETMAC

This command is used specifically to find the Media Access Control (MAC) address of the network adapters. It's a unique hardware ID.

- **Command:** getmac
- **Observation:** It displayed the physical addresses.

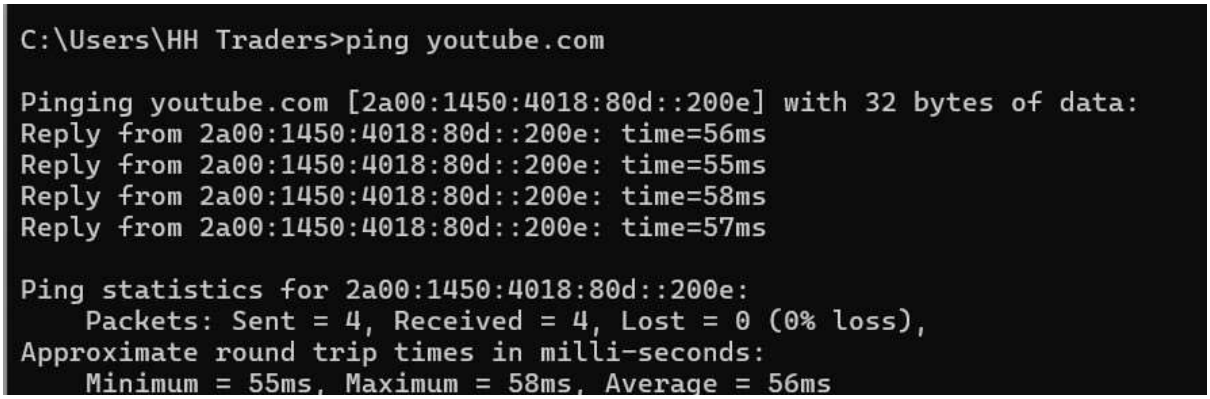
```
C:\Users\HH Traders> getmac

Physical Address    Transport Name
=================== ============================================================
54-E1-AD-48-6C-FC   Media disconnected
E4-A4-71-E3-6B-AB   \Device\Tcpip_{0C1433F4-0E38-408A-9E62-CF163C8BBE1D}
0A-00-27-00-00-08   \Device\Tcpip_{412760A2-0423-4C33-A3F6-7C030E2CB934}
```

5. PING

This command is used to test if a host is reachable. I used it to check if my internet was working by pinging a public server.

- **Command:** ping google.com
- **Observation:** My computer sent 4 packets of data. I received 4 replies, which means the connection is stable with 0% packet loss.



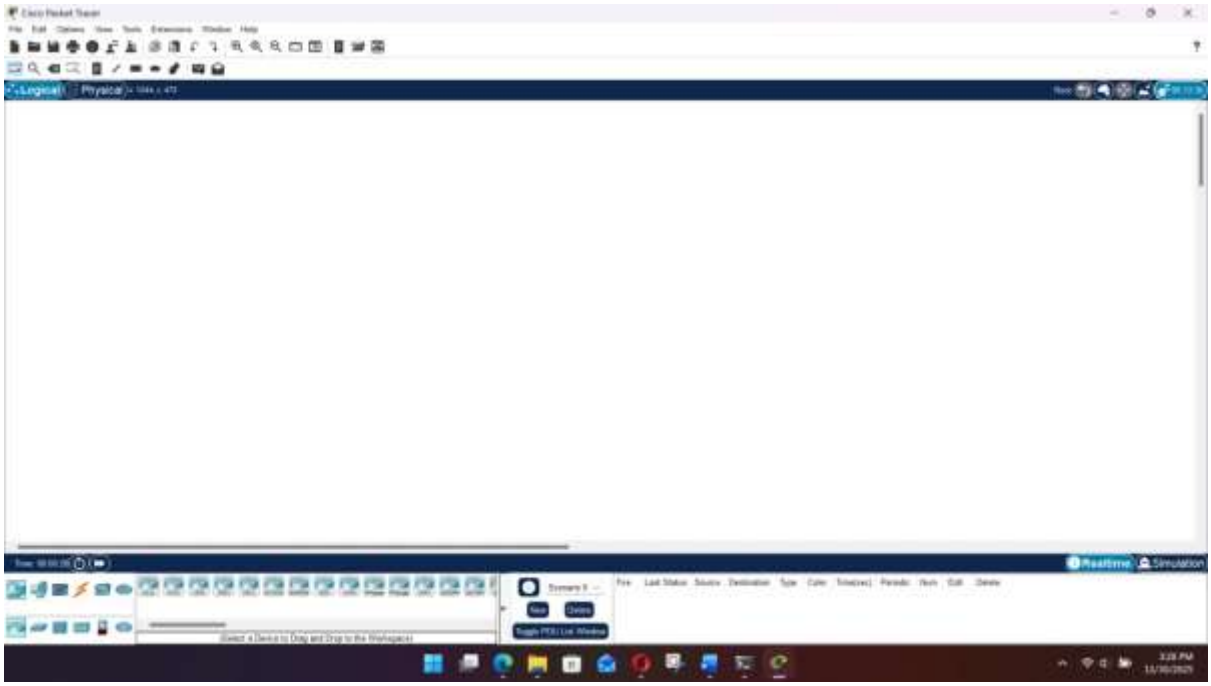
CHAPTER # 01

TOPIC: INTRODUCTION TO CISCO PACKET TRACER

Introduction:

Cisco Packet Tracer is a simulation tool we use in the lab to create virtual networks. It allows us to experiment with routers, switches, and PCs without needing physical hardware. It helps in troubleshooting and understanding how data flows. **Key Interface Areas:**

1. **Workspace:** The big white area where we draw the network.



2. **Device Box (Bottom Left):** This is where we pick devices like Routers, Switches, Hubs, Connections (cables), and End Devices (PCs/Laptops).

3. **Top Menu:** Contains standard options like Save, Print, and Zoom.

Common Tools I Used:

- **Place Note (N):** I used this to write labels (like IP addresses) next to the PCs so I don't forget them.

- **Delete (Del):** To remove a wrong wire or device.
- **Inspect:** To look at the routing tables or ARP tables.
- **Add Simple PDU:** This is basically a "virtual ping" message to test if PC1 can talk to PC2.

CHAPTER # 02

TOPIC: NETWORK TOPOLOGIES

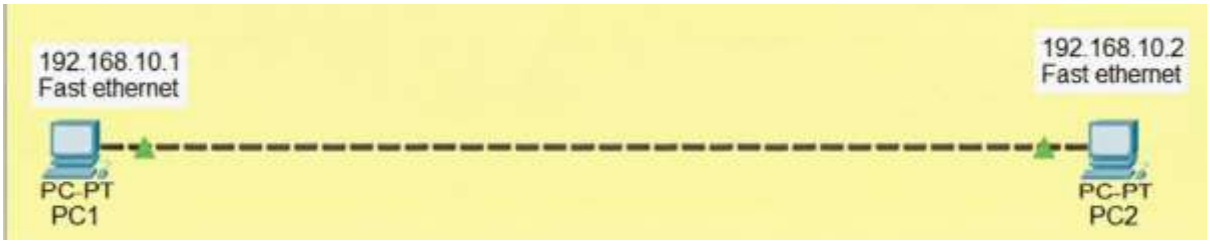
Objective:
To design and simulate different network layouts (topologies) to understand how nodes are connected.

1. Peer-to-Peer (P2P)

This is the simplest connection between just two computers. ● **Setup:** I connected PC0 and

PC1 directly using a **Copper Cross-Over Cable**.

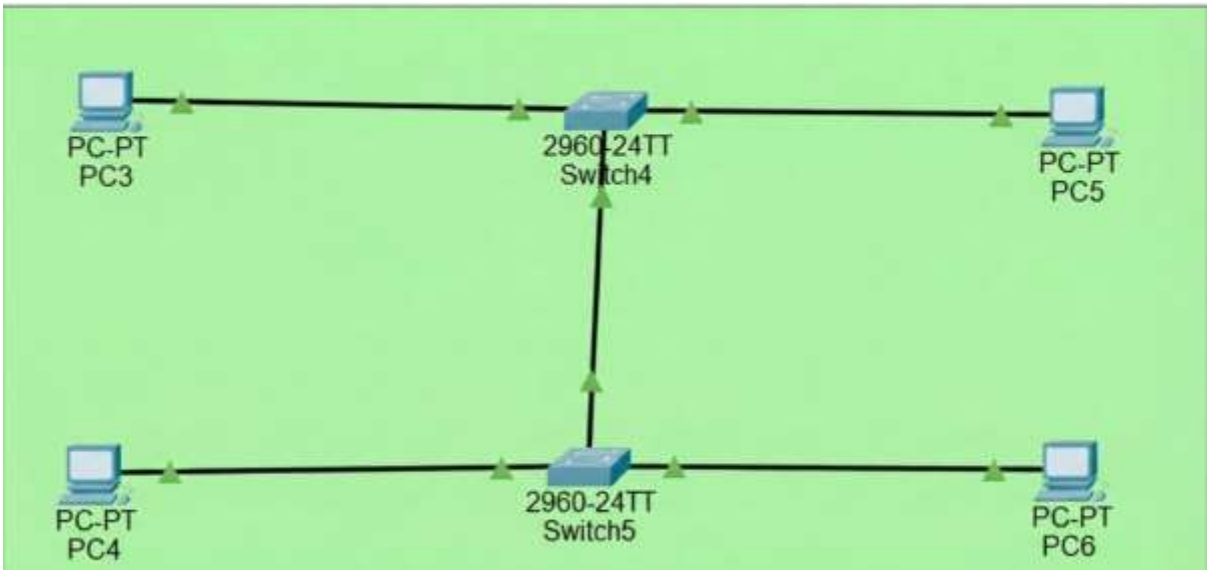
- **Configuration:**
  - PC1 IP: 192.168.10.1
  - PC2 IP: 192.168.10.2
- **Result:** I was able to ping PC1 from PC2 successfully.



2. Bus Topology

In this layout, all devices share a single communication line (backbone).

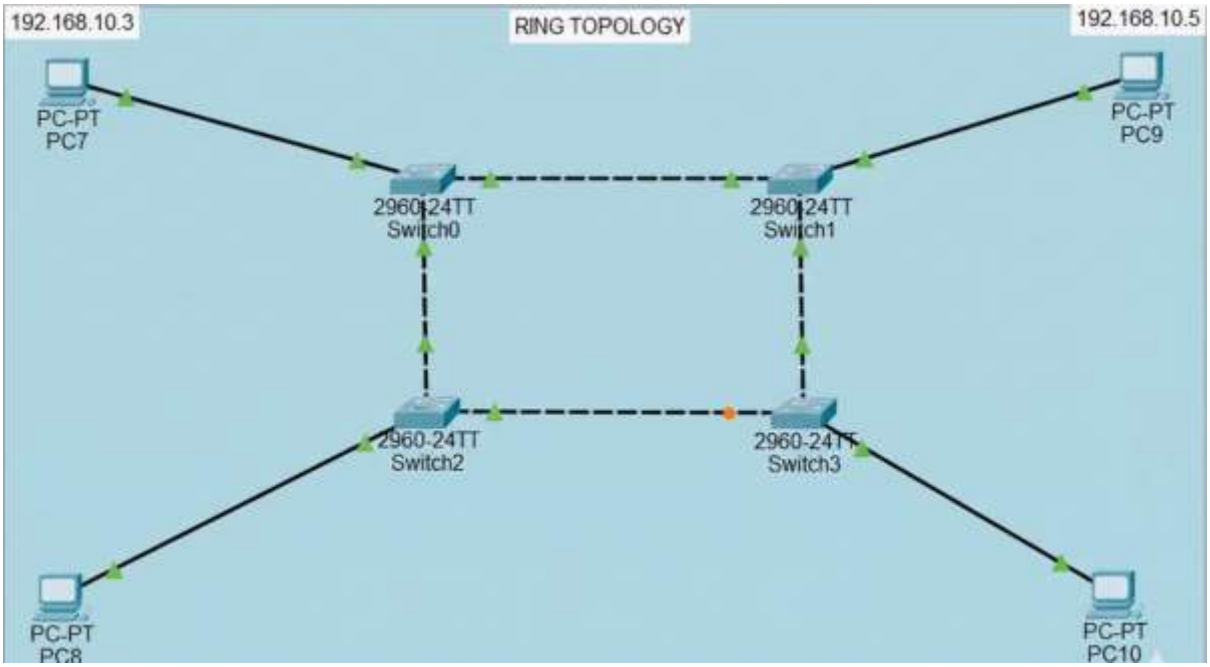
- **Setup:** Since Packet Tracer doesn't have a specific "Bus" cable, I simulated this using a series of Switches/Hubs connected in a line.
- **Observation:** If the main link breaks, the communication stops. It’s not very robust.



3. Ring Topology

Here, every device has exactly two neighbors for communication purposes.

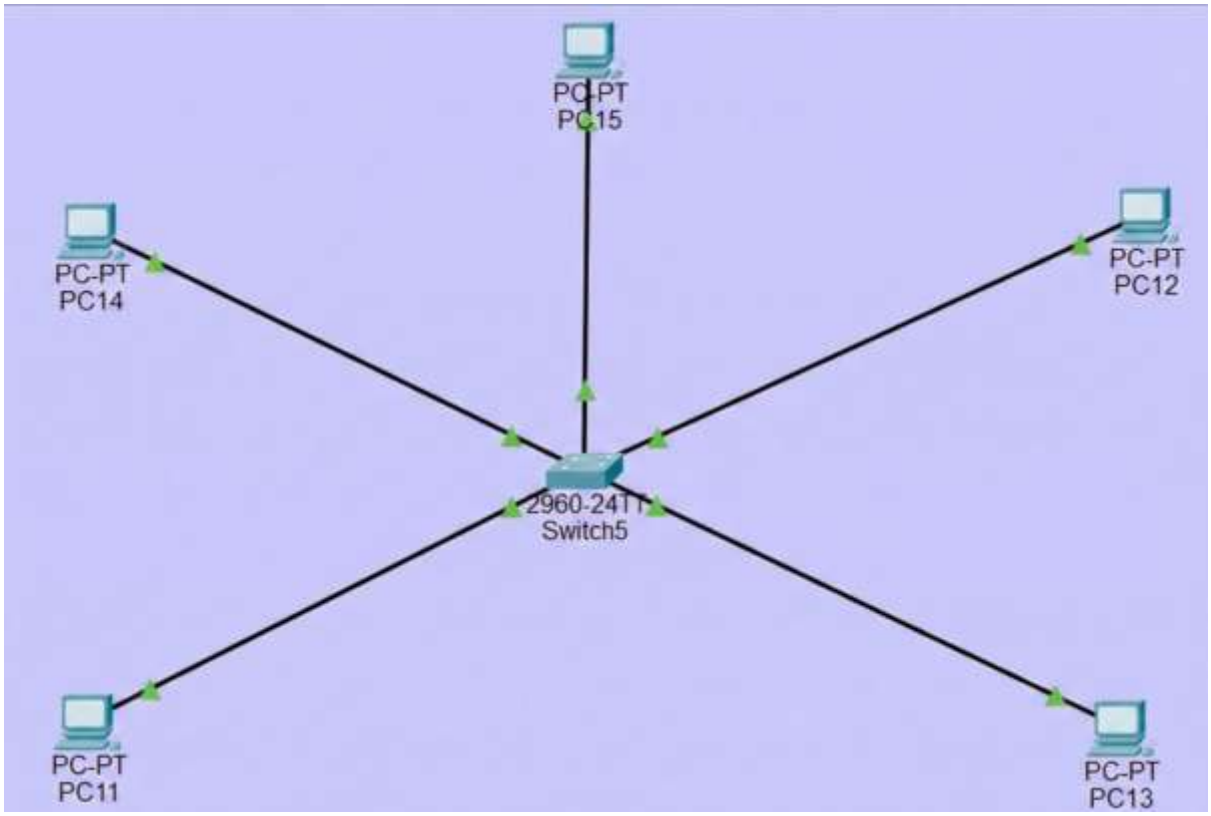
- **Setup:** I connected 4 Switches in a circle (closed loop) and attached one PC to each switch.
- **Observation:** Data travels in a circle. In a real Token Ring, a token is passed around, but here we just simulated the circular physical connection.



4. Star Topology

This is the most common one. All devices connect to a central device (Switch).

- **Setup:**
  - Central Device: 2960 Switch.
  - End Devices: 5 PCs connected via Copper Straight-Through cables.
- **IP Addressing:** 192.168.10.1 to 192.168.10.5
- **Result:** This is the best topology because if one cable breaks, only that PC goes down. The rest of the network stays up.



CHAPTER # 03

TOPIC: STUDY OF NETWORK DEVICES

Objective:

To understand the difference between various connecting devices used in our labs.

1. Repeater

- **Function:** It works at the Physical Layer. Its main job is to regenerate weak signals so they can travel longer distances.
- **Lab Scenario:** I placed a repeater between two distant wire segments to extend the network range.

2. Hub

- **Function:** It is a basic device that connects multiple PCs.
- **Key Characteristic:** It is "dumb." When it receives data, it broadcasts it to **everyone** on the network.
- **Security Risk:** I noticed in simulation mode that when PC1 sent a message to PC2, PC3 also received it. This makes the hub insecure.

3. Switch

- **Function:** It looks like a hub but is "intelligent." It uses MAC addresses to decide where to send data.
- **Observation:** Unlike the Hub, the Switch only sent the data to the specific destination PC. This reduces traffic and improves security.

4. Router

- **Function:** Used to connect two *different* networks (e.g., connecting a LAN to the Internet).
- **Lab Scenario:** I used a Router to connect a network with IP 192.168.1.0 to a network with IP 10.0.0.0.

CHAPTER # 04

TOPIC: VLAN CONFIGURATION (Virtual LAN)

Objective:

To logically separate a single physical switch into different networks (Sales and HR) for better management and security. **Scenario:**

- **VLAN 10:** Named "SALES" (Ports 1-2)
- **VLAN 20:** Named "HR" (Ports 3-4) **Steps I Followed:**

1. Enable Switch Configuration:

First, I went into the CLI of the switch and entered privileged mode.

Switch> enable

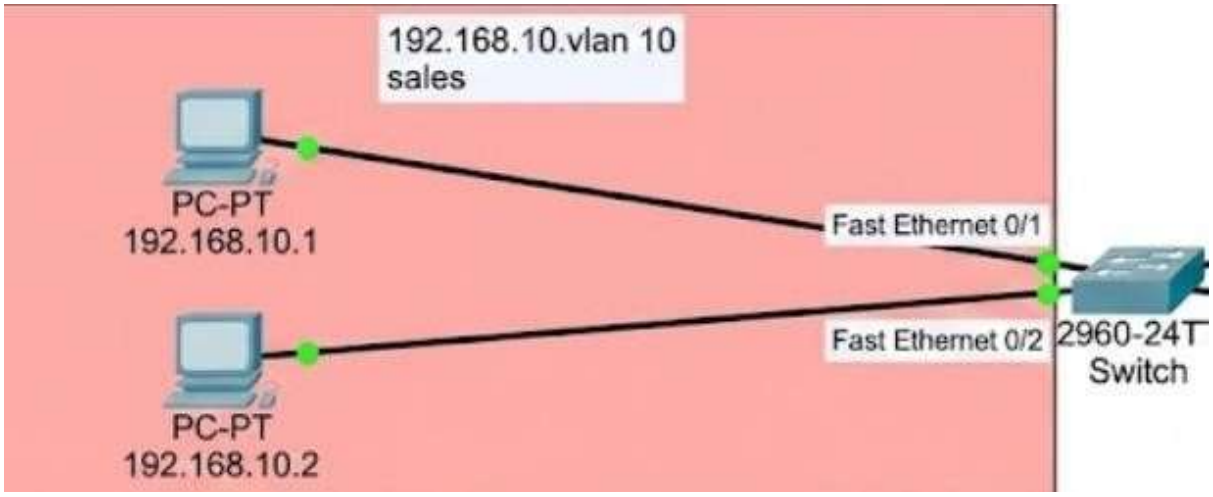
Switch# configure terminal

2. Creating the VLANs:

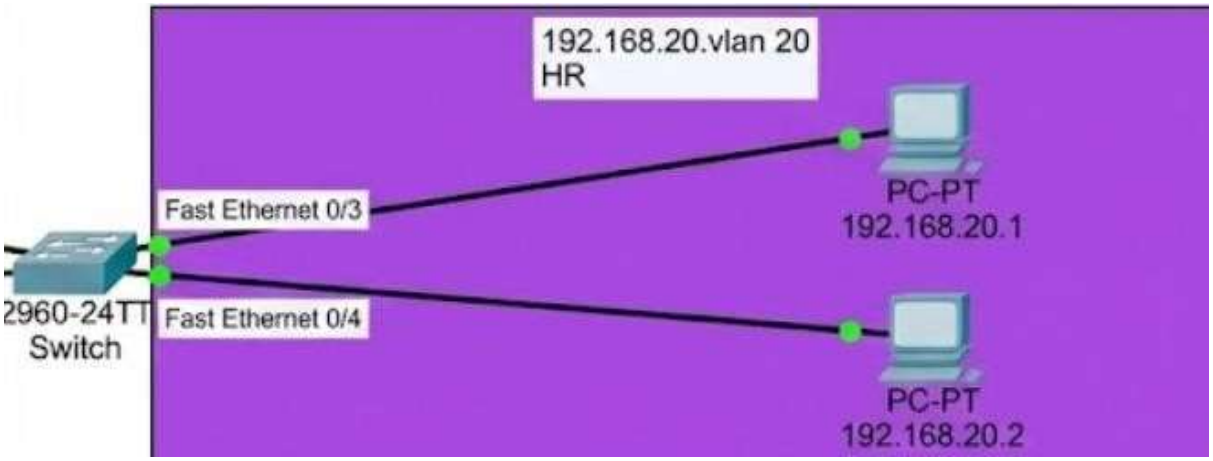
I created the two separate groups. Switch(config)# vlan 10

Switch(config-vlan)# name SALES

Switch(config-vlan)# exit



Switch(config)# vlan 20
Switch(config-vlan)# name HR
Switch(config-vlan)# exit



3. Assigning Ports to VLANs:
I assigned the first two ports to the Sales team.
Switch(config)# interface fastEthernet 0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# exit

(I repeated this for port 0/2)

Then I assigned the next ports to the HR team.

Switch(config)# interface fastEthernet 0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 20
Switch(config-if)# exit

CHAPTER # 5:

OSI Layers

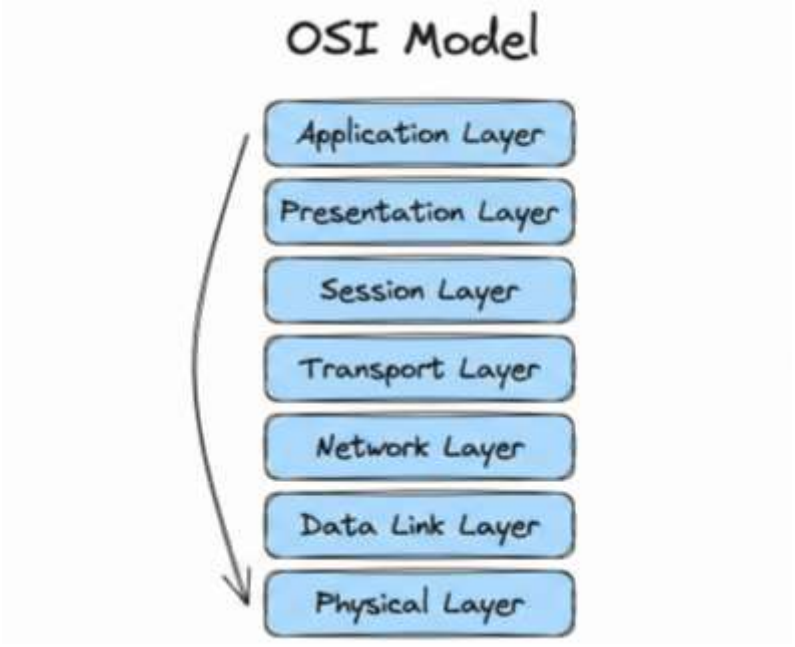
OSI MODEL:

The OSI Model (Open Systems Interconnection Model) is a simple guide that shows how computers talk to each other over a network. It's like a rulebook that helps all devices understand each other, no matter what brand or type they are.

The ISO (International Organization for Standardization) created this model so that every device could communicate using the same standard method.

The OSI Model has 7 layers. Think of it like a 7-step process where each step does one specific job. All these layers work together like a team to make sure your messages and data reach the right place safely.

The best part? By breaking communication into these simple layers, it's much easier to understand how networks work. It also helps engineers create new devices that can easily connect with existing technology. Everyone follows the same plan, so everything just works smoothly together.



OSI MODEL LAYERS

Layer 7: Application Layer:

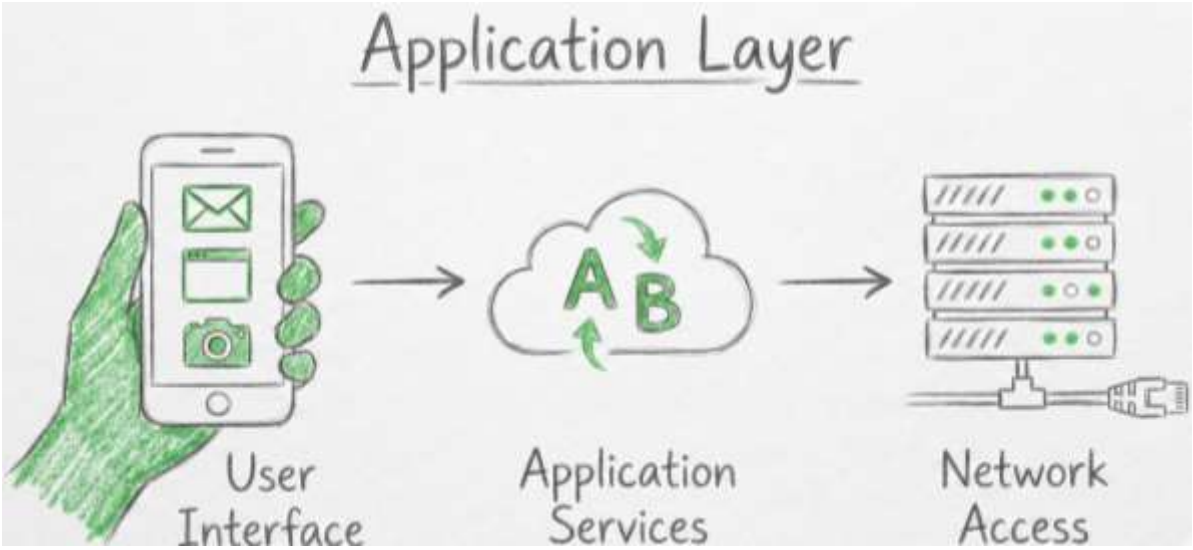
The Application Layer sits at the very top of the OSI Model – it's the layer closest to us, the actual users. This is where all the apps and programs we use every day do their work, like email, web browsers, and file-sharing tools.

This layer creates the data that needs to be sent across the network. For example, when you send an email or download a file, it starts here.

You can think of the Application Layer as a window or gateway. It lets your apps connect to the network and send information out. It also receives data coming back and displays it to you in a way you can understand and use.

Some common protocols (which are like specific rules or languages) used in this layer include:

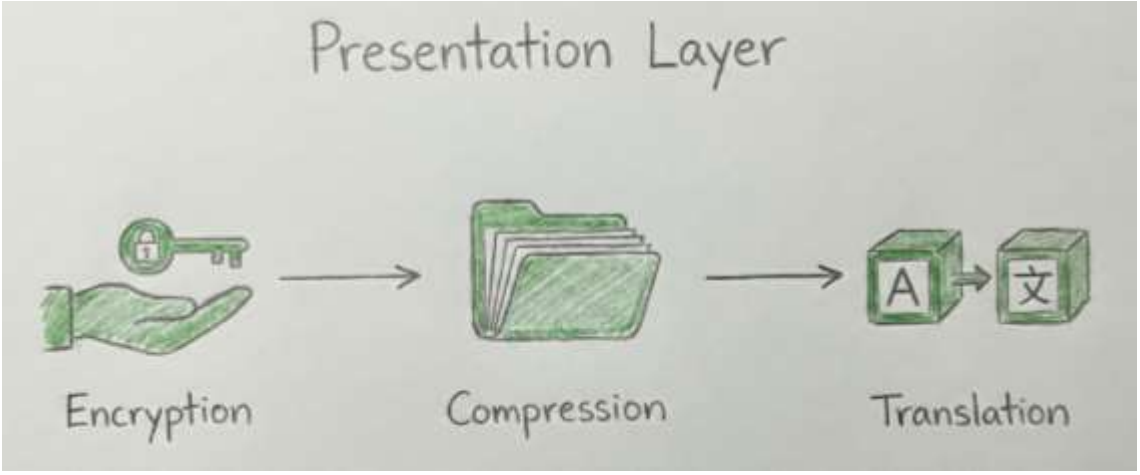
- **SMTP:** for sending emails
- **FTP:** for transferring files
- **DNS:** for finding website addresses



Layer 6: Presentation Layer

The Presentation Layer is sometimes called the Translation Layer because it translates data into the right format. Think of it as a translator that makes sure information is in a language that both the sender and receiver can understand.

When data comes from the Application Layer above, this layer takes it and converts or formats it properly so it can travel smoothly across the network. It's like packaging a gift before you ship it you need to wrap it



the right way!

This layer also handles important jobs like:

- **Encryption:** scrambling data to keep it secure and private
- **Compression:** making files smaller so they're faster to send
- **Formatting:** arranging data in standard formats that other devices can recognize

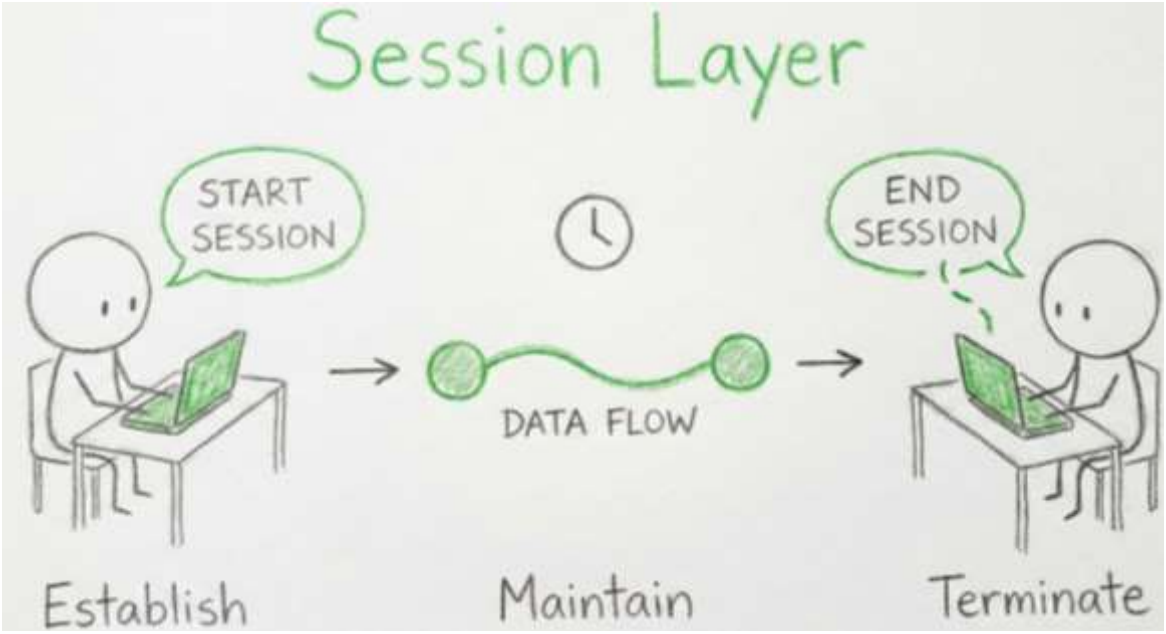
Some common protocols and formats used in this layer include:

- **TLS/SSL:** for securing data and keeping your online activity safe (like when you see "https" in a web address)
- **JPEG, MPEG, GIF:** these are file formats for pictures and videos that help display images and media correctly

Layer 5: Session Layer

The Session Layer is like a manager that controls conversations between two devices. It's responsible for starting, maintaining, and ending communication sessions between computers.

Think of it like a phone call this layer helps "dial" the connection, keeps the call going smoothly while



you're talking, and then properly "hangs up" when you're done.

This layer also takes care of important security tasks like:

- **Authentication:** making sure the devices talking to each other are really who they say they are
- **Security:** protecting the connection from unauthorized access

Some common protocols (rules) used in this layer include:

- **NetBIOS:** helps computers find and talk to each other on a local network
- **PPTP:** used for creating secure VPN connections

Layer 4: Transport Layer:

The Transport Layer acts as a bridge it receives data from the Application Layer above and passes it down to the Network Layer below. In this layer, data is called Segments (just small chunks of information).

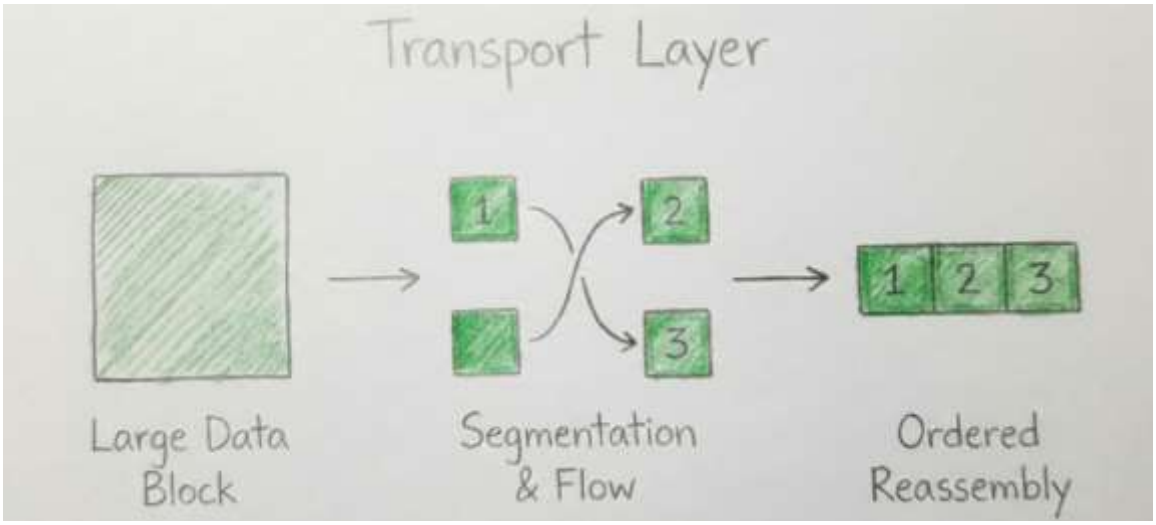
The main job of this layer is to make sure your complete message gets delivered from start to finish from your device all the way to the destination device. Think of it like a delivery service that ensures your package arrives safely and completely at the right address.

Here's what this layer does:

- **Breaks data into smaller pieces:** It divides large messages into smaller segments that are easier to send
- **Ensures reliable delivery:** It checks that all the segments arrive correctly and in the right order
- **Error checking:** If something goes wrong or gets lost along the way, this layer detects it and resends the missing pieces
- **Flow control:** It makes sure data isn't sent too fast or too slow, keeping everything smooth

Some common protocols used in this layer include:

- **TCP (Transmission Control Protocol):** ensures reliable, ordered delivery
- **UDP (User Datagram Protocol):** faster but doesn't guarantee delivery

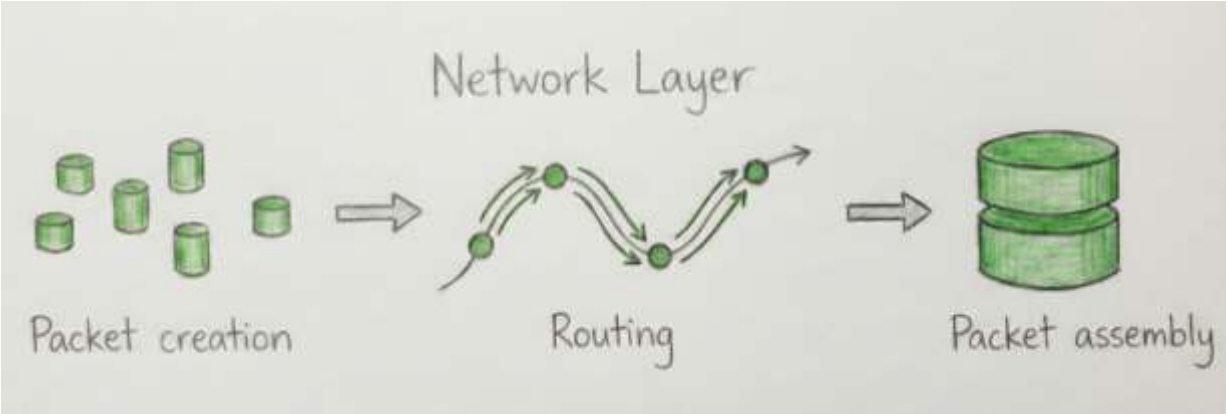


Layer 3: Network Layer

The Network Layer is responsible for moving data from one computer to another, especially when they're on different networks. Think of it like a GPS system that figures out the best route to get your data where it needs to go.

Here's what this layer does:

- **Packet routing:** It chooses the shortest and most efficient path to send your data through multiple networks. If there are several possible routes, it picks the best one, just like choosing the fastest route on a map.
- **Addressing:** This layer adds important address labels to your data. It puts both the sender's IP address and the receiver's IP address in the header (like writing a return address and destination address on an envelope). This way, the data knows where it's coming from and where it needs to go.
- **Data is called Packets:** In this layer, the data segments are now called Packets.
- **Uses network devices:** Devices like routers and switches work at this layer to direct traffic and make sure packets reach the correct destination.



Layer 2: Data Link Layer

The Data Link Layer is responsible for moving data directly from one device (node) to another that are connected nearby. Its main job is to make sure the data travels without any errors across the physical connection between two devices.

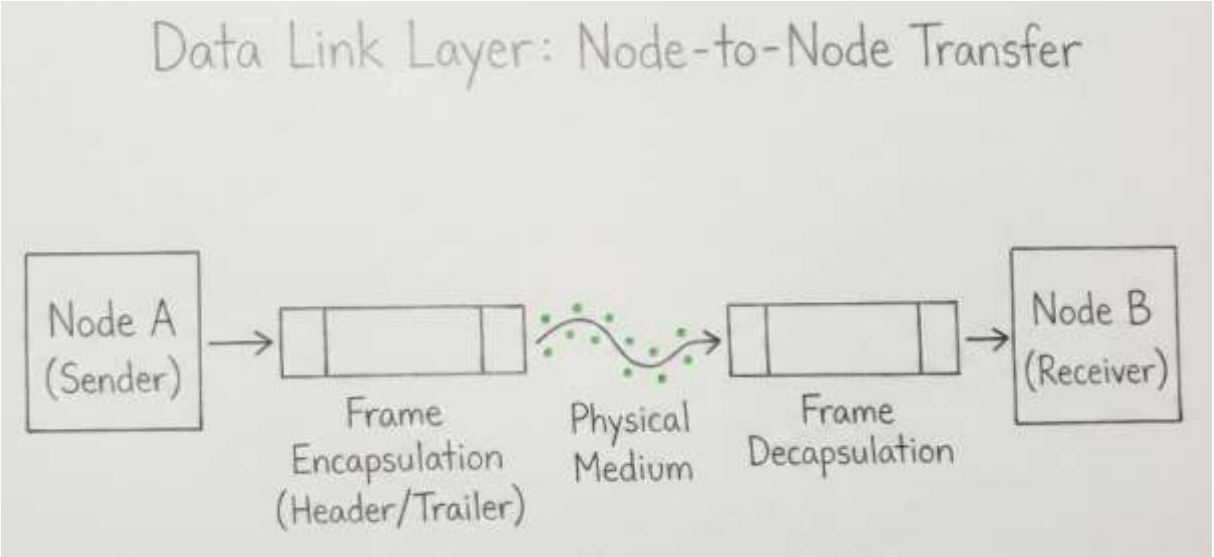
Here's what this layer does:

Error-free delivery: It checks the data carefully to make sure no mistakes happen when it moves from one device to another on the same network.

Uses MAC addresses: When data arrives at a network, this layer uses the MAC address (a unique ID for each device, like a serial number) to deliver it to the correct device or host.

Data is called Frames: In this layer, packets are now called Frames. Think of frames as data wrapped in a protective envelope for safe delivery.

Uses network devices: Common devices that work at this layer include Switches and Bridges, which help direct frames to the right place within a local network.



Layer 1: Physical Layer:

The Physical Layer is the very bottom layer of the OSI Model. It deals with the actual physical connection between devices things like cables, wires, signals, and electrical pulses. This layer handles information in its most basic form: bits (which are just 0s and 1s).

Here's what this layer does:

- **Transmits raw bits:** It sends individual bits (0s and 1s) from one device to another through physical means like cables, radio waves, or fiber optics.
- **Handles the physical connection:** This includes everything you can physically touch or see, like network cables, connectors, voltage levels, and light signals.
- **Converts signals:** When data arrives, this layer receives the electrical, light, or radio signals and converts them into 0s and 1s (digital bits). Then it passes these bits up to the Data Link Layer, which puts them back together into frames.

Common examples at this layer include:

- **Ethernet cables**
- **Wi-Fi signals**
- **USB connections**
- **Fiber optic cables**