**Name -** Jimmy George

**Email Id-** minigeorge10031976@gmail.com

**DAY 1-**

**Question 1:** Comparison between different processors: Single core, Dual core and Quad Core.

**Answer 1:** Single core, Dual core, and Quad-core processors refer to the number of processing units (cores) within an integrated CPU chip:

1. **Single Core**: A single-core processor has only one core, meaning it can only execute one instruction at a time. These processors are older and less common in modern devices because they're not as efficient in handling multiple tasks simultaneously.
2. **Dual Core**: Dual-core processors have two cores, allowing them to handle two instructions simultaneously. This increases performance compared to single-core processors, especially in multitasking scenarios where multiple applications are running simultaneously.
3. **Quad Core**: Quad-core processors have four cores, enabling them to handle four instructions at once. They provide even better performance than dual-core processors, particularly for demanding tasks such as gaming, video editing, and running virtual machines.

**Question 2: Explain different Intel I family ( i5 and i7 ) ?**

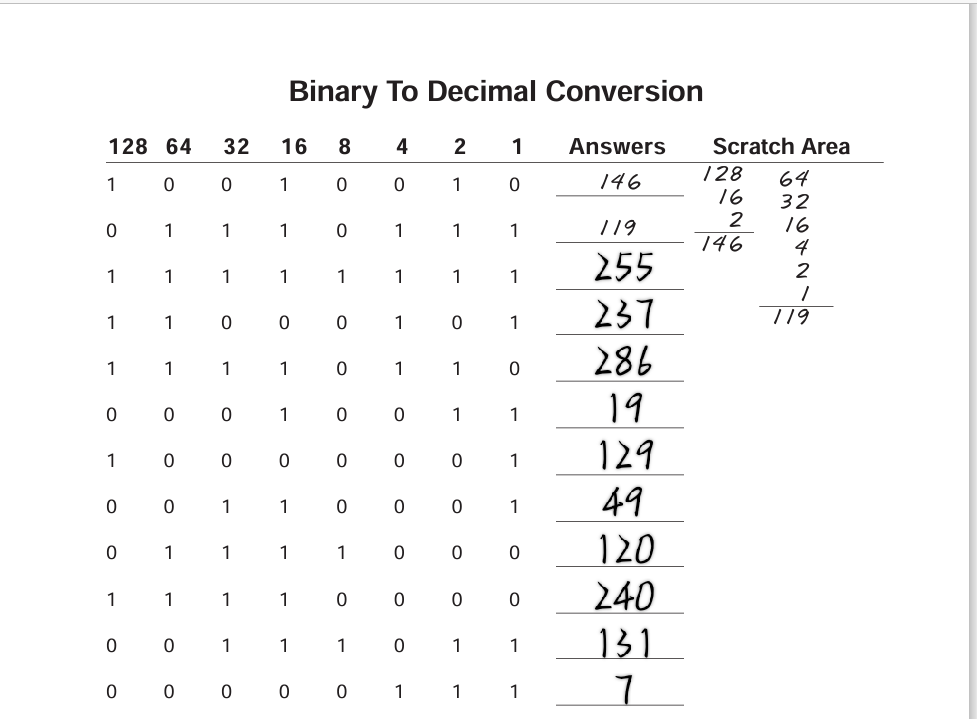
**Answer 2:** The differences between Intel Core i5 and Core i7 processors primarily lie in their performance capabilities, core counts, clock speeds, cache sizes, and additional features. Here's a breakdown of the key distinctions:

1. **Performance**: Core i7 processors generally offer higher performance compared to Core i5 processors. They often have more cores and threads, allowing them to handle more simultaneous tasks and process data more quickly.
2. **Core Count**: Core i5 processors typically have fewer cores compared to Core i7 processors. However, this isn't always the case, as some Core i5 models may have the same number of cores as certain Core i7 models.
3. **Hyper-Threading**: Many Core i7 processors feature hyper-threading technology, which allows each physical core to handle two threads simultaneously. This effectively doubles the number of threads the processor can execute, improving multitasking performance. Core i5 processors generally lack hyper-threading, though there are exceptions in certain models.
4. **Clock Speed**: Core i7 processors may have higher base and turbo boost clock speeds compared to Core i5 processors. This means they can execute instructions at a faster rate, resulting in improved performance for tasks that require high CPU frequency.
5. **Cache Size**: Core i7 processors typically have larger cache sizes compared to Core i5 processors. The cache is a small, high-speed memory bank integrated into the CPU that stores frequently accessed data and instructions, reducing latency and improving performance.
6. **Price**: Core i7 processors are generally more expensive than Core i5 processors due to their higher performance and additional features. They are often targeted at power users, enthusiasts, and professionals who require top-tier performance for demanding tasks.
7. **Use Case**: Core i7 processors are well-suited for demanding tasks such as gaming, video editing, 3D rendering, software development, and running virtual machines. Core i5 processors, while still capable of handling these tasks, may offer slightly lower performance and may be more suitable for mainstream computing needs like office work, web browsing, and light multitasking.

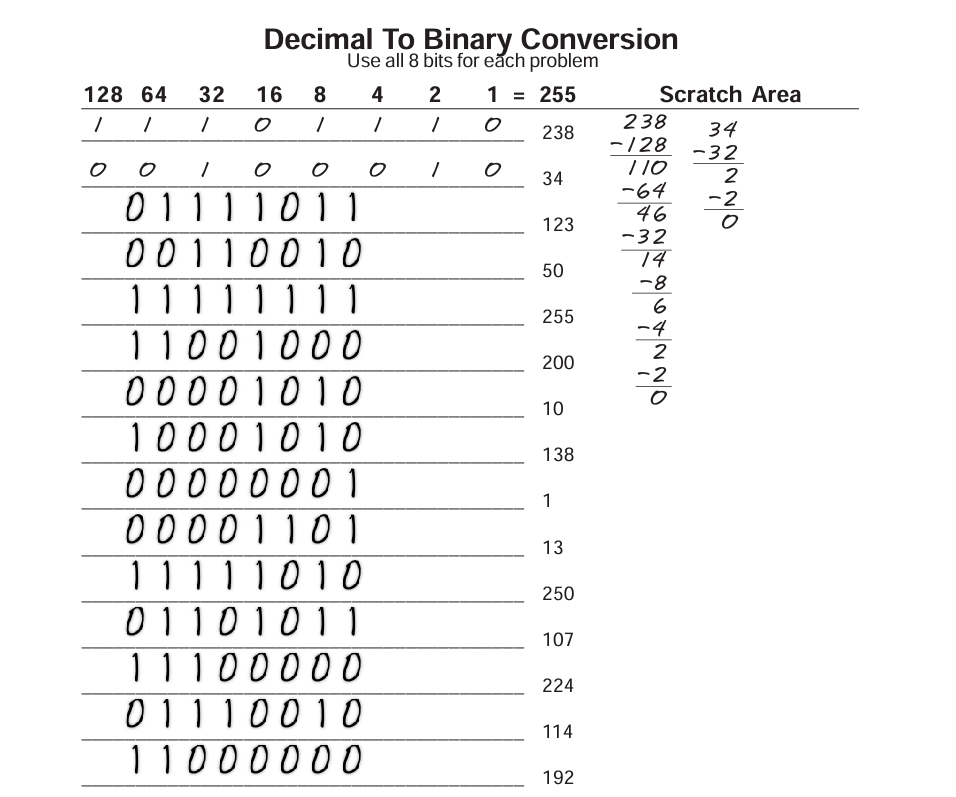
In summary, Core i7 processors offer higher performance, more cores, higher clock speeds, larger cache sizes, and additional features like hyper-threading compared to Core i5 processors. However, the specific performance difference between individual models within each series can vary, so it's essential to consider the specific requirements of your computing tasks when choosing between them.

**DAY 2-**

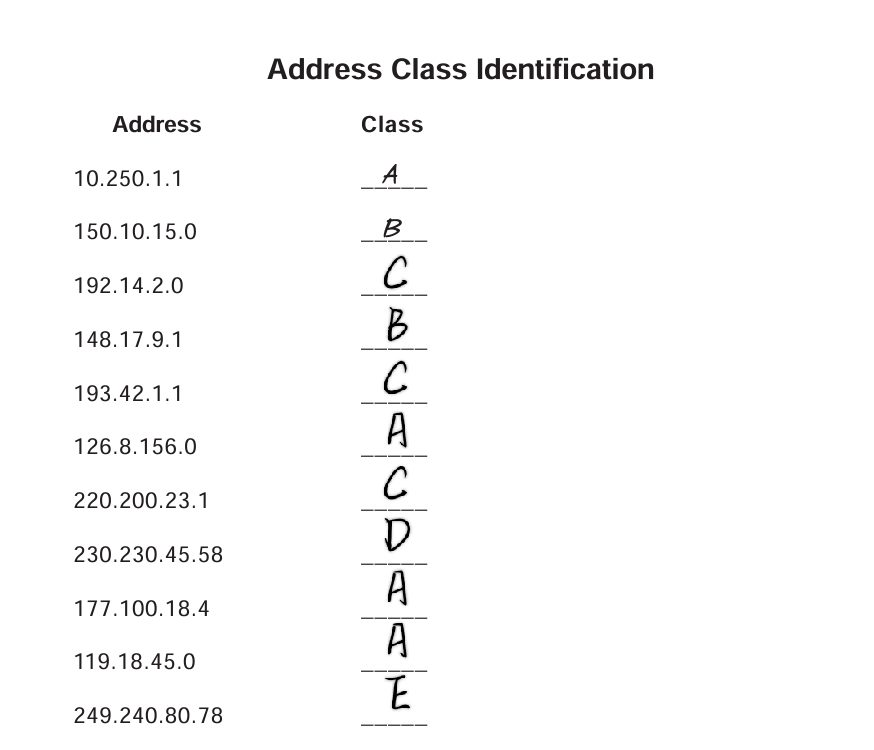
1. **Binary to Decimal conversion.**

****

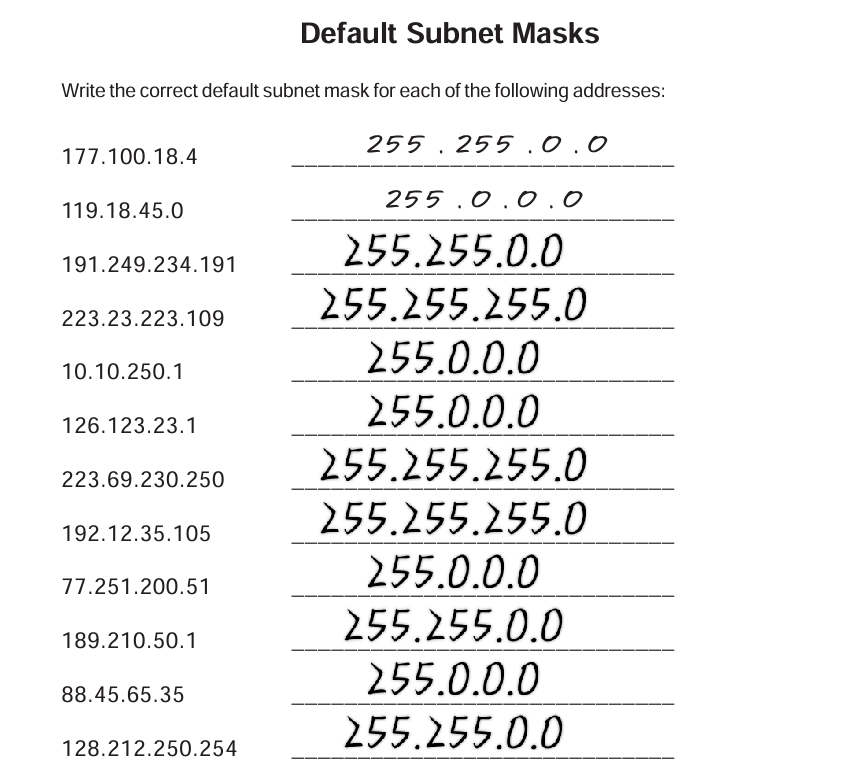
1. **Decimal to Binary conversion.**

****

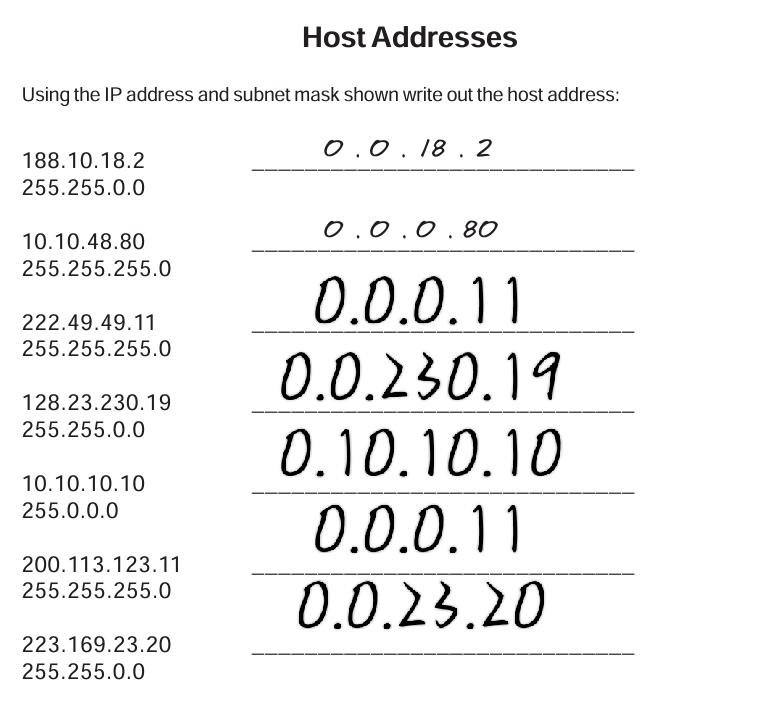
1. **Address Class Identification**

****

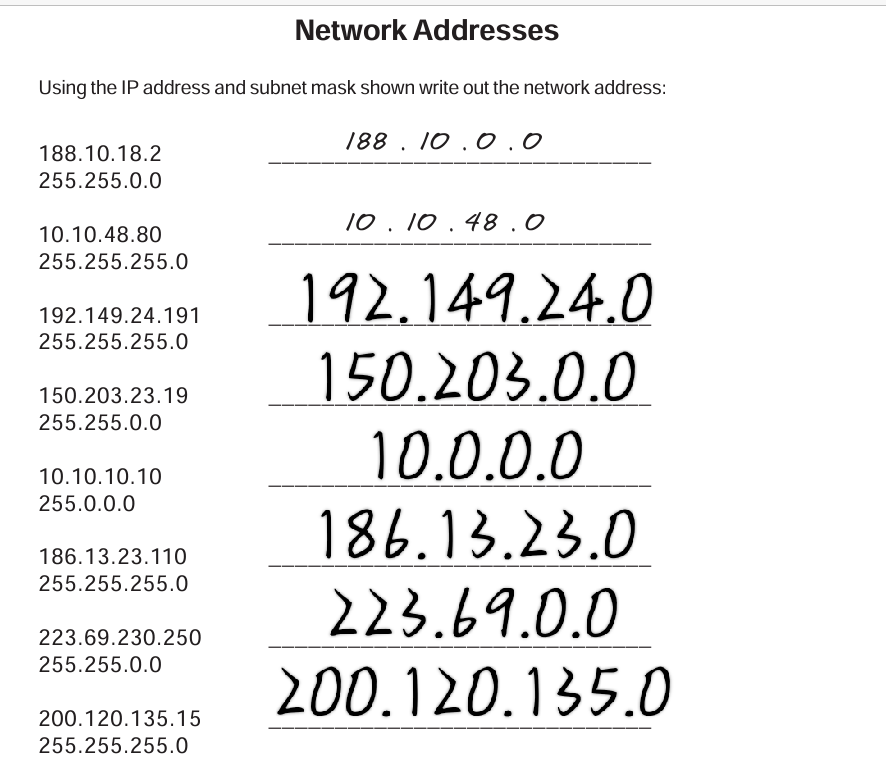
1. **Default Subnet Mask**

****

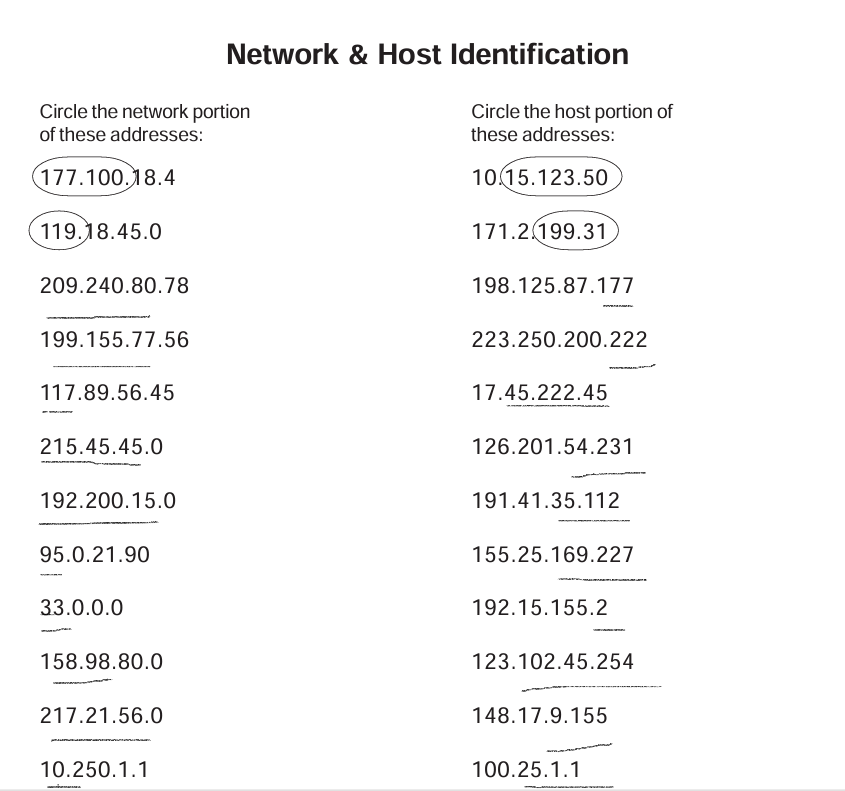
1. **Host Addresses**

****

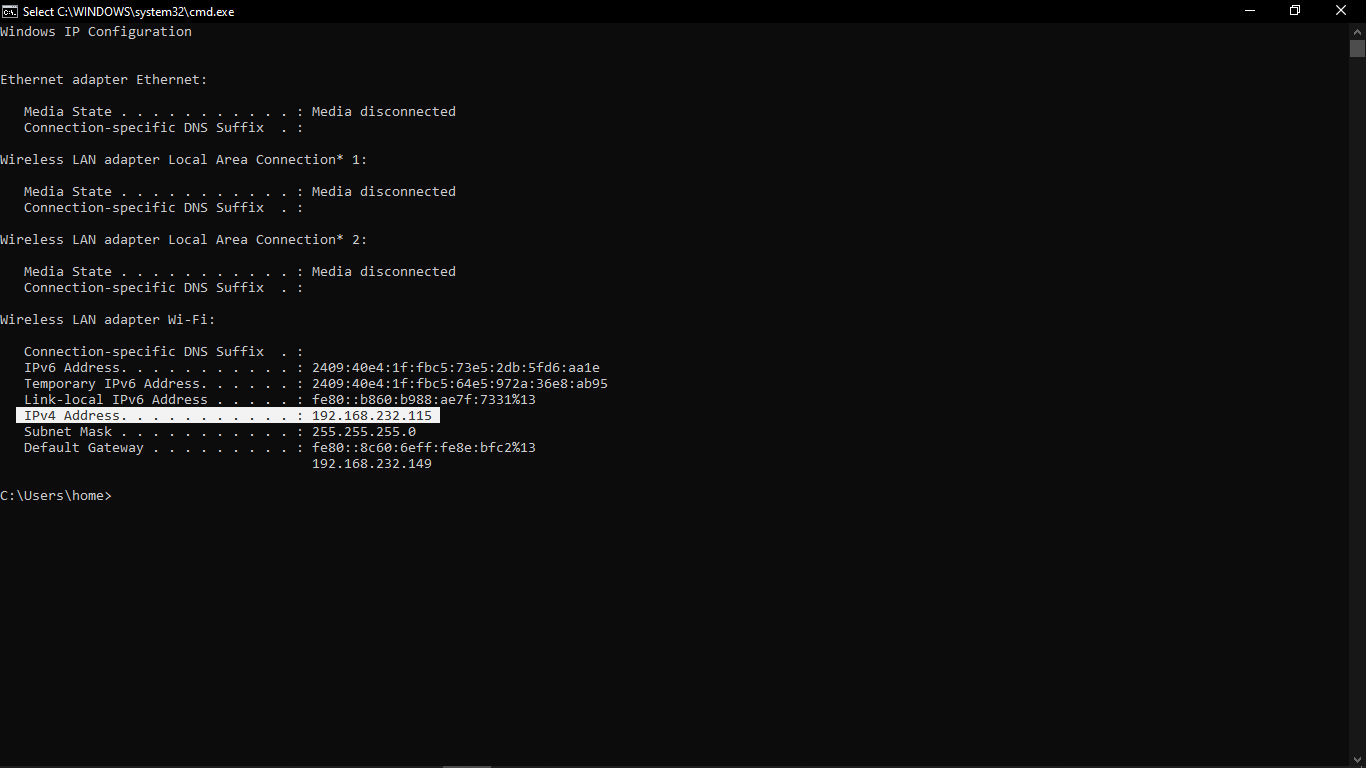
1. **Network Address**

****

1. **Network & Host Identification**

****

1. **Obtain IPv4 Address from cmd panel in PC.**

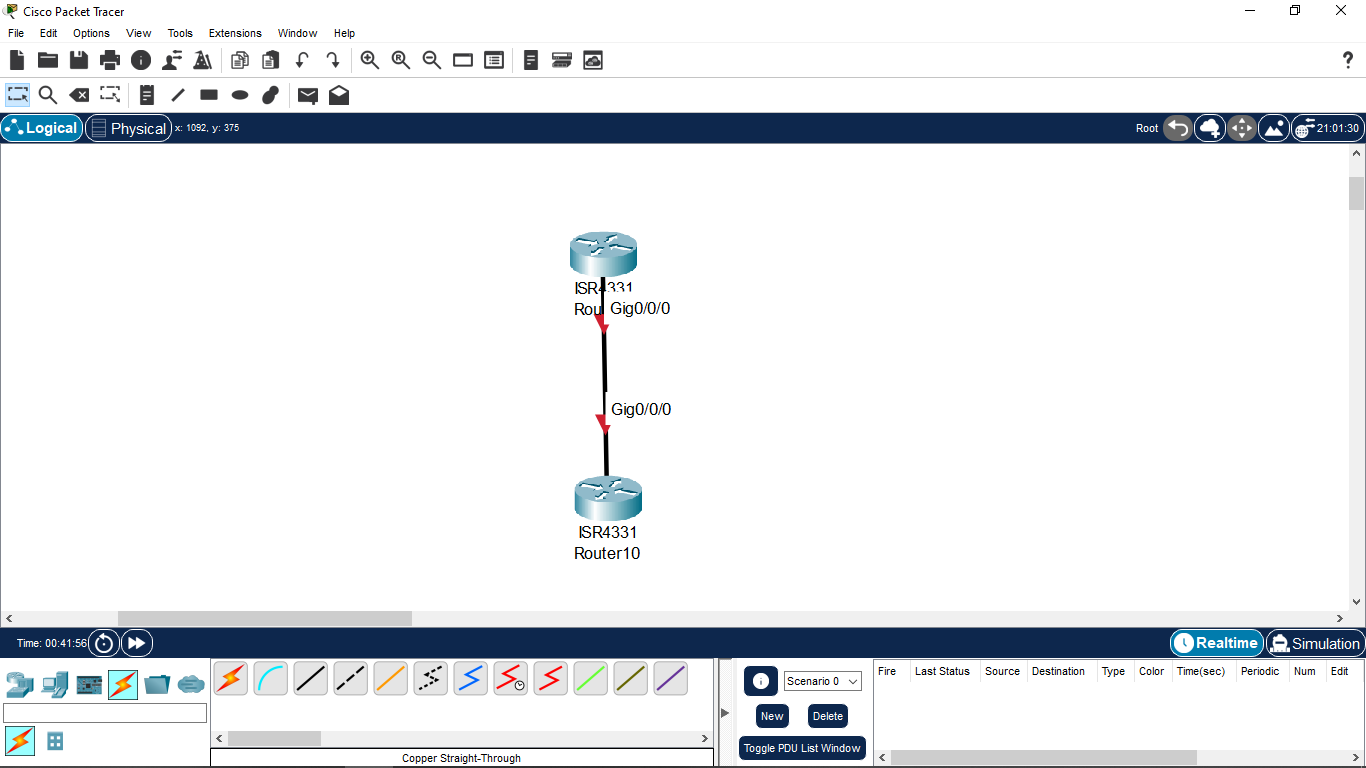
****

**DAY 3 –**

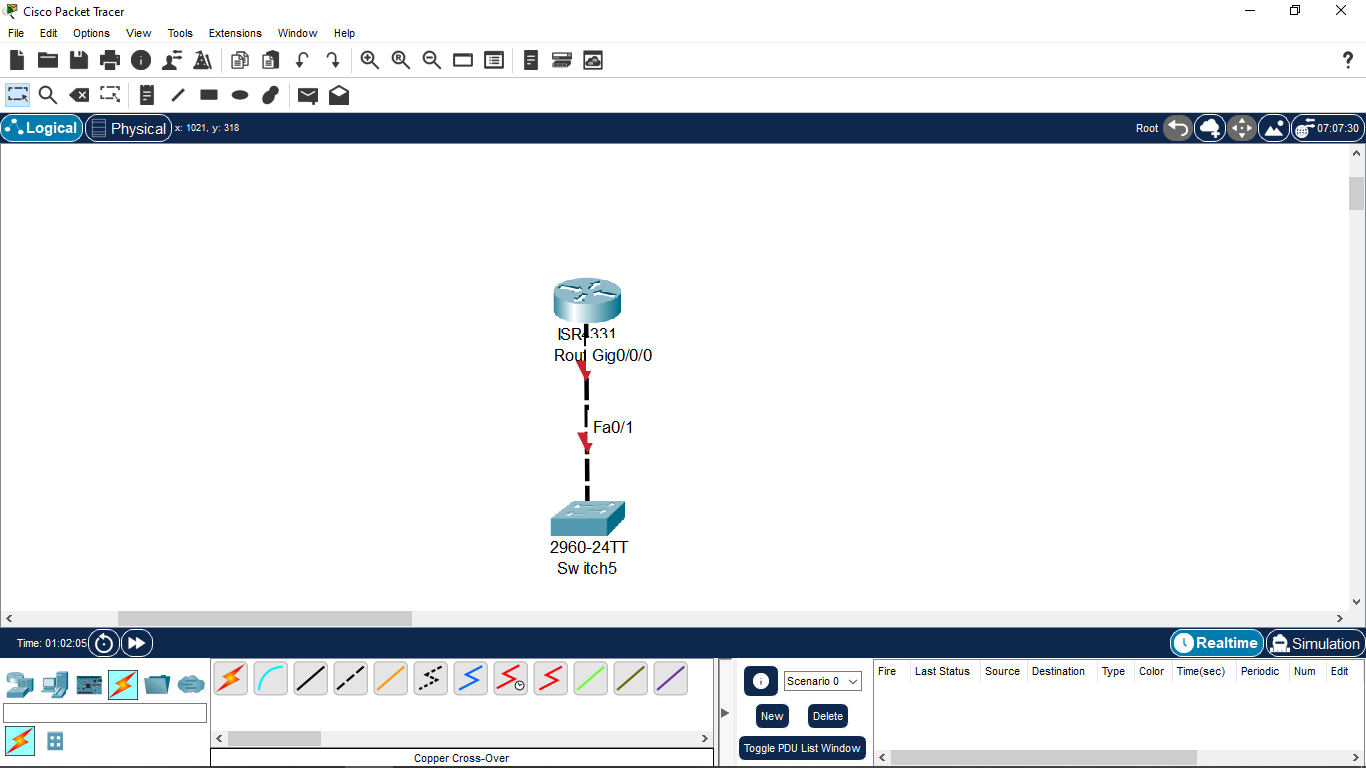
**LAB 1**

**Activity - Lab Activities with CISCO PACKET TRACER.**

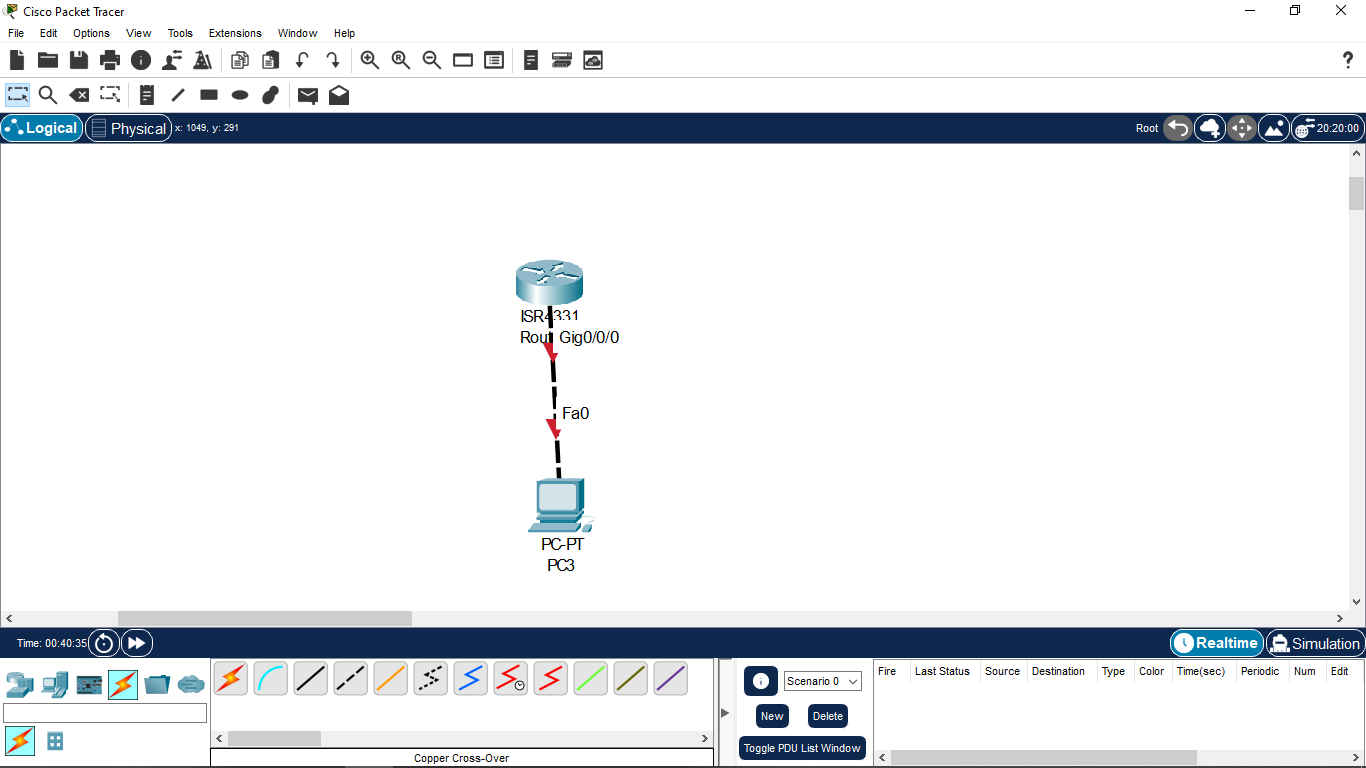
1. **Connect Two Routers.**

****

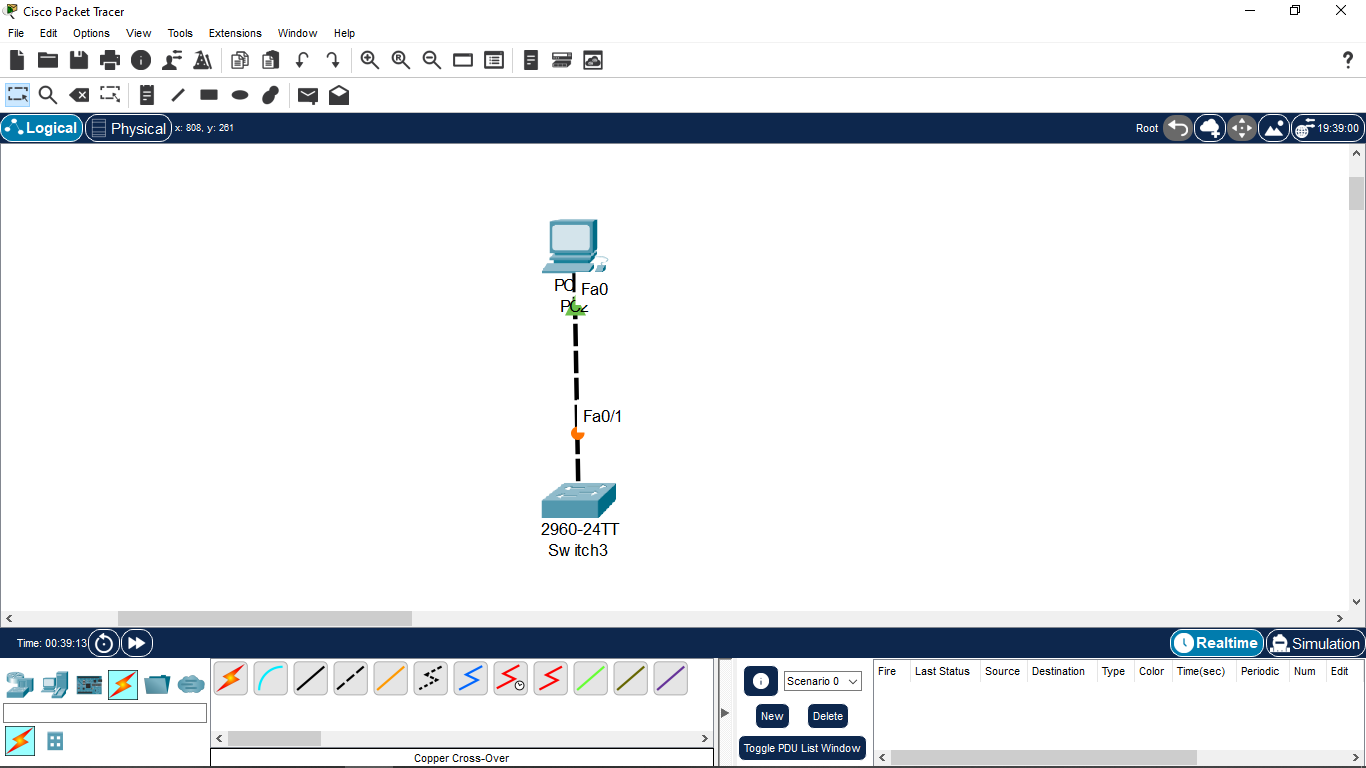
1. **Connect Router with Switch.**

****

1. **Connect Router with PC.**

****

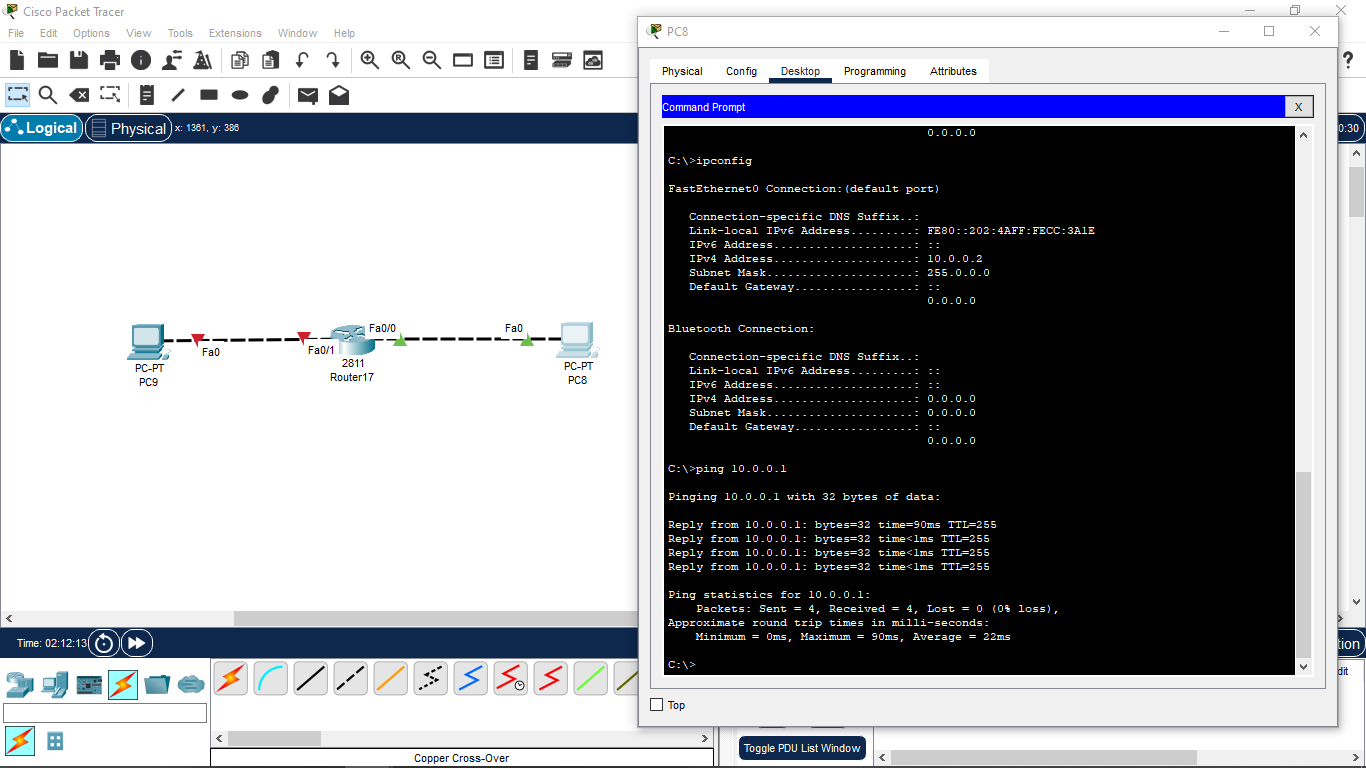
1. **Connect PC with Switch.**

****

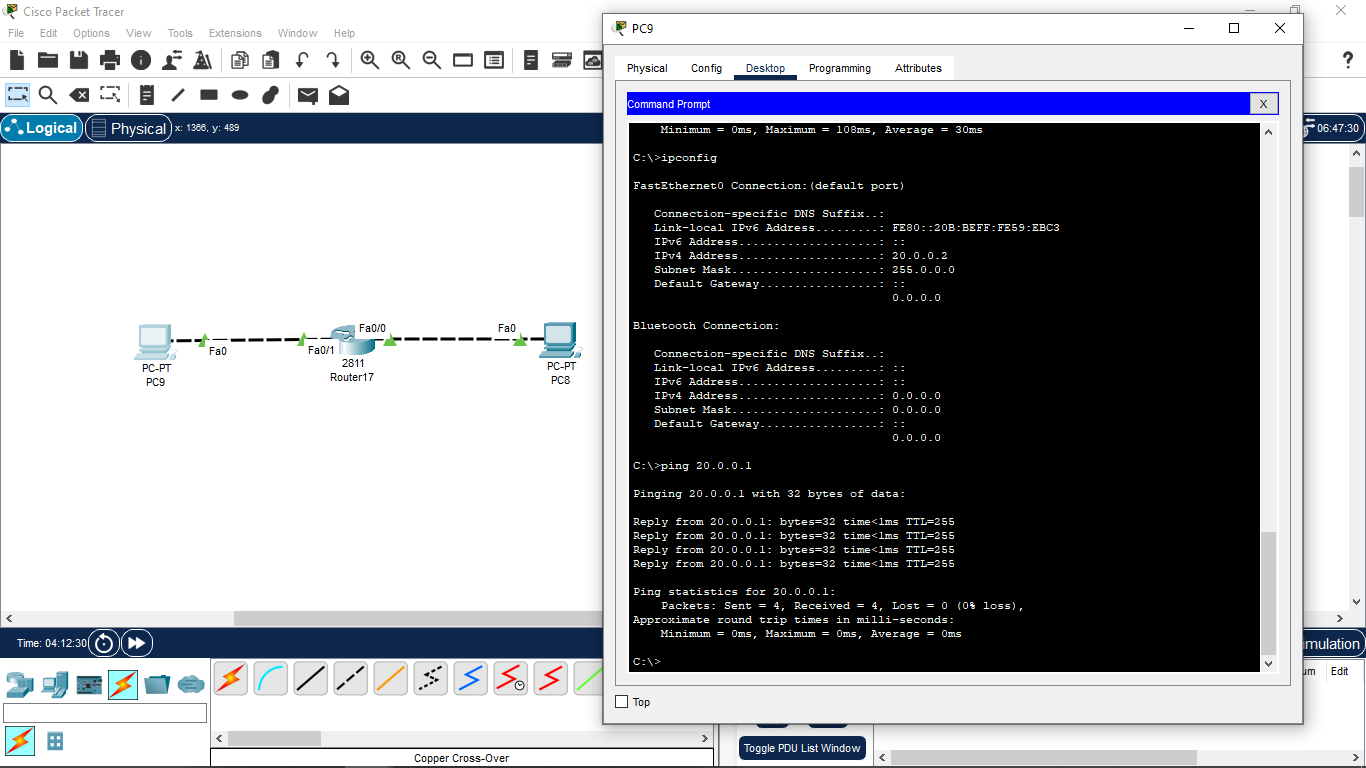
**LAB 2**

1. **Configuration of IP Address. [ Router - > PC ]**

**(i) From Router 2811 to PC 8.**

****

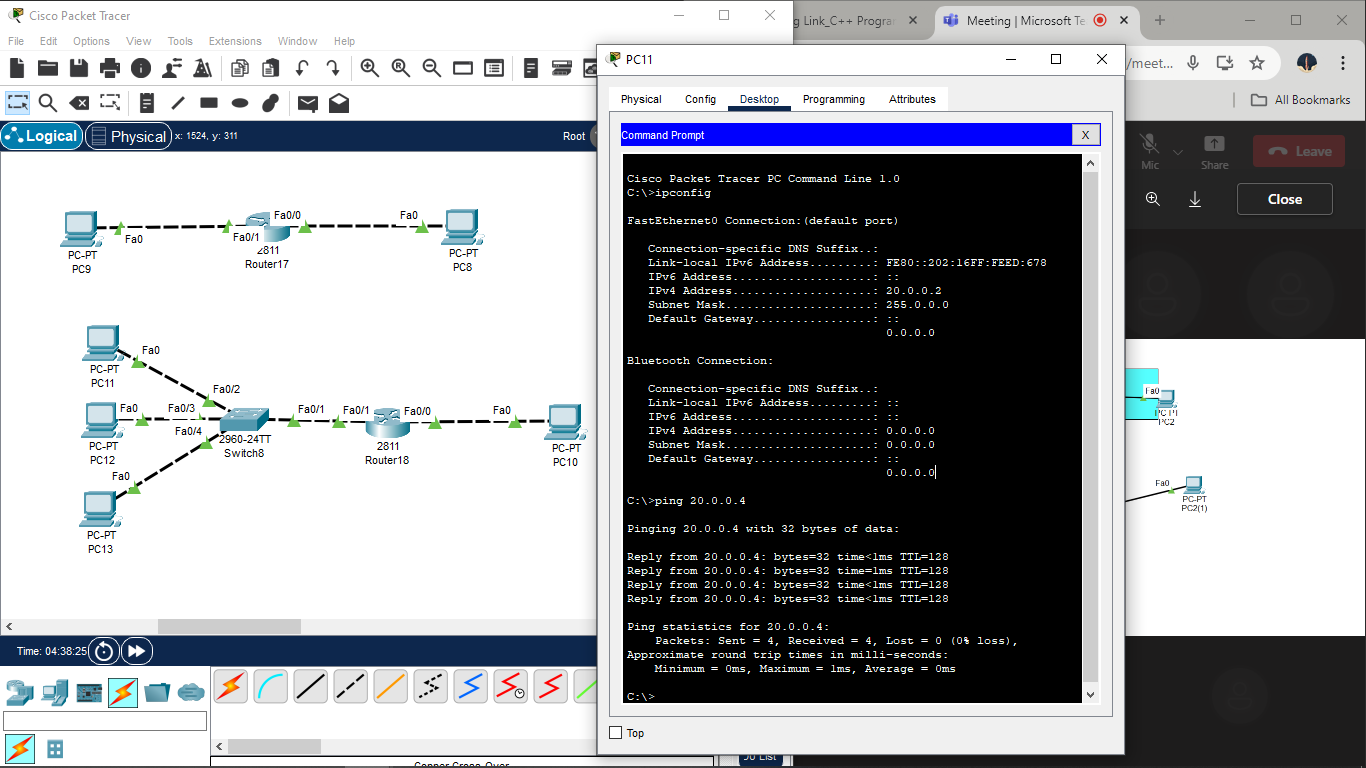
**(ii) From Router 2811 to PC 9.**

****

**DAY 4 –**

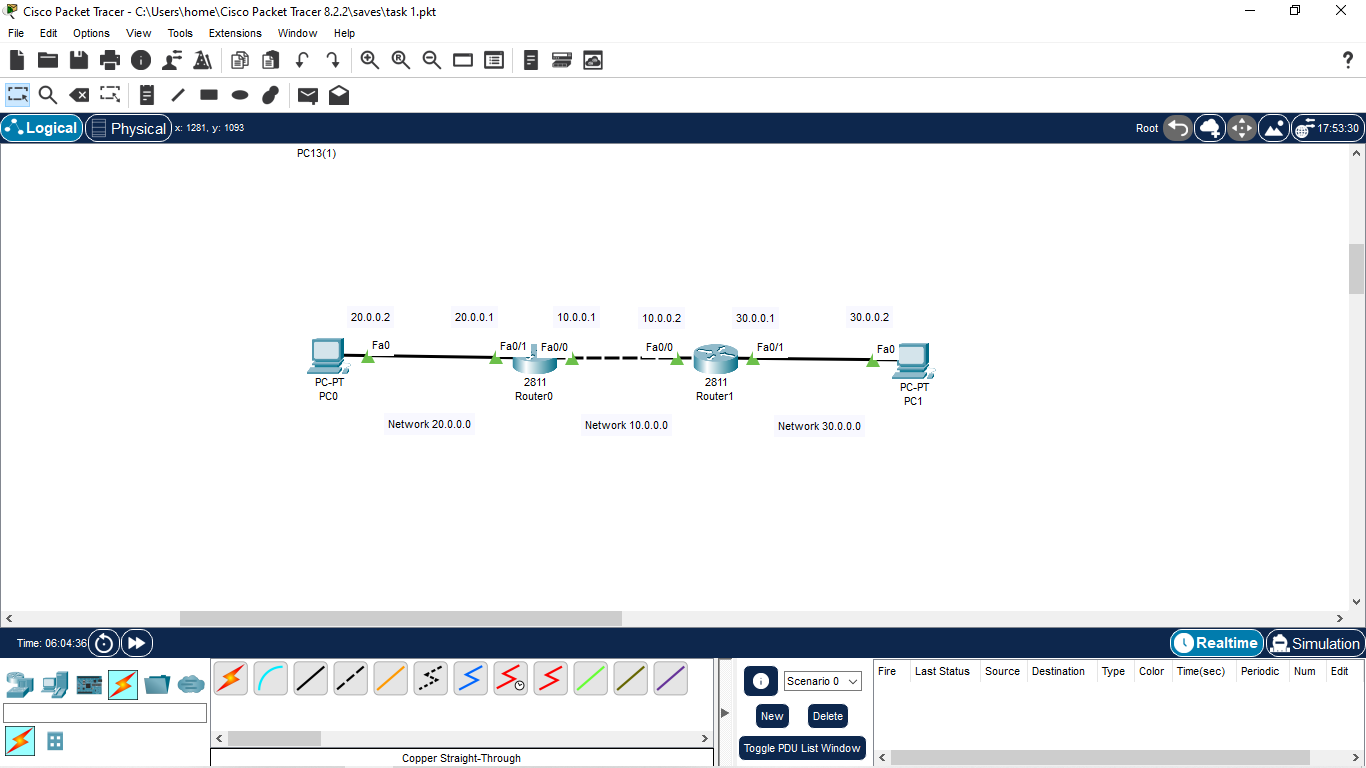
**LAB 3**

1. **Configuration of IP Address to multiple PC’s within network. [ Router - > PC ]**
2. **From Router to Multiple PC’s connecting to same networks’ IP Address.**

****

**LAB 4**

1. **Prepare an IP Configuration Table for below PC 0 & PC 1.**

****

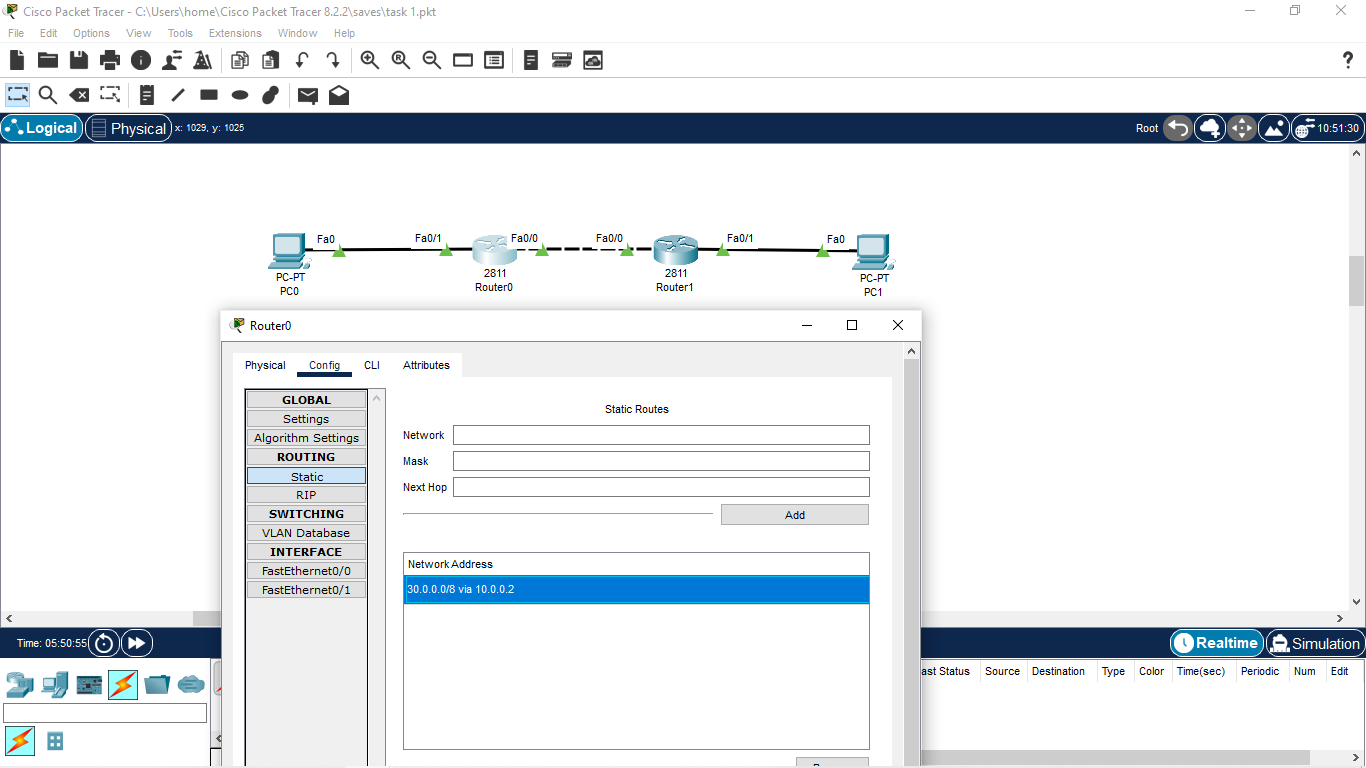
**Table –**

|  |  |  |  |
| --- | --- | --- | --- |
| **Source** | **Destination** | **Ping Status** | **Directory** |
| **PC 0** | Fa0 | Working | Access **IP Address 20.0.0.2** |
|  | Fa0/1 | Working | Access **IP Address 20.0.0.1** |
|  | Fa0/0  ( Router 0) | Working | Access **IP Address 10.0.0.1**  **( Default Gateway Access**  **[ IP Address – 10.0.0.1 ] )** |
|  | Fa0/0  ( Router 1) | Working | Access **IP Address 10.0.0.2**  **( After Routing )** |
|  | Fa0/1  ( Router 1 ) | Working | Access **IP Address 30.0.0.1**  **( After Routing )** |
|  | Fa0  ( Router 1 ) | Working | Access **IP Address 30.0.0.2**  **( After Routing )** |
| **PC 1** | Fa0 | Working | Access **IP Address 30.0.0.2** |
|  | Fa0/1 | Working | Access **IP Address 30.0.0.1** |
|  | Fa0/0  ( Router 1) | Working | Access **IP Address 10.0.0.2**  **( Default Gateway Access**  **[ IP Address – 10.0.0.2] )** |
|  | Fa0/0  ( Router 0 ) | Working | Access **IP Address 10.0.0.1**  **( After Routing )** |
|  | Fa0/1  ( Router 0 ) | Working | Access **IP Address 20.0.0.1**  **( After Routing )** |
|  | Fa0  ( Router 0 ) | Working | Access **IP Address 20.0.0.2**  **( After Routing )** |

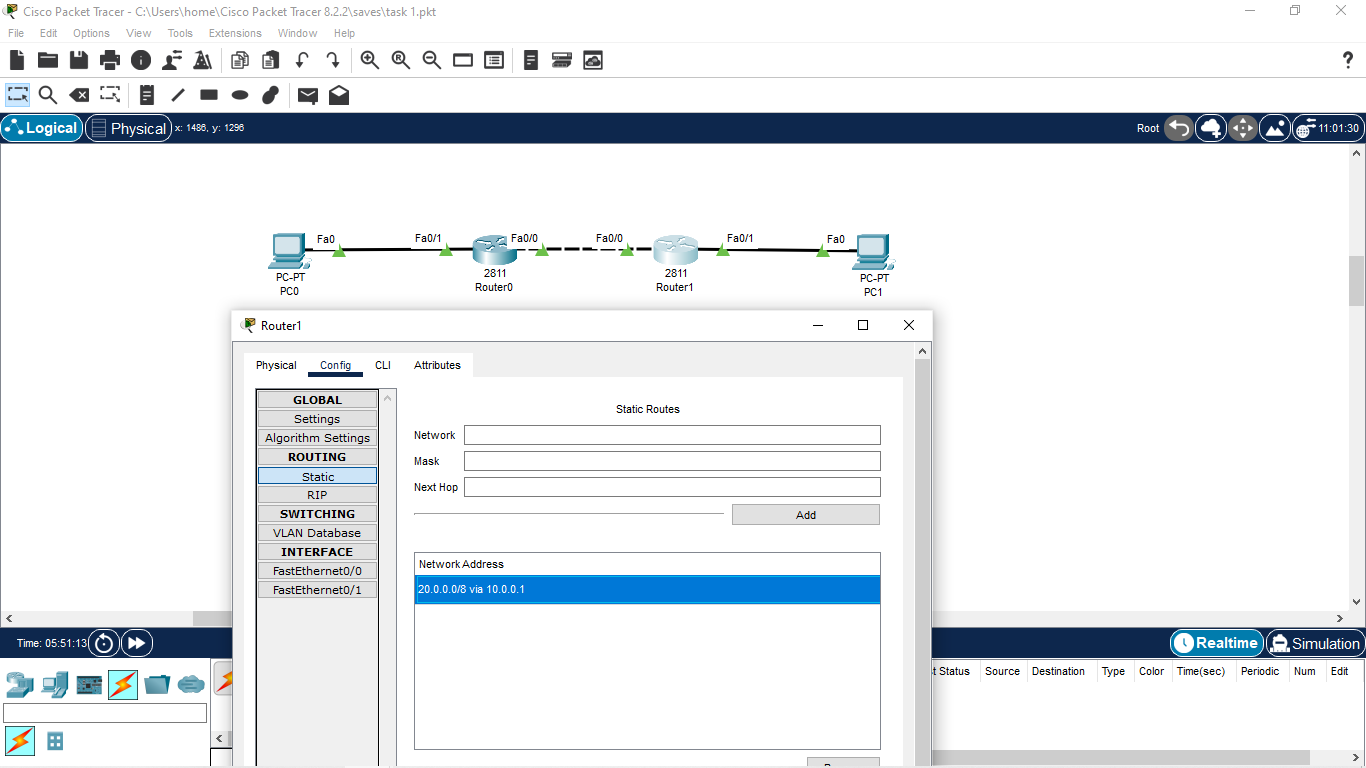
**Those IP’s that were not accessible earlier are possible now due the concept of Routing.**

**Routing Steps in Cisco Packet Tracer:**

1. **For Router 0 (PC 0) -**

****

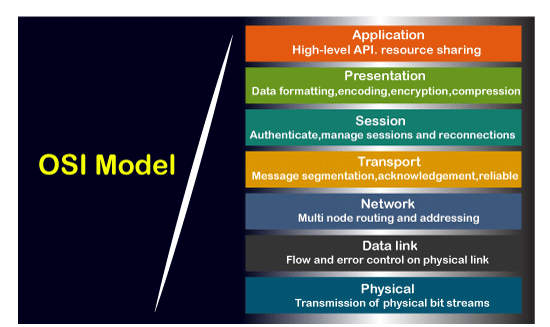
1. **For Router 1 (PC 1) –**

****

**DAY 5 –**

**Question 1: Explain Different Layers of OSI Model/Protocol.**

**Answer 1:** The OSI (Open Systems Interconnection) model is a conceptual framework used to understand and describe network communication between devices. It consists of seven layers, each responsible for specific functions in the process of transmitting data across a network. Here’s an overview of each layer:

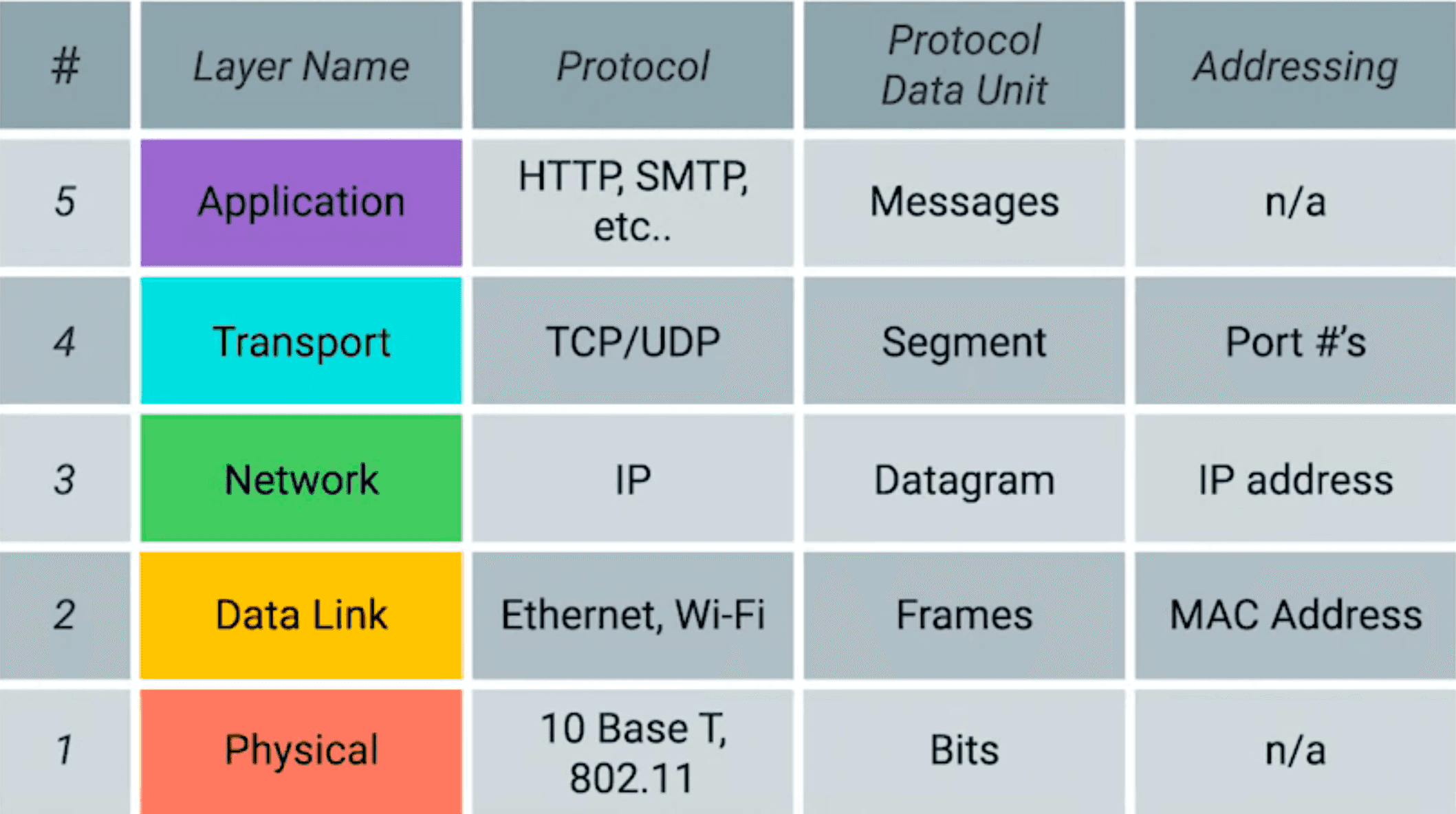


1. **Physical Layer (Layer 1)**:
   1. **Function**: This layer deals with the physical connection between devices. It defines the electrical, mechanical, and procedural aspects of data transmission.
   2. **Examples**: Ethernet cables, voltage levels, connectors.
2. **Data Link Layer (Layer 2)**:
   1. **Function**: Responsible for node-to-node communication, ensuring data packets are delivered error-free. It handles framing, error detection, and flow control.
   2. **Examples**: MAC addresses (e.g., Ethernet MAC), switches.
3. **Network Layer (Layer 3)**:
   1. **Function**: Manages logical addressing and routing of data packets between different networks. It determines the optimal path for data to travel from the source to the destination.
   2. **Examples**: IP addresses (e.g., IPv4, IPv6), routers.
4. **Transport Layer (Layer 4)**:
   1. **Function**: Provides end-to-end communication, ensuring data packets are delivered reliably and in sequence. It handles error detection & recovery and flow control.
   2. **Examples**: TCP (Transmission Control Protocol), UDP (User Datagram Protocol).
5. **Session Layer (Layer 5)**:
   1. **Function**: Establishes, manages, and terminates sessions between applications. It controls the dialogues (sessions) between computers.
   2. **Examples**: NetBIOS, RPC (Remote Procedure Call).
6. **Presentation Layer (Layer 6)**:
   1. **Function**: Ensures that data sent from the application layer of one system can be read by the application layer of another. It deals with data translation, encryption, and compression.
   2. **Examples**: JPEG, MPEG, SSL (Secure Sockets Layer).
7. **Application Layer (Layer 7)**:
   1. **Function**: Provides an interface between the user applications and the network. It supports communication services for applications and end-user processes.
   2. **Examples**: HTTP, FTP, DNS.

These layers collectively provide a standardized way for different systems to communicate effectively over networks. Each layer builds upon the services provided by the layer below it, adding more abstraction and functionality as you move up the stack. This division of functions helps in designing, understanding, and troubleshooting complex network systems.

**Question 2: Explain Different Layers of TCP/IP Model/Protocol.**

**Answer 2:** The TCP/IP (Transmission Control Protocol/Internet Protocol) model, unlike the OSI model, consists of four layers. These layers define the protocols used for communication over the Internet and are fundamental to modern networking. Here's an explanation of each layer in the TCP/IP protocol suite:



1. **Application Layer**:
   1. **Function**: The application layer is responsible for providing network services directly to end-users or applications. It enables communication between applications running on different devices.
   2. **Examples**: HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), DNS (Domain Name System).
2. **Transport Layer**:
   1. **Function**: The transport layer ensures end-to-end communication between devices. It segments and reassembles data into packets, provides error checking mechanisms, and manages flow control to ensure reliable data delivery.
   2. **Protocols**: TCP (Transmission Control Protocol) provides reliable, connection-oriented communication with error-checking and retransmission capabilities. UDP (User Datagram Protocol) offers a connectionless, unreliable transport service suitable for applications where speed is more critical than reliability.
3. **Internet Layer** (also known as Network Layer in the OSI model):
   1. **Function**: The internet layer handles addressing, routing, and packaging of data packets for transmission across networks. It determines the best path for data to travel from the source to the destination across multiple networks.
   2. **Protocols**: IP (Internet Protocol) is the primary protocol used in this layer, providing logical addressing (IPv4 and IPv6) to uniquely identify devices on a network and facilitating packet forwarding by routers.
4. **Link Layer** (also known as Network Interface Layer or Network Access Layer in some references):
   1. **Function**: The link layer is responsible for the physical transmission of data over a specific medium (e.g., Ethernet, WI-Fi). It defines protocols for accessing the physical network, transmitting data frames, and error detection at the bit level.
   2. **Examples**: Ethernet, WI-Fi (IEEE 802.11), PPP (Point-to-Point Protocol).

**Question 3: Explain the Difference between OSI and TCP/IP Model.**

**Answer 3:-**

| **Parameters** | **OSI Model** | **TCP/IP Model** |
| --- | --- | --- |
| **Full Form** | OSI stands for Open Systems Interconnection. | TCP/IP stands for Transmission Control Protocol/Internet Protocol. |
| **Layers** | It has 7 layers. | It has 4 layers. |
| **Usage** | It is low in usage. | It is mostly used. |
| **Approach** | It is vertically approached. | It is horizontally approached. |
| **Delivery** | Delivery of the package is guaranteed in OSI Model. | Delivery of the package is not guaranteed in TCP/IP Model. |
| **Replacement** | Replacement of tools and changes can easily be done in this model. | Replacing the tools is not easy as it is in OSI Model. |
| **Reliability** | It is less reliable than TCP/IP Model. | It is more reliable than OSI Model |

In summary, while both models provide frameworks for understanding network communication, the **OSI model** is more theoretical and structured, aiming for standardization, whereas the **TCP/IP model** is more practical and has been widely adopted in real-world networking, particularly in the context of the Internet.

1. **Different Port Numbers in Application Layer –**

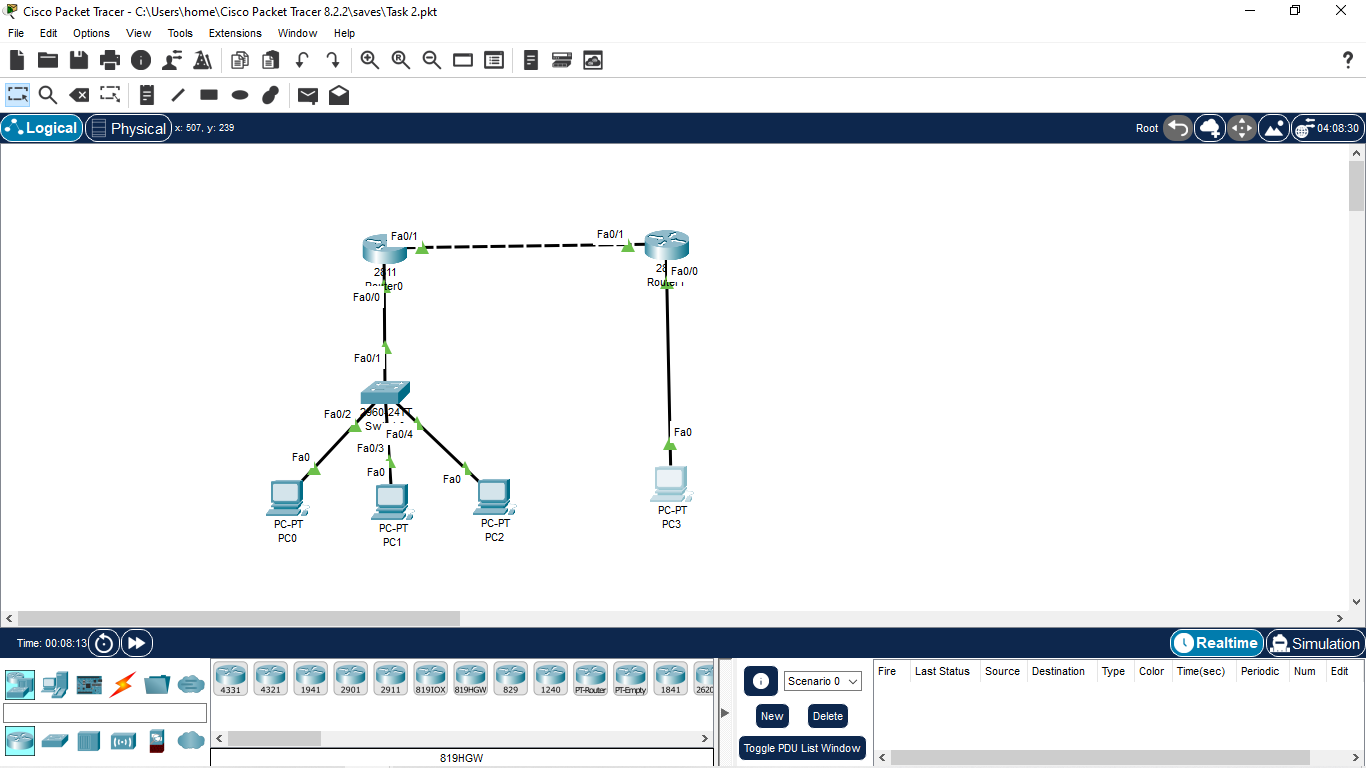
|  |  |  |
| --- | --- | --- |
| **Application** | **Protocol** | **Port Number** |
| HTTP | TCP/UDP | 80 |
| FTP DATA / FTP CONTROL | TCP/UDP | 20 , 21 |
| SSH | TCP/UDP | 22 |
| Telnet | TCP/UDP | 23 |
| SMPT | TCP | 25 |
| DNS | TCP/UDP | 53 |
| HTTPS | TCP/UDP | 443 |
| SMPTS | TCP | 465 |
| RDP **(Remote Desktop Protocol)** | TCP/UDP | 3389 |

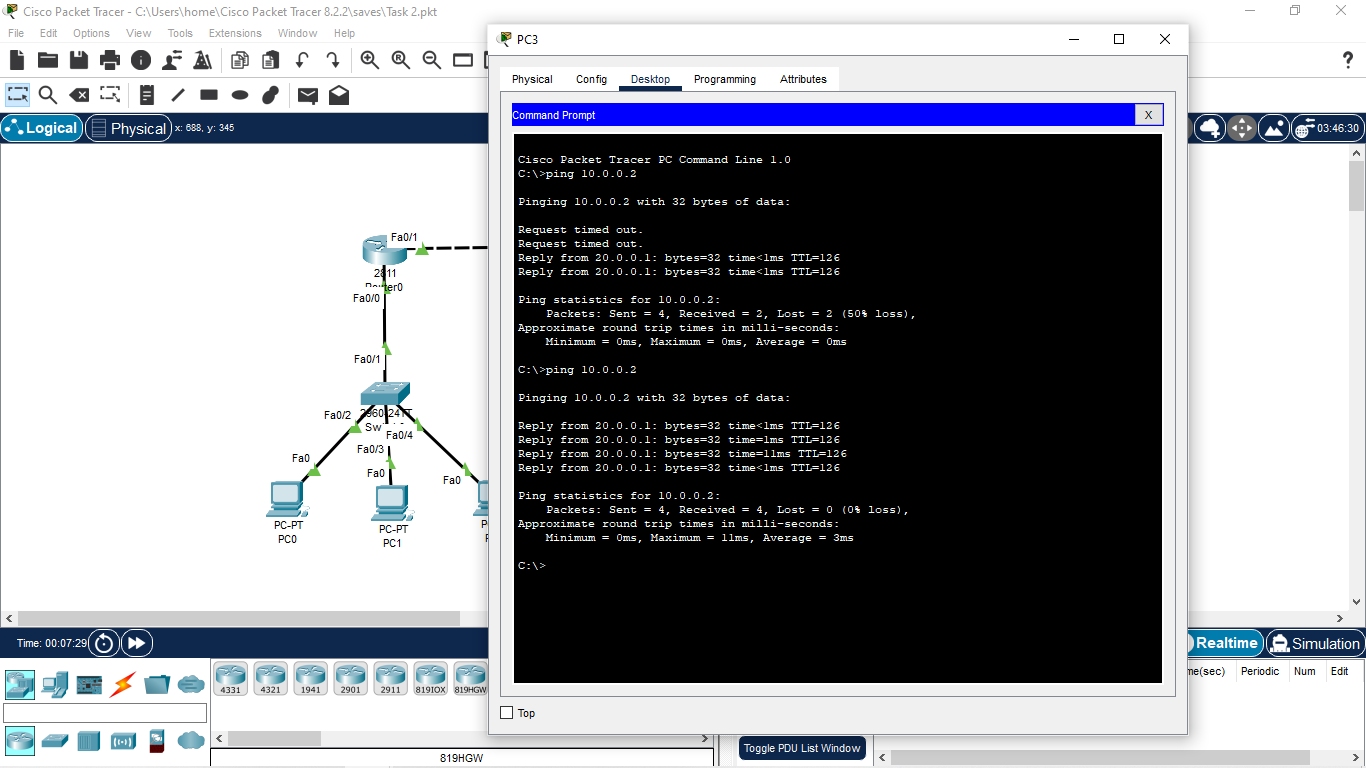
**Question 4: Identify the layer which they begin.**

1. **YouTube link (** [**www.youtube.com**](http://www.youtube.com) **)** – Layer 7 ( Application Layer ).
2. **Email** – Layer 7 (Application Layer) since it uses SMTP protocol.
3. **Taking Remote Access** - Layer 7 (Application Layer uses RDP-Remote Desktop Protocol, SSH -Secure Shell).
4. **D.N.S** – If we need to access and web-based application/links it will uses Layer 7 but if we need to **ping** web address we just need to go with Layer 3 which in turn for IP will trace back to Layer 7(DNS).

**LAB 5**

1. **Enable the concept of NAT (Network Address Translation).**

****

****

Top of Form

Bottom of Form