**Task 1:**

**What do you understand by SDLC?**

SDLC stands for Software development life cycle. It is a structured or well-defined process used by software development teams to design, develop (build) and maintain software. It is a process of breaking down software development process into various steps or stages to ensure delivery of high-quality software.

**Task 2:**

**Why is SDLC?**

As we know SDLC is a structured process of working on a software. It helps the software teams to build software with Improved quality, identifying and managing risks early, plan the project with a limited budget, improve customer satisfaction etc.

Structured Approach: the phases are well structured

Improved quality: It helps in achieving higher quality of software

Mitigating risks early: It helps in identifying ang mitigating the risks early in the project

Working in the budget: It helps in completing the project in a budget

**Task 3:**

**What are the stages of SDLC ? write 2 lines about each.**

Different stages of SDLC are:

1. **Planning and Requirement analysis**: In this stage, we define the project scope, objectives, customer requirements, calculating cost and budget, making scheduling and resource planning.
2. **Design**: Team then works on creating a detailed design of the software, database, interface, architecture.
3. **Coding or implementation**: In this stage the team starts working on the software (code) based on the customer requirements and scope
4. **Testing**: This stage involves testing the software thoroughly to identify bugs, issues or any functional errors in the code. They use different testing techniques.
5. **Deployment**: After passing the testing stage, the code is now deployed and is available for the customer to use.
6. **Maintenance**: Post deployment software is continuously maintained to ensure its working properly.

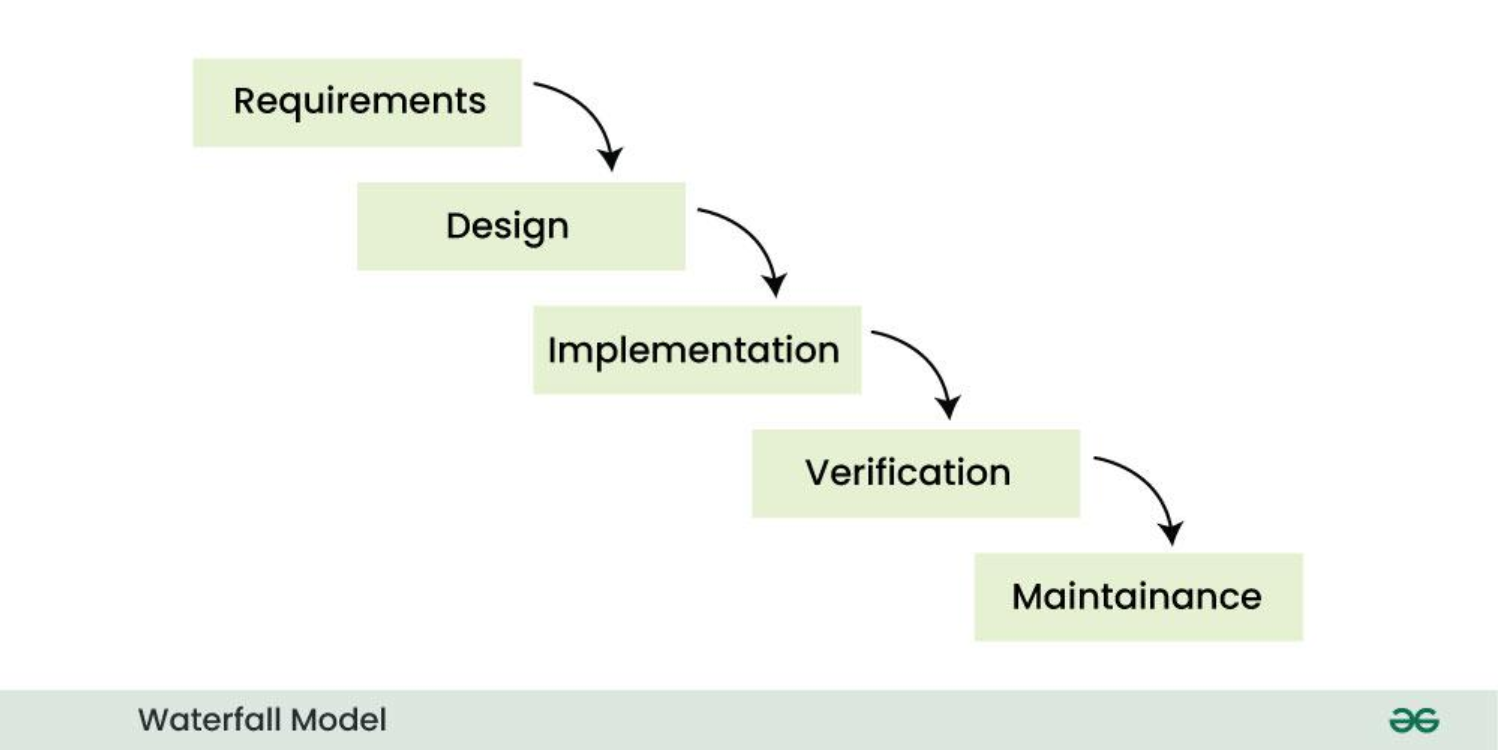
**Task 4:**

**SDLC Models: Def, applications & advantages, disadvantages**

**List them, description - 4 lines min and with a image:**

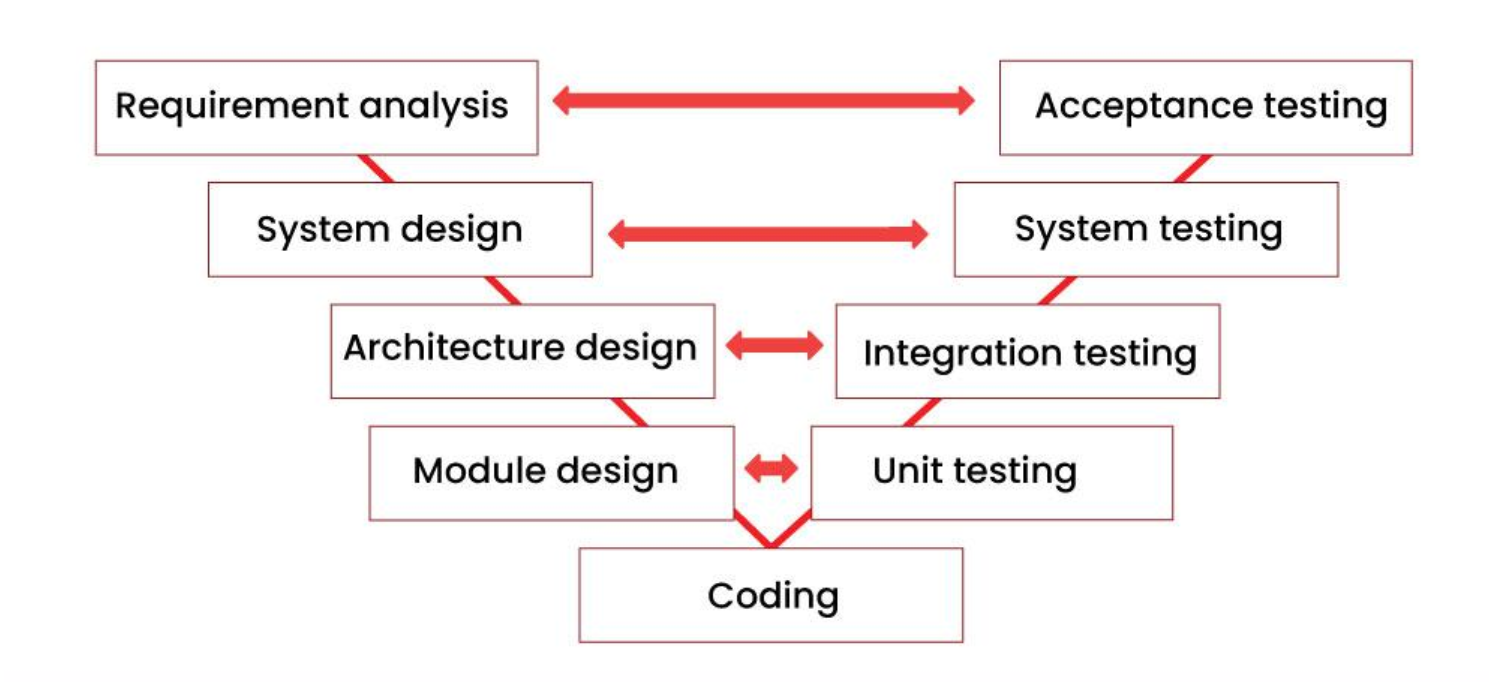
Different models of SDLC:

1. **Waterfall model**: It is a sequential model as it follows a strict linear. We cannot start next phase until the previous phase is completely completed.



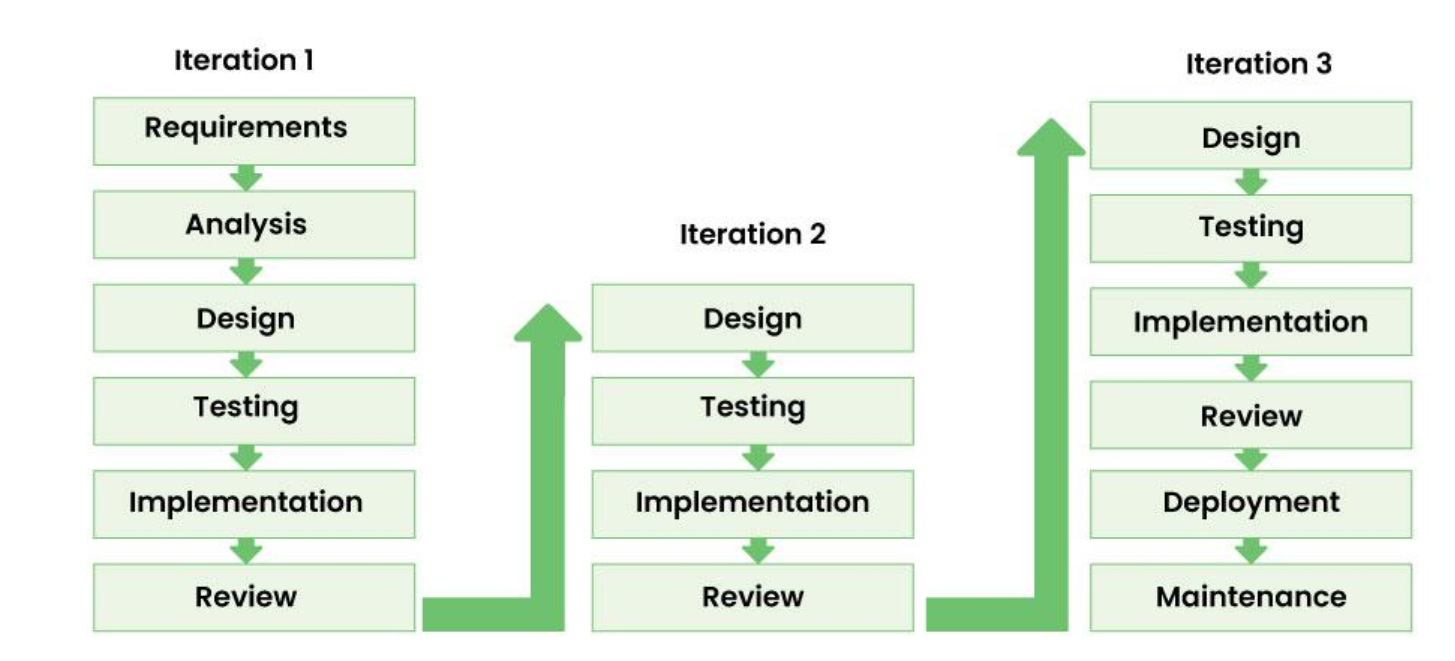
* 1. Applications: It is used when we have a clear and fixed requirement, short & simple project & stable environment.
  2. Pros: Clear & defines process, predictable timelines, easy documentation, focus on upfront planning.
  3. Cons: Due to its nature, its inflexible to changes, there is potential for errors going unnoticed, not ideal for complex projects.

1. **V- Model (Verification & Validation model):** In this model, the execution of each phase is sequential, new phase starts only after the previous phase ends.

****

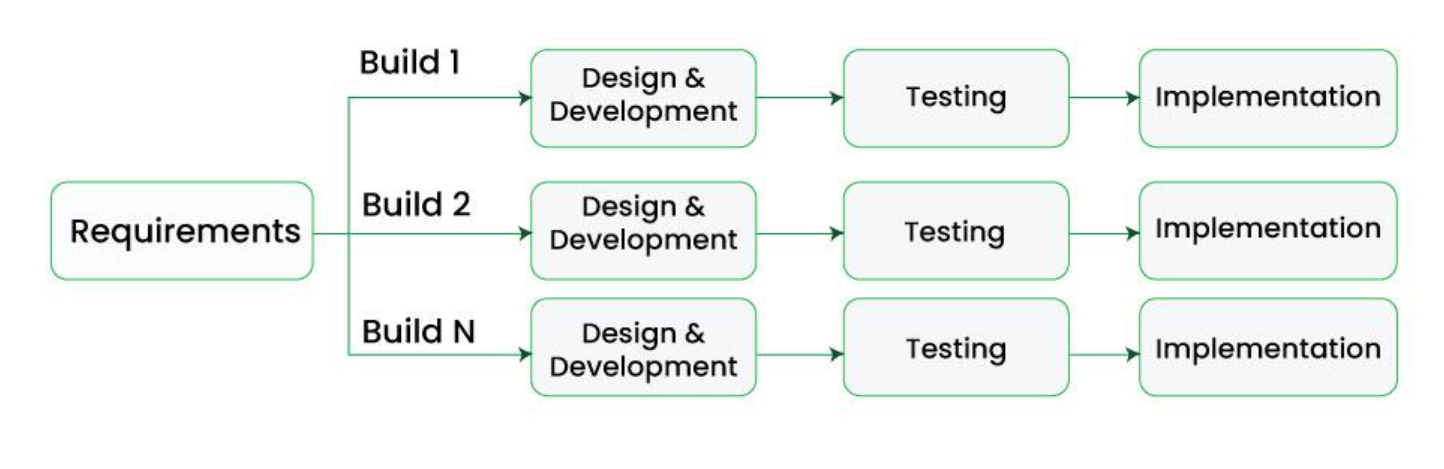
* 1. Applications: It is used when we have strong focus on testing, when we need to correspondingly work on testing phases, suitable for stable projects, early defect detection.
  2. Pros: Early defect detection, structured approach, improved quality
  3. Cons: inflexibility, limited used feedback,

1. **Iterative model**: In this model, the software is developed in iterations. If there needs to be any changes, we can add them in an iteration.



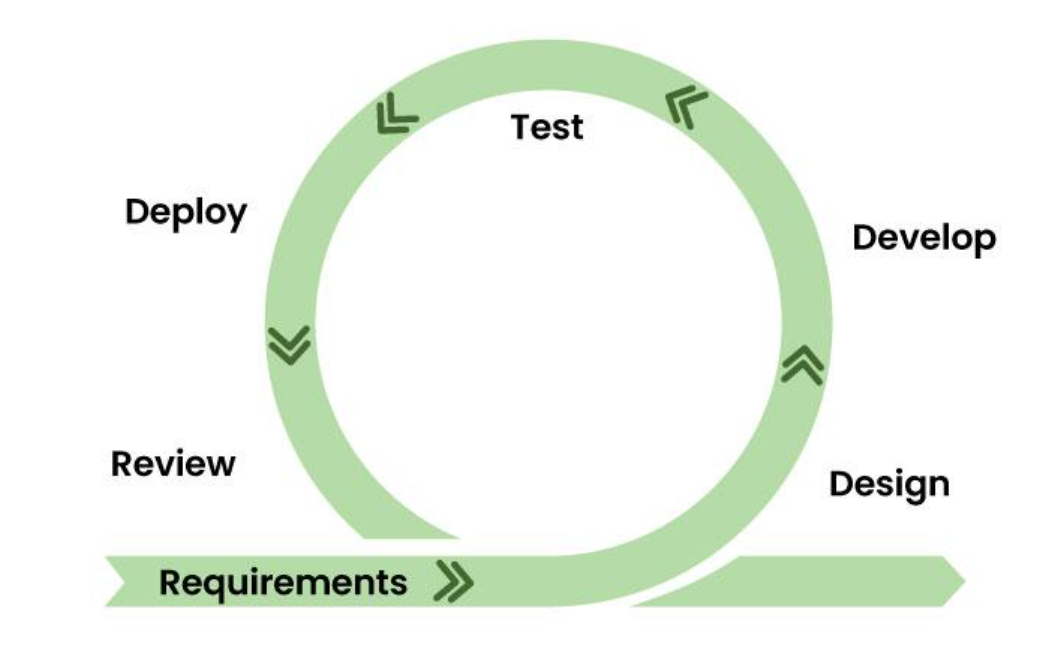
* 1. Applications: It is used when we have evolving requirements, large projects, early used feedback, continuous improvement, industries with continuous improvement.
  2. Pros: Adaptable to change, early user feedback, reduced risk, Improved quality.
  3. Cons: Unpredictable timelines, increased management overhead, potential for scope creep, need highly skilled resources.

1. **Incremental Model**: in this model, the software development process is divided into small increments and the same phases are followed in each increment.



* 1. Applications: It is used when we have evolving requirements, early delivery and partial functionality, complex projects.
  2. Pros: Early delivery, flexibility, reduced risk, Improved efficiency
  3. Cons: Highly skilled resources required, Potential for scope creep.

1. **Agile Model:** Agile methodology is a combination of both iterative and incremental models. It is used when the requirements are uncertain or evolving, complex projects and it focuses on collaboration.

****

* 1. Applications: It is used when requirements are uncertain or evolving, complex projects, need for rapid delivery feedback, focus on collaborations.
  2. Pros: it is adaptable to change, early and continuous feedback, reduced risk
  3. Cons: unpredictable timelines, requires self-organizing teams

**Task 7: What is Scrum in Agile**

**Scrum:**  Scrum is a project management framework within Agile. Scrum focuses on breaking down the project into small, manageable pieces and always focusing on continuous adaptation and improvement. Scrum consists of some key components like Sprints, Product backlog, sprint backlog.

Scrum consists of various events like Daily scrum, sprint planning, Sprint review and sprint retrospective.

Also, there are different roles in the scrum like Product Owner, Scrum master, Scrum team.

Pros: Improved collaboration between teams, Increased transparency, quick adaptation to changing requirements

Cons: Require highly skilled resources, need for scrum master, there is a potential for scope creep.

**Task 8**: **Sprint**

Sprint is a short time-based iteration in which a team sets an achievable goal and works towards delivering a working product for the customer. It is generally 1-4 weeks long.

Key components: Sprint planning, sprint review, daily scrum and sprint retrospective.

**Task 9: 3 dos and don’ts while in Sprint**

**Do’s:**

1. Focusing on the goal
2. Identify blockers
3. Setting up a priority order for tasks at hand and backlog
4. Learning from the lessons learnt from previous sprint

**Don’ts:**

1. Always setting the realistically (achievable) and should not change scope mid sprint
2. Should not ignore blockers and dependencies
3. Ignoring lessons from previous sprints
4. Ignoring priority order and focusing on individual tasks

**Task 10: Backlog & Stories in scrum**

**Backlog:** In scrum, backlog is a priority list of items to be completed to deliver a product or service. These items can be like features, any technical tasks, fixing bugs etc. Clearing backlog is an important part of scrum.

**Stories:** Stories in scrum refers to the short descriptions or explanations of a feature or functionality from the end user perspective.

Ex: As an end user I want a credit card payment option in the app, so that I can directly pay using my credit card instead of paying with UPI every time.

**Task 11:**

**Scrum Artifacts**

**Product Backlog**

**Sprint Backlog**

**Burn-Down Chart**

**Increment**

**Can you brief the above Artifacts?**

**Product Backlog:** This is the main backlog, it is a prioritized list of all the tasks that need to be delivered.

**Sprint Backlog:** This is the list of items that are to be completed in the current sprint.

**Burn-Down Chart:** Burn down chart is a visual representation of the remaining work in the sprint which is plotted against time.

**Increment:** Increment is similar to a sprint, which is like a valuable piece or stepping stone towards of the product goal.

**Networking Concepts**

**Task 12: Ports and Protocols**

**Ports:** A port is a logical communication endpoint or a virtual point where network connections start and end. These ports are managed by a computers operating system. They are identified by numeric ID’s.

**Protocols:** Protocols are defined as rules of communication. They define the rules for actions like sending and receiving data and ensure that different devices communicate with each other effectively.

**Task 13: What are the different network types**

Different types of networks are:

1. Personal Area network (PAN)
2. Local Area Network (LAN)
3. Wireless Local Area Network (WLAN)
4. Wide Area Network (WAN)
5. Virtual Private Network (VPN)
6. Metropolitan Area Network (MAN)

**Task 14:**

**What are the types of servers?**

Different types of Servers:

1. Web Servers: These servers host websites and web applications, delivering web pages to users.
2. Database Servers: These servers are dedicated to storing and managing data.
3. DNS Servers: These servers translate domain names to Ip addresses
4. Mail Servers: These servers handle receiving, sending and also storing of email.
5. File Servers: These servers handle sharing of files across networks and also store files.
6. Application Servers: These servers host software applications. These act as bridge between users and back-end systems.
7. Proxy Servers: These servers act as an intermediary between the client and other servers.
8. FTP Server: These servers helps file transfer between computers.

**Task 15:**

**What do you know about DNS? Domain Name System**

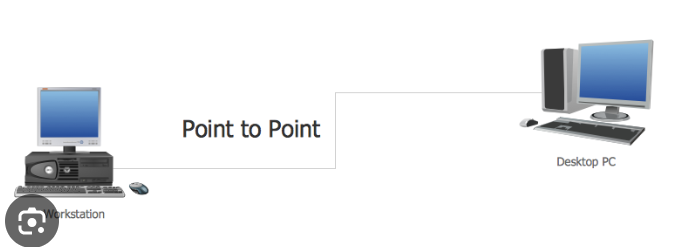
Domain name system (DNS) server is a critical component of the internet which acts like a translator which translates domain names (like google.com) into the IP Addresses which the computer uses for communication.

DNS Server acts like a translator, phonebook etc.

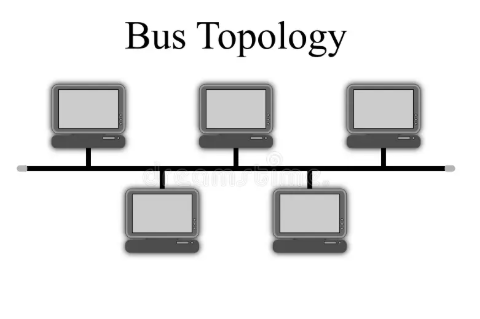
**Task 16: What are the different network topologies?**

Different types of topologies:

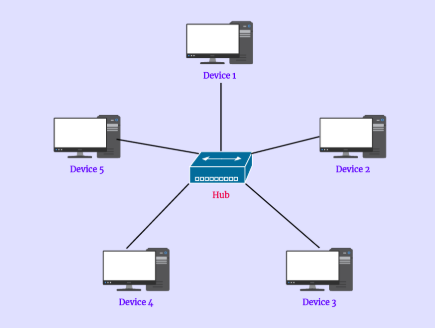
1. **Point-to-point**: It is a direct link between two nodes. It is simple and cost effective, but it is not scalable.



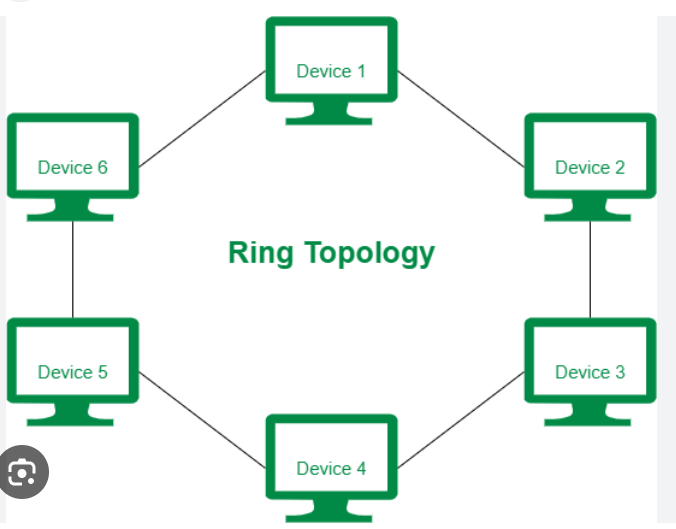
1. **Bus Topology**: When all devices are connected to a single shared cable i.e. bus. It is simple and cost effective, but can be slow at times because many devices are connected to a single cable.



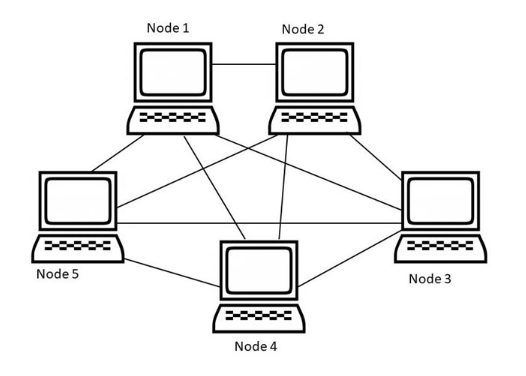
1. **Star Topology:** When all the devices are connected to a central switch. Easy to troubleshoot, scalable, and reliable. But can fail due to failure of switch.

****

1. **Ring Topology**: Each device connects to two other devices, forming a ring. It can be efficient and reliable, but a failure at single point can disrupt the entire ring

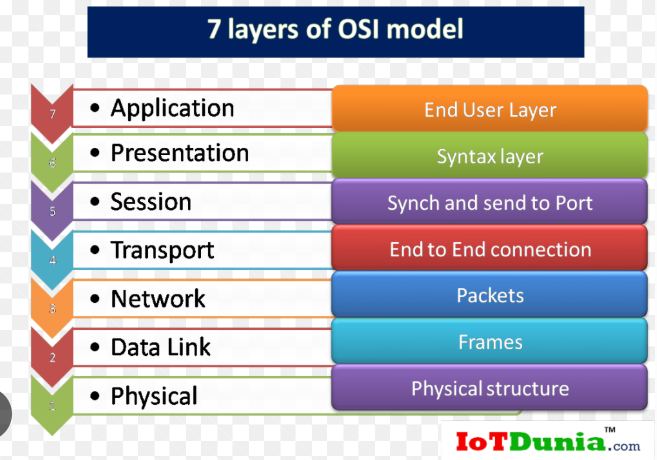


1. **Mesh Topology**: Each device is connected to multiple other devices, creating multiple paths for the data transmission. It is highly reliant but expensive and complex.



**Task 17: What is OSI Model? Describe the 7 layers with description.**

**OSI Model:** It stands for open system interconnection. It is a framework that describes how data travels across a network. It has 7 layers. Each layer has a specific function and the data passes from top layer to bottom layer and goes back up.



Different type of layers:

1. **Physical layer (Layer 1)**: This is the lowest layer. It is responsible for transmitting raw bits of data over a physical medium.
2. **Data Link Layer (Layer 2):** This layer transmits data between two physically connected nodes on a network. It has 2 sub layers, Media access control (MAC) and Logical link control (LLC)
3. **Network Layer (Layer 3):** This layer handles routing data across networks. It provides addressing and delivery services.
4. **Transport Layer (Layer 4):** This layer ensures reliable delivery of data between two nodes. It uses TCP (transmission control protocol) and UDP (User datagram protocol)
5. **Session Layer (Layer 5):** This layer manages connections between applications. It establishes, maintains, and terminates connections, and handles authentication and authorization.
6. **Presentation Layer (Layer 6):** This layer translates data into a format that the application layer can understand. It handles encryption, compression, and data formatting.
7. **Application Layer (Layer 7):** This is the highest layer, closest to the end-user application. It provides services like email, file transfer, and web browsing.