

MUST-DO Questions for Interviews (DBMS, CN and OS)

SDE-Sheet(Core) by **Take U Forward** (Striver_79)

Disclaimer: All the below solutions are directly collected from trustable resources like wikipedia , geeksforgeeks, javatpoint, tutorialspoint , guru99 , Apni kaksha notes, etc..
Sole purpose of sharing this is to help people who don't have enough time to collect answers for every question.

Thank you Striver for this question bank!

Operating Systems

1. What is the main purpose of an operating system? Discuss different types?

An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs. So it manages the computer's memory, processes, devices, files, and security aspects of the system. It also allows us to communicate with the computer without knowing how to speak the computer's language. Without an operating system, a computer is not useful.

Types of Operating Systems:

<https://www.geeksforgeeks.org/types-of-operating-systems/>

1. **Batch OS** (eg: payroll system, bank statements, data entry, etc.)
Batch OS is the first operating system for second-generation computers. This OS does not directly interact with the computer. Instead, an operator takes up similar jobs and groups them together into a batch, and then these batches are executed one by one based on the first-come, first, serve principle.
2. **Distributed OS** (eg: LOCUS)
In a distributed OS, various computers are connected through a single communication channel. These independent computers have their memory unit and CPU and are known as loosely coupled systems. Failure of one system will not affect the other systems because all the computers are independent of each other.
3. **Multitasking OS** (eg: UNIX)
The multitasking OS is also known as the time-sharing operating system as each task is given some time so that all the tasks work efficiently. Each task gets equal time for execution. The idle time for the CPU will be the lowest.
4. **Network OS** (eg: Microsoft Windows server 2008, LINUX, etc.)
Network operating systems are the systems that run on a server and manage all the networking functions. They allow sharing of various files, applications,

printers, security, and other networking functions over a small network of computers like LAN or any other private network.

5. **Real-Time OS** (eg: Medical imaging systems, robots,)

These operating systems are useful where many events occur in a short time or certain deadlines, such as real-time simulations. (More details in Q.15)

6. **Mobile OS** (eg. Android OS, ios)

A mobile OS is an operating system for smartphones, tablets, and PDA's. It is a platform on which other applications can run on mobile devices.

2. What is a socket, kernel and monolithic kernel ?

Socket:

A socket is defined as an endpoint for communication, A pair of processes communicating over a network employ a pair of sockets ,one for each process. A socket is identified by an IP address concatenated with a port number.

The server waits for incoming client requests by listening to a specified port. Once a request is received, the server accepts a connection from the client socket to complete the connection.

Kernel is the central core component of an operating system that manages operations of computer and hardware. Kernel

- Establishes communication between user level application and hardware.
- Manages memory and CPU time
- Decides state of incoming processes.
- Controls Disk, Memory, Task Management

Types of kernels:

❖ **Monolithic Kernel** (provides good performance but lots of lines of code)

It is one of the types of kernel where all operating system services operate in kernel space. It has dependencies between system components. It has huge lines of code which is complex.

Example : Unix, Linux, Open VMS, XTS-400 etc.

(few more types includes : Micro Kernel (more stable but lots of system calls,context switch), Hybrid Kernel(but this is still similar to monolithic kernel), Exo kernel(fewest hardware abstractions hence more work for app developers))

3. Difference between process and program and thread? Different types of process.

Program	Process	Thread
<p>Program is a set of instructions to perform a certain task.</p> <p>Eg: chrome.exe, notepad.exe</p>	<p>Process is an instance of an executing program.</p> <p>For example, we write our computer programs in a text file and when we execute this program, it becomes a process which performs all the tasks mentioned in the program.</p>	<p>Thread is a path of execution within a process. A thread is also known as a lightweight process. The idea is to achieve parallelism by dividing a process into multiple threads.</p> <p>For example, Word processor uses multiple threads: one thread to format the text, another thread to process inputs,</p>
Program is a passive entity as it resides in the secondary memory	Process is a active entity as it is created during execution and loaded into the main memory	Thread is also active entity as it executes a part of a process
Program exists at a single place and continues to exist until its deleted	Process exists for a limited span of time as it gets terminated after the completion of task	A thread goes through various stages in its lifecycle. For example, a thread is born, started, runs, and then dies.
Program does not have its own control block	Process has its own control block called Process Control Block, Stack and Address Space,	Thread has Parents' PCB, its own Thread Control Block and Stack and common Address space.
	Process run in separate memory space	Threads within the process run in a shared memory space.
	Process switching needs interaction with the operating system.	Thread switching does not need to interact with operating system , hence lesser context switching time as well
	Process is heavy weight or resource intensive.	Thread is lightweight, taking fewer resources than a process.
		Threads are implemented in following two ways –

		User Level Threads – User managed threads. Kernel Level Threads – Operating System managed threads acting on kernel, an operating system core.
--	--	---

4. Define virtual memory, thrashing, threads.

Virtual Memory:

A computer can address more memory than the amount physically installed on the system. This extra memory is actually called virtual memory and it is a section of a hard disk that's set up to emulate the computer's RAM.

The main visible advantage of this scheme is that programs can be larger than physical memory. Virtual memory serves two purposes. First, it allows us to extend the use of physical memory by using a disk. Second, it allows us to have memory protection, because each virtual address is translated to a physical address.

Thrashing:

Thrashing is a condition or a situation when the system is spending a major portion of its time in servicing the page faults, but the actual processing done is very negligible. High degree of multiprogramming (if number of processes keeps on increasing in the memory), lack of frames (if a process is allocated too few frames, then there will be too many and too frequent page faults) causes Thrashing.

Threads:

A thread is a single sequential flow of execution of tasks of a process so it is also known as thread of execution or thread of control.

5. What is RAID ? Different types.

RAID mode	Description	Operation	Advantages	Disadvantages	Recovery
RAID 0	Striped disks	Data is split evenly between two or more disks.	Large size and the fastest speed.	No redundancy.	If one or more drives fails, this results in array failure.
RAID 1	Mirrored disks	Two or more drives have identical data on them.	A single drive failure will not result in data loss.	Speed and size is limited by the slowest and smallest disk.	Only one drive is needed for recovery.
RAID 3	Striped set with dedicated parity	Data is split evenly between two or more disks, plus a dedicated drive for parity storage	High speeds for sequential read/write operations.	Poor performance for multiple simultaneous instructions.	A single drive failure will rebuild.

RAID, or “Redundant Arrays of Independent Disks” is a technique which makes use of a combination of multiple disks instead of using a single disk for increased performance, data redundancy or both. Data redundancy, although taking up extra space, adds to disk reliability. This means, in case of disk failure, if the same data is also backed up onto another disk, we can retrieve the data and go on with the operation. Here are the important RAID types:

Few other RAID types: JBOD, Clone, Big

6. What is a deadlock ? Different conditions to achieve a deadlock.

A Deadlock is a situation where each of the computer processes waits for a resource which is being assigned to some other process. In this situation, none of the processes gets executed since the resource it needs is held by some other process which is also waiting for some other resource to be released.

Deadlock is an infinite waiting.

How deadlock is achieved:

Deadlock happens when Mutual exclusion, hold and wait, No preemption and circular wait occurs simultaneously.

Mutual Exclusion:

A resource can only be shared in a mutually exclusive manner. It implies that two processes cannot use the same resource at the same time.

Hold and Wait:

A process waits for some resources while holding another resource at the same time.

No preemption:

The process which once scheduled will be executed till the completion. No other process can be scheduled by the scheduler meanwhile.

Circular Wait:

All the processes must be waiting for the resources in a cyclic manner so that the last process is waiting for the resource which is being held by the first process.

7. What is fragmentation? Types of fragmentation.

Fragmentation:

An unwanted problem in the operating system in which the processes are loaded and unloaded from memory, and free memory space is fragmented. Processes can't be assigned to memory blocks due to their small size, and the memory blocks stay unused. It is also necessary to understand that as programs are loaded and deleted from memory, they generate free space or a hole in the memory. These small blocks cannot be allotted to new arriving processes, resulting in inefficient memory use.

The conditions of fragmentation depend on the memory allocation system. As the process is loaded and unloaded from memory, these areas are fragmented into small pieces of memory that cannot be allocated to incoming processes. It is called fragmentation.

Causes of Fragmentation:

User processes are loaded and unloaded from the main memory, and processes are kept in memory blocks in the main memory. Many spaces remain after process loading and swapping that another process cannot load due to their size. Main memory is available, but its space is insufficient to load another process because of the dynamical allocation of main memory processes

Types of fragmentation :

Internal Fragmentation

When a process is allocated to a memory block, and if the process is smaller than the amount of memory requested, a free space is created in the given memory block. Due to this, the free space of the memory block is unused, which causes internal fragmentation.

Let's suppose a process P1 with a size of 3MB arrives and is given a memory block of 4MB. As a result, the 1MB of free space in this block is unused and cannot be used to allocate memory to another process. It is known as internal fragmentation.

Solution to internal fragmentation: Dynamic Partitioning

External Fragmentation

External fragmentation happens when a dynamic memory allocation method allocates some memory but leaves a small amount of memory unusable. The quantity of available memory is substantially reduced if there is too much external fragmentation. There is enough memory space to complete a request, but it is not contiguous. It's known as external fragmentation

One of the solution to external fragmentation : Compaction : combining all the free memory into a single large block

8. What is spooling ?

SPOOL is an acronym for simultaneous peripheral operations online.

Spooling is a process in which data is temporarily held to be used and executed by a device, program, or system.

In spooling, there is no interaction between the I/O devices and the CPU. That means there is no need for the CPU to wait for the I/O operations to take place. Such operations take a long time to finish executing, so the CPU will not wait for them to finish.

The biggest example of Spooling is printing. The documents which are to be printed are stored in the SPOOL and then added to the queue for printing. During this time,

many processes can perform their operations and use the CPU without waiting while the printer executes the printing process on the documents one-by-one.

9. What is semaphore and mutex (Differences might be asked)? Define Binary semaphore.

Mutex and Semaphore both provide synchronisation services.

Mutex is a mutual exclusion object that synchronises access to a resource. It is created with a unique name at the start of a program. The Mutex is a locking mechanism that makes sure only one thread can acquire the Mutex at a time and enter the critical section. This thread only releases the Mutex when it exits the critical section.

```
wait (mutex);  
...  
Critical Section  
...  
signal (mutex);
```

A Mutex is different from a semaphore as it is a locking mechanism while a semaphore is a signalling mechanism. A binary semaphore can be used as a Mutex but a Mutex can never be used as a semaphore.

Semaphore: A semaphore is a signalling mechanism and a thread that is waiting on a semaphore can be signalled by another thread. This is different from a mutex as the mutex can be signalled only by the thread that called the wait function. A semaphore uses two atomic operations, wait and signal for process synchronisation.

A semaphore restricts the number of simultaneous users of a shared resource upto a maximum number. Threads can request access to the resource (decrementing the semaphore), and can signal that they have finished using the resource (incrementing the semaphore).

The wait operation decrements the value of its argument S, if it is positive. If S is negative or zero, then no operation is performed.

```
wait(S)  
{  
    while (S<=0);  
    S--;  
}
```

The signal operation increments the value of its argument S.

```
signal(S)  
{
```

```
S++;  
}
```

There are mainly two types of semaphores i.e. counting semaphores and binary semaphores.

Counting Semaphores are integer value semaphores and have an unrestricted value domain. These semaphores are used to coordinate the resource access, where the semaphore count is the number of available resources.

The binary semaphores are like counting semaphores but their value is restricted to 0 and 1. The wait operation only works when the semaphore is 1 and the signal operation succeeds when semaphore is 0.

10. Belady's Anomaly

Bélády's anomaly is the name given to the phenomenon where increasing the number of page frames results in an increase in the number of page faults for a given memory access pattern.

Solution to fix Belady's Anomaly:

Implementing alternative page replacement algo helps eliminate Belady's Anomaly.. Use of stack based algorithms, such as Optimal Page Replacement Algorithm and Least Recently Used (LRU) algorithm, can eliminate the issue of increased page faults as these algorithms assign priority to pages.

11. Starving and Ageing in OS

Starving/Starvation(also called Lived lock):

Starvation is the problem that occurs when low priority processes get jammed for an unspecified time as the high priority processes keep executing. So starvation happens if a method is indefinitely delayed.

Solution to Starvation : Ageing

Ageing is a technique of gradually increasing the priority of processes that wait in the system for a long time.

12. Why does thrashing occur?

High degree of multiprogramming(if number of processes keeps on increasing in the memory) , lack of frames(if a process is allocated too few frames, then there will be too many and too frequent page faults.) causes Thrashing.

13. What is paging and why do we need it?

Paging is a memory management scheme that eliminates the need for contiguous allocation of physical memory. This scheme permits the physical address space of a process to be non – contiguous.

Paging is used for faster access to data. When a program needs a page, it is available in the main memory(RAM) as the OS copies a certain number of pages from your storage device to main memory. Paging allows the physical address space of a process to be noncontiguous.

14. Demand Paging, Segmentation

Demand paging is a method of virtual memory management which is based on the principle that pages should only be brought into memory if the executing process demands them. This is often referred to as lazy evaluation as only those pages demanded by the process are swapped from secondary storage to main memory. So demand paging works opposite to the principle of loading all pages immediately.

Segmentation is a memory management technique in which the memory is divided into the variable size parts. Each part is known as a segment which can be allocated to a process.

The details about each segment are stored in a table called a segment table. Segment table is stored in one (or many) of the segments.

Segment table contains mainly two information about segment:

Base: It is the base address of the segment

Limit: It is the length of the segment.

15. Real Time Operating System, types of RTOS.

A real-time operating system (RTOS) is a special-purpose operating system used in computers that has strict time constraints for any job to be performed and is intended to serve real time applications that possess data as it comes in , typically without buffer delays.

Types of RTOS:

Hard Real Time:

In Hard RTOS, the deadline is handled very strictly which means that a given task must start executing on specified scheduled time, and must be completed within the assigned time duration.

Example: Medical critical care system, Aircraft systems, etc.

Firm Real Time:

These types of RTOS also need to follow the deadlines. However, missing a deadline may not have a big impact but could cause undesired effects, like a huge reduction in quality of a product.

Example: Various types of Multimedia applications.

Soft Real Time:

Soft Real time RTOS, accepts some delays by the Operating system. In this type of RTOS, there is a deadline assigned for a specific job, but a delay for a small amount of time is acceptable. So, deadlines are handled softly by this type of RTOS.

Example: Online Transaction system and Livestock price quotation System.

16. Difference between main memory and secondary memory.

Parameter	Primary Memory	Secondary Memory
Nature	The primary memory is categorized as volatile & nonvolatile memories.	The secondary memory is always a non-volatile memory.
Alias	These memories are also called internal memory.	Secondary memory is known as a Backup memory or Additional memory or Auxiliary memory.
Access	Data is directly accessed by the processing unit.	Data cannot be accessed directly by the processor. It is first copied from secondary memory to primary memory. Only then CPU can access it.
Formation	It's a volatile memory meaning data cannot be retained in case of power failure.	It's a non-volatile memory so that that data can be retained even after power failure.
Storage	It holds data or information that is currently being used by the processing unit. Capacity is usually in 16 to 32 GB	It stores a substantial amount of data and information. Capacity is generally from 200GB to terabytes.
Accesses	Primary memory can be accessed by the data bus.	Secondary memory is accessed by I/O channels.
Expense	Primary memory is costlier than secondary memory.	Secondary memory is cheaper than primary memory.

17. Dynamic Binding

Static binding happens when the code is compiled, while dynamic bind happens when the code is executed at run time.

Static Binding:

When a compiler acknowledges all the information required to call a function or all the values of the variables during compile time, it is called “static binding”. As all the required information is known before runtime, it increases the program efficiency and it also enhances the speed of execution of a program. Static Binding makes a program very efficient, but it declines the program flexibility, as ‘values of variable’ and ‘function calling’ are predefined in the program. Static binding is implemented in a program at the time of coding. Overloading a function or an operator is the example of compile time polymorphism i.e. static binding.

Dynamic Binding Calling a function or assigning a value to a variable, at run-time is called “Dynamic Binding”. Dynamic binding can be associated with run time ‘polymorphism’ and ‘inheritance’ in OOP. Dynamic binding makes the execution of a program flexible as it can decide what value should be assigned to the variable and which function should be called, at the time of program execution. But as this information is provided at run time it makes the execution slower as compared to static binding.

18. FCFS Scheduling and SJF Scheduling

First Come First Served (FCFS)	Shortest Job First (SJF)
First Come First Served (FCFS) executes the processes in the order in which they arrive i.e. the process that arrives first is executed first.	Shortest Job First (SJF) executes the processes based upon their burst time i.e. in ascending order of their burst times.
FCFS is non preemptive in nature.	SJF is also non-preemptive but its preemptive version is also there called Shortest Remaining Time First (SRTF) algorithm.
FCFS results in quite long waiting time for the processes and thus increases average waiting time.	The average waiting time for given set of processes is minimum.
FCFS leads to the convoy effect.	It does not lead to the convoy effect.
FCFS algorithm is the easiest to implement in any system.	The real difficulty with SJF is knowing the length of the next CPU request or burst.
A process may have to wait for quite long to get executed depending on the burst time of the processes that have arrived first.	A long process may never get executed and the system may keep executing the short processes.
FCFS lead to lower device and CPU utilization thereby decreasing the efficiency of the system.	SJF leads to higher effectiveness of the system due to lower average waiting time.
FCFS results in minimal overhead.	In case of SJF, elapsed time should be recorded, results in more overhead on the processor.

19. SRTF Scheduling

SRTF Scheduling is a preemptive version of SJF scheduling. In SRTF, the execution of the process can be stopped after a certain amount of time. At the arrival of every process, the short term scheduler schedules the process with the least remaining burst time among the list of available processes and the running process.

20. LRTF Scheduling

This is a preemptive version of Longest Job First (LJF) scheduling algorithm. In this scheduling algorithm, we find the process with the maximum remaining time and then process it. We check for the maximum remaining time after some interval of time (say 1 unit each) to check if another process having more Burst Time arrived up to that time.

21. Priority Scheduling

Priority Scheduling is a method of scheduling processes that is based on priority. In this algorithm, the scheduler selects the tasks to work as per the priority.

The processes with higher priority should be carried out first, whereas jobs with equal priorities are carried out on a round-robin or FCFS basis. Priority depends upon memory requirements, time requirements, etc.

Priority scheduling divided into two main types:

Preemptive Scheduling

In Preemptive Scheduling, the tasks are mostly assigned with their priorities. Sometimes it is important to run a task with a higher priority before another lower priority task, even if the lower priority task is still running. The lower priority task holds for some time and resumes when the higher priority task finishes its execution.

Non-Preemptive Scheduling

In this type of scheduling method, the CPU has been allocated to a specific process. The process that keeps the CPU busy, will release the CPU either by switching context or terminating. It is the only method that can be used for various hardware platforms. That's because it doesn't need special hardware (for example, a timer) like preemptive scheduling.

22. Round Robin Scheduling

In Round-robin scheduling, each ready task runs turn by turn only in a cyclic queue for a limited time slice. This algorithm also offers starvation free execution of processes. Widely used preemptive scheduling method in traditional OS. All the jobs get a fair allocation of CPU. Cons include : Finding a correct time quantum is a quite

difficult task in this system, Round-robin scheduling doesn't give special priority to more important tasks.

23. Producer Consumer Problem

About Producer-Consumer problem

The Producer-Consumer problem is a classic problem that is used for multi-process synchronisation i.e. synchronisation between more than one processes.

The job of the Producer is to generate the data, put it into the buffer, and again start generating data. While the job of the Consumer is to consume the data from the buffer.

What's the problem here?

The following are the problems that might occur in the Producer-Consumer:

The producer should produce data only when the buffer is not full. If the buffer is full, then the producer shouldn't be allowed to put any data into the buffer.

The consumer should consume data only when the buffer is not empty. If the buffer is empty, then the consumer shouldn't be allowed to take any data from the buffer.

The producer and consumer should not access the buffer at the same time.

We can solve this problem by using semaphores.

24. Banker's Algorithm

It is a banker algorithm used to avoid deadlock and allocate resources safely to each process in the computer system. The 'S-State' examines all possible tests or activities before deciding whether the allocation should be allowed to each process. It also helps the operating system to successfully share the resources between all the processes. The banker's algorithm is named because it checks whether a person should be sanctioned a loan amount or not to help the bank system safely simulate allocation resources.

25. Explain Cache

Cache memory is an extremely fast memory type that acts as a buffer between RAM and the CPU. It holds frequently requested data and instructions so that they are immediately available to the CPU when needed.

26. Diff between direct mapping and associative mapping

In Cache memory, data is transferred as a block from primary memory to cache memory. This process is known as Cache Mapping.

Direct Mapping

In direct mapping cache, instead of storing total address information with data in cache only part of address bits is stored along with data.

The new data has to be stored only in a specified cache location as per the mapping rule for direct mapping. So it doesn't need a replacement algorithm.

Advantages of direct mapping

Direct mapping is the simplest type of cache memory mapping.

Here only tag fields are required to match while searching words that is why it is the fastest cache.

Direct mapping cache is less expensive compared to associative cache mapping.

Disadvantages of direct mapping

The performance of direct mapping cache is not good as requires replacement for data-tag value.

In associative mapping both the address and data of the memory word are stored.

The associative mapping method used by cache memory is a very flexible one as well as very fast.

This mapping method is also known as a fully associative cache.

Advantages of associative mapping

Associative mapping is fast.

Associative mapping is easy to implement.

Disadvantages of associative mapping

Cache Memory implementing associative mapping is expensive as it requires storing addresses along with the data.

27. Diff between multitasking and multiprocessing

S	Multi-tasking	Multiprocessing
No.		
1.	The execution of more than one task simultaneously is known as multitasking.	The availability of more than one processor per system, that can execute several set of instructions in parallel is known as multiprocessing.
2.	The number of CPU is one.	The number of CPUs is more than one.
3.	It takes moderate amount of time.	It takes less time for job processing.
4.	In this, one by one job is being executed at a time.	In this, more than one process can be executed at a time.
5.	It is economical.	It is economical.
6.	The number of users is more than one.	The number of users is can be one or more than one.
7.	Throughput is moderate.	Throughput is maximum.
8.	Its efficiency is moderate.	Its efficiency is maximum.
9.	It is of two types: Single user multitasking and Multiple user multitasking.	It is of two types: Symmetric Multiprocessing and Asymmetric Multiprocessing.
10.	Number of user tasks is more than one.	Number of user tasks can be one or more than one.

DBMS

1. What is DBMS ? Mention advantages.

Database Management System (DBMS) is a software for storing and retrieving users' data while considering appropriate security measures. It consists of a group of programs which manipulate the database. The DBMS accepts the request for data from an application and instructs the operating system to provide the specific data. In large systems, a DBMS helps users and other third-party software to store and retrieve data.

Advantages:

- Improved data sharing
- Improved data security
- Better data integration
- Minimised data inconsistency
- Improved data access
- Improved decision making
- Improved end-user productivity

2. What is Database and a database system?

A database is an organised collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a database management system (DBMS). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just a database.

3. What is RDBMS ? Properties.

A Relational Database Management system (RDBMS) is a database management system that is based on the relational model. It has the following major components: Table, Record/Tuple/Row, Field, and Column/Attribute. Examples of the most popular RDBMS are MYSQL, Oracle, IBM DB2, and Microsoft SQL Server database.

Relational databases have the following properties:

- Values are atomic.
- All of the values in a column have the same data type.
- Each row is unique.
- The sequence of columns is insignificant.
- The sequence of rows is insignificant.
- Each column has a unique name.
- Integrity constraints maintain data consistency across multiple tables.

4. Types of database languages

1. Data Definition Language

DDL stands for Data Definition Language. It is used to define database structure or pattern.

It is used to create schema, tables, indexes, constraints, etc. in the database.

Using the DDL statements, you can create the skeleton of the database.

Data definition language is used to store the information of metadata like the number of tables and schemas, their names, indexes, columns in each table, constraints, etc.

Here are some tasks that come under DDL:

- Create: It is used to create objects in the database.
- Alter: It is used to alter the structure of the database.
- Drop: It is used to delete objects from the database.
- Truncate: It is used to remove all records from a table.
- Rename: It is used to rename an object.
- Comment: It is used to comment on the data dictionary.

These commands are used to update the database schema that's why they come under Data definition language.

2. Data Manipulation Language

DML stands for Data Manipulation Language. It is used for accessing and manipulating data in a database. It handles user requests.

Here are some tasks that come under DML:

- Select: It is used to retrieve data from a database.
- Insert: It is used to insert data into a table.
- Update: It is used to update existing data within a table.
- Delete: It is used to delete all records from a table.
- Merge: It performs UPSERT operation, i.e., insert or update operations.
- Call: It is used to call a structured query language or a Java subprogram.
- Explain Plan: It has the parameter of explaining data.
- Lock Table: It controls concurrency.

3. Data Control Language

DCL stands for Data Control Language. It is used to retrieve the stored or saved data.

The DCL execution is transactional. It also has rollback parameters.

(But in Oracle database, the execution of data control language does not have the feature of rolling back.)

Here are some tasks that come under DCL:

- Grant: It is used to give user access privileges to a database.
- Revoke: It is used to take back permissions from the user.

There are the following operations which have the authorization of Revoke:

CONNECT, INSERT, USAGE, EXECUTE, DELETE, UPDATE and SELECT.

4. Transaction Control Language

TCL is used to run the changes made by the DML statement. TCL can be grouped into a logical transaction.

Here are some tasks that come under TCL:

- Commit: It is used to save the transaction on the database.
- Rollback: It is used to restore the database to originality since the last commit.

5. ACID properties (VVVVV IMP)

To ensure the consistency of the database, certain properties are followed by all the transactions occurring in the system.

These properties are called ACID Properties of a transaction.

1. Atomicity
 - This property ensures that either the transaction occurs completely or it does not occur at all.
 - In other words, it ensures that no transaction occurs partially.
2. Consistency
 - This property ensures that integrity constraints are maintained.
 - In other words, it ensures that the database remains consistent before and after the transaction.
3. Isolation
 - This property ensures that multiple transactions can occur simultaneously without causing any inconsistency.
 - The resultant state of the system after executing all the transactions is the same as the state that would be achieved if the transactions were executed serially one after the other.
4. Durability
 - This property ensures that all the changes made by a transaction after its successful executions are written successfully to the disk.
 - It also ensures that these changes exist permanently and are never lost even if there occurs a failure of any kind.

6. Difference between vertical and horizontal scaling

Scaling alters the size of a system. In the scaling process, we either compress or expand the system to meet the expected needs. The scaling operation can be achieved by adding resources to meet the smaller expectation in the current system, or by adding a new system in the existing one, or both.

Vertical scaling keeps your existing infrastructure but adds computing power. Your existing pool of code does not need to change — you simply need to run the same code on machines with better specs. By scaling up, you increase the capacity of a single machine and increase its throughput. Vertical scaling allows data to live on a single node, and scaling spreads the load through CPU and RAM resources for your machines.

Horizontal scaling simply adds more instances of machines without first implementing improvements to existing specifications. By scaling out, you share the processing power and load balancing across multiple machines.

Horizontal Scaling	Vertical Scaling
When new server racks are added in the existing system to meet the higher expectation, it is known as horizontal scaling.	When new resources are added in the existing system to meet the expectation, it is known as vertical scaling
It expands the size of the existing system horizontally.	It expands the size of the existing system vertically.
It is difficult to implement	It is easy to implement
It is costlier, as new server racks comprises of a lot of resources	It is cheaper as we need to just add new resources
It takes more time to be done	It takes less time to be done

7. What is sharding

Sharding is a method of splitting and storing a single logical dataset in multiple databases. By distributing the data among multiple machines, a cluster of database systems can store larger dataset and handle additional requests. Sharding is necessary if a dataset is too large to be stored in a single database. Moreover, many sharding strategies allow additional machines to be added. Sharding allows a database cluster to scale along with its data and traffic growth.

<https://www.mongodb.com/features/database-sharding-explained>

<https://youtu.be/5faMjKuB9bc>

8. Keys in DBMS

Keys:

A key is a set of attributes that can identify each tuple uniquely in the given relation.

Types of Keys:

- Super Key - A superkey is a set of attributes that can identify each tuple uniquely in the given relation. A super key may consist of any number of attributes.
- Candidate Key - A set of minimal attribute(s) that can identify each tuple uniquely in the given relation is called a candidate key.
- Primary Key - A primary key is a candidate key that the database designer selects while designing the database. Primary Keys are unique and NOT NULL.
- Alternate Key - Candidate keys that are left unimplemented or unused after implementing the primary key are called alternate keys.
- Foreign Key - An attribute 'X' is called as a foreign key to some other attribute 'Y' when its values are dependent on the values of attribute 'Y'. The relation in which attribute 'Y' is present is called the referenced relation. The relation in which attribute 'X' is present is called the referencing relation.
- Composite Key - A primary key composed of multiple attributes and not just a single attribute is called a composite key.
- Unique Key - It is unique for all the records of the table. Once assigned, its value cannot be changed i.e. it is non-updatable. It may have a NULL value.

9. Types of relationship

Relationship:

A relationship is defined as an association among several entities.

- Unary Relationship Set - Unary relationship set is a relationship set where only one entity set participates in a relationship set.
- Binary Relationship Set - Binary relationship set is a relationship set where two entity sets participate in a relationship set.
- Ternary Relationship Set - Ternary relationship set is a relationship set where three entity sets participate in a relationship set.
- N-ary Relationship Set - N-ary relationship set is a relationship set where 'n' entity sets participate in a relationship set.

10. Data abstraction in DBMS, three levels of it

Data Abstraction is a process of hiding unwanted or irrelevant details from the end user. It provides a different view and helps in achieving data independence which is used to enhance the security of data.

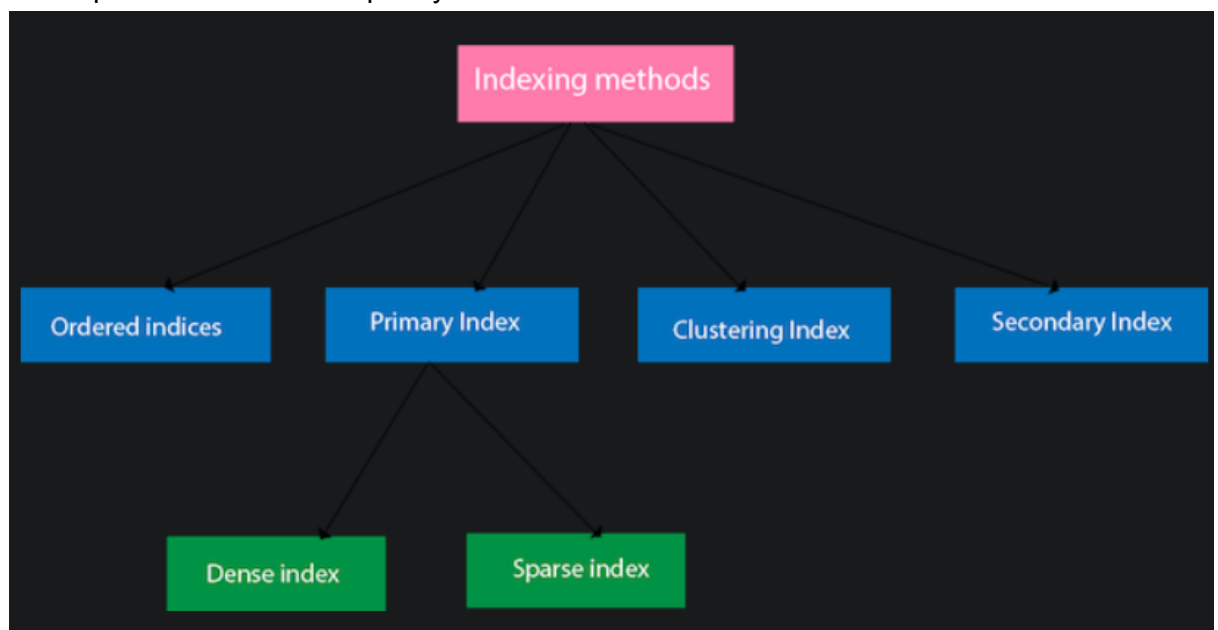
Database systems include complex data-structures. In terms of retrieval of data, reduce complexity in terms of usability of users and in order to make the system efficient, developers use levels of abstraction that hide irrelevant details from the users. Levels of abstraction simplify database design.

Mainly there are three levels of abstraction for DBMS, which are as follows:

1. **Physical or Internal Level:** It is the lowest level of abstraction for DBMS which defines how the data is actually stored, it defines data-structures to store data and access methods used by the database.
2. **Logical or Conceptual Level:** Logical level is the intermediate level or next higher level. It describes what data is stored in the database and what relationship exists among those data.
3. **View or External Level:** It is the highest level. In view level, there are different levels of views and every view only defines a part of the entire data

11. Indexing in DBMS

Indexing is a way to optimise the performance of a database by minimising the number of disk accesses required when a query is processed. It is a data structure technique which is used to quickly locate and access the data in a database.



12. What is DDL (Data Definition Language)

Discussed earlier

13. What is DML (Data Manipulation Language)

Discussed earlier

14. What is normalisation ? Types of them (*Refer Javatpoint for all*)

Normalisation:

In DBMS, database normalisation is a process of making the database consistent by-

- Reducing the redundancies
- Ensuring the integrity of data through lossless decomposition

Normal Forms:

1. First Normal Form (1NF) - A given relation is called in First Normal Form (1NF) if and only if
 - There are **only Single Valued** Attributes.
 - Attribute Domain does not change.
 - There is a unique name for every Attribute/Column.
 - The order in which data is stored does not matter.
2. Second Normal Form (2NF) - A given relation is called in Second Normal Form (2NF) if and only if
 - Relation already exists in 1NF.
 - **No partial dependency** exists in the relation.
No non-prime attribute (attributes which are not part of any candidate key) is dependent on any proper subset of any candidate key of the table.
3. Third Normal Form (3NF) - A given relation is called in Third Normal Form (3NF) if and only if
 - Relation already exists in 2NF.
 - **No transitive dependency** exists for non-prime attributes.
if it holds at least one of the following conditions for every non-trivial functional dependency $X \rightarrow Y$.
 - X is a super key.
 - Y is a prime attribute, i.e., each element of Y is part of some candidate key.
4. Boyce-Codd Normal Form - A given relation is called in BCNF if and only if
 - Relation already exists in 3NF.
 - For every non-trivial functional dependency ' $A \rightarrow B$ ', A is a **super key** of the relation.
5. Fourth Normal Form (4NF) - A relation will be in 4NF if it is
 - in Boyce Codd normal form

- **No multivalued dependency.**

For a dependency $A \twoheadrightarrow B$, if for a single value of A, multiple values of B exists, then the relation will be a multivalued dependency.

6. Fifth Normal Form (5NF) - A relation is in 5NF if it is

- in 4NF
- does not contain any join dependency
- joining should be lossless

15. What is denormalization ?

<https://www.geeksforgeeks.org/denormalization-in-databases/>

Denormalization is a database optimization technique in which we add redundant data to one or more tables. This can help us avoid costly joins in a relational database. Note that denormalization does not mean not doing normalisation. It is an optimization technique that is applied after doing normalisation.

16. What is functional dependency ?


A functional dependency is a constraint that specifies the relationship between two sets of attributes where one set can accurately determine the value of other sets. It is denoted as $X \rightarrow Y$, where X is a set of attributes that is capable of determining the value of Y. The attribute set on the left side of the arrow, X is called Determinant, while on the right side, Y is called the Dependent.

17. E-R Model ?

ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system. It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.

In ER modelling, the database structure is portrayed as a diagram called an entity-relationship diagram.

18. Conflict Serializability in DBMS

 [Lec-77: Conflict Equivalent Schedules with Example | Transaction concurrency ...](#)

Serializability is a concept that helps us to check which schedules are serializable. A serializable schedule is the one that always leaves the database in consistent state.

A schedule is called conflict serializability if after swapping of non-conflicting operations, it can transform into a serial schedule.

The schedule will be a conflict serializable if it is conflict equivalent to a serial schedule.

Conflicting Operations

The two operations become conflicting if all conditions satisfy:

- a) Both belong to separate transactions.
- b) They have the same data item.
- c) They contain at least one write operation

19. Explain Normal forms in DBMS

Discussed earlier

20. What is CCP ? (Concurrency Control Protocols)

Concurrency Control is the management procedure that is required for controlling concurrent execution of the operations that take place on a database

The concurrency control protocols ensure the atomicity, consistency, isolation, durability and serializability of the concurrent execution of the database transactions.

Therefore, these protocols are categorised as:

1. Lock Based Concurrency Control Protocol
2. Timestamp Concurrency Control Protocol
3. Validation Based Concurrency Control Protocol

21. Entity, Entity Type, Entity Set, Weak Entity Set.

Entity in DBMS can be a real-world object with an existence, For example, in a College database, the entities can be Professor, Students, Courses, etc.

The entity type is a collection of the entity having similar attributes.

Types of Entity type

1. Strong Entity Type
Strong entities are those entity types which have a key attribute. The primary key helps in identifying each entity uniquely. It is represented by a rectangle.
2. Weak Entity Type
Weak entity type doesn't have a key attribute. Weak entity types can't be identified on their own. It depends upon some other strong entity for its distinct identity.

Entity Set:

Entity Set is a collection of entities of the same entity type. We can say that entity type is a superset of the entity set as all the entities are included in the entity type.

<https://www.geeksforgeeks.org/weak-entity-set-in-er-diagrams/>

22. What are SQL commands ? Types of them.

<https://www.javatpoint.com/dbms-sql-command>

23. Nested Queries in SQL ?

<https://www.tutorialspoint.com/explain-about-nested-queries-in-dbms>

24. What is JOIN ? Explain types of JOINS

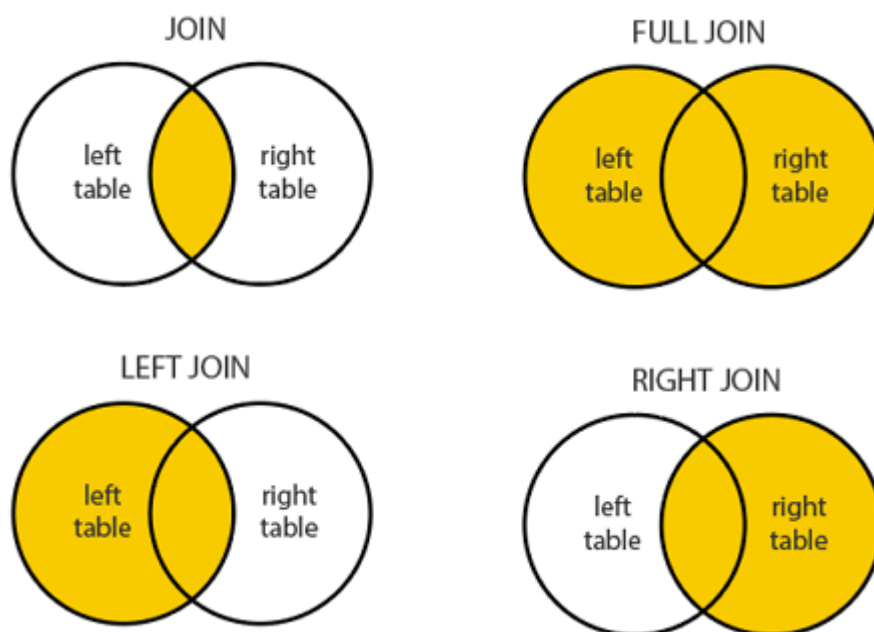
<https://www.javatpoint.com/dbms-sql-joins>

25. Inner and Outer Join

Different Types of SQL JOINS

Here are the different types of the JOINS in SQL:

1. (INNER) JOIN:
Returns records that have matching values in both tables
2. LEFT (OUTER) JOIN:
Returns all records from the left table, and the matched records from the right table
3. RIGHT (OUTER) JOIN:
Returns all records from the right table, and the matched records from the left table
4. FULL (OUTER) JOIN:
Returns all records when there is a match in either left or right table



26. Practice sql queries from leetcode

27. Diff between 2 tier and 3 tier architecture

[Difference Between Two-Tier And Three-Tier Database Architecture - GeeksforGeeks](https://www.geeksforgeeks.org/difference-between-two-tier-and-three-tier-database-architecture/)

28. Diff between TRUNCATE and DELETE command ..

Comparison Basis	DELETE	TRUNCATE
Definition	The delete statement is used to remove single or multiple records from an existing table depending on the specified condition.	The truncate command removes the complete data from an existing table but not the table itself. It preserves the table structure or schema.
Language	It is a DML (Data Manipulation Language) command.	It is a DDL (Data Definition Language) command.
WHERE	It can use the WHERE clause to filter any specific row or data from the table.	It does not use the WHERE clause to filter records from the table.
Permission	We need to have DELETE permission to use this command.	We need to have ALTER permission to use this command.
Working	This command eliminates records one by one.	This command deletes the entire data page containing the records.

29. Difference between Intension and Extension in a DataBase

Intension is the permanent part of the relation and comprises two things: relation schema and the integrity constraints. Relation schema defines the name and attributes of the relation, and integrity constraints define key constraints, referential constraints. etc.

As it corresponds to the schema of the relation, it provides definition to all the extensions of the relation and is time independent.

For example : Intension of student :

Student (RollNo Number(4) Not NULL, Name Char(20), Age Number(2), Course Char(15))

Extension is the snapshot of the system at a particular time. It displays values for tuples in a relation at the particular instance of time. It is dependent on time and it keeps on changing as the tuples are added, deleted, edited.

For example, extension of student at time t1 when two tuples were added

RollNo	Name	Age	Course
101	Amit	21	B.Tech.
102	Annanya	20	BCA

extension of student at time t2 when one more tuple was added and one tuple was updated

RollNo	Name	Age	Course
101	Amit	21	B.Tech.
102	Annanya	20	BSc
103	Ankit	24	MCA

30. Difference between share lock and exclusive lock, definition of lock

A lock is a data variable which is associated with a data item. This lock signifies operations that can be performed on the data item. Locks in DBMS help synchronise access to the database items by concurrent transactions. All lock requests are made to the concurrency-control manager

Difference between Shared Lock and Exclusive Lock :

S.No.	Shared Lock	Exclusive Lock
1.	Lock mode is read only operation.	Lock mode is read as well as write operation.
2.	Shared lock can be placed on objects that do not have an exclusive lock already placed on them.	Exclusive lock can only be placed on objects that do not have any other kind of lock.
3.	Prevents others from updating the data.	Prevents others from reading or updating the data.
4.	Issued when transaction wants to read item that do not have an exclusive lock.	Issued when transaction wants to update unlocked item.
5.	Any number of transaction can hold shared lock on an item.	Exclusive lock can be hold by only one transaction.
6.	S-lock is requested using lock-S instruction.	X-lock is requested using lock-X instruction.

Computer Networks

1. Define network

A network is a set of devices that are connected with a physical media link. In a network, two or more nodes are connected by a physical link or two or more networks are connected by one or more nodes. A network is a collection of devices connected to each other to allow the sharing of data.

2. What do you mean by network topology and explain types of them

Network topology specifies the layout of a computer network. It shows how devices and cables are connected to each other

Types of network topology : Ring, Bus, Mesh, Tree , Hybrid

i) Star :

- Star topology is a network topology in which all the nodes are connected to a single device known as a central device.
- Star topology requires more cable compared to other topologies. Therefore, it is more robust as a failure in one cable will only disconnect a specific computer connected to this cable.
- If the central device is damaged, then the whole network fails.
- Star topology is very easy to install, manage and troubleshoot. It is commonly used in office and home networks.

ii) Ring :

1. Ring topology is a network topology in which nodes are exactly connected to two or more nodes and thus, forming a single continuous path for the transmission.
2. It does not need any central server to control the connectivity among the nodes.
3. If the single node is damaged, then the whole network fails.
4. Ring topology is very rarely used as it is expensive, difficult to install and manage.
5. Examples of Ring topology are SONET network, SDH network, etc.

iii) Bus :

1. Bus topology is a network topology in which all the nodes are connected to a single cable known as a central cable or bus.
2. It acts as a shared communication medium, i.e., if any device wants to send the data to other devices, then it will send the data over the bus which in turn sends the data to all the attached devices.
3. Bus topology is useful for a small number of devices.
4. As if the bus is damaged then the whole network fails.
(This topology is no longer used. But there was a time when this topology used to be the first choice among the network administrators.)

iv) Mesh :

1. Mesh topology is a network topology in which all the nodes are individually

connected to other nodes.

2. It does not need any central switch or hub to control the connectivity among the nodes.

3. Mesh topology is categorised into two parts: Fully connected mesh topology:

In this topology, all the nodes are connected to each other. Partially connected

mesh topology: In this topology, all the nodes are not connected to each other.

4. It is robust as a failure in one cable will only disconnect the specified computer connected to this cable.

5. Mesh topology is rarely used as installation and configuration are difficult when connectivity gets more.

6. Cabling cost is high as it requires bulk wiring.

(Mesh topology is commonly used in the WAN network for backup purposes. This topology is not used in the LAN network implementations.)

v)Tree :

1. Tree topology is a combination of star and bus topology. It is also known as the expanded star topology.

2. In tree topology, all the star networks are connected to a single bus.

3. Ethernet protocol is used in this topology.

4. In this, the whole network is divided into segments known as star networks which can be easily maintained. If one segment is damaged, there is no effect on other segments.

5. Tree topology depends on the "main bus," and if it breaks, then the whole network gets damaged

(Tree topology is suitable for large networks, spread into many branches. Example: Big university campuses, hospitals etc. Main disadvantage of tree topology is that the connectivity between tree branches are dependent on main backbone switches)

vi)Hybrid :

1. A hybrid topology is a combination of different topologies to form a resulting topology.

2. If star topology is connected with another star topology, then it remains a star topology. If star topology is connected with different topology, then it becomes a Hybrid topology.

3. It provides flexibility as it can be implemented in a different network environment

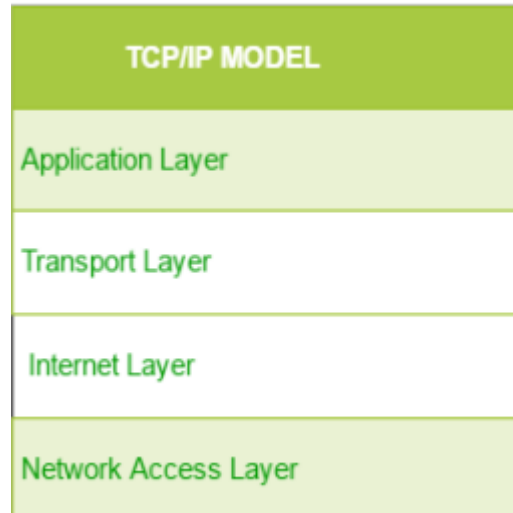
3. Define bandwidth, node and link ?

Bandwidth is the data transfer capacity of a computer network in bits per second (Bps). The term may also be used colloquially to indicate a person's capacity for tasks or deep thoughts at a point in time.

A network is a connection setup of two or more computers directly

connected by some physical mediums like optical fibre or coaxial cable. This physical medium of connection is known as a link, and the computers that it is connected to are known as nodes

4. Explain TCP model



It is a compressed version of the OSI model with only 4 layers. It was developed by the US Department of Defence (DoD) in the 1960s. The name of this model is based on 2 standard protocols used i.e. TCP (Transmission Control Protocol) and IP (Internet Protocol).

1. Network Access/Link layer : Decides which links such as serial lines or classic Ethernet must be used

to meet the needs of the connectionless internet layer. Ex - Sonet, Ethernet

2. Internet : The internet layer is the most important layer which holds the whole architecture together. It delivers the IP packets where they are supposed to be delivered. Ex - IP, ICMP.

3. Transport : Its functionality is almost the same as the OSI transport layer. It enables peer entities on the network to carry on a conversation. Ex - TCP, UDP (User Datagram Protocol)

4. Application : It contains all the higher-level protocols. Ex - HTTP, SMTP, RTP, DNS

5. Layers of OSI model

OSI MODEL
Application Layer
Presentation Layer
Session Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer

It is a network architecture model based on the ISO standards. It is called the OSI model as it deals with connecting the systems that are open for communication with other systems. The OSI model has seven layers.

The principles used to arrive at the seven layers can be summarised briefly as below:

1. Create a new layer if a different abstraction is needed.
2. Each layer should have a well-defined function.
3. The function of each layer is chosen based on internationally standardised protocols.

- Seven Layers :

1. Physical Layer

- It is the lowest layer of the OSI reference model.
- It is used for the transmission of an unstructured raw bit stream over a physical medium.
- Physical layer transmits the data either in the form of electrical/optical or mechanical form.
- The physical layer is mainly used for the physical connection between the devices, and such physical connection can be made by using twisted-pair cable, fibre-optic or wireless transmission media.

2. DataLink Layer

- It is used for transferring the data from one node to another node.
- It receives the data from the network layer and converts the data into data frames and then attaches the physical address to these frames which are sent to the physical layer.
- It enables the error-free transfer of data from one node to another node.

Functions of Data-link layer:

- Frame synchronisation: Data-link layer converts the data into frames, and it ensures that the destination must recognize the starting and ending of each frame.
- Flow control: Data-link layer controls the data flow within the network.
- Error control: It detects and corrects the error occurred during the transmission from source to destination.
- Addressing: Data-link layers attach the physical address with the data frames so that the individual machines can be easily identified.
- Link management: Data-link layer manages the initiation, maintenance and termination of the link between the source and destination for the effective exchange of data.

3. Network Layer

- Network layer converts the logical address into the physical address.
- The routing concept means it determines the best route for the packet to travel from source to the destination.

Functions of network layer :

- Routing: The network layer determines the best route from source to destination. This function is known as routing.
- Logical addressing: The network layer defines the addressing scheme to identify each device uniquely.
- Packetizing: The network layer receives the data from the upper layer and converts the data into packets. This process is known as packetizing.
- Internetworking: The network layer provides the logical connection between the different types of networks for forming a bigger network.
- Fragmentation: It is a process of dividing the packets into fragments..

4. Transport Layer

- It delivers the message through the network and provides error checking so that no error occurs during the transfer of data.
- It provides two kinds of services:
 - Connection-oriented transmission: In this transmission, the receiver sends the acknowledgement to the sender after the packet has been received.
 - Connectionless transmission: In this transmission, the receiver does not send the acknowledgement to the sender.

5. Session Layer

- The main responsibility of the session layer is beginning, maintaining and ending the communication between the devices.
- Session layer also reports the error coming from the upper layers.
- Session layer establishes and maintains the session between the two users.

6. Presentation Layer

- The presentation layer is also known as a Translation layer as it translates the data from one format to another format.
- At the sender side, this layer translates the data format used by the application layer to the common format and at the receiver side, this layer translates the common format into a format used by the application layer.

Functions of presentation layer:

- Character code translation
- Data conversion
- Data compression
- Data encryption

7. Application Layer

- Application layer enables the user to access the network.
- It is the topmost layer of the OSI reference model.
- Application layer protocols are file transfer protocol, simple mail transfer protocol, domain name system, etc.
- The most widely used application protocol is HTTP(Hypertext transfer protocol). A user sends the request for the web page using HTTP.

6. Significance of Data Link Layer

(discussed in the previous question)

7. Define gateway, difference between gateway and router ..

A node that is connected to two or more networks is commonly known as a gateway. It is also known as a router. It is used to forward messages from one network to another. Both the gateway and router regulate the traffic in the network. Differences between gateway and router: A router sends the data between two similar networks while gateway sends the data between two dissimilar networks

8. What does ping command do ?

The "ping" is a utility program that allows you to check the connectivity between the network devices. You can ping devices using its IP address or name.

9. What is DNS, DNS forwarder, NIC, ?

DNS (Imp) :

1. DNS is an acronym that stands for Domain Name System. DNS was introduced by Paul Mockapetris and Jon Postel in 1983.

2. It is a naming system for all the resources over the internet which includes physical nodes and applications. It is used to locate resources easily over a network.

3. DNS is an internet which maps the domain names to their associated IP addresses. 4. Without DNS, users must know the IP address of the web page that you wanted to access.

- Working of DNS (Imp): If you want to visit the website of "shaurya", then the user will

type "https://www.shaurya.com" into the address bar of the web browser. Once the domain name is entered, then the domain name system will translate the domain name

into the IP address which can be easily interpreted by the computer. Using the IP address, the computer can locate the web page requested by the user.

- DNS Forwarder : A forwarder is used with a DNS server when it receives DNS queries

that cannot be resolved quickly. So it forwards those requests to external DNS servers

for resolution. A DNS server which is configured as a forwarder will behave differently than the DNS server which is not configured as a forwarder.

NIC stands for Network Interface Card. It is a peripheral card attached to the PC to connect to a network. Every NIC has its own MAC address that identifies the

PC on the network. It provides a wireless connection to a local area network. NICs were

mainly used in desktop computers.

10. What is a MAC address ?

A media access control address (MAC address) is a unique identifier assigned to a network interface controller (NIC) for use as a network address in communications within a network segment.

MAC address and IP address (Imp) :

1. Both MAC (Media Access Control) Address and IP Address are used to uniquely define a device on the internet. NIC Card's Manufacturer provides the MAC Address, On the other hand, Internet Service Providers provide IP Addresses.
2. The main difference between MAC and IP address is that MAC Address is used to ensure the physical address of a computer. It uniquely identifies the devices on a network. While IP addresses are used to uniquely identify the connection of a network with that device taking part in a network.

11. What is IP address, private IP address, public IP address, APIPA ?

An IP address is a unique address that identifies a device on the internet or a local network. IP stands for "Internet Protocol," which is the set of rules governing the format of data sent via the internet or local network.

Private IP Address - There are three ranges of IP addresses that have been reserved for IP addresses. They are not valid for use on the internet. If you want to access the internet on these private IPs, you must use a proxy server or NAT server.

Public IP Address - A public IP address is an address taken by the Internet Service Provider which facilitates communication on the internet.

APIPA stands for Automatic Private IP Addressing (APIPA). It is a feature or characteristic in operating systems (eg. Windows) which enables computers to self-configure an IP address and subnet mask automatically when their DHCP(Dynamic Host Configuration Protocol:A DHCP Server is a network server that automatically provides and assigns IP addresses, default gateways and other network parameters to client devices. It relies on the standard protocol known as Dynamic Host Configuration Protocol) server isn't reachable

12. Difference between IPv4 and IPv6

IPv4	IPv6
IPv4 has a 32-bit address length	IPv6 has a 128-bit address length
Address representation of IPv4 is in decimal	Address Representation of IPv6 is in hexadecimal
In IPv4 Encryption and Authentication facility not provided	In IPv6 Encryption and Authentication are provided
It can generate 4.29×10^9 address space	Address space of IPv6 is quite large it can produce 3.4×10^{38} address space
It Supports Manual and DHCP address	It supports Auto and renumbering

configuration	address configuration
IPv4 sites load less faster compared to IPv6	

13. What is subnet ?

A subnet is a network inside a network achieved by the process called subnetting which helps divide a network into subnets. It is used for getting a higher routing efficiency and enhances the security of the network. It reduces the time to extract the host address from the routing table.

14. Firewalls

The firewall is a network security system that is used to monitor the incoming and outgoing traffic and blocks the same based on the firewall security policies. It acts as a wall between the internet (public network) and the networking devices (a private network). It is either a hardware device, software program, or a combination of both. It adds a layer of security to the network

15. Different type of delays

The delays, here, means the time for which the processing of a particular packet takes place. We have the following types of delays in computer networks:

1)Transmission Delay:

The time taken to transmit a packet from the host to the transmission medium is called Transmission delay.

Let B bps is the bandwidth and L bit is the size of the data then transmission delay is,

$$T_t = L/B$$

2) Propagation delay:

After the packet is transmitted to the transmission medium, it has to go through the medium to reach the destination. Hence the time taken by the last bit of the packet to reach the destination is called propagation delay. $T_p = \text{Distance} / \text{Velocity}$

3) Queueing delay:

If the packet is received by the destination, the packet will not be processed by the destination immediately. It has to wait in a queue in something called a buffer. So the amount of time it waits in queue before being processed is called queueing delay.

In general, we can't calculate queueing delay because we don't have any formula for that.

4.)Processing delay:

Now the packet will be taken for the processing which is called processing delay.

Time is taken to process the data packet by the processor, that is the time required by intermediate routers to decide where to forward the packet, update TTL, and perform header checksum calculations.

It also doesn't have any formula since it depends upon the speed of the processor and the speed of the processor varies from computer to computer.

Note: $\text{Total} = T_t + T_p + T_q + T_{\text{pro}}$

$\text{Total} = T_t + T_p$

(when taking T_q (queuing delay) and T_{pro} (processing delay) equals to 0)

16. 3 way handshaking

Three-Way HandShake or a TCP 3-way handshake is a process which is used in a TCP/IP network to make a connection between the server and client. It is a three-step process that requires both the client and server to exchange synchronisation and acknowledgment packets before the real data communication process starts.

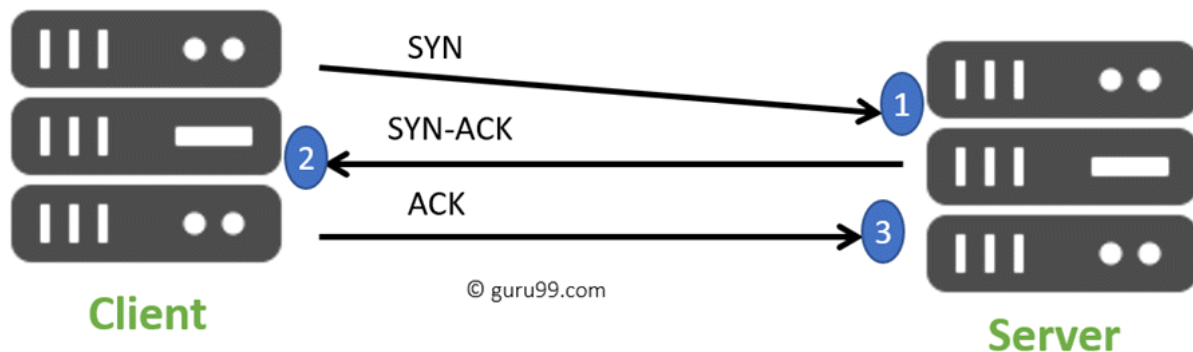
Three-way handshake process is designed in such a way that both ends help you to initiate, negotiate, and separate TCP socket connections at the same time. It allows you to transfer multiple TCP socket connections in both directions at the same time.

TCP message types

Message	Description
Syn	Used to initiate and establish a connection. It also helps you to synchronise sequence numbers between devices.
ACK	Helps to confirm to the other side that it has received the SYN.
SYN-ACK	SYN message from local device and ACK of the earlier packet.
FIN	Used to terminate a connection.

TCP Three-Way Handshake Process

TCP traffic begins with a three-way handshake. In this TCP handshake process, a client needs to initiate the conversation by requesting a communication session with the Server:



3 way Handshake Diagram

- **Step 1:** In the first step, the client establishes a connection with a server. It sends a segment with SYN and informs the server that the client should start communication, and with what should be its sequence number.
- **Step 2:** In this step server responds to the client request with SYN-ACK signal set. ACK helps you to signify the response of the segment that is received and SYN signifies what sequence number it should be able to start with the segments.
- **Step 3:** In this final step, the client acknowledges the response of the Server, and they both create a stable connection and begin the actual data transfer process.

17. Server-side load balancer

Q. Why do we need load balancing?

Ans. When using scalable microservices, a client needs to be able to route its requests to one of the multiple backend server instances. Multiple requests from the client(s) need to be load-balanced across the backend servers so that no single backend server gets overloaded.

There are 2 approaches for load balancing:

1. **Server-side load-balancing:** All backend server instances are registered with a central load balancer. A client requests this load balancer which then routes the request to one of the server instances using various algorithms like round-robin. AWS ELB(Elastic Load Balancing) is a prime example of server-side load-balancing that registers multiple EC2 instances launched in its auto-scaling group and then routes the client requests to one of the EC2 instances.

Advantages of server-side load balancing:

Simple client configuration: only need to know the load-balancer address.
Clients can be untrusted: all traffic goes through the load-balancer where it can be looked at. Clients are not aware of the backend servers.

2. Client-side load-balancing: The load balancing decision resides with the client itself. The client can take the help of a naming server (eg. Netflix Eureka) to get the list of registered backend server instances, and then route the request to one of these backend instances using client-side load balancing libraries like Netflix Ribbon.

Advantages of client-side load balancing:

No more single point of failure as in the case of the traditional load balancer approach.

Reduced cost as the need for server-side load balancer goes away.

Less network latency as the client can directly invoke the backend servers removing an extra hop for the load balancer.

18. RSA Algorithm

RSA algorithm is an asymmetric cryptography algorithm. Asymmetric actually means that it works on two different keys i.e. Public Key and Private Key. As the name describes, the Public Key is given to everyone and the Private key is kept private.

An example of asymmetric cryptography :

A client (for example browser) sends its public key to the server and requests for some data.

The server encrypts the data using the client's public key and sends the encrypted data.

Client receives this data and decrypts it.

Since this is asymmetric, nobody else except the browser can decrypt the data even if a third party has the public key of the browser.

19. What is HTTP and HTTPS protocol ?

HTTP is the HyperText Transfer Protocol which defines the set of rules and standards on how the information can be transmitted on the World Wide Web (WWW). It helps the web browsers and web servers for communication. It is a 'stateless protocol' where each command is independent with respect to the previous command. HTTP is an application layer protocol built upon the TCP. It uses port 80 by default.

HTTPS is the HyperText Transfer Protocol Secure or Secure HTTP. It is an advanced and a secured version of HTTP. On top of HTTP, SSL/TLS protocol is used to provide security. It enables secure transactions by encrypting the communication and also helps identify network servers securely. It uses port 443 by default.

20. What is SMTP protocol ?

SMTP is the Simple Mail Transfer Protocol. SMTP sets the rule for communication between servers. This set of rules helps the software to transmit emails

over the internet. It supports both End-to-End and Store-and-Forward methods. It is in always-listening mode on port 25.

21. TCP and UDP protocol, prepare differences

TCP is a connection-oriented protocol, whereas UDP is a connectionless protocol. A key difference between TCP and UDP is speed, as TCP is comparatively slower than UDP. Overall, UDP is a much faster, simpler, and efficient protocol, however, retransmission of lost data packets is only possible with TCP

TCP provides extensive error checking mechanisms. It is because it provides flow control and acknowledgment of data. UDP has only the basic error checking mechanism using checksums.

22. What happens when you enter "google.com" (very very famous question)

Steps :

- Check the browser cache first if the content is fresh and present in the cache display the same.
- If not, the browser checks if the IP of the URL is present in the cache (browser and OS) if not then requests the OS to do a DNS lookup using UDP to get the corresponding IP address of the URL from the DNS server to establish a new TCP connection.
- A new TCP connection is set between the browser and the server using three-way handshaking.
- An HTTP request is sent to the server using the TCP connection.
- The web servers running on the Servers handle the incoming HTTP request and send the HTTP response.
- The browser processes the HTTP response sent by the server and may close the TCP connection or reuse the same for future requests.
- If the response data is cacheable then browsers cache the same.
- Browser decodes the response and renders the content.

23. Hub vs Switch

Hub: Hub is a networking device which is used to transmit the signal to each port (except one port) to respond from which the signal was received. Hub is operated on a Physical layer. In this packet filtering is not available. It is of two types: Active Hub, Passive Hub.

Switch: Switch is a network device which is used to enable the connection establishment and connection termination on the basis of need. Switch is operated on the Data link layer. In this packet filtering is available. It is a type of full duplex transmission mode and it is also called an efficient bridge

24. VPN, advantages and disadvantages of it

VPN (Virtual Private Network) : VPN or the Virtual Private Network is a private WAN

(Wide Area Network) built on the internet. It allows the creation of a secured tunnel (protected network) between different networks using the internet (public network). By using the VPN, a client can connect to the organisation's network remotely.

Advantages of VPN :

1. VPN is used to connect offices in different geographical locations remotely and is cheaper when compared to WAN connections.
2. VPN is used for secure transactions and confidential data transfer between multiple offices located in different geographical locations.
3. VPN keeps an organisation's information secured against any potential threats or intrusions by using virtualization.
4. VPN encrypts the internet traffic and disguises the online identity

Disadvantages of VPN :

1. Not designed for continuous use
2. Complexity prevents scalability
3. Lack of granular security
4. Unpredictable performance
5. Unreliable availability

25. LAN

A local area network (LAN) is a collection of devices connected together in one physical location, such as a building, office, or home. A LAN can be small or large, ranging from a home network with one user to an enterprise network with thousands of users and devices in an office or school.