Q.1. What is Database?

- A Database is a logical, consistent, and organized collection of data that it can easily be accessed, managed, and updated.
- Databases, also known as electronic databases are structured to provide the facility of creation, insertion, updating of the data efficiently and are stored in the form of a file or set of files, on the magnetic disk, tapes, and another sort of secondary devices.

Q.2. What is DBMS?

- DBMS is a collection of programs that facilitates users to create and maintain a database.
- In other words, DBMS provides us an interface or tool for performing different operations such as the creation of a database, inserting data into it, deleting data from it, updating the data, etc.

Q.3. What is difference between Data and Information?

- Data is a raw and unorganized fact that is required to be processed to make it meaningful.
- Information is defined as a set of data that is processed according to the given requirement in a meaningful way.

Q.4. What are the advantages of DBMS?

- Reduce data redundancy
- Restriction of unauthorized access
- Provides multiple user interfaces
- Provides backup and recovery
- Enforces integrity constraints
- Ensure data consistency
- Easy accessibility
- Easy data extraction and data processing due to the use of queries

Q.5. What do you mean by transparent DBMS?

- The transparent DBMS is a type of DBMS which keeps its physical structure hidden from users.
- Physical structure or physical storage structure implies to the memory manager of the DBMS, and it describes how the data stored on disk.

Q.6. What is the difference between DBMS and RDBMS.

- RDBMS stands for Relational Database Management Systems.
- It is used to maintain the data records and indices in tables. RDBMS is the form of DBMS which uses the structure to identify and access data concerning the other piece of data in the database.
- RDBMS is the system that enables you to perform different operations such as- update, insert, delete, manipulate, and administer a relational database with minimal difficulties. Most of the time RDBMS use SQL language because it is easily understandable and is used for often.

Q.7. What is Relation Schema and a Relation?

- A Relation Schema is specified as a set of attributes.
- It is also known as table schema.
- It defines what the name of the table is.
- Relation schema is known as the blueprint with the help of which we can explain that how the data is organized into tables. This blueprint contains no data.
- A relation is specified as a set of tuples.
- A relation is the set of related attributes with identifying key attributes

Q.8. What is a degree of Relation?

- The degree of relation is number of attributes of its relation schema.
- A degree of relation is also known as Cardinality it is defined as the number of occurrences of one entity which is connected to the number of occurrences of other entity.
- There are three degree of relation they are one-to-one (1:1), one-to-many (1:M), many-to-one (M:M).

Q.9. What is data abstraction in DBMS?

- Data abstraction in DBMS is a process of hiding irrelevant details from users.
- Because database systems are made of complex data structures so, it makes accessible the user interaction with the database.

Q.10. What are the three levels of data abstraction?

Following are three levels of data abstraction:

- Physical level: It is the lowest level of abstraction. It describes how data are stored.
- Logical level: It is the next higher level of abstraction. It describes what data are stored in the database and what the relationship among those data is.
- View level: It is the highest level of data abstraction. It describes only part of the entire database.
- For example- User interact with the system using the GUI and fill the required details, but the user does not have any idea how the data is being used. So, the abstraction level is absolutely high in VIEW LEVEL.
- Then, the next level is for PROGRAMMERS as in this level the fields and records are visible and the programmer has the knowledge of this layer. So, the level of abstraction here is a little low in VIEW LEVEL.
- And lastly, physical level in which storage blocks are described.

Q.11. What is the difference between a database schema and a database state?

- The collection of information stored in a database at a particular moment in time is called database state while the overall design of the database is called the database schema.

Q.12. What are the three levels of data abstraction?

- Physical level: It is the lowest level of abstraction. It describes how data are stored.
- Logical level: It is the next higher level of abstraction. It describes what data are stored in the database and what relationship among those data.
- View level: It is the highest level of data abstraction. It describes only part of the entire database.

Q.13. What are the differences between network and hierarchical database model?

- Network Database Model:
  - ➢ Each parent node can have multiple children's nodes and vice versa.
  - ➢ Supports one-to-one, one-to-many, and many-to-many relationships
- Hierarchical Database Model:
  - ➢ A top-down structure where each parent node can have many child nodes. But a child node can have only a single parent node.
  - ➢ Supports one-tone and one-to-many relationships"

Q.14. What is Data Warehousing?

- A data warehouse is a centralized storage system that allows for the storing, analysing, and interpreting of data in order to facilitate better decision-making.

- Data Warehouse is the Central Data Store within your company.
- There is a need for Data Warehouse for all the enterprises that want to make data-driven decisions because a Data Warehouse is the "Single Source of Truth" for all the data in the company.

Q.15. What is Data dictionary and why it is required?
- A data dictionary contains metadata i.e data about the database.
- The data dictionary is very important as it contains information such as what is in the database, who is allowed to access it, where is the database physically stored etc.
- The users of the database normally don't interact with the data dictionary, it is only handled by the database administrators.

Q.16. What are the different types of Data dictionary?
- Active Data Dictionary
    - The data dictionary is automatically updated by the database management system when any changes are made in the database. This is known as an active data dictionary as it is self-updating.
- Passive Data Dictionary
    - The passive data dictionary has to be manually updated to match the database. This needs careful handling or else the database and data dictionary are out of sync.

Q.17. What are the unary operations in Relational Algebra?
- The select, project, and rename operations are called unary operations, because they operate on one relation.

Q.18. What are the binary operations in Relational Algebra?
- Union, intersection, difference, cartesian product, join, division comes under binary operation, because they operate on two tables.

Q.19. What is Relational Algebra?
- Relational Algebra is a Procedural Query Language which contains a set of operations that take one or two relations as input and produce a new relationship.
- Relational algebra is the basic set of operations for the relational model.
- There are few fundamental operations of relational algebra:
    - select
    - project
    - set difference
    - union
    - rename

Q.20. What do you understand by Data Model?
- The Data model is specified as a collection of conceptual tools for describing data, data relationships, data semantics and constraints.
- These models are used to describe the relationship between the entities and their attributes.
- There are various data models:
    - Hierarchical data model
    - network model
    - relational model

➢ Entity-Relationship model

Q.21. What do you understand by Data Model?

- The Data model is specified as a collection of conceptual tools for describing data, data relationships, data semantics and constraints.
- These models are used to describe the relationship between the entities and their attributes.
- There are various data models:
  - ➢ Hierarchical data model
  - ➢ network model
  - ➢ relational model
  - ➢ Entity-Relationship model

Q.22. What is the Relationship?

- The Relationship is defined as an association among two or more entities.
- There are three type of relationships in DBMS-
  - ➢ One-To-One: Here one record of any object can be related to one record of another object.
  - ➢ One-To-Many (many-to-one): Here one record of any object can be related to many records of other object and vice versa.
  - ➢ Many-to-many: Here more than one records of an object can be related to n number of records of another object.

Q.23. What is the E-R model?

- E-R model is a short name for the Entity-Relationship model.
- This model is based on the real world.
- It contains necessary objects (known as entities) and the relationship among these objects.
- Here the primary objects are the entity, attribute of that entity, relationship set, an attribute of that relationship set can be mapped in the form of E-R diagram.
- In E-R diagram, entities are represented by rectangles, relationships are represented by diamonds, attributes are the characteristics of entities and represented by ellipses, and data flow is represented through a straight line.

Q.24. What is an entity and entity set?

- An entity can be a real-world object which physically exists in this world.
- All the entities have their attribute which in the real world considered as the characteristics of the object.
- For example: In the employee database of a company, the employee, department, and the customer can be considered as the entities. These entities have some characteristics which will be the attributes of the corresponding entity.
- An entity may be concrete (a person or a book, for example) or abstract (like a holiday or a concept). An entity set is a set of entities of the same type (e.g., all persons having an account at a bank).

Q.25. What is difference between strong entity set and weak entity set?

- The strong entity has a primary key where as Weak entities are dependent on strong entity.
- A strong entity is independent, whereas a weak entity is reliant on another entity.

Q.26. What is an attribute?

- It is used to describe the property of an entity.

- An attribute can be defined as the characteristics of the entity.
- Entities can be uniquely identified using the attributes.
- Attributes represent the instances in the row of the database.
- For example: If a student is an entity in the table then name, age, height, weight will be the attribute of that student.

Q.27. Differentiate specialization and generalization
- Generalization collects the common features of multiple entities to form a new entity whereas Specialization divides an entity to form multiple new entities that inherit some feature of the splitting entity.
- Generalization is a bottom-up manner approach whereas Specialization is a top-down manner approach.

Q.28. In what situations should we use ER diagrams vs UML class diagrams?
- ER diagrams are used to model the data stored in a database, while UML class diagrams are used to model the structure of a system.
- If you are trying to model the relationships between the data in a database, then you would use an ER diagram. If you are trying to model the relationships between the different components of a system, then you would use a UML class diagram.

Q.29. Is it possible to define inheritance hierarchies with ER diagrams?
- Yes, it is possible to define inheritance hierarchies using ER diagrams.
- This is typically done by using a superclass/subclass relationship, where the superclass is the parent class in the inheritance hierarchy and the subclass is the child class.

Q.30. What is an identifying relationship?
- An identifying relationship is a relationship in which the primary key of the parent table is also a foreign key in the child table.
- This means that the child table cannot exist without the parent table, and the child table cannot be linked to more than one parent table.

Q.31. What do you understand about aggregation, composition, and association?
- Aggregation is a relationship between two classes where one class is a part of the other.
- Composition is a relationship between two classes where one class contains the other.
- Association is a relationship between two classes where one class is related to the other.

Q.32. What is a recursive relationship?
- A recursive relationship is one in which an entity is related to itself.
- For example, a self-employed person is both an employee and an employer.

Q.33. What is functional Dependency? Why functional dependency?
- Functional dependency in DBMS refers to a relationship that is present between attributes of any table that are dependent on each other.
- Functional dependency helps ensure the same data doesn't exist repetitively across a database or network of databases.

Q.34. What is normalization?
- Normalization is a database design technique that reduces data redundancy and eliminates undesirable characteristics like Insertion, Update and Deletion Anomalies.

- Normalization rules divides larger tables into smaller tables and links them using relationships.
- The purpose of Normalisation in SQL is to eliminate redundant (repetitive) data and ensure data is stored logically.
- There most used normal forms are:
  - First Normal Form(1NF)
  - Second Normal Form(2NF)
  - Third Normal Form(3NF)
  - Boyce & Codd Normal Form(BCNF)
  - Forth Normal Form(4NF)
  - Fifth Normal Form(5NF)

Q.35. What is Denormalization?

- Denormalization is the process of adding precomputed redundant data to an otherwise normalized relational database to improve read performance of the database.

Q.36. What is 1NF?

- A relation will be 1NF if it contains an atomic value.
- It states that an attribute of a table cannot hold multiple values. It must hold only single-valued attribute.
- First normal form disallows the multi-valued attribute, composite attribute, and their combinations.

Q.37. Why 1NF is used.

- 1NF is used because it removes the unnecessary repeating values by creating a separate table and hence does not create any issues while inserting, deleting, and updating the values in the database.

Q.38. What is 2NF?

- 2NF is the Second Normal Form. A table is said to be 2NF if it follows the following conditions:
  - The table is in 1NF, i.e., firstly it is necessary that the table should follow the rules of 1NF.
  - Every non-prime attribute is fully functionally dependent on the primary key.

Q.39. What is 3NF?

- 3NF stands for Third Normal Form. A database is called in 3NF if it satisfies the following conditions:
  - It is in second normal form.
  - There is no transitive functional dependency.
    We can represent transitive functional dependency as:
    If: $A \rightarrow B$ and $B \rightarrow C$ Then: $A \rightarrow C$

Q.40. What is BCNF?

- BCNF stands for Boyce-Codd Normal Form. It is an advanced version of 3NF, so it is also referred to as 3.5NF.
- BCNF is stricter than 3NF.
- A table complies with BCNF if it satisfies the following conditions:
  - It is in 3NF.
  - For every functional dependency X->Y, X should be the super key of the table. It merely means that X cannot be a non-prime attribute if Y is a prime attribute.

Q.41. What do you understand by query optimization?
- The term query optimization specifies an efficient execution plan for evaluating a query that has the least estimated cost.
- The concept of query optimization came into the frame when there were several methods, and algorithms existed for the same task then the question arose that which one is more efficient and the process of determining the efficient way is known as query optimization.
- There are many benefits of query optimization:
  - It reduces the time and space complexity.
  - More queries can be performed as due to optimization every query comparatively takes less time.
  - User satisfaction as it will provide output fast

Q.42. What is the main goal of RAID technology?
- RAID stands for Redundant Array of Inexpensive (or sometimes "Independent") Disks.
- RAID is a method of combining several hard disk drives into one logical unit (two or more disks grouped together to appear as a single device to the host system).
- It is a technology which is used to connect multiple secondary storage devices for increased performance, data redundancy or both.
- It gives you the ability to survive one or more drive failure depending upon the RAID level used.

Q.43. What is Index in SQL?
- The Index in SQL is a special table used to speed up the searching of the data in the database tables.
- It also retrieves a vast amount of data from the tables frequently.
- The INDEX requires its own space in the hard disk.
- The index concept in SQL is same as the index concept in the novel or a book.

Q.44. What is index types of indexes in SQL?
- There are two types of indexing in SQL.
  - Clustered index
  - Non-clustered index.

Q.45. What is difference between clustered and non-clustered Indexes?

Clustered Indexes	Non-clustered Indexes
A Clustered index is a type of index in which table records are physically reordered to match the index.	A Non-Clustered index is a special type of index in which the logical order of the index does not match the physical stored order of the rows on the disk.
The size of primary clustered index is large.	The size of the non-clustered index is smaller.
Primary Keys of the table by default are clustered indexes.	The composite key when used with unique constraints of the table act as the non-clustered index.
A clustered index is faster.	A non-clustered index is slower.

Q.46. What is a checkpoint in DBMS?
- The Checkpoint is a type of mechanism where all the previous logs are removed from the system and permanently stored in the storage disk.

Q.47. Why do We Need Checkpoints?

- Whenever transaction logs are created in a real-time environment, it eats up lots of storage space.
- Also keeping track of every update and its maintenance may increase the physical space of the system.
- Eventually, the transaction log file may not be handled as the size keeps growing.
- This can be addressed with checkpoints.
- The methodology utilized for removing all previous transaction logs and storing them in permanent storage is called a Checkpoint.

Q.48. What are the different types of Checkpoints?

- There are basically two main types of Checkpoints:
  - Automatic Checkpoint
  - Manual Checkpoint
- Automatic Checkpoint: These checkpoints occur very frequently like every hour or every day. These intervals are set by the database administrator. They are generally used by heavy databases as they are frequently updated, and we can recover the data easily in case of failure.
- Manual Checkpoint: These are the checkpoints that are manually set by the database administrator. Manual checkpoints are generally used for smaller databases. They are updated very less frequently only when they are set by the database administrator.

Q.49. What is Transaction? Why do we need transactions in DBMS?

- A transaction is a single unit of logic or work, sometimes made up of multiple operations.
- Transactions are required:
  - to manage concurrency
  - to satisfy ACID properties.
  - to maintain database integrity.
  - for protecting the user data from system failures.

Q.50. What is ACID properties?

- ACID properties are some basic rules, which must be satisfied by every transaction to preserve the integrity.
- These properties and rules are:
  - ATOMICITY: This property state that either the entire transaction takes place at once or does not happen at all. There is no midway i.e., transactions do not occur partially.
  - CONSISTENCY: This property state that the database must be consistent before and after the transaction.
  - ISOLATION: This property states that the number of the transaction can be executed concurrently without leading to the inconsistency of the database state.
  - DURABILITY: This property ensures that once the transaction is committed it will be stored in the non-volatile memory and system crash can also not affect it anymore.

Q.51. What is a Schedule?

- A Schedule is creating a group of multiple parallel transactions and executing them one after one.
- The order in which the instructions appear should be preserved in every transaction.
- In case two transactions are executed simultaneously, the result of one transaction may affect the output of the other.

Q.52. What is a Deadlock? how it can be resolved?

- A deadlock is a condition where two or more transactions are waiting indefinitely for one another to give up locks.
- Deadlock is said to be one of the most feared complications in DBMS as no task ever gets finished and is in waiting state forever.
- Deadlocks can be prevented by making all the transactions acquire all the locks at the same instance of time. So, once deadlock occurs, the only way to cure is to abort one of the transactions and remove the partially completed work.

Q.53. What is Starvation?

- Starvation is a situation where all the low-priority processes get blocked.
- In any system, requests for high and low-priority resources keep happening dynamically. Therefore, some policy is required to decide who gets support and when.

Q.54. What is the difference between a shared lock and exclusive lock?

- Shared lock: Shared lock is required for reading a data item. In the shared lock, many transactions may hold a lock on the same data item. When more than one transaction is allowed to read the data items then that is known as the shared lock.
- Exclusive lock: When any transaction is about to perform the write operation, then the lock on the data item is an exclusive lock. Because, if we allow more than one transaction then that will lead to the inconsistency in the database.

Q.55. What is a Live Lock? How live lock is differed from deadlock?

- In the case of a livelock, the states of the processes involved in a live lock scenario constantly change. On the other hand, the processes still depend on each other and can never finish their tasks.
- In a deadlock, processes involved in a deadlock are stuck indefinitely and do not make any state change. However, in a live lock scenario, processes block each other and wait indefinitely but they change their resource state continuously.
- The notable point is that the resource state change has no effect and does not help the processes make any progress in their task.

Q.56. How many components of SQL are?

- There are four types of components of SQL:
  - ➢ **Data Definition Language (DDL)**: DDL changes the structure of the table like creating a table, deleting a table, altering a table, etc.
  - ➢ All the command of DDL is auto-committed that means it permanently save all the changes in the database.
  - ➢ Here are some commands that come under DDL:
    - ▪ CREATE
    - ▪ ALTER
    - ▪ DROP
    - ▪ TRUNCATE
  - ➢ **Data Manipulation Language (DML):** DML commands are used to modify the database. It is responsible for all form of changes in the database.
  - ➢ The command of DML is not auto-committed that means it cannot permanently save all the changes in the database. They can be rollback.
  - ➢ Here are some commands that come under DML:

- INSERT
- UPDATE
- DELETE
- ➢ **DATA Control Language (DCL):** DCL commands are used to grant and take back authority from any database user.
- ➢ Here are some commands that come under DCL:
  - Grant
  - Revoke
- ➢ **Transaction Control Language (TCL)**: TCL commands can only use with DML commands like INSERT, DELETE and UPDATE only.
- ➢ These operations are automatically committed in the database that is why they cannot be used while creating tables or dropping them.
- ➢ Here are some commands that come under TCL:
  - COMMIT
  - ROLLBACK
  - SAVEPOINT

Q.57. What is Starvation?

- Starvation is a situation where all the low-priority processes get blocked.

In any system, requests for high and low-priority resources keep happening dynamically. Therefore, some policy is required to decide who gets support and when.

Q.58. What is the functionality of DML Compiler.

- The DML Compiler translates DML statements into a query language that the query evaluation engine can understand.
- DML Compiler is required because the DML is the family of syntax element which is very similar to the other programming language which requires compilation. So, it is essential to compile the code in the language which query evaluation engine can understand and then work on those queries with proper output.

Q.59. What are the integrity rules in DBMS?

- Integrity constraints are a set of rules. It is used to maintain the quality of information.
- Integrity constraints ensure that the data insertion, updating, and other processes must be performed in such a way that data integrity is not affected.
- Thus, integrity constraint is used to guard against accidental damage to the database.
  1. Domain constraints
     Domain constraints can be defined as the definition of a valid set of values for an attribute.
     The data type of domain includes string, character, integer, time, date, currency, etc.
     The value of the attribute must be available in the corresponding domain.
  2. Entity integrity constraints
     The entity integrity constraint states that primary key value cannot be null.
     This is because the primary key value is used to identify individual rows in relation an if the primary key has a null value, then we cannot identify those rows.
     A table can contain a null value other than the primary key field.
  3. Referential Integrity Constraints
     A referential integrity constraint is specified between two tables.

In the Referential integrity constraints, if a foreign key in Table 1 refers to the Primary Key of Table 2, then every value of the Foreign Key in Table 1 must be null or be available in Table 2.

4. Key constraints
Keys are the entity set that is used to identify an entity within its entity set uniquely. An entity set can have multiple keys, but out of which one key will be the primary key. A primary key can contain a unique and null value in the relational table.

Q.60. What is Join?

- Join is used to combine rows from two or more tables, based on a related/common column between them.
- Join is used to join/combine more than one table to get some specific results out of those tables.
- There are following types of joins:
  - ➢ Inner joins: Inner join is of 3 categories. They are:
  - ➢ Theta join
  - ➢ Natural join
  - ➢ Equi join
  - ➢ Outer joins: Outer join have three types.
    - ▪ Left outer join
    - ▪ Right outer join
    - ▪ Full outer join

Q.61. What is the difference between inner join and outer join?

- INNER JOIN returns the common and the matching records between the tables. OUTER JOIN returns all the records from the both tables.

Q.62. What is the difference between inner and outer join in SQL performance?

- INNER JOIN is generally faster than OUTER JOIN.
- Generally, an OUTER JOIN is slower than an INNER JOIN as it needs to return more number of records when compared to INNER JOIN.

Q.63. What is stored procedure? What is the use of stored procedure?

- A stored procedure is a group of SQL statements that have been created and stored in the database.
- The stored procedure increases the reusability as here the code or the procedure is stored into the system and used again and again that makes the work easy, takes less time in processing and decreases the complexity of the system.
- So, if you have a code which you need to use again and again then save that code and call that code whenever it is required.

Q.64. What is the difference between a DELETE command and TRUNCATE command?

- Using delete command we can delete records using WHERE clause where as in truncate we cannot delete a particular record using where condition.
- DELETE command can be rolled back where as TRUNCATE command cannot be rolled back.
- DELETE command maintain a log, that is why it is slow whereas TRUNCATE command does not maintain a log. That is why it is fast.

Q.65. What is stored procedure? What is the use of stored procedure?

- A stored procedure is a group of SQL statements that have been created and stored in the database.
- The stored procedure increases the reusability as here the code or the procedure is stored into the system and used again and again that makes the work easy, takes less time in processing and decreases the complexity of the system.
- So, if you have a code which you need to use again and again then save that code and call that code whenever it is required.

Q.66. How do you communicate with an RDBMS?

- You must use Structured Query Language (SQL) to communicate with the RDBMS.
- Using queries of SQL, we can give the input to the database and then after processing of the queries database will provide us the required output.

Q.67. Describe the types of keys?

- There are following types of keys:
  - ➢ Primary key: The Primary key is an attribute in a table that can uniquely identify each record in a table. It is compulsory for every table.
  - ➢ Candidate key: The Candidate key is an attribute or set of an attribute which can uniquely identify a tuple. The Primary key can be selected from these attributes.
  - ➢ Super key: The Super key is a set of attributes which can uniquely identify a tuple. Super key is a superset of the candidate key.
  - ➢ Foreign key: The Foreign key is a primary key from one table, which has a relationship with another table. It acts as a cross-reference between tables.

Q.68. What is the difference between primary key and unique constraints?

- The primary key cannot have NULL value, the unique constraints can have NULL values.
- There is only one primary key in a table, but there can be multiple unique constrains.

Q.69. What is stored procedure? What is the use of stored procedure?

- A stored procedure is a group of SQL statements that have been created and stored in the database.
- The stored procedure increases the reusability as here the code or the procedure is stored into the system and used again and again that makes the work easy, takes less time in processing and decreases the complexity of the system.
- So, if you have a code which you need to use again and again then save that code and call that code whenever it is required.

Q.70. What is the difference between having and where clause?

- WHERE is used to filter records before any groupings take place that is on single rows.
- GROUP BY aggregates/ groups the rows and returns the summary for each group. HAVING is used to filter values after they have been groups.

Q.71. What is Identity?

- Identity (or AutoNumber) is a column that automatically generates numeric values.
- A start and increment value can be set, but most DBA leave these at 1. A GUID column also generates numbers; the value of this cannot be controlled.
- Identity/GUID columns do not need to be indexed.

Q.72. What is a view in SQL? Why we need view?

- A view is a virtual table based on the result-set of an SQL statement. We can create it using create view syntax.
- Database views are saved in the database as named queries and can be used to save frequently used, complex queries.

Q.73. What is a Trigger? Why we need trigger?

- A database trigger is procedural code that is automatically executed in response to certain events on a particular table or view in a database.
- The trigger is mostly used for maintaining the integrity of the information on the database.

Q.74. What are the different types of trigger?

- We can categorize the triggers in SQL Server in mainly three types:
    1. Data Definition Language (DDL) Triggers: DDL triggers are fired in response to the DDL events, such as CREATE, ALTER, and DROP statements. We can create these triggers at the database level or server level, depending on the type of DDL events.
    2. Data Manipulation Language (DML) Triggers: DML triggers are fired in response to DML events like INSERT, UPDATE, and DELETE statements in the user's table or view. It can also be executed in response to DML-like operations performed by system-defined stored procedures.
    3. Logon Triggers: Logon triggers are fires in response to a LOGON event. The LOGON event occurs when a user session is generated with an SQL Server instance, which is made after the authentication process of logging is completed but before establishing a user session.

Q.75. What are the benefits of using SQL triggers?

- The benefits of using SQL triggers include:
    - ➢ Data integrity: Triggers allow you to enforce complex business rules and constraints at the database level, ensuring that data remains consistent and accurate.
    - ➢ Automation: Triggers can automate repetitive or complex tasks by executing predefined actions whenever a specified event occurs. This reduces the need for manual intervention and improves efficiency.
    - ➢ Audit trails: Triggers can be used to track changes made to data, such as logging modifications in a separate audit table. This helps in auditing and maintaining a history of data changes.
    - ➢ Data validation: Triggers can perform additional validation checks on data before it is inserted, updated, or deleted, ensuring that only valid and conforming data is stored in the database.

Q.76. What is the difference between Trigger and Stored Procedure?

- Unlike Stored Procedures, Triggers cannot be called directly. They can only be associated with queries.

Q.77. What are indexes?

- A database index is a data structure that improves the speed of data retrieval operations on a database table at the cost of additional writes and the use of more storage space to maintain the extra copy of data.

- Data can be stored only in one order on a disk. To support faster access according to different values, faster search like binary search for different values is desired, For this purpose, indexes are created on tables.
- These indexes need extra space on the disk, but they allow faster search according to different frequently searched values.

Q.78. What is CLAUSE in SQL?
- A clause in SQL is a part of a query that lets you filter or customize how you want your data to be queried to you.

Q.79. What is cursor and what is need of cursor.
- Cursors allow us to process data row-by-row, which can be useful when we need to perform complex calculations or transformations on the data.
- Cursor is used when we want to retrieve data, one row at a time, from a result set and process each record one by one.

Q.80. What are the different types of cursor?
- There are 2 types of Cursors: Implicit Cursors, and Explicit Cursors. These are explained as following below.
  - ➤ Implicit Cursors: Implicit Cursors are also known as Default Cursors of SQL SERVER. These Cursors are allocated by SQL SERVER when the user performs DML operations.
  - ➤ Explicit Cursors: Explicit Cursors are Created by Users whenever the user requires them. Explicit Cursors are used for Fetching data from Table in Row-By-Row Manner.

Q.81. Under what conditions should you use an index?
- Indexes should not be used on small tables.
- Indexes should not be used on columns that return a high percentage of data rows when used as a filter condition in a query's WHERE clause.

Q.82. What is the use of the DROP command and what are the differences between DROP, TRUNCATE and DELETE commands?
- DROP command is a DDL command which is used to drop/delete the existing table, database, index, or view from the database.
- The major difference between DROP, TRUNCATE and DELETE commands are:
- DROP and TRUNCATE commands are the DDL commands which are used to delete tables from the database.
- And when we make use of a DROP command, the tables get deleted permanently all the privileges and indexes that are related to the table also get deleted. This operation cannot be rolled back and so should be used only when necessary.
- However in case of TRUNCATE, only the data stored in a table is deleted and the structure of the table is preserved and you can re-insert data by the use of "INSERT INTO clause". It can be rolled back until the commit has been made.
- DELETE command, on the other hand, is a DML Command which is used to delete rows from the table and this can be rolled back, however it's considered slower than truncate. Using the delete command, we can delete 1 or more specific rows from the table."

Q.83. What is sub-query? How it is differ from Correlated sub-query.
- A sub-query is a query that appears inside another query statement.
- A sub-query obtains its results independently of its containing (outer) statement.

- A correlated sub-query requires values from its outer query in order to execute.

Q.84. What is the main difference between UNION and UNION ALL?

- Union combines the result set from multiple tables and returns distinct records into a single result set whereas union all combines the result set from multiple tables and returns all records into a single result set.

Q.85. How is the pattern matching done in the SQL?

- SQL pattern matching can be done using the LIKE operator.
- _ is used to match a single element and % is used to match a zero or more number of elements.
- The NOT operator is used to select elements that do not match the given pattern.

Q.86. What are temporary tables? When are they useful?

- Temporary tables exist solely for a particular session, or whose data persists for the duration of the transaction.
- The temporary tables are generally used to support specialized rollups or specific application processing requirements.
- Unlike a permanent table, space is not allocated to a temporary table when it is created. Space will be dynamically allocated for the table as rows are inserted. The CREATE GLOBAL TEMPORARY TABLE command is used to create a temporary table in Oracle.

Q.87. Do we consider NULL values the same as that of blank space or zero?

- A NULL value is not at all same as that of zero or a blank space.
- The NULL value represents a value which is unavailable, unknown, assigned or not applicable whereas zero is a number and blank space is a character.

Q.88. What are the differences between Hash join and Merge join?

- Merge join is used when projections of the joined tables are sorted on the join columns. Merge joins are faster and uses less memory than hash joins.
- Hash join is used when projections of the joined tables are not already sorted on the join columns. In this case, the optimizer builds an in-memory hash table on the inner table's join column.
- The optimizer then scans the outer table for matches to the hash table, and joins data from the two tables accordingly.
- The cost of performing a hash join is low if the entire hash table can fit in memory. Cost rises significantly if the hash table must be written to disk.

Q.89. What kind of data structure is an index?

- In most of the cases, B-Tree data structure is used to store the indexes.
- This is because of the time efficiency of B-Trees.
- In other words, B-trees are traversed, searched, inserted, deleted and updated in logarithmic time. In addition, B-Tree data are always sorted and stored.
- Hence it makes searching and inserting the data in known fraction of time.

Q.90. What do you mean by Self Join?

- In self-join operation, a table is joined with itself to retrieve the desired data.
- Every join operation needs two tables as a basic rule.
- Therefore, in self-join, a table is joined with an instance of the same table.

- By doing this, values of the two table columns are compared with each other, and the desired data is retrieved as the result set.

Q.91. What are the SQL constraints?
- SQL constraints specify conditions for a column or table to manage the data stored in tables effectively.
- The following are the commonly used SQL constraints.
  - NOT NULL - This condition ensures columns won't accept a NULL value.
  - UNIQUE - It ensures that all the values in a column must be unique.
  - CHECK - It ensures that all the column fields obey a specific condition.
  - DEFAULT - It provides a default value for the fields of a column unless no value is specified for the fields
  - CREATE INDEX - It ensures creating an index for tables so that retrieving data from the tables becomes easier
  - PRIMARY KEY - It must identify every row of a table
  - FOREIGN KEY - It must link tables based on common attributes

Q.92. What is Alias in SQL?
- SQL aliases help to assign temporary names for a table or column.
- It is used to simplify table or column names. And aliases can exist only for that query period.
- It can be created using the 'AS' keyword.
- Know that creation of an alias is in no way affecting the column names in the database.
- It can be applied when more than one table is involved in a query.

Q.93. Differentiate CHAR and VARCHAR data types in SQL?
- Char is a fixed-length character string data type whereas varchar is a variable-length character string data type.
- Char data type can be used when the character length is known whereas varchar data type is used when the character length is not clear

Q.94. What do you mean by DELETE CASCADE constraint?
- When a foreign key is created under this option, and if a referenced row in the parent table is deleted, the referencing row(s) in a child table also gets deleted.
- On similar tracks, when a referenced row is updated in a parent table, the referencing row(s) in a child table is also updated.

Q.95. What is MAGIC TABLE in triggers?
- Magic Tables are invisible tables or virtual tables.
- You can see them only with the help of Triggers in the SQL Server.
- Magic Tables are those tables that allow you to hold inserted, deleted and updated values during insert, delete and update DML operations on a table in SQL Server.