1. What is index; types of indices; pros and cons
-Indexes are not always appropriate. Again, for tables with only a few rows, it's actually quicker for MySQL to run through a sequential search than to make use of indexes. Similarly, when you want most or all of the rows in a table returned, a sequential search will be faster.

2. What's the difference between Primary key and Unique constraint?
-PRIMARY KEY constraint differs from the UNIQUE constraint in that; you can create multiple UNIQUE constraints in a table, with the ability to define only one SQL PRIMARY KEY per each table. Another difference is that the UNIQUE constraint allows for one NULL value, but the PRIMARY KEY does not allow NULL values.

3. Tell me about check constraint
-The CHECK constraint is used to limit the value range that can be placed in a column.

4. Difference between temp table and table variable
-A Temp table is easy to create and back up data. Table variable involves the effort when you usually create the normal tables. Temp table result can be used by multiple users. Table variable can be used by the current user only.

5. Difference between WHERE and HAVING
-A WHERE clause is used is filter records from a result. The filter occurs before any groupings are made.
-A HAVING clause is used to filter values from a group.

6. Difference between RANK() and DenseRank() — value gap
-ROW_NUMBER will always generate unique values without any gaps, even if there are ties.
-RANK can have gaps in its sequence and when values are the same, they get the same rank.
-DENSE_RANK also returns the same rank for ties, but doesn't have any gaps in the sequence.

7. COUNT(*) vs. COUNT(colName)
-As you've already learned, COUNT(*) will count all the rows in the table, including NULL values. On the other hand, COUNT(column name) will count all the rows in the specified column while excluding NULL values.

8. What's the difference between left join and inner join? JOIN and Subquery, which one has a better performance, why?
-Left join is part of outer join. Outer join has left join, right join, and full join.
-Subqueries can be used to return either a scalar (single) value or a row set; whereas, joins are used to return rows.

9. What is correlated subquery
-Correlated subqueries are used for row-by-row processing. Each subquery is executed once for every row of the outer query.
-A correlated subquery is evaluated once for each row processed by the parent statement. The parent statement can be a SELECT, UPDATE, or DELETE statement.

10. What is a CTE, why do we need CTE?
-CTE is stand for common table expression, which is use for create a recursive query.
if you need to reference/join the same data set multiple times you can do so by defining a CTE.

11. What does SQL Profiler do?
-SQL Server Profiler is an interface to create and manage traces and analyze and replay trace results. Events are saved in a trace file that can later be analyzed or used to replay a specific series of steps when diagnosing a problem.

12. What is SQL injection, how to avoid SQL injection?
-SQL Injection (SQLi) is a type of an injection attack that makes it possible to execute malicious SQL statements. These statements control a database server behind a web application. Attackers can use SQL Injection vulnerabilities to bypass application security measures.

13. Difference between SP and user defined function? When to use SP when to use function?
-SPs can change database objects. Inline User-Defined Functions can be treated like views with parameters and can be used in row set operations and JOINs. Cannot JOIN the output of a Stored procedure. UDF can be used in the SQL statements anywhere in the WHERE / HAVING / SELECT sections.

14. Criteria of Union and Union all? Difference between UNION and UNION ALL
-The only difference between Union and Union All is that Union extracts the rows that are being specified in the query while Union All extracts all the rows including the duplicates (repeated values) from both the queries.

15. Steps you take to improve SQL Queries
Improve SQL Query Performance. ...
Avoid Multiple Joins in a Single Query. ...
Eliminate Cursors from the Query. ...

Avoid Use of Non-correlated Scalar Sub Query. ...
Avoid Multi-statement Table Valued Functions (TVFs) ...
Creation and Use of Indexes. ...
Understand the Data. ...
Create a Highly Selective Index.
Position a Column in an Index
Drop Unused Indexes
Statistic Creation and Updates
Revisit Your Schema Definitions
Summary

16. concurrency problem in transaction
-Concurrency problems occur when multiple transactions execute concurrently in an uncontrolled manner.

17. what is deadlock, how to prevent
-Deadlock can be prevented by eliminating any of the four necessary conditions, which are mutual exclusion, hold and wait, no preemption, and circular wait. Mutual exclusion, hold and wait and no preemption cannot be violated practically. Circular wait can be feasibly eliminated by assigning a priority to each resource.

18. what is normalization, 1NF - BCNF, benefits using normalization
-Normalization organizes the columns and tables of a database to ensure that database integrity constraints properly execute their dependencies.

19. what are the system defined databases?
-System databases are defined by Microsoft and are needed for SQL Server to operate. These databases include Master, Model, MSDB, TempDB, Resource, Distribution database used in replication as well as the ReportServer and ReportServerTempDB databases used for Reporting Services.

20. composite key
-In database design, a composite key is a candidate key that consists of two or more attributes (table columns) that together uniquely identify an entity occurrence (table row). A compound key is a composite key for which each attribute that makes up the key is a foreign key in its own right.

21. candidate key
-A candidate key is a specific type of field in a relational database that can identify each unique record independently of any other data. Experts describe a candidate key of having "no redundant attributes" and being a "minimal representation of a tuple" in a relational database table.

22. DDL vs. DML

-The basic difference between DDL and DML is that DDL (Data Definition Language) is used to Specify the database schema database structure. On the other hand, DML (Data Manipulation Language) is used to access, modify or retrieve the data from the database.

23. ACID property

-A transaction is a single logical unit of work that accesses and possibly modifies the contents of a database. Transactions access data using read and write operations.
In order to maintain consistency in a database, before and after the transaction, certain properties are followed. These are called ACID properties.

24. table scan vs. index scan

-Table scan means iterate over all table rows. Index scan means iterate over all index items, when item index meets search condition, table row is retrived through index. Usualy index scan is less expensive than a table scan because index is more flat than a table.

25. Difference between Union and JOIN

-Join combine data into new columns, Unions combine data into new rows.