1. What is index; types of indices; pros and cons

**Index is an on-disk structure associated with a table that increase retrieval speed of rows from the table.**

**There two types of indexes:**

1. **Clustered index: physically sort the data; one table can only have one clustered index**
2. **Non clustered index: will not sort the data, will be sorted separately**

**Pros: index will help us improve retrieving speed**

**Cons: slow down other DML statements, and make extra space**

2. What's the difference between Primary key and Unique constraint?

**There are 4 different points:**

1. **Unique key can accept one and only one null value; Primary Key cannot accept any null values**
2. **One table can have multiple Unique Key, but only one Primary Key**
3. **Primary Key sort the data by default (ascending order), but Unique Key will not**
4. **Primary Key will by default create clustered index, and Unique Key will create non-clustered index.**

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

**There are 4 different points:**

1. **Both temp table and table variables are stored in tempdb.**
2. **Scope: temp table is in the local/global; table variable is in current batch**
3. **Size: temp table is more than 100 rows; table variable is less than 100 rows**
4. **Usage: temp table do not use in SP/FUNCTION; table variable can be used in SP/FUNCTION**

5. Difference between WHERE and HAVING

**There are 3 different points:**

1. **Both are used as filters, but HAVING applies only to group as a whole, and only filters on aggregation functions; WHERE applies to individual rows.**
2. **WHERE goes before aggregations, but HAVING filters after the aggregations.**
3. **WHERE can be used with SELECT, UPDATE or DELETE, but HAVING will only be used in SELECT.**

6. Difference between RANK() and DenseRank() — value gap

**If there is the same rank using RANK(), then there will be a gap for the next rank.**

**DENSERANK() will not skip any number in rank if there is a tie.**

7. COUNT(\*) vs. COUNT(colName)

**COUNT(\*) will include null values, but COUNT(ColumnName) will not include null values.**

8. What's the difference between left join and inner join? JOIN and Subquery, which one has a better performance, why?

**INNER JOIN is to return the records that have matching values in both tables.**

**LEFT JOIN is to return all records from the left table, and the matching records from the right table. For the non-matching records in the right table, the result set will return us null values**

**There are 3 different points between join and subquery:**

1. **JOIN can only be used in FROM; SUBQUERY can be used in SELECT, WHERE, FROM, and ORDER BY**
2. **SUBQUERY is easy to understand and maintain**
3. **Usually JOIN will have a better performance than SUBQUERY since SQL Server will have an optimizer in JOIN, not in SUBQUERY.**

9. What is correlated subquery

**A correlated subquery is that inner query is dependent on the outer query.**

10. What is a CTE, why do we need CTE?

**CTE stands for common table expression to specify a temporary named result set.**

**We need CTE is because it can improve readability and manageability of complex SQL statements.**

11. What does SQL Profiler do?

**SQL Profiler can help us to trace all the changes in SQL Server.**

12. What is SQL injection, how to avoid SQL injection?

**SQL injection is a code injection technique that might destroy your database.**

**Stored Procedure (SP) can take parameters, so that it can be used to prevent SQL injection.**

13. Difference between SP and user defined function? When to use SP when to use function?

**There are 4 different points between SP and user defined function:**

1. **Usage: SP for DML; User defined function for calculations**
2. **How to call: SP will be called by its name; User defined function will be called in SQL query**
3. **Output: SP may or may not have output; User defined function must return some value**
4. **SP can call User defined function, but User defined function cannot call SP**

14. Criteria of Union and Union all? Difference between UNION and UNION ALL

**The Criteria of UNION and UNION ALL is the number of columns must be the same and column's types must be identical.**

**There are 3 different points between UNION and UNION ALL:**

1. **UNION remove all duplicate records, but UNION ALL will not**
2. **UNION sort the first column ascendingly, but UNION ALL will not**
3. **UNION cannot be used in recursive CTE, but UNION ALL can**

15. Steps you take to improve SQL Queries

1. **look at the execution plan**
2. **choose index wisely**
3. **avoid unnecessary joins**
4. **avoid SELECT \***
5. **JOIN to replace subquery**
6. **derived table to avoid a lot of grouping by**

16. concurrency problem in transaction

**1. dirty reads:**

**t1 allows t2 to read uncommitted data and then t1 rolled back;**

**caused by isolation level read uncommitted;**

**solved by isolation level read committed**

**2. lost update:**

**t1 and t2 read and update the same data but t2 finish its work earlier than t1, then t2 will lost their update;**

**caused by isolation level read committed;**

**solved by isolation level repeatable read**

**3. non repeatable read:**

**t1 read the same data twice while t2 is updating the data;**

**caused by isolation level read committed;**

**solved by isolation level repeatable read**

**4. phantom read:**

**t1 reads the same data twice while t2 is inserting records;**

**caused by isolation level repeatable read;**

**solved by isolation level serializable**

17. what is deadlock, how to prevent

**A deadlock occurs when two or more processes lock the separate resource.**

**To Prevent deadlock, we need to cluster the foreign key.**

18. what is normalization, 1NF - BCNF, benefits using normalization

**Database Normalization is a process of organizing data to minimize redundancy (data duplication), which in turn ensures data consistency.**

**First Normal Form: One cell can only have one value without any repeating groups.**

**Second Normal Form: 1NF + remove all the Partial Dependency.**

**Third Normal Form: 2NF + remove all the Transitive Dependency.**

**BCNF (Boyce Codd Normal Form): 3NF + remove all the Functional Dependency.**

**Benefits using normalization:**

1. **It will organize the data that reduce data redundancy**
2. **It can make sure our data is consistent**

19. what are the system defined databases?

**There are 4 system defined databases:**

1. **Master**
2. **Model**
3. **Msdb**
4. **Tempdb**

20. composite key

**When a PRIMARY KEY made by the combination of two or more columns to uniquely identify every row in a table. This is usually seen in JOIN tables.**

21. candidate key

**A candidate key is a key that is not a PK but eligible to be a PK. Mostly, it is a unique key without null values.**

22. DDL vs. DML

**DDL stands for data definition language includes CREATE, ALTER, and DROP. They will be used to define the data structures, including the databases and tables.**

**DML stands for data manipulation language including SELECT, INSERT, UPDATE, and DELETE. This wishes about manipulating data itself.**

23. ACID property

**Atomicity: work is atomic**

**Consistency: whatever happens in the middle of the transaction, this property will never leave your database in half-completed state**

**Isolation: two transactions will be isolated from each other by locking the resource**

**Durability: once the transaction is completed, then the changes it has made to the database will be permanent**

24. table scan vs. index scan

**Table scan means that our table doesn’t have any indexes.**

**Index scan means that our table have an index.**

25. Difference between Union and JOIN

**JOIN is to combine rows from two or more tables, based on a related column between them.**

**UNION is to combine different result sets vertically by adding rows from multiple result sets together.**