**1. What Is SQL Server?**

This is one of the most basic ms SQL server interview questions, so it’s really essential to know it well.

That said, your interviewer is likely looking for you to share that SQL Server is a relational database management system (RDBMS) from Microsoft that’s used to store, process and secure data in a structured way. It also offers features like analysis, reporting, integration, master data management, and even machine learning.

**2. What Are Foreign and Primary Keys in SQL Server?**

This is one of those MSSQL interview questions where I can show my fundamental understanding of how primary and foreign keys differ. It’s also helpful to refer to any [SQL projects](https://hackr.io/blog/best-sql-projects) I may have built to fully demonstrate my knowledge here.

That said, I would summarize the following:

* **Primary key:** Can be one column or a group of columns to uniquely identify table rows.
* **Foreign key:** This can also be one column or a group of columns that refers to the primary key from another. This is essential for referential integrity between tables in relational databases.

**21. What is the difference between UNION and UNION ALL?**

* UNION: To select related information from two tables UNION command is used. It is similar to JOIN command.
* UNION All: The UNION ALL command is equal to the UNION command, except that UNION ALL selects all values. It will not remove duplicate rows, instead it will retrieve all rows from all tables.

### 35. What will be the maximum number of index per table?

For SQL Server 2008 100 Index can be used as maximum number per table. 1 Clustered Index and 999 Non-clustered indexes per table can be used in SQL Server.

1000 Index can be used as maximum number per table. 1 Clustered Index and 999 Non-clustered indexes per table can be used in SQL Server.

1 Clustered Index and 999 Non-clustered indexes per table can be used in SQL Server.

### 36. What is the difference between COMMIT and ROLLBACK?

Every statement between BEGIN and COMMIT becomes persistent to database when the COMMIT is executed. Every statement between BEGIN and ROOLBACK are reverted to the state when the ROLLBACK was executed.

## Difference between Functions and Stored Procedures in SQL Server

|  |  |
| --- | --- |
| Function | Stored Procedure |
| Returns a single value, either as a table or as a scalar, always. | Can return zero, a single value, or several values. |
| Run-time compilation and execution occur for functions. | The database contains stored procedures that have been parsed and compiled. |
| Only Select statements are allowed. Updating and inserting DML statements are allowed. | Capable of carrying out any action on database objects, such as DML and select statements. |
| Only input parameters are permitted. Output parameters are not supported. | Both input and output parameters are supported. |
| Does not permit the usage of Try...Catch blocks are used to handle exceptions. | Allows the use of Try...Catch blocks are used to handle exceptions. |
| Transactions are not permitted within a function. | A stored procedure can contain transactions. |
| A function cannot call a stored procedure. | A stored procedure can be called a function. |
| A Select statement can invoke functions. | Stored procedures can't be accessed by Select/Where or Having statements. To run a stored procedure, use the Execute statement. |
| In JOIN clauses, functions can be used. | JOIN clauses can't use stored procedures. |

**3. What Is a Clustered and Non-Clustered Index in SQL Server?**

Clustered indexes are like a well-organized filing system for data. They determine the physical order of data in a table based on the primary key or a unique column. This makes retrieving information more efficient.

On the other hand, non-clustered indexes act as quick access paths for frequently queried columns. They don't affect the physical order of data but help locate specific information faster.

In a nutshell, clustered indexes bring order to data, while non-clustered indexes speed up data retrieval.

**4. Explain What SQL Profiler Is.**

This handy SQL Server tool captures and analyzes SQL Server events, enabling database administrators and developers to monitor SQL Server activities, including executed queries, [stored procedures](https://hackr.io/blog/stored-procedures), and error reports.

**5. What Are Normalization and Denormalization in SQL Server?**

Normalization involves breaking a large table into smaller tables and establishing relationships using foreign keys to minimize data redundancy and dependency.

Denormalization is the opposite process, and it involves adding redundant data to a normalized database to improve query performance. This is because it excludes the need for expensive joins or subqueries.

**6. What Is An Index In SQL Server, And How Does It Improve Query Performance?**

This is another of those relatively simple questions on SQL server that you need to know well.

Your interviewer would like to hear that this is a database structure that uses keys and pointers to locate and access data quickly. By creating indexes on columns that are frequently used, SQL Server can avoid scanning the entire table and perform efficient index seeks or scans, which leads to faster query performance.

**7. What Is a Transaction in SQL Server?**

This is a logical unit of one or more database operations that are executed together to maintain consistency and integrity. Transactions ensure data integrity by allowing multiple changes to be made as a single atomic operation.

**8. How Does a Stored Procedure Improve Performance in SQL Server?**

It improves performance by being pre-compiled, optimized for performance, reducing network traffic, and enhancing security.

**9. How Would You Define a Trigger in an SQL Server? And List The Types of Triggers.**

This is a special type of database object that contains a set of SQL statements that are automatically executed after specific data manipulation events like INSERT, UPDATE, or DELETE.

The different types of triggers in SQL Server are:

* DML triggers for data manipulation
* DDL triggers for data definition
* CLR triggers for executing CLR code
* Logon triggers for user logins
* Database triggers for database-level events.

**10. What Is A Cursor In SQL Server, And When Would You Use It?**

It is a database object used to retrieve and manipulate data from a result set one row at a time. It is used in situations where complex data manipulations cannot be easily accomplished with a single SQL statement. For instance:

* When iterating through a result set and performing calculations or transformations on each row individually
* Updating or deleting rows based on complex criteria
* Processing data in a specific order
* Performing multiple operations on a single row of data before moving to the next row.

**11. How Would You Define the CHECK Constraint In SQL Server?**

As with all types of constraints, the CHECK constraint is used to specify rules for the data in a table. It is a type of constraint that ensures a condition is met before a row can be inserted or updated in a table.

This condition is defined using a Boolean expression that evaluates to TRUE or FALSE and can reference one or more columns in the table.

**12. What Is a Subquery in SQL Server, And How Does It Differ From A Regular Query?**

A subquery is a nested query that is used to retrieve data for the main query. It is enclosed in parentheses and typically appears within a WHERE or HAVING clause.

A subquery differs from a regular query because it is nested within another query, while a regular query stands alone.

**13. Mention The Most Common Types Of Queries In SQL Server.**

With this question, you can show that you’re aware of various types of SQL command, such as DDL, DML, DCL, TCL, and DQL. You can then mention the most common types of query:

* **SELECT query:** retrieves data from one or more tables.
* **INSERT query:** inserts new rows of data into a table.
* **UPDATE query:** modifies existing rows of data in a table.
* **DELETE query:** removes rows of data from a table.
* **MERGE query:** combines insert, update, or delete operations on a table based on join results with another table.

**14. When Would You Use A CTE (Common Table Expression) In SQL Server?**

Here are some common scenarios where you might use a CTE:

* + **Recursive Queries:** a situation where a recursive query is necessary to navigate through hierarchical data structures (e.g., organizational charts or bill of materials).
  + **Simplifying Complex Queries:** When faced with complex queries, CTEs can simplify these by enhancing readability and maintainability.
  + **Reusing Subqueries:** If you need to reference a subquery multiple times within a larger query, a CTE can help you define the subquery once and refer to it multiple times.

**15. What Is The Difference Between A Temporary Table And A Table Variable In SQL Server?**

|  |  |
| --- | --- |
| **Temporary Table** | **Table Variable** |
| Stored in the tempdb database | Stored in memory |
| Can be created and dropped like regular tables | Automatically dropped at the end of the scope of the query or stored procedure |
| Can have indexes and constraints | Cannot have indexes or constraints, but can have primary keys |
| Can be used by multiple users and sessions | Can only be used by the session that created it |
| Can store large amounts of data | Can store small amounts of data |
| Can be used with SELECT INTO to create a new table based on the results of a query | Cannot be used with SELECT INTO to create a new table based on the results of a query |

**17. Explain What Scheduled Tasks Refer To In SQL Server.**

Scheduled tasks are automated jobs or processes that run at specific times or intervals. These tasks can include various activities like executing stored procedures, running scripts or queries, performing database maintenance, and more.

SQL Server also provides a built-in scheduling system called SQL Server Agent which you can use to create and manage scheduled tasks. This is a great way for administrators to automate routine database operations and optimize system performance.

**18. What Is The Method For Handling Exceptions In SQL Server Programming?**

The TRY-CATCH block is essential for managing exceptions gracefully. As a developer, it allows you to define code that might throw an exception within a "try" block and subsequently manage any resulting exceptions in a corresponding "catch" block.

In the catch block, it is possible to analyze the exception and execute relevant actions, such as logging an error message or undoing a transaction. For more intricate error-handling situations, the TRY-CATCH block can also be nested.

**Intermediate SQL Server Interview Questions**

**21. How Would You Delete A Table Using SQL Server Management Studio?**

To delete a table in SQL Server Management Studio, follow these steps:

* Open SQL Server Management Studio and connect to the server that contains the database with the table you want to delete.
* Expand the Databases folder to display a list of databases on the server.
* Expand the database that contains the table you want to delete.
* Expand the Tables folder to display a list of tables in the database.
* Right-click the table you want to delete and select Delete from the context menu.
* A dialog box will appear, asking you to confirm the deletion. If you are sure you want to delete the table, click OK.
* The table will be deleted from the database.

**28. How Would You Optimize A Slow-Running SQL Server Query?**

Optimizing a slow-running SQL Server query involves identifying the performance bottlenecks and addressing them. Here are some ways to optimize a slow-running SQL Server query:

* **Examine the query execution plan** to identify performance bottlenecks like table scans, expensive joins, or missing indexes.
* **Ensure appropriate indexes** are used to support query filtering, sorting, and joining operations.
* **Rewrite the query**to simplify complex expressions by removing redundant joins or subqueries, using query restructuring or common table expressions (CTEs).
* **Update statistics** to ensure optimal query plans via the UPDATE STATISTICS command.
* **Ensure sufficient resources** are available to handle the query workload and monitor performance with SQL Server Profiler or Performance Monitor.
* **Consider query caching**if the query is executed frequently or use materialized views to precompute and store complex query results.
* **Partitioning or data archiving** can improve query performance by reducing the data volume accessed during query execution.
* **Test the optimized query** to ensure changes have resulted in improved performance.

**29. How Do You Perform Pagination In SQL Server?**

You have two options here, with the first being the OFFSET-FETCH clause and the second being the ROW\_NUMBER() function, which also needs to use the ORDER BY and WHERE clauses.

Depending on your preference, OFFSET-FETCH fetches a range of rows, while ROW\_NUMBER() assigns a unique integer to each row returned by the query.

**30. Where Are Usernames And Passwords Stored In SQL Server?**

Usernames and hashed passwords are stored in the "sys.sql\_logins" system catalog view, which is a dynamic management view (DMV) in the "master" database. The passwords are not stored in clear text; instead, they are stored as a salted hash value for security purposes.

**35. How Do You Create An SQL Server Backup And Restore It?**

To create a SQL Server backup and restore it, you can follow these general steps:

To backup:

* Open SQL Server Management Studio and connect to the required database.
* Right-click on the database and select "Tasks" > "Back Up".
* In the "Back Up Database" window, choose backup type, destination, and backup options, then click "OK" to start the backup process

To restore:

* To restore the database, right-click on the "Databases" node in Object Explorer and select "Restore Database".
* In the "Restore Database" window, select the source of the backup, the destination database name and location, and the restore options.
* Click "OK" to start the restore process.

**38. How Do You Create A Stored Procedure In SQL Server?**

To create a stored procedure, you can use the CREATE PROCEDURE statement followed by the name of the stored procedure, input parameters, and procedure logic, as shown in the code block below.

CREATE PROCEDURE proc\_name

@param1 data\_type = def\_val,

@param\_n data\_type = def\_val

AS

BEGIN

*-- Add logic and statements here*

END;

**43. What Is the Meaning of SSRS in SQL Server?**

SSRS refers to SQL Server Reporting Services, which is a server-based reporting platform that you can use to create, manage, and deliver a wide range of reports for data analysis and decision-making.

This differs from a traditional index (which covers the entire table) because you can create this index on a specific subset of data that meets your filter criteria, resulting in a smaller index size and improved query performance.