

T-SQL Sample Questions

1. What is RDBMS?

Relational Data Base Management Systems (RDBMS) are database management systems that maintain data records and indices in tables. Relationships may be created and maintained across and among the data and tables. In a relational database, relationships between data items are expressed by means of tables. Interdependencies among these tables are expressed by data values rather than by pointers. This allows a high degree of data independence. An RDBMS has the capability to recombine the data items from different files, providing powerful tools for data usage.

2. What is PRIMARY KEY?

A PRIMARY KEY constraint is a unique identifier for a row within a database table. Every table should have a primary key constraint to uniquely identify each row and only one primary key constraint can be created for each table. The primary key constraints are used to enforce entity integrity.

3. What is UNIQUE KEY constraint?

A UNIQUE constraint enforces the uniqueness of the values in a set of columns, so no duplicate values are entered. The unique key constraints are used to enforce entity integrity as the primary key constraints.

4. What is FOREIGN KEY?

A FOREIGN KEY constraint prevents any actions that would destroy links between tables with the corresponding data values. A foreign key in one table points to a primary key in another table. Foreign keys prevent actions that would leave rows with foreign key values when there are no primary keys with that value. The foreign key constraints are used to enforce referential integrity.

5. What is CHECK Constraint?

A CHECK constraint is used to limit the values that can be placed in a column. The check constraints are used to enforce domain integrity.

6. What is NOT NULL Constraint?

A NOT NULL constraint enforces that the column will not accept null values. The not null constraints are used to enforce domain integrity, as the check constraints

7. What's the difference between a primary key and a unique key?

Both primary key and unique key enforces uniqueness of the column on which they are defined. But by default primary key creates a clustered index on the column, where unique creates a nonclustered index by default. Another major difference is that, primary key doesn't allow NULLs, but unique key allows one NULL only.

8. What is normalization?

Database normalization is a data design and organization processes applied to data structures based on rules that help build relational databases. In relational database design, the process of organizing data to minimize redundancy. Normalization usually involves dividing a database into two or more tables and defining relationships between the tables. The objective is to isolate data so that additions, deletions, and modifications of a field can be made in just one table and then propagated through the rest of the database via the defined relationships.

9. What are different normalization forms?

- 1NF: Eliminate Repeating Groups Make a separate table for each set of related attributes, and give each table a primary key. Each field contains at most one value from its attribute domain.
- 2NF: Eliminate Redundant Data If an attribute depends on only part of a multi-valued key, remove it to a separate table.
- 3NF: Eliminate Columns Not Dependent On Key If attributes do not contribute to a description of the key, remove them to a separate table. All attributes must be directly dependent on the primary key
- BCNF: Boyce-Codd Normal Form If there are non-trivial dependencies between candidate key attributes, separate them out into distinct tables.

10. What is Stored Procedure?

A stored procedure is a named group of SQL statements that have been previously created and stored in the server database. Stored procedures accept input parameters so that a single procedure can be used over the network by several clients using different input data. And when the procedure is modified, all clients automatically get the new version. Stored procedures reduce network traffic and improve performance. Stored procedures can be used to help ensure the integrity of the database.
e.g. sp_helpdb, sp_renamedb, sp_depends etc.

11. What are the advantages of using Stored Procedures?

- Stored procedure can reduced network traffic and latency, boosting application performance.
- Stored procedure execution plans can be reused, staying cached in SQL Server's memory, reducing server overhead.
- Stored procedures help promote code reuse.
- Stored procedures can encapsulate logic. You can change stored procedure code without affecting clients.
- Stored procedures provide better security to your data.

12. What is Trigger?

A trigger is a SQL procedure that initiates an action when an event (INSERT, DELETE or UPDATE) occurs. Triggers are stored in and managed by the DBMS. Triggers are used to maintain the referential integrity of data by changing the data in a systematic fashion. A trigger cannot be called or executed; the DBMS automatically fires the trigger as a result of a data modification to the associated table. Triggers can be viewed as similar to stored procedures in that both consist of procedural logic that is stored at the database level. Stored procedures, however, are not event-drive and are not attached to a specific table as triggers are. Stored procedures are explicitly executed by invoking a CALL to the procedure while triggers are implicitly executed. In addition, triggers can also execute stored procedures. Nested Trigger: A trigger can also contain INSERT, UPDATE and DELETE logic within itself, so when the trigger is fired because of data modification it can also cause another data modification, thereby firing another trigger. A trigger that contains data modification logic within itself is called a nested trigger.

13. What is View?

A simple view can be thought of as a subset of a table. It can be used for retrieving data, as well as updating or deleting rows. Rows updated or deleted in the view are updated or deleted in the table the view was created with. It should also be noted that as data in the original table changes, so does data in the view, as views are the way to look at part of the original table. The results of using a view are not permanently stored in the database. The data accessed through a view is actually constructed using standard T-SQL select command and can come from one to many different base tables or even other views.

14. What is Index?

An index is a physical structure containing pointers to the data. Indices are created in an existing table to locate rows more quickly and efficiently. It is possible to create an index on one or more columns of a table, and each index is given a name. The users cannot see the indexes; they are just used to speed up queries. Effective indexes are one of the best ways to improve performance in a database application. A table scan happens when there is no index available to help a query. In a table scan SQL Server examines every row in the table to satisfy the query results. Table scans are sometimes unavoidable, but on large tables, scans have a terrific impact on performance.

15. What are the difference between clustered and a non-clustered index?

- A **Clustered index** is a special type of index that reorders the way records in the table are physically stored. Therefore table can have only one clustered index. The leaf nodes of a clustered index contain the data pages.
- 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 disk. The leaf node of a non-clustered index does not consist of the data pages. Instead, the leaf nodes contain index rows.

16. Difference between Function (UDF) and Stored Procedure (SP)?

- SP-Must be executed explicitly with EXECUTE Statement whereas UDF-Must be called from SELECT Statement (Implicit calling)
- SP may or may not return any value whereas UDF must return something (scalar/table-valued)
- SP can have OUT parameters whereas UDF do not have any OUT parameters
- SP cannot return 2D Data (must always be scalar INT) with RETURN statement whereas UDF can return 2D data along with scalar data
- SP can CREATE temporary tables whereas UDF do not have any access to temporary tables
- SP can include any DDL and DML statements whereas UDF cannot have DDL and cannot do DML with permanent tables/views.
- UDF can do DML only with table variables created inside of it whereas SP can call UDF but UDF cannot call a SP

17. What is OLTP (Online Transaction Processing)?

In OLTP - online transaction processing systems relational database design use the discipline of data modeling and generally follow the Codd rules of data normalization in order to ensure absolute data integrity. Using these rules complex information is broken down into its most simple structures (a table) where all of the individual atomic level elements relate to each other and satisfy the normalization rules.

18. What is difference between DELETE and TRUNCATE commands?

Delete command removes the rows from a table based on the condition that we provide with a WHERE clause. Truncate will actually remove all the rows from a table and there will be no data in the table after we run the truncate command.

TRUNCATE:

- TRUNCATE is faster and uses fewer system and transaction log resources than DELETE.
- TRUNCATE removes the data by deallocating the data pages used to store the table's data, and only the page deallocations are recorded in the transaction log.

- TRUNCATE removes all rows from a table, but the table structure, its columns, constraints, indexes and so on, remains. The counter used by an identity for new rows is reset to the seed for the column.
- You cannot use TRUNCATE TABLE on a table referenced by a FOREIGN KEY constraint. Because TRUNCATE TABLE is not logged, it cannot activate a trigger.
- TRUNCATE cannot be rolled back.
- TRUNCATE is DDL Command.
- TRUNCATE Resets identity of the table.

DELETE:

- DELETE removes rows one at a time and records an entry in the transaction log for each deleted row.
- If you want to retain the identity counter, use DELETE instead. If you want to remove table definition and its data, use the DROP TABLE statement.
- DELETE Can be used with or without a WHERE clause
- DELETE Activates Triggers.
- DELETE can be rolled back.
- DELETE is DML Command.
- DELETE does not reset identity of the table.

Note: DELETE and TRUNCATE both can be rolled back when surrounded by TRANSACTION if the current session is not closed. If TRUNCATE is written in Query Editor surrounded by TRANSACTION

19. What is the difference between a HAVING CLAUSE and a WHERE CLAUSE?

They specify a search condition for a group or an aggregate. But the difference is that HAVING can be used only with the SELECT statement. HAVING is typically used in a GROUP BY clause. When GROUP BY is not used, HAVING behaves like a WHERE clause. Having Clause is basically used only with the GROUP BY function in a query whereas WHERE Clause is applied to each row before they are part of the GROUP BY function in a query.

20. Types of SQL Statement:

- **DDL** (Data Definition Language, contains **Create, Alter, Drop**) --- Used for Table creation.
E.g. create table student (StudentID Int, Name Varchar (50), Class Varchar (20);
- **DML** (Data Manipulation Language, contains **Insert, Update, Delete**) --- Used for data management.

E.g. Insert into student (StudentID, Name, Class) Values (1,'Alex', 'Graduation');
Update table Student Set Name='Tom' where StudentID= 111;
Delete from Student where StudentID = 111;
- **DQL** (Data Query Language, contain **Select**) --- Used data retrieval.
E.g. Select Name from Student where StudentID =111;
Select * from Student;
- **DCL** (Data Control Language, contains **Grant, Revoke, Deny**) --- Used for granting and revoking permissions to users.