**Difference between primary key, foreign key and unique key?**

Basic, Null value acceptance, number of keys that can be defined in the table, foreign key,index

**A primary key** is used to server as a unique identifier for each row in a table. The primary key dose not allow duplicate values and Null values. Only one primary key can be defined in the table. A primary key can be referenced by foreign key of another table. A clustered index automatically created when a primary key is defined.

**Unique key** uniquely determines a row in the table. Unique key does allow one Null value for column. A table can have more than one unique key. Unique key can be also referenced by foreign key of another table. Unique key generates the non-clustered index.

**Foreign key:** a key used to link two tables together. A FOREIGN KEY is a field (or collection of fields) in one table that refers to the PRIMARY KEY in another table. Foreign key can contain duplicate values and null values. More then one foreign key are allowed in a table.

**What is the difference between inner join and left outer join? What is join?**

A SQL Join statement is used to combine data or rows from two or more tables based on a common field between them. Four types of Joins are:

**Inner join**: Inner Join basically gives us records that have matching values in two tables. It is used to return all the rows from multiple tables where the join condition is satisfied. Let us suppose, we have two tables Table A and Table B. When we apply Inner Join on these two tables, we will get only those records that are common to both Table A and Table B.

Left Join:  Left Join in MySQL is used to return all the rows from the left table but only the matching rows from the right table where the join condition is meet.

**Right Join:**Right Join in MySQL is used to return all the rows from the right table but only the matching rows from the left table where the join condition is meet.

**Full Join:** Full join returns all the records when there is a match in any of the tables. Therefore, it returns all the rows from the left-hand side table and all the rows from the right-hand side table.

**INNER JOIN** keyword selects all rows from both tables as long as there is a match between the columns in both tables

**LEFT JOIN** keyword returns the matching rows from both tables and mismatching rows from the left table. The data is null is Null on the right side when there is no match.

**RIGHT JOIN** keyword returns the matching rows from both tables and mismatching rows from the right table. The data is null is Null on the left side when there is no match.

**FULL OUTER JOIN** keyword returns all rows from the left table and from the right table. It combines the result of both LEFT and RIGHT joins

**CROSS JOIN** Returns the Cartesian product of rows from tables in the join. In other words, it will produce rows which combine each row from the first table with each row from the second table

**Self Join** A self-join is joining a table to itself

**What is Normalization?**

Normalization rules divides larger tables into smaller tables and links them using relationships. The purpose of Normalization in SQL is to eliminate redundant (repetitive) data and ensure data is stored logically. There are five normal forms, but we commonly use first three. For the first Normal Form, it has single value for each column and all the columns in a table should have unique names. For the second normal form, it is based on 1NF (all the columns depend on the table’s primary key) and it should not have partial dependency. (all the columns are not transitively dependent on the primary key) Third Normal Form is based on 2NF and it doesn’t have transitive dependency.

Suppose we have a table contains employee and department attributes. The company just hired a person and hasn’t decide which department the new employee will be assigned to. I can’t insert this new employee into the database, because we have the department information of this person. Let’s say the mechanical department of the company only has one person. However, one day this employee decide to leave the company. In the database, we will not only delete the person information but we will also delete mechanical department. This is not what we want to do. My company is small company, all the engineers belongs to engineering department. As the company expand, we hired more and engineers. We decide to separate engineering department into IT department and mechanical department. Hence, we need to update lost of data and this could also lead to data inconsistency. This is why need to use Normalization to split the data.

**Denormalization** is a strategy that database managers use to increase the performance of a database infrastructure. It involves adding redundant data to a normalized database. The main purpose of denormalization is to significantly speed up data retrieval.

**Difference between View & Temp Table?**

Create view view\_name as

**A view** is just a select sql query statement stored in the views folder of a database. A view can be consider as a virtual table which means it does not store any set of data in a database. If you have a sub query nested in a query, we can replace that sub query with a view to simply the query statement. This also gives us good reusability of code. View doesn’t allow user to access the underlying base tables directly, so view is like a protection mechanism.

Create table #table\_name( ); ##table\_name

**Temporary tables** are similar to the permanent tables. Permanent tables get created and remain in the database permanently until user drop them. Temporary tables get created in the tempdb and are automatically deleted, when they are no longer used.

Local temporary table: a local temporary table is available, only for the connection that has created the table. If I create another session to access the table, this is not allowed.

**A local temporary table** is automatically dropped, when the connection that has created the table, is closed. If the temporary table, is created inside the stored procedure, it is dropped automatically upon the completion of stored procedure execution. It is also possible for different connection, to create a local temporary table with the same name.

**Global temporary table:** Global temporary tables are visible to all connections of sql server, and are only destroyed then the lase connection referencing the table is closed. A global temporary table has unique name across multiple connections.

**Difference**

**A view** is just a select sql query statement stored in the views folder of a database. A view can be consider as a virtual table which means it does not store any set of data in a database.**Temporary** table is real table gets created in the database and it will automatically deleted, when they are no longer used.

**Performance**

views exist only for a single query, and each time you generate a view it is recreated from current data. In contrast, a temp table exists for the each session, and once you create a temp table it is always there until the session ends. That’s why temp table has better performance then a view.

**Update restrictions**

When a user tries to update rows of a view, the DBMS must translate the request into an update on rows of the underlying source tables. This is possible for simple views, but more complex views are often restricted to read-only. However, for the temp table, you can update, delete, insert as a common table.

**Difference between truncate and drop?**

The **DELETE**(DML) command is used to remove rows from a table based on WHERE condition. **TRUNCATE**(DDL) command is used to delete complete data from an existing table/Truncate removes all rows from a table. You can also use **DROP**(DDL) command to delete complete table but it would remove complete table structure form the database. Delete can be rollback where as Truncate and Drop can not be rollback. Truncate is faster then delete.

**Difference between Stored Procedure and Function?**

Stored procedure is used to do specific tasks. User defined function is used to define a value. Function muse return a value. Stored procedure could return a value or not. User defined function only allow select statements, where as stored procedure allows select statements and DML statements. That’ why you can not put stored procedures inside and transactions a function but functions can be called and transactions can be used within a stored procedure. function will allow only input parameters, doesn't support output parameters. Stored procedure can have both input and output parameters. Functions can be called from a select statement where as Procedures can't be called from Select statement.

**Difference between Where and Having?**

Where clause is used for filtering individual rows where as HAVING clause is used to filter groups.

WHERE comes before GROUP BY. HAVING comes after GROUP BY. That means where can’t be used with aggregates functions but having can be used with aggregates function. From a performance standpoint, having is slower than where and should be avoided when possible.

.We have a student table. In this table, we have students belong to different classes and come from different country. I want to find In Which country the total number of student is greater than a certain number in class 1. select from student where class = class 1 group by country where count(Id) > number.

**Different join and union?**

SQL joins are used to combine rows from two or more tables, based on a common fields between two tables. Union combines rows form 2 or more tables when you use join, you would see table becomes wider. When you use join, you would see table becomes longer.

**Difference between Union & Union ALL?**

Union combines rows form 2 or more tables, where as Joins combine columns from 2 or more tables.

Union and Union ALL operations are used to combine the result-set of select queries. Union removes duplicate rows, where as Union ALL does not. Union has to perform distinct sort to remove duplicates, which means it less faster than Union ALL.

For union and union all to work, the number, data types and the order of the columns in the select statements must be same.

**What is trigger?**

A trigger is a special type of stored procedure that automatically runs when an event occurs in the database server. DML triggers run when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements on a table or view.