1.The three types of joins used in SQL are:

(i) Inner Join: Inner Join returns only the matching rows from the two tables being joined. The join is performed based on the condition specified using the ON keyword or using the WHERE clause. If there is no matching row between the tables, then those rows will not be returned.

(ii) Left Join: Left Join returns all the rows from the left table and the matching rows from the right table. If there are no matching rows in the right table, then the result will have NULL values for the columns from the right table.

(iii) Right Join: Right Join returns all the rows from the right table and the matching rows from the left table. If there are no matching rows in the left table, then the result will have NULL values for the columns from the left table.

2. The difference between a NULL value, a 0 value and a blank value in SQL:

In SQL, NULL, 0, and blank values are all different types of data values that can be stored in a column of a table:

(i) NULL value: A NULL value represents the absence of a value or the lack of a value in a column. NULL is not the same as an empty string or a zero value. A NULL value indicates that the data is unknown or not applicable. When comparing a NULL value with any other value (including another NULL), the result is always unknown.

(ii) 0 value: A 0 value is a numeric value that represents the number zero. It is not the same as a NULL value or a blank value. In most cases, 0 is used to represent an actual value, such as the number of items in a list or the amount of money in an account.

(iii) Blank value: A blank value, also known as an empty string, is a text value that contains no characters. It is not the same as a NULL value or a 0 value. In some cases, an empty string may be used to represent a value that is not applicable or unknown.

3. The difference between HAVING and WHERE clause in SQL:

In SQL, both the HAVING and WHERE clauses are used to filter the data in a query. However, there are some differences between them.

(i) WHERE clause: The WHERE clause is used to filter rows based on a condition. It is used with the SELECT, UPDATE, and DELETE statements. The WHERE clause is applied before any grouping or aggregation is performed. The condition specified in the WHERE clause determines which rows are selected from the table to be included in the result set.

(ii) HAVING clause: The HAVING clause is used to filter groups based on a condition. It is used with the GROUP BY clause in a SELECT statement. The HAVING clause is applied after the groups have been formed, and it filters the groups based on the conditions specified. The condition specified in the HAVING clause determines which groups are included in the result set.

4. To sort data in SQL, you can use the ORDER BY clause in a SELECT statement. The ORDER BY clause is used to specify the column or columns by which the result set should be sorted. The default sorting order is(ASC) ascending (from the lowest value to the highest value), but you can also sort in descending order by using the DESC keyword.

5. In SQL, a table, a view, and a stored procedure are all database objects that serve different purposes:

(i) Table: A table is a collection of data that is organized into rows and columns. Tables are used to store data in a structured way, and they can be manipulated using SQL statements such as SELECT, INSERT, UPDATE, and DELETE. Tables are created using the CREATE TABLE statement.

(ii) View: A view is a virtual table that is based on a SELECT statement. Views do not store data, but rather, they are a way to present data from one or more tables in a specific way. Views can be used to simplify complex queries or to provide a different perspective on the data. Views are created using the CREATE VIEW statement.

(iii) Stored procedure: A stored procedure is a precompiled set of SQL statements that are stored in the database and can be executed on demand. Stored procedures can be used to encapsulate complex logic or business rules and provide a reusable and efficient way to perform database operations. Stored procedures are created using the CREATE PROCEDURE statement.

6. You can use a cursor to loop through records in a result set returned by a SELECT statement. A cursor is a database object that allows you to iterate over a result set row by row and perform some action on each row.

While cursors can be useful for looping through records in a result set, they can also have a negative impact on performance, especially for large result sets. As much as possible, it's best to use set-based approach which involves performing operations on entire sets of rows at once. This can often be achieved using SQL statements such as SELECT, UPDATE, DELETE, and INSERT.

7.A decimal number is a number that has a fractional part, while a whole number is a number that does not have a fractional part.

In SQL, the data type used for storing decimal numbers is usually "decimal" or "numeric". These data types are used to store numbers with a fixed precision and scale. The precision is the total number of digits that can be stored, while the scale is the number of digits that can be stored to the right of the decimal point.

On the other hand, whole numbers are typically stored using the "integer" data type in SQL. The integer data type can store whole numbers between -2,147,483,648 and 2,147,483,647 in most database systems. If you need to store larger numbers, you can use the "bigint" data type.

8. A stored procedure is a set of SQL statements that have been precompiled and stored in the database. A stored procedure is essentially a reusable program that can be called from other SQL statements or from application code to perform a specific task or set of tasks.

9. In SQL, "auto increment" (or "auto-incrementing") is a feature that allows a unique, sequential value to be automatically generated and assigned to a column in a table when a new row is inserted. The column is typically used as the primary key for the table, which means that it uniquely identifies each row.

The auto-increment feature is often used in situations where you need to generate unique, sequential IDs for rows in a table without having to manually manage the values.

10. To select unique records from a table in SQL, you can use the DISTINCT keyword in a SELECT statement. The DISTINCT keyword filters out duplicate values from the result set, returning only the unique values.

11. To query elements stored in a JSON object in SQL, you can use the "JSON_VALUE" function. The "JSON_VALUE" function allows you to extract a specific value from a JSON string based on a specified path.

12. The "LIKE" operator is used for pattern matching in queries. The "LIKE" operator allows you to match a specific pattern of characters in a column of text data.