1.What are Joints?

In SQL, a "joint" or "join" refers to the process of combining data from two or more database tables into a single result set based on a related column or columns between them.

There are different types of joins in SQL, including:

1. Inner Join: This type of join returns only the rows that have matching values in both tables being joined.
2. Left Outer Join: This type of join returns all the rows from the left table and matching rows from the right table, and null values for the non-matching rows in the right table.
3. Right Outer Join: This type of join returns all the rows from the right table and matching rows from the left table, and null values for the non-matching rows in the left table.
4. Full Outer Join: This type of join returns all the rows from both tables, and null values for the non-matching rows in either table.

Joins are important in SQL because they allow you to combine data from different tables in a meaningful way, and can be used to answer complex questions that require information from multiple tables.

2.What are the keywords?

In SQL, keywords are reserved words that have a specific meaning and cannot be used as identifiers (such as table or column names) unless enclosed in double quotation marks.

Here are some common SQL keywords:

1. SELECT: used to retrieve data from one or more tables.
2. FROM: used to specify the table or tables to retrieve data from.
3. WHERE: used to specify conditions that must be met for a row to be included in the result set.
4. ORDER BY: used to sort the result set based on one or more columns.
5. GROUP BY: used to group rows based on one or more columns.
6. HAVING: used to specify conditions that must be met by a group in order to be included in the result set.
7. JOIN: used to combine rows from two or more tables based on a related column or columns.
8. INSERT INTO: used to insert data into a table.
9. UPDATE: used to update data in a table.
10. DELETE FROM: used to delete data from a table.

These are just a few examples of the many keywords available in SQL. It's important to become familiar with them in order to write effective SQL queries.

3. Uses of key words?

In SQL, keywords are used to perform various operations on data, such as retrieving, manipulating, or deleting data from one or more tables. Here are some common uses of SQL keywords:

1. SELECT: The SELECT keyword is used to retrieve data from one or more tables. You can use this keyword to select specific columns from a table or to retrieve all columns from a table.
2. WHERE: The WHERE keyword is used to filter rows based on a specific condition. You can use this keyword to retrieve only the rows that meet certain criteria.
3. ORDER BY: The ORDER BY keyword is used to sort the result set based on one or more columns. You can use this keyword to sort the result set in ascending or descending order.
4. GROUP BY: The GROUP BY keyword is used to group rows based on one or more columns. You can use this keyword to group rows together based on a specific value.
5. JOIN: The JOIN keyword is used to combine rows from two or more tables based on a related column or columns. You can use this keyword to combine rows from different tables to retrieve data from multiple tables at once.
6. INSERT INTO: The INSERT INTO keyword is used to insert data into a table. You can use this keyword to add new rows to a table.
7. UPDATE: The UPDATE keyword is used to update data in a table. You can use this keyword to change the value of a column in one or more rows.
8. DELETE FROM: The DELETE FROM keyword is used to delete data from a table. You can use this keyword to remove one or more rows from a table.

Overall, SQL keywords are used to perform a wide range of operations on data stored in a database, and they are essential for working with SQL databases effectively.

4.Create and merge a table?

To create a table in SQL, you can use the CREATE TABLE statement followed by the table name and column definitions. Here's an example:

CREATE TABLE Customers (

CustomerID INT PRIMARY KEY,

CustomerName VARCHAR(255),

ContactName VARCHAR(255),

Country VARCHAR(255)

);

This statement creates a table called "Customers" with four columns: CustomerID, CustomerName, ContactName, and Country.

To merge tables in SQL, you can use the JOIN keyword to combine rows from two or more tables based on a related column or columns. Here's an example:

SELECT \*

FROM Orders

JOIN Customers

ON Orders.CustomerID = Customers.CustomerID;

This statement retrieves data from two tables, "Orders" and "Customers", and combines them based on the "CustomerID" column. The resulting table will contain all columns from both tables for rows where the CustomerID values match.

Note that in order to join tables, the tables must have a common column with matching values. In this example, both tables have a "CustomerID" column, which is used to join the tables.

5.Keys in SQL?

In SQL, a "key" is a column or set of columns in a table that uniquely identifies each row in that table. Keys are important because they ensure data integrity and help maintain consistency in the database.

Here are some common types of keys in SQL:

1. Primary Key: A primary key is a column or set of columns in a table that uniquely identifies each row in that table. The primary key constraint ensures that the values in the column or columns are unique and not null.
2. Foreign Key: A foreign key is a column or set of columns in one table that refers to the primary key of another table. This establishes a relationship between the two tables and ensures referential integrity.
3. Candidate Key: A candidate key is a column or set of columns that could be used as a primary key, but is not currently designated as such.
4. Unique Key: A unique key is a column or set of columns in a table that contains unique values, but is not designated as the primary key. The unique key constraint ensures that the values in the column or columns are unique.
5. Composite Key: A composite key is a combination of two or more columns in a table that together uniquely identify each row in that table.

Using keys in SQL is important because they ensure that the data is accurate and consistent, and can help improve query performance. When designing a database schema, it's important to carefully consider which columns should be designated as keys in order to ensure data integrity and maintainability.

6.Conditional statement in SQL?

In SQL, conditional statements are used to execute different SQL statements based on a specific condition. The most commonly used conditional statement in SQL is the "IF" statement, which allows you to execute a block of code if a specified condition is true.

Here's an example of an IF statement in SQL:

Sql

IF (SELECT COUNT(\*) FROM Customers) > 1000

BEGIN

PRINT 'There are more than 1000 customers in the database'

END

In this example, the IF statement checks the number of rows in the "Customers" table, and if the count is greater than 1000, it prints a message to the console.

You can also use the "CASE" statement in SQL to perform conditional operations. The CASE statement allows you to evaluate multiple conditions and execute different SQL statements based on each condition. Here's an example:

sql

SELECT

CustomerName,

CASE

WHEN Country = 'USA' THEN 'Domestic'

ELSE 'International'

END AS 'Customer Type'

FROM

Customers;

In this example, the CASE statement checks the value of the "Country" column for each row in the "Customers" table. If the value is "USA", it returns "Domestic". Otherwise, it returns "International" as the "Customer Type" for each row.

Overall, conditional statements in SQL are useful for executing different SQL statements based on specific conditions, and can help you perform complex operations and queries on your data.

7.Second highest salary program -need to write query?

SELECT MAX(salary) as 'Second Highest Salary'

FROM employees

WHERE salary < (SELECT MAX(salary) FROM employees);

In this query, we first find the highest salary in the "employees" table using the MAX function. Then, we select the maximum salary value that is less than the highest salary using a subquery. Finally, we use the MAX function again to find the second highest salary in the table.

Note that this query assumes that the "employees" table has a "salary" column that contains the salary information for each employee. If your table has a different structure or column names, you may need to adjust the query accordingly.

8.What is CRUD?

CRUD stands for Create, Read, Update, and Delete, which are the four basic operations that can be performed on data in a database. These operations are used to manage the data stored in a database and are supported by almost all database management systems, including SQL.

Here's a brief explanation of each CRUD operation in SQL:

1. Create: The Create operation is used to insert new data into a table in the database. This is done using the SQL INSERT statement, which allows you to specify the values to be inserted into each column of the table.
2. Read: The Read operation is used to retrieve data from a table in the database. This is done using the SQL SELECT statement, which allows you to specify the columns to be retrieved and any conditions for filtering the data.
3. Update: The Update operation is used to modify existing data in a table in the database. This is done using the SQL UPDATE statement, which allows you to specify the columns to be updated and the new values for each column.
4. Delete: The Delete operation is used to remove data from a table in the database. This is done using the SQL DELETE statement, which allows you to specify the conditions for filtering the data to be deleted.

Overall, the CRUD operations are fundamental to working with data in a SQL database and are used extensively in applications and systems that rely on database management.