

TOP 50

SQL

INTERVIEW QUESTION



Created by- **TOPPERWORLD**

Q 1. What is SQL?

Ans: SQL stands for **Structured Query Language**.

- It is a language used to interact with the database, i.e to create a database, to create a table in the database, to retrieve data or update a table in the database, etc.
- SQL is an ANSI(American National Standards Institute) standard. Using SQL, we can do many things.

For example – we can execute queries, we can insert records into a table, can update records, can create a database, can create a table, can delete a table, etc.

Q 2. What is a database?

Ans: A Database is defined as a structured form of data storage in a computer or a collection of data in an organized manner and can be accessed in various ways.

It is also the collection of schemas, tables, queries, views, etc.

Databases help us with easily storing, accessing, and manipulating data held on a computer.

The Database Management System allows a user to interact with the database.

Q 3. What are the differences between SQL and PL/SQL?

Ans: Some common differences between SQL and PL/SQL are as shown below:

SQL	PL/SQL
-----	--------

SQL is a query execution or commanding language.	PL/SQL is a complete programming language
SQL is a data-oriented language.	PL/SQL is a procedural language
SQL is very declarative in nature.	PL/SQL has a procedural nature.
It is used for manipulating data.	It is used for creating applications.
We can execute one statement at a time in SQL.	We can execute blocks of statements in PL/SQL
SQL tells databases, what to do?	PL/SQL tells databases how to do.
We can embed SQL in PL/SQL.	We can not embed PL/SQL in SQL

Q 4. Write an SQL query to find the names of employees starting with 'A' .

Ans: The LIKE operator of SQL is used for this purpose.

It is used to fetch filtered data by searching for a particular pattern in the where clause.



The Syntax for using LIKE is,

```
SELECT column1, column2 FROM table_name WHERE  
column_name LIKE pattern;
```

LIKE: operator name **pattern:** exact value extracted from the pattern to get related data in

result set.

The required query is:

```
SELECT * FROM Employees WHERE EmpName like 'A%';
```

Q 5. What is the difference between CHAR and VARCHAR datatype in SQL?

Ans :

1. CHAR Datatype:

It is a datatype in SQL which is used to store character string of fixed length specified.

If the length of the string is less than set or fixed-length then it is padded with extra blank spaces so that its length became equal to the set length when **PAD_CHAR_TO_FULL_LENGTH SQL mode** is enabled.

The storage size of the CHAR datatype is n bytes(set length). We should use this datatype when we expect the data values in a column are of the same length.



Example:

Consider the Query:

```
CREATE TABLE Student(Name VARCHAR(30), Gender
CHAR(6));
INSERT into Student VALUES('Herry', 'Male');
INSERT into Student VALUES('Mahi', 'Female');
SELECT LENGTH(Gender) FROM Student;
```

OUTPUT:

```
LENGTH(Gender)
6
6
```

2. VARCHAR Datatype:

It is a datatype in SQL which is used to store character string of variable length but a maximum of the set length specified.

If the length of the string is less than set or fixed-length then it will store as it is without padded with extra blank spaces.

The storage size of the VARCHAR datatype is equal to the actual length of the entered string in bytes.

We should use this datatype when we expect the data values in a column are of variable length.

Example:

Consider the Query:

```
CREATE TABLE Student(Name VARCHAR(20), Gender
CHAR(6));
INSERT into Student VALUES('Herry', 'Male');
INSERT into Student VALUES('Mahi', 'Female');
SELECT LENGTH(Name) FROM Student;
```

OUTPUT:

```
LENGTH(Name)
5
4
```

Q 6. What do you mean by data manipulation language?

Ans: Data manipulation Language or DML is used to access or manipulate data in the database.

It allows us to perform the below-listed functions:

- 1) Insert data or rows in a database
- 2) Delete data from the database
- 3) Retrieve or fetch data
- 4) Update data in a database.



LEARN DATA STRUCTURE & ALGORITHMS

www.topperworld.in

CLICK HERE

Topic wise PDF

The advertisement features a dark blue background with a glowing brain icon on the right, symbolizing learning and algorithms. A yellow hand cursor points to a red 'CLICK HERE' button. The text 'LEARN DATA STRUCTURE & ALGORITHMS' is prominently displayed in yellow and white. The website URL 'www.topperworld.in' is shown in a blue pill-shaped button. The phrase 'Topic wise PDF' is written in green near the brain icon.

Q 7. What is the view in SQL?

Views in SQL are a kind of virtual table. A view also has rows and columns as they are on a real table in the database.

We can create a view by selecting fields from one or more tables present in the database. A View can either have all the rows of a table or specific rows based on certain conditions.

The CREATE VIEW statement of SQL is used for creating views.

Basic Syntax:

```
CREATE VIEW view_name AS
SELECT column1, column2.....
FROM table_name
WHERE condition;
```

view_name : Name for the View
table_name: Name of the table
condition : Condition to select rows

Q 8. What do you mean by foreign key?

Ans : A Foreign key is a field that can uniquely identify each row in another table. And this constraint is used to specify a field as a Foreign key. That is this field points to the primary key of another table. This usually creates a kind of link between the two tables.

Consider the two tables as shown below:

Orders:

O_ID	ORDER_NO	C_ID
1	2253	3
2	3325	3
3	4521	2
4	8532	1

Customers:

C_ID	NAME	ADDRESS
1	RAMESH	DELHI
2	SURESH	NOIDA
3	DHARMESH	GURGAON

As we can see clearly, that the field C_ID in the Orders table is the primary key in the Customers' table, i.e. it uniquely identifies each row in the Customers table.

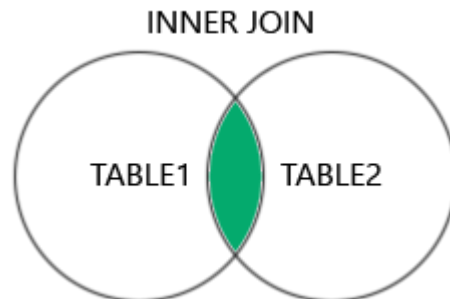
Syntax:

```
CREATE TABLE Orders
(
O_ID int NOT NULL,
ORDER_NO int NOT NULL,
C_ID int,
PRIMARY KEY (O_ID),
FOREIGN KEY (C_ID) REFERENCES Customers(C_ID)
)
```

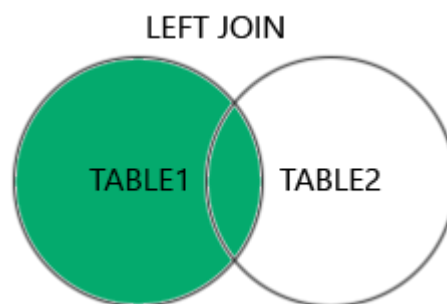
Q 9. What is a join in SQL? What are the types of joins?

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

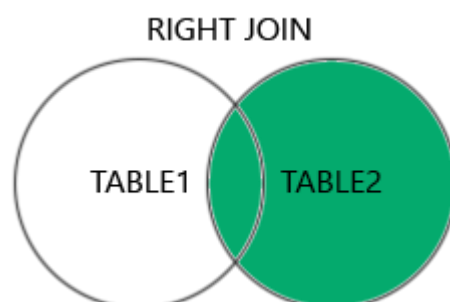
- **INNER JOIN:** The INNER JOIN keyword selects all rows from both tables as long as the condition is satisfied. This keyword will create the result set by combining all rows from both the tables where the condition satisfies i.e. the value of the common field will be the same.



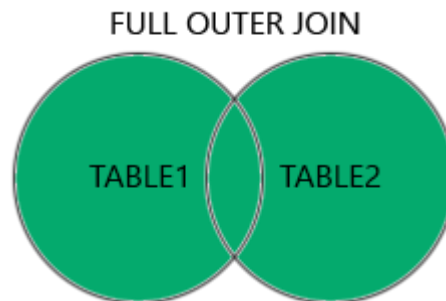
- **LEFT JOIN:** This join returns all the rows of the table on the left side of the join and matching rows for the table on the right side of the join. For the rows for which there is no matching row on the right side, the result set will be null. LEFT JOIN is also known as LEFT OUTER JOIN.



- **RIGHT JOIN:** RIGHT JOIN is similar to LEFT JOIN. This join returns all the rows of the table on the right side of the join and matching rows for the table on the left side of the join. For the rows for which there is no matching row on the left side, the result set will contain null. RIGHT JOIN is also known as RIGHT OUTER JOIN.



- **FULL JOIN:** FULL JOIN creates the result set by combining the results of both LEFT JOIN and RIGHT JOIN. The result set will contain all the rows from both tables. For the rows for which there is no matching, the result set will contain NULL values.



Q 10. What is normalization?

Ans: It is a process of analyzing the given relation schemas based on their functional dependencies and primary keys to achieve the following desirable properties:

- ✓ Minimizing Redundancy
- ✓ Minimizing the Insertion, Deletion, And Update Anomalies

Relation schemas that do not meet the properties are decomposed into smaller relation schemas that could meet desirable properties.

Q 11. What is Denormalization?

Ans : Denormalization is a database optimization technique in which we add redundant data to one or more tables. This can help us avoid costly joins in a relational database.



Note that denormalization does not mean not doing normalization. It is an optimization technique that is applied after normalization.

In a traditional normalized database, we store data in separate logical tables and attempt to minimize redundant data. We may strive to have only one copy of each piece of data in the database.

Q 12. Explain WITH clause in SQL?

Ans : The WITH clause provides a way relationship of defining a temporary relationship whose definition is available only to the query in which the with clause occurs.

SQL applies predicates in the WITH clause after groups have been formed, so aggregate functions may be used.

Q 13. What are all the different attributes of indexes?

Ans: The indexing has various attributes:

- **Access Types:** This refers to the type of access such as value-based search, range access, etc.
- **Access Time:** It refers to the time needed to find a particular data element or set of elements.
- **Insertion Time:** It refers to the time taken to find the appropriate space and insert new data.
- **Deletion Time:** Time is taken to find an item and delete it as well as update the index structure.
- **Space Overhead:** It refers to the additional space required by the index.

Q 14. What is a query?

Ans : An SQL query is used to retrieve the required data from the database. However, there may be multiple SQL queries that yield the same results but with different levels of efficiency.

An inefficient query can drain the database resources, reduce the database speed or result in a loss of service for other users. So it is very important to optimize the query to obtain the best database performance.

Q 15. What are the different operators available in SQL?

Ans: Generally, there are three types of operators that are used in SQL.

1. Arithmetic Operators
2. Comparison Operators
3. Logical Operators

➤ Arithmetic SQL Operators

Arithmetic operators are used to perform arithmetic operations such as addition, subtraction, division, and multiplication. These operators usually accept numeric operands.

➤ Comparison SQL Operators

Comparison operators in SQL are used to check the equality of two expressions. It checks whether one expression is identical to another. Comparison operators are generally used in the WHERE clause of a SQL query. The result of a comparison operation may be TRUE, FALSE or UNKNOWN. When one or both the expression is NULL, then the operator returns UNKNOWN. These operators could be used on all types of expressions except expressions that contain a text, ntext or an image.

➤ Logical SQL Operators

Logical operators are those operators that take two expressions as operands and return TRUE or False as output. While working with complex SQL statements and queries, comparison operators come in handy and these operators work in the same way as logic gates do.

Q 16. What is the difference between DELETE and TRUNCATE commands?

The DELETE statement removes rows one at a time and records an entry in the transaction log for each deleted row.	TRUNCATE TABLE removes the data by deallocating the data pages used to store the table data and records only the page deallocations in the transaction log.
DELETE command is slower than the identity TRUNCATE command.	While the TRUNCATE command is faster than the DELETE command.
To use Delete you need DELETE permission on the table.	To use Truncate on a table we need at least ALTER permission on the table.
The identity of the column retains the identity after using DELETE Statement on the table.	The identity of the column is reset to its seed value if the table contains an identity column.

The delete can be used with indexed views.

Truncate cannot be used with indexed views.

Q 17. What are local and global variables and their differences?

Ans :

- **Global Variable:**

Global variables are variables that are defined outside of functions. These variables have global scope, so they can be used by any function without passing them to the function as parameters.

- **Local Variable:**

Local variables are variables that are defined within functions. They have local scope, which means that they can only be used within the functions that define them.



Global Variable	Local Variable
Global variables are declared outside all the function blocks.	Local Variables are declared within a function block.
The scope remains throughout the program.	The scope is limited and remains within the function only in which they are declared.

Any change in global variable affects the whole program, wherever it is being used.	Any change in the local variable does not affect other functions of the program.
A global variable exists in the program for the entire time the program is executed.	A local variable is created when the function is executed, and once the execution is finished, the variable is destroyed.
It can be accessed throughout the program by all the functions present in the program.	It can only be accessed by the function statements in which it is declared and not by the other functions.
If the global variable is not initialized, it takes zero by default.	If the local variable is not initialized, it takes the garbage value by default.
Global variables are stored in the data segment of memory.	Local variables are stored in a stack in memory.
We cannot declare many variables with the same name.	We can declare various variables with the same name but in other functions.

Q 18. What is Auto Increment?

Ans : Sometimes, while creating a table, we do not have a unique identifier within the table, hence we face difficulty in choosing Primary Key.

So as to resolve such an issue, we ' ve to manually provide unique keys to every record, but this is often also a tedious task.

So we can use the Auto-Increment feature that automatically generates a numerical Primary key value for every new record inserted. The Auto Increment feature is supported by all the Databases.

Q 19. What is the difference between Cluster and Non-Cluster Index?

Ans :

CLUSTERED INDEX	NON-CLUSTERED INDEX
The clustered index is faster.	The non-clustered index is slower.
The clustered index requires less memory for operations.	The non-Clustered index requires more memory for operations.
In a clustered index, the index is the main data.	In the Non-Clustered index, the index is a copy of data.
A table can have only one clustered index.	A table can have multiple non-clustered indexes.
The clustered index has an inherent ability to store data on the disk.	The non-Clustered index does not have the inherent ability to store data on the disk.
Clustered indexes store pointers to block not data.	The non-Clustered index store both value and a pointer to the the the actual row that holds data.
In Clustered index leaf nodes are actual data itself.	In a Non-Clustered index, leaf nodes are not the actual data itself rather they only contain included columns.
In the Clustered index, the Clustered key defines the order of data within the table.	In the Non-Clustered index, the index key defines the order of data within the index.

CLUSTERED INDEX	NON-CLUSTERED INDEX
A Clustered index is a type of index in which table records are physically reordered to match the index.	A Non-Clustered index is a special type of index in which the logical order of index does not match the physical stored order of the rows on the disk.

Q 20. What is MySQL collation?

Ans : A MySQL collation is a well-defined set of rules which are used to compare characters of a particular character set by using their corresponding encoding.

Each character set in MySQL might have more than one collation, and has, at least, one default collation.

Two character sets cannot have the same collation.

Q 21. What are user-defined functions?

We can use User-defined functions in PL/SQL or Java to provide functionality that is not available in SQL or SQL built-in functions. SQL functions and User-defined functions can appear anywhere, that is, wherever an expression occurs.

Q 22. What are all types of user-defined functions?

Ans : User-Defined Functions allow people to define their own T-SQL functions that can accept 0 or more parameters and return a single scalar data

value or a table data type.

Different Kinds of User-Defined Functions created are:

1. Scalar User-Defined Function A Scalar user-defined function returns one of the scalar data types. Text, image, and timestamp data types are not supported. These are the type of user-defined functions that most developers are used to in other programming languages. You pass in 0 to many parameters and you get a return value.

2. Inline Table-Value User-Defined Function An Inline Table-Value user-defined function returns a table data type and is an exceptional alternative to a view as the user-defined function can pass parameters into a T-SQL select command and, in essence, provide us with a parameterized, non-updateable view of the underlying tables.

3. Multi-statement Table-Value User-Defined Function A Multi-Statement Table-Value user-defined function returns a table and is also an exceptional alternative to a view, as the function can support multiple T-SQL statements to build the final result where the view is limited to a single SELECT statement.

Q 23. What are Union, minus, and Intersect commands?

Ans : Set Operations in SQL eliminate duplicate tuples and can be applied only to the relations which are union compatible. Set Operations available in SQL are :

- Set Union
- Set Intersection
- Set Difference

•

UNION Operation: This operation includes all the tuples which are present in either of the relations.

For example: To find all the customers who have a loan or an account or both in a bank.

```
SELECT CustomerName FROM Depositor
UNION
SELECT CustomerName FROM Borrower ;
```

The union operation automatically eliminates duplicates. If all the duplicates are supposed to be retained, UNION ALL is used in place of UNION.

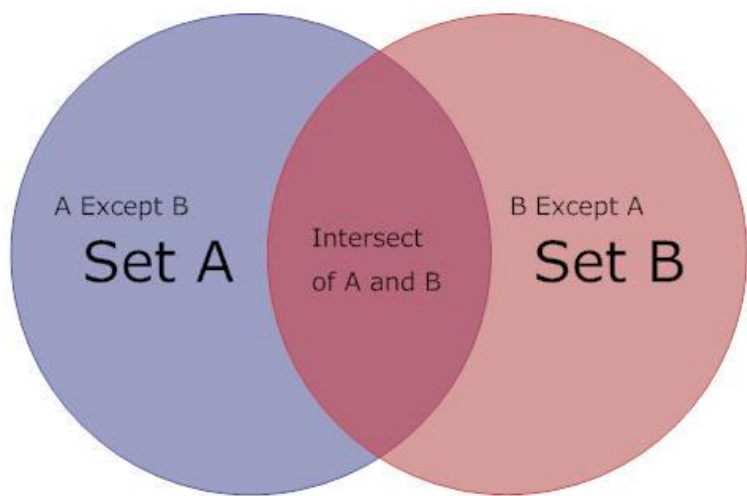
INTERSECT Operation: This operation includes the tuples which are present in both of the relations.

For example: To find the customers who have a loan as well as an account in the bank:

```
SELECT CustomerName FROM Depositor
INTERSECT
SELECT CustomerName FROM Borrower ;
```

The Intersect operation automatically eliminates duplicates.

If all the duplicates are supposed to be retained, INTERSECT ALL is used in place of INTERSECT.



EXCEPT for Operation: This operation includes tuples that are present in one relationship but should not be present in another relationship.

For example: To find customers who have an account but no loan at the bank:

```
SELECT CustomerName FROM Depositor  
EXCEPT  
SELECT CustomerName FROM Borrower ;
```

The Except operation automatically eliminates the duplicates. If all the duplicates are supposed to be retained, EXCEPT ALL is used in place of EXCEPT.

Q 24. What is an ALIAS command ?

Ans : Aliases are the temporary names given to a table or column for the purpose of a particular SQL query.

It is used when the name of a column or table is used other than its original name, but the modified name is only temporary.

- Aliases are created to make table or column names more readable.
- The renaming is just a temporary change and the table name does not change in the original database.
- Aliases are useful when table or column names are big or not very readable.
- These are preferred when there is more than one table involved in a query.



Q 25. What is the difference between TRUNCATE and DROP statements?

Ans :

S.NO.	DROP	TRUNCATE
1.	The DROP command is used to remove the table definition and its contents.	Whereas the TRUNCATE command is used to delete all the rows from the table.
2.	In the DROP command, table space is freed from memory.	While the TRUNCATE command does not free the table space from memory.
3.	DROP is a DDL(Data Definition Language) command.	Whereas the TRUNCATE is also a DDL(Data Definition Language) command.
4.	In the DROP command, a view of the table does not exist.	While in this command, a view of the table exists.
5.	In the DROP command, integrity constraints will be removed.	While in this command, integrity constraints will not be removed.
6.	In the DROP command, undo space is not used.	While in this command, undo space is used but less than DELETE.
7.	The DROP command is quick to perform but gives rise to complications.	While this command is faster than DROP.

Q 26. What are aggregate and scalar functions?

Ans : For doing operations on data SQL has many built-in functions, they are categorized into two categories and further sub-categorized into seven different functions under each category. The categories are:

Aggregate functions:

These functions are used to do operations from the values of the column and a single value is returned.

Scalar functions:

These functions are based on user input, these too return a single value.

Q 27. Explain SQL AND OR statement with an example?

Ans : In SQL, the AND & OR operators are used for filtering the data and getting precise results based on conditions.

The AND and OR operators are used with the WHERE clause.

These two operators are called conjunctive operators.

AND Operator: This operator displays only those records where both conditions condition 1 and condition 2 evaluate to True.

OR Operator: This operator displays the records where either one of the conditions condition 1 and condition 2 evaluates to True. That is, **either condition1 is True or condition2 is True.**

Q 28. Why do we use Commit and Rollback commands?

Ans :

COMMIT	ROLLBACK
COMMIT permanently saves the changes made by the current transaction.	ROLLBACK undo the changes made by the current transaction.
The transaction can not undo changes after COMMIT execution.	Transaction reaches its previous state after ROLLBACK.
When the transaction is successful, COMMIT is applied.	When the transaction is aborted, ROLLBACK occurs.

Q 29. What is a T-SQL?

Ans : T-SQL is an abbreviation for Transact Structure Query Language. It is a product by Microsoft and is an extension of SQL Language which is used to interact with relational databases.

It is considered to perform best with Microsoft SQL servers. T-SQL statements are used to perform the transactions to the databases. T-SQL has huge importance since all the communications with an instance of an SQL server are done by sending Transact-SQL statements to the server. Users can also define functions using T-SQL.

Types of T-SQL functions are :

- **Aggregate** functions.
- **Ranking** functions. There are different types of ranking functions.
- **Rowset** function.
- **Scalar** functions.

Q 30. What is the need for a MERGE statement?

Ans : The MERGE command in SQL is actually a combination of three SQL statements:

INSERT, UPDATE, and DELETE.

In simple words, the MERGE statement in SQL provides a convenient way to perform all these three operations together which can be very helpful when it comes to handling large running databases. But unlike INSERT, UPDATE, and DELETE statements MERGE statement requires a source table to perform these operations on the required table which is called a target table.

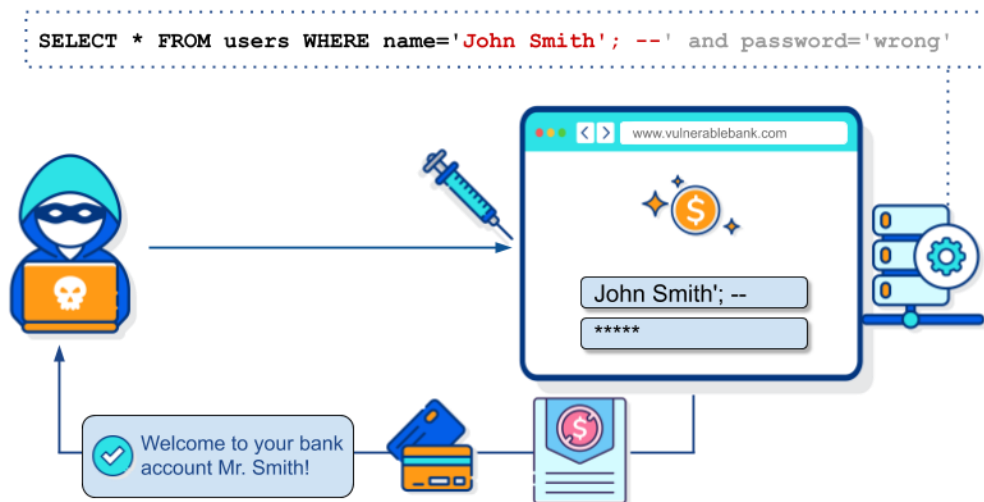
Q 31. What are the advantages of PL/SQL functions?

Ans : The advantages of PL / SQL functions are as follows:

- We can make a single call to the database to run a block of statements. Thus, it improves the performance against running SQL multiple times. This will reduce the number of calls between the database and the application.
- We can divide the overall work into small modules which becomes quite manageable, also enhancing the readability of the code.
- It promotes reusability.
- It is secure since the code stays inside the database, thus hiding internal database details from the application(user). The user only makes a call to the PL/SQL functions. Hence, security and data hiding is ensured.

Q 32. What is SQL injection?

Ans : SQL injection is a technique used to exploit user data through web page inputs by injecting SQL commands as statements. Basically, these statements can be used to manipulate the application ' s web server by malicious users.



- SQL injection is a code injection technique that might destroy your database.
- SQL injection is one of the most common web hacking techniques.
- SQL injection is the placement of malicious code in SQL statements, via web page input.

Q 33. Can we disable a trigger? If yes, how?

Ans : Yes, we can disable a trigger in PL/SQL. If consider temporarily disabling a trigger and one of the following conditions is true:

- An object that the trigger references is not available.
- We must perform a large data load and want it to proceed quickly without firing triggers.
- We are loading data into the table to which the trigger applies.
- We disable a trigger using the ALTER TRIGGER statement with the DISABLE option.
- We can disable all triggers associated with a table at the same time using the ALTER TABLE statement with the DISABLE ALL TRIGGERS option.

Q 34. Name the operator which is used in the query for appending two strings?

Ans : In SQL for appending two strings, the " Concatenation operator" is used and its symbol is " || " .

Q 35. What is the difference between COALESCE() & ISNULL()?

Ans : **COALESCE():** COALESCE function in SQL returns the first non-NULL expression among its arguments.

If all the expressions evaluate to null, then the COALESCE function will return null.

Syntax:

```
SELECT column(s),  
CAOLESCE(expression_1,.....,expression_n)  
FROM table_name;
```

ISNULL(): The ISNULL function has different uses in SQL Server and MySQL.

In SQL Server, ISNULL() function is used to replace NULL values.

Syntax:

```
SELECT column(s), ISNULL(column_name,  
value_to_replace)  
FROM table_name;
```



Q 36. What is Case WHEN in SQL?

Control statements form an important part of most languages since they control the execution of other sets of statements.

These are found in SQL too and should be exploited for uses such as query filtering and query optimization through careful selection of tuples that match our requirements.

In this post, we explore the Case-Switch statement in SQL. The CASE statement is SQL's way of handling if/then logic.

syntax: 1

```
CASE case_value
  WHEN when_value THEN statement_list
  [WHEN when_value THEN statement_list] ...
  [ELSE statement_list]
END CASE
```

syntax: 2

```
CASE
  WHEN search_condition THEN statement_list
  [WHEN search_condition THEN statement_list] ...
  [ELSE statement_list]
END CASE
```

Q 37. The difference between NVL and NVL2 functions?

Ans : These functions work with any data type and pertain to the use of null values in the expression list. These are all single-row functions i.e. provide one result per row.

NVL(expr1, expr2): In SQL, NVL() converts a null value to an actual value. Data types that can be used are date, character, and number. Data types must match with each other. i.e. expr1 and expr2 must be of the same data type.

NVL2(expr1, expr2, expr3): The NVL2 function examines the first expression. If the first expression is not null, then the NVL2 function returns the second expression. If the first expression is null, then the third expression is returned i.e. If expr1 is not null, NVL2 returns expr2. If expr1 is null, NVL2 returns expr3. The argument expr1 can have any data type.

Q 38. How do we avoid getting duplicate entries in a query without using the distinct keyword?

Ans :

DISTINCT is useful in certain circumstances, but it has drawbacks that it can increase the load on the query engine to perform the sort (since it needs to compare the result set to itself to remove duplicates). We can remove duplicate entries using the following options:

- Remove duplicates using row numbers.
- Remove duplicates using self-Join.
- Remove duplicates using group by.

Q 39. What is a Live Lock?

Ans : Livelock occurs when two or more processes continually repeat the same interaction in response to changes in the other processes without doing any useful work.

These processes are not in the waiting state, and they are running concurrently.

This is different from a deadlock because in a deadlock all processes are in the waiting state.

Q 40. How to copy tables in SQL?

Ans : Sometimes, in SQL, we need to create an exact copy of an already defined (or created) table. MySQL enables you to perform this operation. Because we may need such duplicate tables for testing the data without having any impact on the original table and the data stored in it.

```
CREATE TABLE Contact List(Clone_1) LIKE Original_table;
```

Q 41. How to find the available constraint information in the table?

Ans : In SQL Server the **data dictionary** is a set of database tables used to store information about a database ' s definition.

One can use these data dictionaries to check the constraints on an already existing table and to change them(if possible).

Q 42. What is the SQL query to display the current date?

Ans : CURRENT_DATE returns to the current date. This function returns the same value if it is executed more than once in a single statement, which means that the value is fixed, even if there is a long delay between fetching rows in a cursor.

Syntax:

```
CURRENT_DATE
```

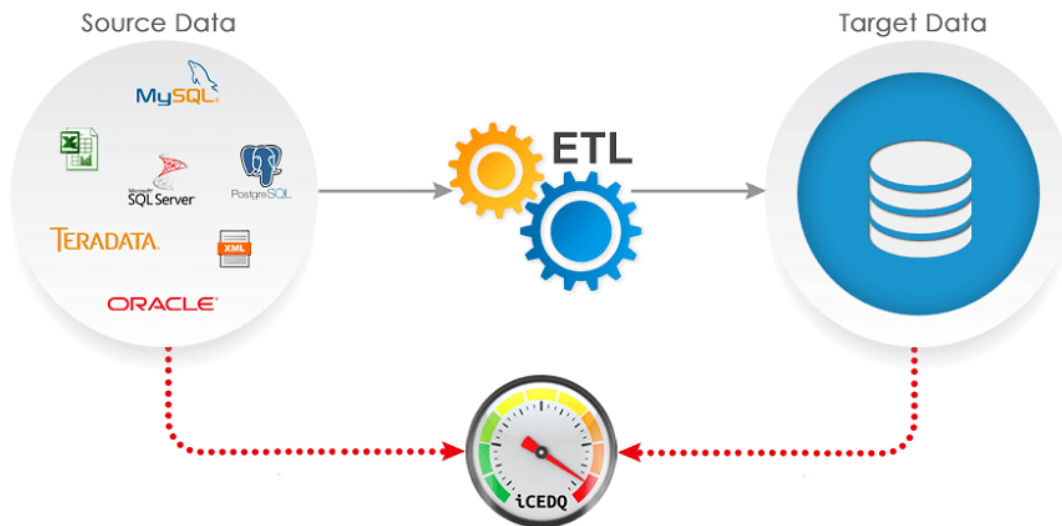
```
or
```

```
CURRENT DATE
```

Q 43. What is ETL in SQL?

Ans : ETL is a process in Data Warehousing.

It stands for **Extract, Transform, and Load**. It is a process in which an ETL tool extracts the data from various data source systems, transforms it in the staging area, and then finally, loads it into the Data Warehouse system.



These are three database functions that are incorporated into one tool to pull data out from one database and put data into another database.

Q 44. What are Nested Triggers?

Ans : 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.



Q 44. How to find the available constraint information in the table?

Ans : In SQL Server the **data dictionary** is a set of database tables used to store information about a database ' s definition.

One can use these data dictionaries to check the constraints on an already existing table and to change them(if possible).

Q 45. How can you fetch common records from two tables?

Ans : The below statement could be used to get data from multiple tables, so, we need to use join to get data from multiple tables.

Syntax :

```
SELECT tablename1.columnname,  
tablename2.columnname  
FROM tablename1  
JOIN tablename2  
ON tablename1.columnname =  
tablename2.columnname  
ORDER BY columnname;
```

Q 46. Are NULL values the same as zero or a blank space?

Ans : In SQL, zero or blank space can be compared with another zero or blank space. whereas one null may not be equal to another null. null means data might not be provided or there is no data.

Q 47. What is the need for group functions in SQL?

Ans : In database management, group functions, also known as aggregate functions, is a function where the values of multiple rows are grouped together as input on certain criteria to form a single value of more significant meaning.

Various Group Functions

- Count()
- Sum()
- Avg()
- Min()
- Max()

Q 48. What are ACID properties?

Ans : A transaction is a single logical unit of work that accesses and possibly modifies the contents of a database. Transactions access data using read-and-write operations. In order to maintain consistency in a database, before and after the transaction, certain properties are followed. These are called **ACID** properties. ACID (Atomicity, Consistency, Isolation, Durability) is a set of properties that guarantee that database transactions are processed reliably.

Q 49. Define SQL Order by the statement?

Ans : The ORDER BY statement in SQL is used to sort the fetched data in either ascending or descending according to one or more columns.

- By default ORDER BY sorts the data in **ascending order**.
- We can use the keyword DESC to sort the data in descending order and the keyword ASC to sort in ascending order.

Q 50. Explain SQL Having statement?

Ans : HAVING is used to specify a condition for a group or an aggregate function used in the select statement.

The WHERE clause selects before grouping. The HAVING clause selects rows after grouping. Unlike the HAVING clause, the WHERE clause cannot contain aggregate functions.

ABOUT US

At TopperWorld, we are on a mission to empower college students with the knowledge, tools, and resources they need to succeed in their academic journey and beyond.

➤ Our Vision

- ❖ Our vision is to create a world where every college student can easily access high-quality educational content, connect with peers, and achieve their academic goals.
- ❖ We believe that education should be accessible, affordable, and engaging, and that's exactly what we strive to offer through our platform.

➤ **Unleash Your Potential**

- ❖ In an ever-evolving world, the pursuit of knowledge is essential. TopperWorld serves as your virtual campus, where you can explore a diverse array of online resources tailored to your specific college curriculum.
- ❖ Whether you're studying science, arts, engineering, or any other discipline, we've got you covered.
- ❖ Our platform hosts a vast library of e-books, quizzes, and interactive study tools to ensure you have the best resources at your fingertips.

➤ **The TopperWorld Community**

- ❖ Education is not just about textbooks and lectures; it's also about forming connections and growing together.
- ❖ TopperWorld encourages you to engage with your fellow students, ask questions, and share your knowledge.
- ❖ We believe that collaborative learning is the key to academic success.

➤ **Start Your Journey with TopperWorld**

- ❖ Your journey to becoming a top-performing college student begins with TopperWorld.
- ❖ Join us today and experience a world of endless learning possibilities.
- ❖ Together, we'll help you reach your full academic potential and pave the way for a brighter future.
- ❖ Join us on this exciting journey, and let's make academic success a reality for every college student.

“UNLOCK YOUR POTENTIAL”

With- **TOPPERWORLD**

Explore More



www.topperworld.in

DSA TUTORIAL

C TUTORIAL

C++ TUTORIAL

JAVA TUTORIAL

PYTHON TUTORIAL

Follow Us On



E-mail



topperworld.in@gmail.com

