SQL Questions

1. What do you mean by table and field in SQL?

A table in a relational database is much like a table on paper: It consists of rows and columns. The number and order of the columns is fixed, and each column has a name. The number of rows is variable — it reflects how much data is stored at a given moment. SQL does not make any guarantees about the order of the rows in a table. When a table is read, the rows will appear in an unspecified order, unless sorting is explicitly requested.

Tables contain rows and columns, where the rows are known as records and the columns are known as fields. A column is a set of data values of a particular type (like numbers or alphabets), one value for each row of the database, for example, Age, Student_ID, or Student_Name.

2. What are joins in SQL? 11. List the different type of joins? And explain them.

JOIN is an SQL clause used to query and access data from multiple tables, based on logical relationships between those tables. In other words, JOINS indicate how SQL Server should use data from one table to select the rows from another table.

(INNER) JOIN : Returns records that have matching values in both tables. LEFT (OUTER) JOIN : Returns all records from the left table, and the matched records from the right table. RIGHT (OUTER) JOIN : Returns all records from the right table, and the matched records from the left table.

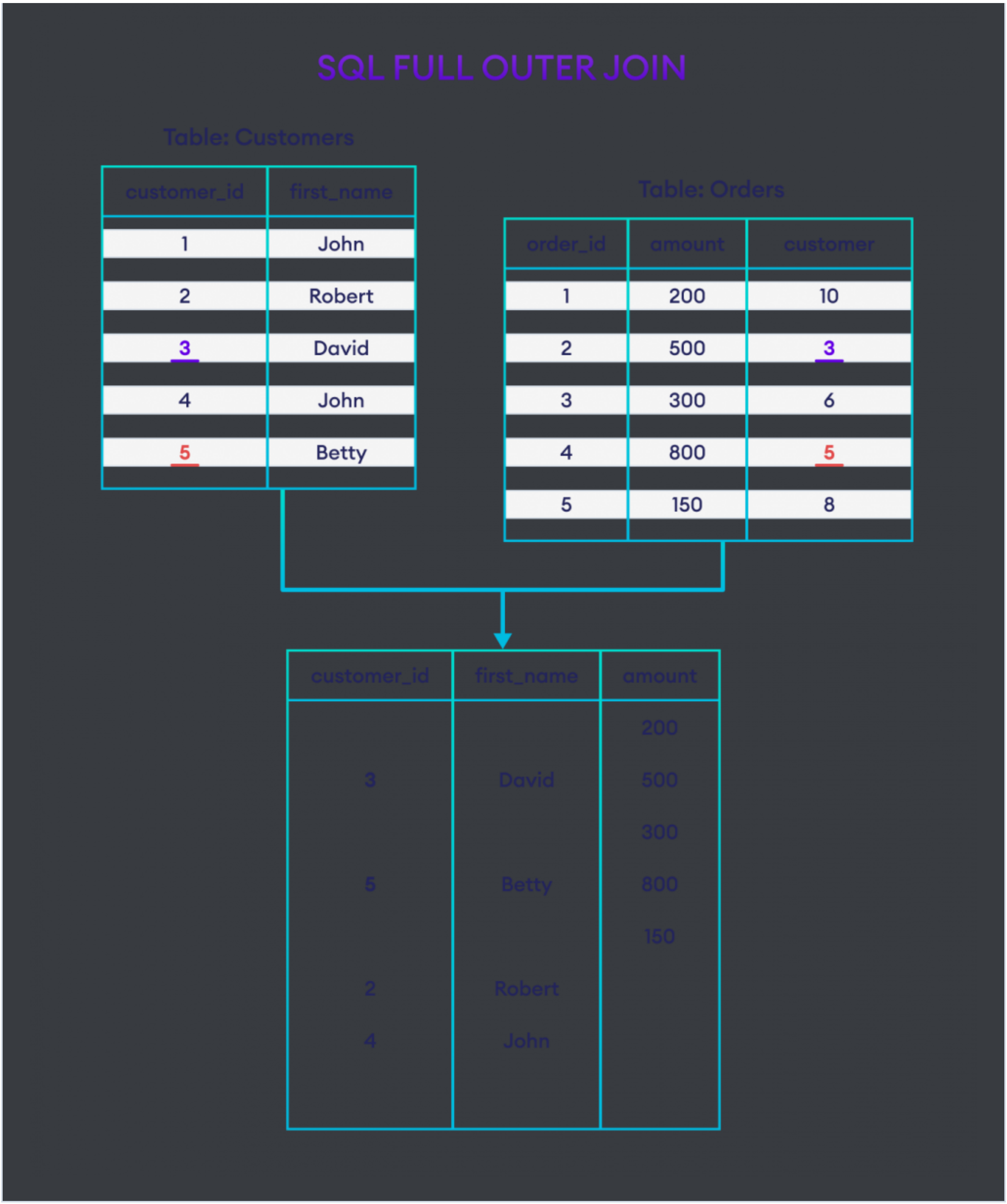
Example:

```
SELECT Customers.customer_id, Customers.first_name, Orders.amount
FROM Customers
JOIN Orders
ON Customers.customer_id = Orders.customer;

SELECT Customers.customer_id, Customers.first_name, Orders.amount
FROM Customers
LEFT JOIN Orders
ON Customers.customer_id = Orders.customer;

SELECT Customers.customer_id, Customers.first_name, Orders.amount
FROM Customers
RIGHT JOIN Orders
ON Customers.customer_id = Orders.customer;

SELECT Customers.customer_id, Customers.first_name, Orders.amount
FROM Customers
FULL OUTER JOIN Orders
ON Customers.customer_id = Orders.customer;
```



SQL FULL OUTER JOIN

Table: Customers

customer_id	first_name
1	John
2	Robert
3	David
4	John
5	Betty

Table: Orders

order_id	amount	customer
1	200	10
2	500	3
3	300	6
4	800	5
5	150	8

customer_id	first_name	amount
		200
3	David	500
		300
5	Betty	800
		150
2	Robert	
4	John	

3. Write an SQL statement to get all the columns from the Customers table

* SELECT column1, column2, ... FROM table_name;
* SELECT * FROM table_name;

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

VARCHAR is variable length, while CHAR is fixed length. CHAR is a fixed length string data type, so any remaining space in the field is padded with blanks(заполняется пробелами). CHAR takes up 1 byte per character. So, a CHAR(100) field (or variable) takes up 100 bytes on disk, regardless of the string it holds.

5. What is the difference between VARCHAR and NVARCHAR datatype in SQL Server?

The key difference between varchar and nvarchar is the way they are stored, varchar is stored as regular 8-bit data(1 byte per character) and nvarchar stores data at 2 bytes per character. Due to this reason, nvarchar can hold upto 4000 characters and it takes double the space as SQL varchar

6. What is a Primary key?

A primary key, also called a primary keyword, is a key in a relational database that is unique for each record. It is a unique identifier, such as a driver license number, telephone number (including area code), or vehicle identification number (VIN). A relational database must always have one and only one primary key.

7. What are Constraints? 8. What is a Unique key? 9. What is a Foreign key?

Constraints are the rules enforced on the data columns of a table. These are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the database.Constraints could be either on a column level or a table level. The column level constraints are applied only to one column, whereas the table level constraints are applied to the whole table.

- NOT NULL Constraint − Ensures that a column cannot have NULL value.
- DEFAULT Constraint − Provides a default value for a column when none is specified.
- UNIQUE Constraint − Ensures that all values in a column are different.
- PRIMARY Key − Uniquely identifies each row/record in a database table.
- FOREIGN Key − Uniquely identifies a row/record in any of the given database table.
- CHECK Constraint − The CHECK constraint ensures that all the values in a column satisfies certain conditions.
- INDEX − Used to create and retrieve data from the database very quickly.

Constraints can be specified when a table is created with the CREATE TABLE statement or you can use the ALTER TABLE statement to create constraints even after the table is created.

```
ALTER TABLE EMPLOYEES DROP CONSTRAINT EMPLOYEES_PK;
```

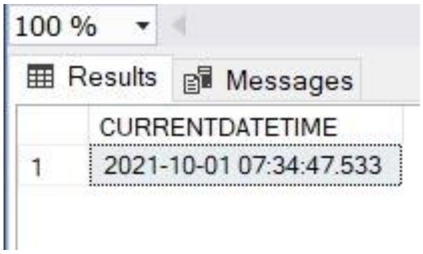
10. Write a SQL query to display the current date?

GETDATE() function is mostly used to find the current Date. It will return the DATETIME data type. This means it will Return the Current date with the current Time.

Query:

```
Select GetDate() AS 'CurrentDATETime';
```

Output:

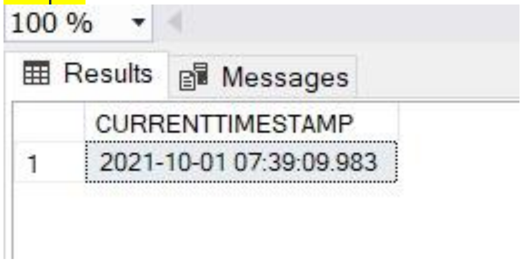


```
   CURRENTDATETIME
1  2021-10-01 07:34:47.533
```

CURRENT_TIMESTAMP:
It is also used to find current TIMESTAMP means current Date and Time. The CURRENT_TIMESTAMP function can be used the same that the GETDATE() function is used CURRENT_TIMESTAMP returns the same result as GETDATE().
Query:

```
Select CURRENT_TIMESTAMP AS "CURRENTTIMESTAMP";
```

Output:



```
   CURRENTTIMESTAMP
1  2021-10-01 07:39:09.983
```

12. What are Entities and Relationships?

An entity–relationship model (or ER model) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between entities (instances of those entity types).

What is the difference between entity and relationship?

An entity is a table in DBMS, and it represents a real-world object. Entities are connected to each other using relationships. Thus, the difference between entity and relationship in DBMS is that the entity is a real-world object while the relationship is an association between the entities.

13. What is an Index?

An Index is the structure or object by which we can retrieve specific rows or data faster. Indexes can be created using one or multiple columns or by using the partial data depending on your query requirement conditions. Index will create a pointer to the actual rows in the specified table

14. What is Normalization and what are the advantages of it?

Data normalization is the process of reorganizing data within a database so that users can utilize it for further queries and analysis. Simply put, it is the process of developing clean data. This includes eliminating redundant and unstructured data and making the data appear similar across all records and fields.

15. What is the difference between DROP and TRUNCATE commands?

TRUNCATE is a DDL(Data Definition Language) command. It is used to delete all the tuples from the table. Like the DROP command, the TRUNCATE command also does not contain a WHERE clause the DROP command is used to remove the whole database or table indexes, data, and more. Whereas the TRUNCATE command is used to remove all the rows from the table.

16. What are the different operators available in SQL?

An operator is a reserved word or a character that is used to query our database in a SQL expression. To query a database using operators, we use a WHERE clause. Operators are necessary to define a condition in SQL, as they act as a connector between two or more conditions. The operator manipulates the data and gives the result based on the operator’s functionality.

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

1. Arithmetic SQL Operators

Arithmetic operators are used to perform arithmetic operations such as addition, subtraction, division, and multiplication. These operators usually accept numeric operands. Different operators that come under this category are given below-

Operator	Operation	Description
+	Addition	Adds operands on either side of the operator
-	Subtraction	Subtracts the right-hand operand from the left-hand operand
*	Multiplication	Multiplies the values on each side
/	Division	Divides left-hand operand by right-hand operand
%	Modulus	Divides left-hand operand by right-hand operand and returns the remainder

2. 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. The table below shows different types of comparison operators in SQL:

Operator	Operation	Description
=	Equal to	Checks if both operands have equal value, if yes, then returns TRUE
>	Greater than	Checks if the value of the left-hand operand is greater than the right-hand operand or not
<	Less than	Returns TRUE if the value of the left-hand operand is less than the value of the right-hand operand
>=	Greater than or equal to	It checks if the value of the left-hand operand is greater than or equal to the value of the right-hand operand, if yes, then returns TRUE
<=	Less than or equal to	Examines if the value of the left-hand operator is less than or equal to the right-hand operand
<> or !=	Not equal to	Checks if values on either side of the operator are equal or not. Returns TRUE if values are not equal
!>	Not greater than	Used to check if the left-hand operator’s value is not greater than or equal to the right-hand operator’s value

!<	Not less than	Used to check if the left-hand operator’s value is not less than or equal to the right-hand operator’s value

3. 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. Different logical operations available in SQL are given in the below table.

Operator	Description
ALL	Compares a value to all other values in a set
AND	Returns the records if all the conditions separated by AND are TRUE
ANY	Compares a specific value to any other values in a set
SOME	Compares a value to each value in a set. It is similar to ANY operator
LIKE	It returns the rows for which the operand matches a specific pattern
IN	Used to compare a value to a specified value in a list

BETWEEN	Returns the rows for which the value lies between the mentioned range
NOT	Used to reverse the output of any logical operator
EXISTS	Used to search a row in a specified table in the database
OR	Returns the records for which any of the conditions separated by OR is true
NULL	Returns the rows where the operand is NULL

17. Are NULL values same as that of zero or a blank space?

A NULL value is not same as zero or a blank space. A NULL value is a value which is 'unavailable, unassigned, unknown or not applicable'. Whereas, zero is a number and blank space is a character.

18. How you can get the number of records in a table?

To counts all of the rows in a table, whether they contain NULL values or not, use COUNT(*). That form of the COUNT() function basically returns the number of rows in a result set returned by a SELECT statement.

```
SELECT count( * ) as  total_record FROM student
```

Output of above query is here.

total_record
35

19. Write a SQL query to find the names of employees from 'Employee' table, that begin with 'A'?

20. Write a SQL query to find the names of employees from 'Employee' table, that ends with 'A'?

```
SELECT *
FROM employees
WHERE emp_name LIKE '%A%';
```

21. Write a SQL query to get the highest salary of an employee from ‘Employee’ table?

```
select *from employee where salary=(select Max(salary) from
employee);
```

22. Write a SQL query to get the third highest salary of an employee from ‘Employee’ table?

ename	sal
A	23000
B	31000
C	24500
D	35000
E	28500
F	31500
G	39800
H	51000
I	39800

Query :

```
select * from(

select ename, sal, dense_rank()

over(order by sal desc)r from Employee)

where r=&n;
```

To find to the 2nd highest sal set n = 2

To find 3rd highest sal set n = 3 and so on.

23. What is the need for group functions(statement) in SQL?

Group functions are mathematical functions to operate on sets of rows to give one result per set. The types of group functions (also called aggregate functions) are: AVG, that calculates the average of the specified columns in a set of rows, COUNT, calculating the number of rows in a set.

24. What is a Relationship and what are they?

A relationship, in the context of databases, is a situation that exists between two relational database tables when one table has a foreign key that references the primary key of the other table. Relationships allow relational databases to split and store data in different tables, while linking disparate data items

What are relationships SQL?

Relationships are the established associations between two or more tables. Relationships are based on common fields from more than one table, often involving primary and foreign keys. A primary key is the field (or fields) that is used to uniquely identify each record in a table

25. How can you insert NULL values in a column while inserting the data?

You can insert NULL value into an int column with a condition i.e. the column must not have NOT NULL constraints. The syntax is as follows. INSERT INTO yourTableName(yourColumnName) values(NULL);

26. What is the main difference between 'BETWEEN' and 'IN' condition operators?

Differences between these operator is that the BETWEEN operator is used to select a range of data between two values while The IN operator allows you to specify multiple values.

27. Why are SQL functions used? And what kind of functions do you know?

SQL functions are simply sub-programs, which are commonly used and re-used throughout SQL database applications for processing or manipulating data. All SQL database systems have DDL (data definition language) and DML (data manipulation language) tools to support the creation and maintenance of databases

There are two types of SQL functions, aggregate functions, and scalar(non-aggregate) functions. Aggregate functions operate on many records and produce a summary, works with GROUP BY whereas non-aggregate functions operate on each record independently.

28. What is the difference between 'HAVING' CLAUSE and a 'WHERE' CLAUSE? Please give example.

Where Clause in SQL	Having Clause in SQL
Filter table based data catering to specific condition	Group based data under set condition
Applicable without GROUP BY clause	Does not function without GROUP BY clause

Row functions	Column functions
Select, update and delete statements	Only select statement
Applied before GROUP BY clause	Used after GROUP BY clause
Used with single row operations such as Upper, Lower and so on	Applicable with multiple row functions such as Sum, count and so on

29. List some case manipulation functions in SQL?

1. CONCAT: Concatenates [joins] the first character value to the second character value; equivalent to concatenation operator (||).
2. SUBSTR: Returns specific characters from character value starting at a specific character position and going specified character positions long
3. INSTR: Returns the numeric position of a named string.
4. LENGTH: Returns the number of characters in the expression
5. LPAD: Pads the left side of a character, resulting in a right-justified value
6. RPAD: Pads the right-hand side of a character, resulting in a left- justified value.
7. TRIM: Removes all specified characters from either the beginning or the ending of a string.
8. REPLACE: Replaces a sequence of characters in a string with another set of characters.

30. How can you select unique records from a table?

The SELECT DISTINCT statement is used to return only distinct (different) values. Inside a table, a column often contains many duplicate values; and sometimes you only want to list the different (distinct) values.

```
SELECT DISTINCT column 1, column 2, ...
FROM table_name;
```