SQL

1) What are Scalar functions and write a query?

The Scalar Functions in SQL are used to return a single value from the given input value.

They are;

1)UCASE() - SELECT UCASE(NAME) FROM EMPLOYEE

2)LCASE() – SELECT LCASE(NAME) FROM EMPLOYEE

3)LEN() – SELECT LENGTH(NAME) FROM EMPLOYE

4)MID() – SELECT MID(NAME,1,4) FROM EMPLOYEE

5)ROUND() – SELECT ROUND(SALARY,0)AS ROUNDEDSALARY FROM EMPLOYEE

6)NOW() – SELECT NAME,SALARY NOW() AS PERDATE FROM EMPLOYEE

7)FORMAT() - SELECT NAME ,SALARY FORMAT(NOW(), ’%m-%d-%Y’) AS DATE FROM EMPLOYEE

2) Explain joins with example and output.

A Join clause is used to combine rows from two or more tables, based on a related column between them.

Different types of joins .They are;

1)Inner join-Returns records that have matching values in both tables

2)Left join- Returns all records from the left table, and the matched records from the right table

3)Right join-Returns all records from the right table, and the matched records from the left table

4)Full join- Returns all records when there is a match in either left or right table

supplier_id	Supplier_name
100	GOOGLE
101	TCS
102	IBM
103	MICROSOFT

Examples:

suppliers

order_id	Supplier_id	Oder_date
2001	100	2013/05/12
2004	101	2013/05/13
3002	104	2013/05/14

oders

1)Inner join;

SELECT suppliers.supplier_id, suppliers.supplier_name, orders.order_date FROM suppliers INNER JOIN orders ON suppliers.supplier_id = orders.supplier_id;

OUTPUT:

Supplier_id	name	Oder_date
100	GOOGLE	2013/05/12
101	TCS	2013/05/13

2)LEFT JOIN;

SELECT suppliers.supplier_id, suppliers.supplier_name, orders.order_date FROM suppliers LEFT JOIN orders ON suppliers.supplier_id = orders.supplier_id;

OUTPUT:

Supplier_id	Supplier_name	Oder_date
100	GOOGLE	2013/05/12
101	TCS	2013/05/13
102	IBM	NULL
103	MICROSOFT	NULL

3)RIGHTJOIN;

SELECT orders.order_id, orders.order_date, suppliers.supplier_name FROM suppliers RIGHT JOIN orders ON suppliers.supplier_id = orders.supplier_id;

OUTPUT:

Oder_id	Oder_date	Supplier_name
2001	2013/05/12	GOOGLE
2004	2013/05/13	TCS

3002	2013/05/14	NULL

4)FULL JOIN;

```
SELECT
suppliers.supplier_id,suppliers.supplier_name,oders.oder_id,oders.supplier_id,oders.ode
r_date FROM suppliers FULL JOIN orders
ON suppliers.supplier_id = orders.supplier_id;
```

OUTPUT:

supplier_id	Supplier_name	order_id	Supplier_id	Oder_date
100	GOOGLE	2001	100	2013/05/12
101	TCS	2004	101	2013/05/13
102	IBM	NULL	NULL	NULL
103	MICROSOFT	NULL	NULL	NULL
NULL	NULL	3002	104	2013/05/14

3) Write a SQL query to Rename the column name.

```
ALTER TABLE table_name RENAME COLUMN old_column_name TO
new_column_name;
```

4) Write a SQL query to find duplicate records.

```
  SELECT column1, column2, COUNT(*)
FROM table_name
GROUP BY column1, column2
HAVING COUNT(*) > 1;
```

5) How do we use the DISTINCT statement . What are its uses?

The DISTINCT statement is used to retrieve unique values from a column or a combination of columns in a table. It is often used in SELECT statements to eliminate duplicate records.

Uses;

I. Retrieving unique values: As mentioned earlier, the primary use of DISTINCT is to retrieve unique values from a column or set of columns in a table. This can be useful for tasks such as finding unique customer names, unique order numbers, or unique product IDs.

II. Filtering duplicate records: DISTINCT can also be used to filter out duplicate records from a result set.

III. Counting unique values: You can also use DISTINCT in conjunction with the COUNT function to count the number of unique values in a column or set of columns.

IV. Aggregating unique values: DISTINCT can be used to aggregate unique values in a column or set of columns using functions like SUM, AVG, or MAX.

6) Remove duplicate from the table

Simply use the DISTINCT keyword after SELECT if you want to select only non-repeated rows. This keyword forces the query to discard any duplicate rows, based only on the columns you listed.

Examples;

I can use the DISTINCT keyword in a SELECT statement to select only the unique records, and then insert them into a new table.

```
CREATE TABLE new_customers AS

SELECT DISTINCT * FROM customers;
```

7) Print max salary for a particular department

```
SELECT DEPT_ID, MAX(SALARY) AS MAXIMUMSALARY FROM EMPLOYEE
GROUP BY DEPT_ID;
```

8) Use different operator in SQL

An operator is a symbol or keyword that is used to perform operations on one or more values, expressions, or conditions.

Different types of operator;

1)Arithmetic operator;

- SELECT 150 + 250; -- O/P = 400
- SELECT 145 - 75; -- O/P = 70
- SELECT 17 * 5; -- O/P = 85
- SELECT 49 / 7; -- O/P = 7.0000
- SELECT 21 % 5; -- O/P = 1

2)Bitwise operator;

- SELECT 15 & 7; --o/p = 7.
- SELECT 10 | 6; --o/p = 14.
- SELECT 8 ^ 3; --o/p = 11.

3)Logical operator;

- SELECT * FROM employee WHERE department = ‘IT’ AND age = 28;

- SELECT * FROM employee WHERE age = 25 OR age = 26;
- SELECT * FROM employee WHERE department = ‘IT’ AND NOT age = 28;

4)Comparison operator;

- SELECT * FROM inventory WHERE quantity < 10;
- SELECT * FROM orders WHERE amount > 1000;
- SELECT * FROM employees WHERE salary <> 50000;
- SELECT * FROM customers WHERE name = 'John';

9) What is Query to display first 5 Records from Employee table?

```
SELECT * FROM employee LIMIT 5;
```

10) What is Query to display last 5 Records from Employee table?

```
SELECT * FROM employee ORDER BY id DESC LIMIT 5;
```

11) How to fetch 3rd highest salary using Rank Function

```
SELECT salary FROM (
   SELECT salary, RANK() OVER (ORDER BY salary DESC) AS rank
   FROM employee
) AS ranked_salary
WHERE rank = 3;
```

12) How Can i create table with same structure with data of Employee table?

```
CREATE TABLE new_employee LIKE employee;
INSERT INTO new_employee SELECT * FROM employee;
```

12) Find Query to get information of Employee where Employee is not assigned to the department

```
   SELECT employee.*
FROM employee
LEFT JOIN department_assignment ON employee.id =
department_assignment.employee_id
```

```
WHERE department_assignment.department_id IS NULL;
```