1. What is the Difference between Aggregate function and Windows function?

| **Feature** | **Aggregate Function** | **Window Function** |
| --- | --- | --- |
| **Purpose** | Summarizes data across groups | Performs calculation across a "window" of rows |
| **Resulting Rows** | **Reduces** number of rows (one per group) | **Keeps** number of rows same |
| **Syntax** | Uses GROUP BY | Uses OVER() clause |
| **Examples** | SUM(), AVG(), COUNT() | ROW\_NUMBER(), RANK(), SUM() OVER(...) |
| **Grouping Scope** | Whole table or group of rows | Sliding or fixed window of rows |

**Query using the Window function :**

Select sum(column) over(order by names) as Sum\_rank

From table

o/p for windows function :

|  |  |
| --- | --- |
| Name | Sum\_win |
| Hello | 3000 |
| Heyy | 3000 |
| Hey | 5000 |

o/p for the Aggregate function :

|  |  |
| --- | --- |
| Name | Sum(salary) |
| Hey | 5000 |

1. Difference btw Union , Union All , Union Distinct ? where does union comes under in order of execution ?

|  |  |  |
| --- | --- | --- |
| UNION | Union ALL | Union Distinct |
| Appends the next table below the left had side table in distinct way  like if the value is repeating more then one time the it takes only one value | Appends the next table below the left had side table , doesn’t bother about the uniqueness | Same as the UNION |

Ex :

Empl\_name :

Vivek

Harsha

Rishee

Vinay

Contractor\_name

Harsha

Rishee

Vinay

Union all Union / Union Distinct

Vivek vivek

Harsha harsha

Rishee rishee

Vinay vinay

Harsha

Rishee

Vinay

Order of execution:  
FROM, ON, JOIN, Where, GROUP BY, HAVING, SELECT, WINDOW FUNCTION, DISTINCT, ORDER BY, LIMIT/OFFSET, UNION/ UNION ALL

1. Led and lag syntax ?

  lead(salary, 1) over (order by salary) as successor,

  lag(salary, 2) over (order by salary) as prev\_2

1. RegEXP ?
2. Get the pattern of for the particular condition

We have :

1. REGEXP\_CONTAINS(column, pattern) – checks where the columns row has the pattern
2. REGEXP\_EXTRACT(column, pattern) – extract all the pattern from the column
3. REGEXP\_REPLACE(column, pattern, replacement) – replace the existing columns with the patten with the thing which I want
4. REGEXP\_EXTRACT\_ALL(column, pattern) – get all the words in the sentences divided by spaces.
5. Corelated Queries , Sub Sequent Queries, Récursive Queries ?
6. Correlated Query :

--Dense\_rank with out window function (corelated query)

-- SELECT

--     e1.id,

--     e1.name,

--     e1.salary,

--     (

--         SELECT COUNT(DISTINCT e2.salary)

--         FROM CTE e2

--         WHERE e2.salary > e1.salary

--     ) + 1 AS dense\_rank

-- FROM CTE e1

-- ORDER BY dense\_rank;

* Subsequent Queries (Nested/Subqueries) :

query which was only runnedonce and the result is used in the code any where , no relation

-- select

-- \*

-- from Cte

-- where salary =(

-- select max(salary) as highest

-- from CTE

-- )

* **Recursive Queries (WITH RECURSIVE) :**
* With RECURSIVE emp\_chain AS (
* -- Anchor
* SELECT id, name, salary, desig, mgr\_id, 1 AS level
* FROM CTE
* WHERE mgr\_id IS NULL
* UNION ALL
* -- Recursive part
* SELECT e.id, e.name, e.salary, e.desig, e.mgr\_id, c.level + 1 AS level
* FROM CTE e
* JOIN emp\_chain c ON e.mgr\_id = c.id
* )
* SELECT \* FROM emp\_chain
* order by level