Date & Time Functions

Agenda:

In today's session, we'll cover essential topics, including:-

- Problem Statement
- Datetime format
- Parse_date()
- Extract()
- Date(), Time()
- Date_add(), Date_sub()
- Date_diff()
- Current_date()

Summary of Previous Lecture:

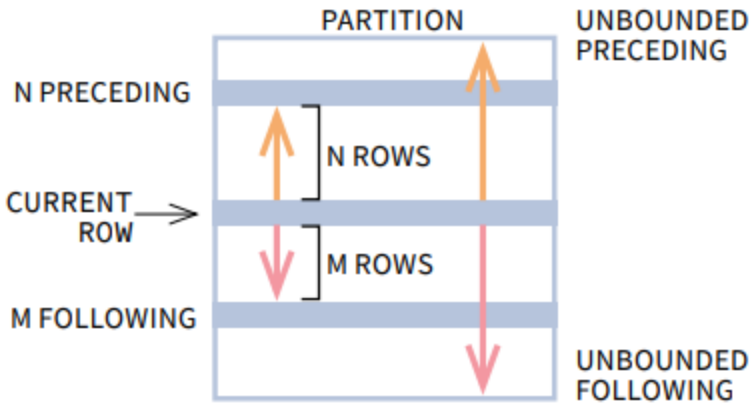
Aggregate Window Functions

- SUM()
- MIN()
- MAX()
- COUNT()
- AVG()
- Example: Calculate the average price for each market date:

```
AVG(original_price) OVER(PARTITION BY market_date) AS avg_price_by_date
```

Window Frame

- A window frame is a set of rows that are somehow related to the current row.
- The window frame is evaluated individually within each partition.



- Syntax for window frame is
  - `ROWS | RANGE BETWEEN lower_bound AND upper_bound`
- The bounds (lower_bound, upper_bound) can be any of the following five options:
  - UNBOUNDED PRECEDING
  - n PRECEDING
  - CURRENT ROW
  - n FOLLOWING
  - UNBOUNDED FOLLOWING

Abbreviation	Meaning
UNBOUNDED PRECEDING	BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW
n PRECEDING	BETWEEN n PRECEDING AND CURRENT ROW
CURRENT ROW	BETWEEN CURRENT ROW AND CURRENT ROW
n FOLLOWING	BETWEEN AND CURRENT ROW AND n FOLLOWING
UNBOUNDED FOLLOWING	BETWEEN CURRENT ROW AND UNBOUNDED FOLLOWING

- Example: Calculate a 5-day moving average of stock prices:

```
AVG(price) OVER (ORDER BY date RANGE BETWEEN 4 PRECEDING
AND CURRENT ROW) AS moving_avg
```

LEAD() - This function allows us to retrieve data from the next row of the current row in the same result set.

- **Syntax**: `LEAD(expression, offset, default_value) OVER(PARTITION BY columns ORDER BY columns)`

LAG() - This function allows us to retrieve data from the preceding row of the current row in the same result set.

- **Syntax**: `LAG(expression, offset, default_value) OVER(PARTITION BY columns ORDER BY columns)`

Note: Both LEAD() and LAG() functions have similar syntax and take three arguments:

- the name of a column or an expression,
- the offset to be skipped below or above, and
- the default value to be returned if the stored value obtained from the row below is empty. It is specified only if you specify the second argument, the offset.

NTILE() - This function divides rows within a partition as equally as possible into n groups and assigns each row its group number.

- Example: Divide rows into 10 equally sized price buckets:

```
    NTILE(10) OVER (ORDER BY original_price DESC) AS
price_ntile
```

NTH_VALUE(): This function returns the value of a given expression from the N^{th} row of the window frame. If that Nth row does not exist, the function returns NULL.

- The syntax of the NTH_VALUE() function is given as

```
NTH_VALUE(expression, N)
```

FIRST_VALUE(): This window function returns the first value in an ordered partition of a result set.

LAST_VALUE(): This window function returns the first value in an ordered partition of a result set.

- Example: Find the second-highest salary in each department:

```
    NTH_VALUE(employee_name, 2) OVER (PARTITION BY department
ORDER BY salary DESC) AS second_highest_salary
```