SQL PROBLEM CHALLENGE

DAY 11

- Problem Statement
- Expected Output
- Solution
- Solution Breakdown

PROBLEM STATEMENT

In the given input table, there are hotel ratings which are either too high or too low compared to the standard ratings the hotel receives each year. We need to identify and exclude these outlier records as shown in expected output.

INPUT		
HOTEL	YEAR	RATING
Radisson Blu	2020	4.8
Radisson Blu	2021	3.5
Radisson Blu	2022	3.2
Radisson Blu	2023	3.4
InterContinental	2020	4.2
InterContinental	2021	4.5
InterContinental	2022	1.5
InterContinental	2023	3.8

EXPECTED OUTPUT

OUTPUT		
HOTEL	YEAR	RATING
Radisson Blu	2021	3.5
Radisson Blu	2022	3.2
Radisson Blu	2023	3.4
InterContinental	2020	4.2
InterContinental	2021	4.5
InterContinental	2023	3.8

SOLUTION

```
WITH hotel_stats as (
    SELECT
         *
        ,AVG(rating) OVER (PARTITION BY hotel order by year
                           range between unbounded preceding and unbounded following) AS Hotel_Avg
        ,STDEV(rating) OVER (PARTITION BY hotel order by year
                           range between unbounded preceding and unbounded following) AS Hotel_Std
    FROM hotel_ratings
)
SELECT hotel,year,rating
FROM hotel_stats AS s
WHERE s.rating BETWEEN s.Hotel_Avg - s.Hotel_Std AND s.Hotel_Avg + s.Hotel_Std
ORDER BY s.hotel desc, s.year;
```

LET’S CHECK OUT THE BREAKDOWN OF THIS SOLUTION

We are using a simple outlier detection technique based on the mean and standard deviation of ratings for each hotel over the years.

PURPOSE:
To identify ratings that significantly deviate from the average for each hotel, as defined by the standard deviation.

1. Calculating Average and Standard Deviation for each HOTEL:
    - **Average (Hotel_Avg):** It computes the average rating of the hotel over the years.
    - **Standard Deviation (Hotel_Std)**: It calculates the standard deviation of ratings for the hotel over the years.

```sql
SELECT
      *
     ,AVG(rating) OVER (PARTITION BY hotel order by year
                        range between unbounded preceding and unbounded following) AS Hotel_Avg
     ,STDEV(rating) OVER (PARTITION BY hotel order by year
                        range between unbounded preceding and unbounded following) AS Hotel_Std
FROM hotel_ratings
```

	hotel	year	rating	Hotel_Avg	Hotel_Std
1	InterContinental	2020	4.2	3.5	1.36381816969859
2	InterContinental	2021	4.5	3.5	1.36381816969859
3	InterContinental	2022	1.5	3.5	1.36381816969859
4	InterContinental	2023	3.8	3.5	1.36381816969859
5	Radisson Blu	2020	4.8	3.825	0.694622199472488
6	Radisson Blu	2021	3.5	3.825	0.694622199472488
7	Radisson Blu	2022	3.2	3.825	0.694622199472488
8	Radisson Blu	2023	3.8	3.825	0.694622199472488

2. Defining an Acceptable Range:

Upper Limit: Mean + 1 * standard deviation.
Lower Limit: Mean – 1 * standard deviation.

Outliers = (Ratings < Lower Limit) && (Ratings > Upper Limit)

Ratings below the lower limit and above the upper limit are considered potential outliers and are excluded from the final result set.

Generally, we use 3 std dev from mean but here we have only few records, so we will be using 1 std dev as the threshold.

```sql
WITH hotel_stats as (
    SELECT
         *
        ,AVG(rating) OVER (PARTITION BY hotel order by year
                           range between unbounded preceding and unbounded following) AS Hotel_Avg
        ,STDEV(rating) OVER (PARTITION BY hotel order by year
                           range between unbounded preceding and unbounded following) AS Hotel_Std
    FROM hotel_ratings
)
SELECT hotel,year,rating
FROM hotel_stats AS s
WHERE s.rating BETWEEN s.Hotel_Avg - s.Hotel_Std AND s.Hotel_Avg + s.Hotel_Std
ORDER BY s.hotel desc, s.year;
```

Filtering the base CTE using upper and Lower Limits gives our final result.

	hotel	year	rating
1	Radisson Blu	2021	3.5
2	Radisson Blu	2022	3.2
3	Radisson Blu	2023	3.8
4	InterContinental	2020	4.2
5	InterContinental	2021	4.5
6	InterContinental	2023	3.8

THANK YOU