

Day56 - April 1st 2024

1. Started my day as usual

2. Began solving leetcode problems **134. Gas Station**

Problem approach doc :

<https://docs.google.com/document/d/1MvEWYtYVzpZ21IG3c2j8CTwUUc0Vqr5uBFCj-rNrHY/edit?usp=sharing>

3. Learned all types of slowly changing dimensions in data modeling

Find the doc here :

<https://docs.google.com/document/d/1RZkiay4BPawjVLWjkOJ50Irp4UwDG4mcuNnyBw00v9Y/edit?usp=sharing>

4.

Customer - dim - tbl

Customer-key	Customer-id	name	Address	State
1	RM101	Manish	Gurgaon	Haryana
2	RM102	Rohan	Gurgaon	Haryana
3	RM103	Rahul	Banaras	UP
4	RM104	Vikash	Amrit	Bihar
5	RM105	Pratham	Coimbatore	Tamil Nadu

17. History retention is very hard to implement

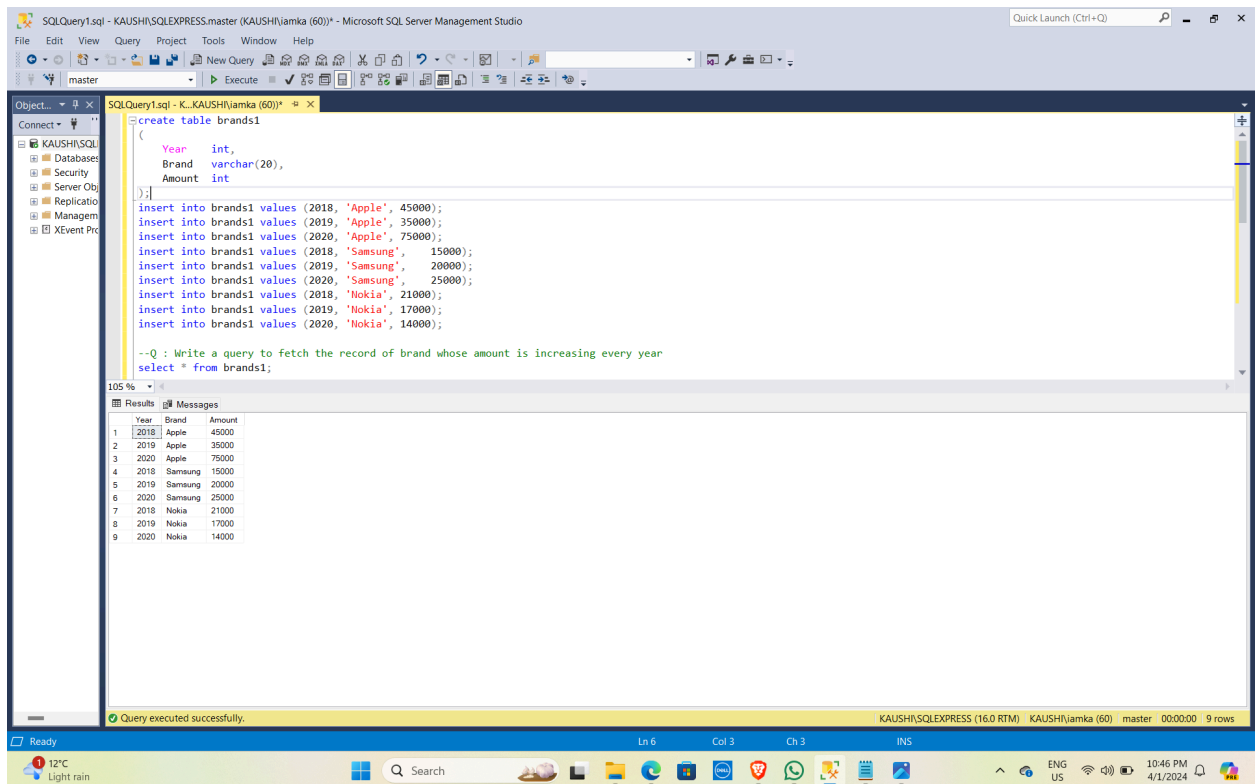
18. Now to implement it..we have to introduce 3 more cols

Customer - dim - tbl

Customer-key	Customer-id	name	Address	State	Status	Start date	End date
1	RM101	Manish	Gurgaon	Haryana	Y	13-11-2023	31-12-2023
2	RM102	Rohan	Gurgaon	Haryana	Y		31-12-2023
3	RM103	Rahul	Banaras	UP	Y		31-12-2023
4	RM104	Vikash	Amrit	Bihar	Y		31-12-2023
5	RM105	Pratham	Coimbatore	Tamil Nadu	Y		31-12-2023

20. Here in end\_date instead of keeping null..we'll set that to the infinite date..which helps us in joining

## 5. Ended my day by solving SQL question from Ankit Bansal YT



The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

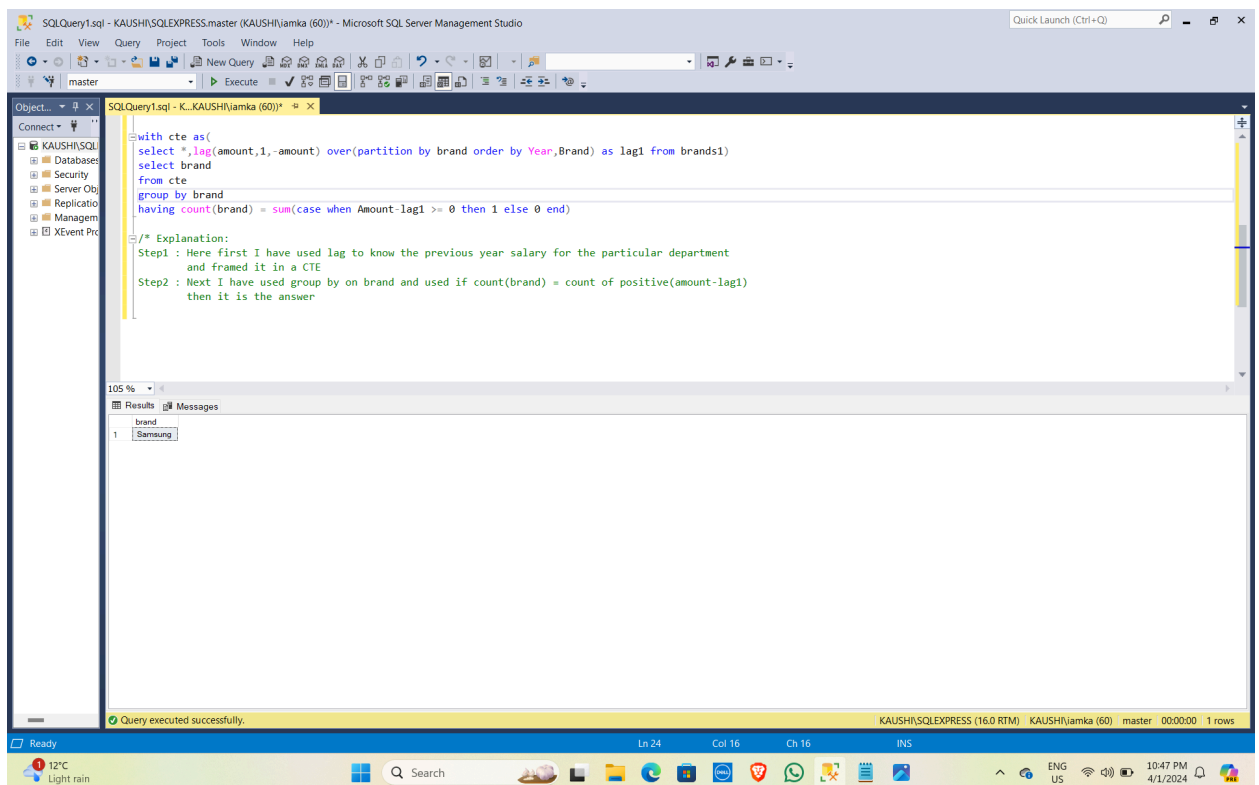
```
create table brands1
(
    Year int,
    Brand varchar(20),
    Amount int
);
insert into brands1 values (2018, 'Apple', 45000);
insert into brands1 values (2019, 'Apple', 35000);
insert into brands1 values (2020, 'Apple', 75000);
insert into brands1 values (2018, 'Samsung', 15000);
insert into brands1 values (2019, 'Samsung', 20000);
insert into brands1 values (2020, 'Samsung', 25000);
insert into brands1 values (2018, 'Nokia', 21000);
insert into brands1 values (2019, 'Nokia', 17000);
insert into brands1 values (2020, 'Nokia', 14000);

--Q : Write a query to fetch the record of brand whose amount is increasing every year
select * from brands1;
```

The Results pane shows the output of the query, displaying a table with 9 rows and 3 columns: Year, Brand, and Amount.

Year	Brand	Amount
2018	Apple	45000
2019	Apple	35000
2020	Apple	75000
2018	Samsung	15000
2019	Samsung	20000
2020	Samsung	25000
2018	Nokia	21000
2019	Nokia	17000
2020	Nokia	14000

The status bar at the bottom indicates "Query executed successfully." and "KAUSHI\SQLEXPRESS (16.0 RTM) KAUSHI\jamka (60) master 00:00:00 9 rows".



The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
with cte as(
    select *, lag(amount, 1, -amount) over(partition by brand order by Year, Brand) as lag1 from brands1)
select brand
from cte
group by brand
having count(brand) = sum(case when Amount-lag1 >= 0 then 1 else 0 end)
```

The Results pane shows the output of the query, displaying a table with 1 row and 1 column: brand.

brand
Samsung

The status bar at the bottom indicates "Query executed successfully." and "KAUSHI\SQLEXPRESS (16.0 RTM) KAUSHI\jamka (60) master 00:00:00 1 rows".