ASSIGNMENT 2

Course- CS 561-A

Submitted By-

Yugaank Arun Sharma

Stevens Id- 10419077

Canvas Id- ysharma

You will continue with evaluating simple report queries and produce the output. As with the assignment #1, you will also express the queries in SQL. The reports below are similar in nature with the reports from the assignment #1; however, there are two main differences between the two: (1) the new reports will require aggregation "outside" the groups (in assignment #1, all of the aggregates were computed for the rows within the groups); (2) some of the aggregates in the new reports will be computed based on other aggregates of the same reports – they are known as "dependent aggregates".

To test my submission, follow these steps-

There are 3 programs, each one generates 3 different outputs which matches the output of the sql queries. Keep in mid to change user name/password etc. that is needed to be done so you can test on your machine. And execute the program.

In the root folder along with the project folder, is another folder called 'SQL Queries' which contains three .sql files which generate the same output that the submitted java programs does.

In these reports, we are aggregating the values outside the groups. Some of the other aggregates are computed based on other aggregates of the report. These are also known as dependent aggregates.

For reports 1 and 2- only one scan is done on the sales table but for report 3- 2 scans were made in order to compare the sales that were between the average sales and the maximum sales as required by the condition.

SQL Queries

- **For 1st report-**

```
select cust as customer, prod as product, avg(quant) as the_avg,
        (select avg(quant) from sales where prod!=a.prod and cust=a.cust) as other_product_avg,
        (select avg(quant) from sales where prod=a.prod and cust!=a.cust) as other_cust_avg
from sales a group by cust, prod order by cust, prod
```

- **For 2nd report-**

```
with

a1 as (select distinct cust, prod, t1.month from sales,(select distinct month from sales) as t1)


select a1.cust as customer, a1.prod as product, a1.month,

        (select avg(quant) as before_avg from sales where cust=a1.cust and prod=a1.prod and month=a1.month-1 group by cust, prod, month),

        (select avg(quant) as after_avg from sales where cust=a1.cust and prod=a1.prod and month=a1.month+1 group by cust, prod, month)

        from a1 order by customer, product, month
```

- For 3rd report-

```
with

a1 as (   select foo1.prod, this, count(before_quant) as BEFORE_TOT from

        (select   prod, month as this, avg(quant) as q1, max(quant) as q2 from sales group by prod, month) as foo1

        left outer join

        (select prod, month as before_month, quant as before_quant from sales) as foo2 on foo1.prod=foo2.prod and foo1.this=foo2.before_month+1 and before_quant between q1 and q2
```

```
        group by foo1.prod, this,q1,q2 order by foo1.prod, this),

a2 as ( select foo1.prod, this, count(after_quant) as AFTER_TOT from

        (select  prod, month as this, avg(quant) as q1, max(quant) as q2 from sales group by prod, month) as foo1

        left outer join

        (select prod, month as after_month, quant as after_quant from sales) as foo3 on foo1.prod=foo3.prod and foo1.this=foo3.after_month-1 and after_quant between q1 and q2

        group by foo1.prod, this,q1,q2 order by foo1.prod, this)


 select prod as PRODUCT, this as MONTH, BEFORE_TOT, AFTER_TOT from a1 natural full outer join a2
```