Stevens Institute of Technology
Castle Point on Hudson
Hoboken, NJ 07030

# CS561 – SQL Programming Assignment 1

**Due Dates: 11/3/2020 (Tue) for Sec. A & 10/29/2020 (Thu) for Sec. B**

## *Objectives*

In this assignment, you will *express* "complex" OLAP queries in SQL. The key point of the exercise is to observe the complexity of expressing the type of such queries despite relatively simple ideas of the queries themselves. Your mission (in addition to writing the SQL queries) is to consider the reasons for the complexity of the *expression* of these queries.

## *Description*

Generate separate reports/output based on the following queries (one report for each of the queries):

1. For each *customer*, compute the *minimum* and *maximum* sales quantities along with the *corresponding products*, *dates* (i.e., dates of those maximum and minimum sales quantities) and the *states* in which the sale transactions took place. For the same *customer*, also compute the *average* sales quantity.
2. For each combination of *customer* and *product*, output the *minimum sales quantities for January* (regardless of the year, that is, both 1/11/2000 and 1/23/2008 are considered sales transactions for January) and *maximum sales quantities for February and March (again, regardless of the year) in 3 separate columns*. Like the first report, display the *corresponding dates* (i.e., dates of those maximum and minimum sales quantities). Furthermore, for January (MIN), include only the sales that occurred after 1999 (that is, not to include sales that occurred in 1999 or earlier); for February (MAX) and March (MAX), include all sales.
3. For *each of the 12 months* (regardless of the year), find the *most "popular"* and *least "popular" products* (those products with most and least total sales quantities) and the corresponding *total sales quantities (i.e., SUMs)*.
4. For each *product*, find the "*most favorable*" *month* (when most amount of the product was sold) and the "*least favorable*" *month* (when the least amount of the product was sold).
5. Show, for each *product* and *customer* combination, the *average sales quantities for the 4 states*, 'CT', 'NJ', 'NY' and 'PA' (in four separate columns). Also compute the *average for the "whole" year* (again ignoring the YEAR component, meaning simply compute AVG) along with the *total quantities* (SUM) and the *counts* (COUNT).

The following are sample output reports – quantities displayed are for illustration only (not the actual values). **For dates (e.g., MAX_DATE, MIN_DATE), you can display 'month', 'day' and 'year' as 3 separate columns**.

Report #1:

| CUSTOMER | MIN_Q | MIN_PROD | MIN_DATE | ST | MAX_Q | MAX_PROD | MAX_DATE | ST | AVG_Q |
|---|---|---|---|---|---|---|---|---|---|
| Bloom | 12 | Pepsi | 01/01/2006 | NJ | 2893 | Apple | 09/25/2001 | NY | 1435 |
| Sam | 1 | Milk | 02/15/2002 | NJ | 259 | Banana | 03/23/2004 | CT | 56 |
| Emily | 1 | Bread | 07/01/2005 | NY | 3087 | Milk | 02/02/2001 | NJ | 1512 |
| . . . . | | | | | | | | | |

Report #2:

| CUSTOMER | PRODUCT | JAN_MIN | JAN_DATE | FEB_MAX | FEB_DATE | MAR_MAX | MAR_DATE |
|---|---|---|---|---|---|---|---|
| Sam | Egg | 8 | 01/11/2001 | 3234 | 02/24/2005 | 2432 | 03/03/2008 |
| Helen | Cookies | 92 | 01/22/2002 | 4342 | 02/14/2000 | 9483 | 03/23/2002 |
| Bloom | Butter | 45 | 01/31/2000 | 1923 | 02/10/2004 | 2596 | 03/11/2006 |
| . . . . | | | | | | | |

Report #3:

| MONTH | MOST_POPULAR_PROD | MOST_POP_TOTAL_Q | LEAST_POPULAR_PROD | LEAST_POP_TOTAL_Q |
|---|---|---|---|---|
| 1 | Eggs | 497214 | Pepsi | 55526 |
| 2 | Milk | 1874794 | Banana | 23126 |
| 3 | Pepsi | 974531 | Milk | 19958 |

. . . .

Report #4:

| PRODUCT | MOST_FAV_MO | LEAST_FAV_MO |
|---|---|---|
| Egg | 4 | 12 |
| Apple | 1 | 11 |
| Banana | 3 | 2 |

. . . .

Report #5:

| PRODUCT | CUSTOMER | CT_AVG | NJ_AVG | NY_AVG | PA_AVG | AVERAGE | TOTAL | COUNT |
|---|---|---|---|---|---|---|---|---|
| Pepsi | Sam | 1923 | 4241 | 2383 | 1325 | 2988 | 38848 | 13 |
| Milk | Emily | 239 | 9872 | 142 | 2435 | 2663 | 21307 | 8 |
| Bread | Helen | 2534 | 981 | 4239 | 1987 | 2781 | 25032 | 9 |

. . . .

## *Grading*

**NOTE: A query with syntax errors will lose 50% of the points for the query**.

## *Submission*

Submit **one file** containing all of the 5 queries with your name and CWID on it on Canvas. The file type must be "TXT".

Please include a "README" section in the same file if any special instructions are required.

You can discuss the "ideas" with your class mates or your friends, but the final queries must be your own work. If I determine that your queries are copies of someone else's, both you and that someone else will be disciplined (you will receive 0 for the entire assignment) and possibly receive additional penalties for the course.