## **Name: Jianwei Su**

**Date:12/13/2022**

# **Exercises**

## **Level 2: Intermediate SQL Syntax**

***Note: For all SQL homeworks, just paste your code under the question in the Word document. No images or proof of output are necessary. Please make your answers bold and blue.***

Sample Question: How would you select all position ids from table TRADE\_DATA\_HIST?

A: **SELECT POSITION\_ID FROM DBO.TRADE\_DATA\_HIST**

### **2.1: CASE WHEN**

1. Return all employees first and last name from EMPLOYEE\_INFO, and add a new column that says “FRONT\_OFFICE” if the employee is in a revenue generating position and “BACK\_OFFICE” if not. A C-Level is neither front or back office. I’ll leave this up to you how to handle this ☺.

**SELECT**

**FIRST\_NAME,LAST\_NAME,**

**CASE WHEN**

**EMPLOYEE\_ID LIKE 'T%' THEN 'FRONT\_OFFICE'**

**WHEN EMPLOYEE\_ID LIKE 'C%' THEN 'NEITHER\_FRONT\_NOR\_BACK\_OFFICE'**

**ELSE 'BACK\_OFFICE'**

**END AS OFFICE\_POSITION**

**FROM**

**DBO.employee\_info**

 Note:In finance, traders, sales, and client facing employees generate revenue for the firm. IT, compliance, HR, and risk don’t generate revenue, they just support the firm.

1. Return DATE, CUSIP, QUANTITY, and a new column that returns SMALL, MEDIUM, and LARGE if QUANTITY is 5,000 < 25,000 < 50,000

**SELECT**

**COB\_DATE,CUSIP,QUANTITY,**

**CASE WHEN**

**QUANTITY < 5000 THEN 'SMALL'**

**WHEN QUANTITY > 50000 THEN 'LARGE'**

**ELSE 'MEDIUM '**

**END AS QUANTITY\_LEVEL**

**FROM**

**DBO.trade\_data\_hist**

### 

### **2.2: ALIAS**

1. Return all of the columns from table SECURITY\_INFO but rename the longer column names to shorter ones

**SELECT**

**START\_DATE AS S\_D,**

**END\_DATE AS E\_D,**

**CUSIP,**

**INSTRUMENT\_TYPE AS I\_T,**

**DELISTED AS DE\_D,**

**TICKER,**

**DESCRIPTION AS D\_P,**

**SECTOR,**

**COUNTRY AS COUN**

**FROM**

**DBO.security\_info**

1. Write a select statement with multiple filters using aliases for both columns and the table you are selecting from

**SELECT**

**S\_I .START\_DATE AS S\_D,**

**S\_I.END\_DATE AS E\_D,**

**S\_I.CUSIP,**

**S\_I.INSTRUMENT\_TYPE AS I\_T,**

**S\_I.DELISTED AS DE\_D,**

**S\_I.TICKER,**

**S\_I.DESCRIPTION AS D\_P,**

**S\_I.SECTOR,**

**S\_I.COUNTRY AS COUN**

**FROM**

**DBO.security\_info AS S\_I**

**WHERE**

**S\_I.CUSIP LIKE 'C5%'**

**AND**

**S\_I.TICKER LIKE 'A%'**

### **2.3: COALESCE**

1. Write a query to return all DATE, CUSIP, DURATION, but when duration is NULL it should return CONVEXITY instead

**SELECT**

**COB\_DATE,**

**CUSIP,**

**COALESCE(DURATION,CONVEXITY) AS DURATION**

**FROM**

**DBO.RISK\_DATA\_HIST**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

When the column has empty cells, we can do it in the following way:

**SELECT**

**COB\_DATE,**

**CUSIP,**

**COALESCE(NULLIF(DURATION,** **' '),CONVEXITY) AS DURATION**

**FROM**

**DBO.RISK\_DATA\_HIST**

1. List all of the tickers in security info that are active. If there is no TICKER return CUSIP. If there is no CUSIP, return DESCRIPTION.

**SELECT**

**COALESCE(TICKER,CUSIP,DESCRIPTION) AS TICKER\_NEW**

**FROM**

**DBO.SECURITY\_INFO**

### **2.4: FUNCTIONS**

1. Return the aggregated quantity of each CUSIP from TRADE\_DATA\_HIST on each day.

**SELECT**

**COB\_DATE,**

**CUSIP,**

**SUM(QUANTITY) QTY\_SUM**

**FROM**

**DBO.TRADE\_DATA\_HIST**

**GROUP BY**

**COB\_DATE,CUSIP**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**If the user only want to show the non empty cusip, we can do it in this way:**

**SELECT**

**COB\_DATE,**

**CUSIP,**

**SUM(QUANTITY) QTY\_SUM**

**FROM**

**DBO.TRADE\_DATA\_HIST**

**WHERE NULLIF(CUSIP,'') IS NOT NULL**

**GROUP BY**

**COB\_DATE,CUSIP**

1. Return the aggregated quantity of each CUSIP for each EMPLOYEE\_ID. (Date does not matter here)

**SELECT**

**EMPLOYEE\_ID,**

**CUSIP,**

**SUM(QUANTITY) QTY\_SUM**

**FROM**

**DBO.TRADE\_DATA\_HIST**

**GROUP BY**

**EMPLOYEE\_ID, CUSIP**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**If the user only want to show the non empty cusip, we can do it in this way:**

**SELECT**

**EMPLOYEE\_ID,**

**CUSIP,**

**SUM(QUANTITY) QTY\_SUM**

**FROM**

**DBO.TRADE\_DATA\_HIST**

**WHERE NULLIF(CUSIP,'') IS NOT NULL**

**GROUP BY**

**EMPLOYEE\_ID, CUSIP**

1. What is the average QUANTITY of CUSIP **C146500095** where the trader is T5?

**SELECT**

**EMPLOYEE\_ID,**

**CUSIP,**

**AVG(QUANTITY) QTY\_AVG**

**FROM**

**DBO.TRADE\_DATA\_HIST**

**WHERE**

**EMPLOYEE\_ID ='T5'**

**AND**

**CUSIP ='C146500095'**

**GROUP BY**

**EMPLOYEE\_ID,CUSIP**

1. What is the average QUANTITY of CUSIP **C146500095** where the trader is T5 and the month is January (could be multiple years, should only check that the month is Jan)?

**SELECT**

**COB\_DATE,**

**EMPLOYEE\_ID,**

**CUSIP,**

**AVG(QUANTITY) QTY\_AVG**

**FROM**

**DBO.TRADE\_DATA\_HIST**

**WHERE**

**(COB\_DATE LIKE '201801%'**

**OR**

**COB\_DATE LIKE '201901%')**

**AND**

**EMPLOYEE\_ID ='T5'**

**AND**

**CUSIP ='C146500095'**

**GROUP BY**

**COB\_DATE, CUSIP, EMPLOYEE\_ID**

### **2.5: DECIMAL PRECISION**

1. Return the full dataset from table RISK\_DATA\_HIST but the last three columns should only have three decimals each using the CAST method.

**SELECT**

**COB\_DATE,**

**POSITION\_ID,**

**CUSIP,**

**CAST(VaR AS DECIMAL(10,3)) AS VAR,**

**CAST(Duration AS DECIMAL(10,3)) AS DURATION,**

**CAST(Convexity AS DECIMAL(10,3)) AS CONVEXITY**

**FROM**

**DBO.RISK\_DATA\_HIST**

1. Do the same as above using ROUND.

**SELECT**

**COB\_DATE,**

**POSITION\_ID,**

**CUSIP,**

**ROUND(VaR,3) AS VAR,**

**ROUND(Duration,3) AS DURATION,**

**ROUND(Convexity,3) AS CONVEXITY**

**FROM**

**DBO.RISK\_DATA\_HIST**