DAX Data Analysis Expressions



Masoud Mirzakhani Senior DW/ ETL/ BI Architect

Data Warehouse Design & Develop



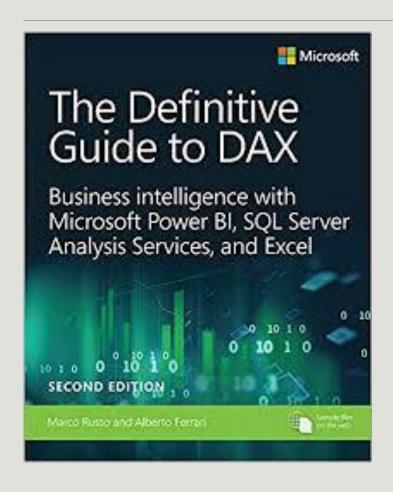
Masoud Mirzakhani Senior DW/ ETL/ BI Architect

- Master of Science in Information Technology
- QAZVIN AZAD UNIVERSITY
- Bachelor of Science in Information Technology
- PARAND AZAD UNIVERSITY
- md.mirzakhani@gmail.com
- @MasoudMirzakhani
- linkedin.com/in/masoudmirzakhani



منابع





DAX



- DAX یک زبان بر اساس فرمول نویسی است.
 - شبیه به توابع Excel رفتار می کند.
- DAX از دو روش در مدل SSAS بارگزاری می شود:
 - calculated column o
 - calculate measure o

DAX Formula



- پس از محاسبه یک مقدار بر می گردانند.
- رای ایجاد column ،kpi ،measure استفاده می شوند.
 - داده را به صورت یک جدول بر می گردانند.
 - برای ایجاد dataset استفاده می شوند.
- و برای تولید مجموعه داده و استفاده در فرمول دیگری استفاده می شوند.

DAX Formula Measure



- Sum_debit_balance: = SUM('BI FactAccountingVoucher'[DebitBalance])
- count_records := COUNT('BI FactAccountingVoucher'[ID])
- SSAS column summarized by options:
 - *Sum
 - Average
 - Count
 - DistinctCount
 - Max
 - * Min

DAX Formula Calculated Columns



- * = 'BI FactAccountingVoucher'[Credit] 'BI FactAccountingVoucher'[Debit]
- = YEAR('BI DimDate'[GregorianDate])

DAX Formula Conditions



- IF(<logical_test>,<value_if_true>,<value_if_false>)
- SWITCH(<expression>,<value>,<result>[, <value2>,<result2>,][,else])
- * =IF('BI FactSales'[SalesDocumentID] = 3, 'BI FactSales'[NetPrice], 0)
- * = SWITCH('BI FactSales'[SalesDocumentID], 1, "Sales", 2, "Return Sales", "Net Sales")

DAX Formula String



- = CONCATENATE("Y", YEAR('BI DimDate'[GregorianDate]))
- LEFT(<text>,<num_chars>)
- RIGHT(<text>,<num_chars>)

DAX Formula Relations



- RELATED(<column>)
- =RELATED('BI DimAccount'[Title])
- RELATEDTABLE(<tableName>)
- = COUNTROWS(RELATEDTABLE('BI FactAccountingVoucher'))

DAX Formula Filter



- FILTER(,<filter>)
- sum_of_net_sales:=SUMX(FILTER('BI FactSales', 'BI FactSales'[SalesDocumentID] = 3), 'BI FactSales'[NetPrice])
- sum_debit_1398:= CALCULATE(SUM('BI FactAccountingVoucher'[Debit]), 'BI DimDate'[CalendarYear] = 1398)
- *=COUNTROWS(FILTER(RELATEDTABLE('BI FactAccountingVoucher'), RELATED('BI DimDate'[CalendarYear]) >= 1398))

DAX Formula Filter



- ALL(|<column>)
 - to remove any conditions that are on your data
 - AllNetSales:= SUMX(ALL('BI FactSales'), 'BI FactSales'[NetSales])



- Select *
 - EVALUATE 'tablename'
 - EVALUATE 'BI FactAccountingVoucher'
 - **EVALUATE**

'BI FactAccountingVoucher'

ORDER BY

[ID] DESC





```
SUMMARIZE

EVALUATE
SUMMARIZE(

'BI FactAccountingVoucher',

'BI DimDate'[CalendarYear],

"Sum of Debit",

SUM('BI FactAccountingVoucher'[Debit])
```





```
❖ SUMMARIZE
 EVALUATE
 SUMMARIZE(
                 'BI FactAccountingVoucher',
                 ROLLUP(
                           'BI DimDate'[CalendarYear],
                           'BI DimAccount'[PrimaryTitle]
                 "Sum of Debit",
                 SUM('BI FactAccountingVoucher'[Debit])
```

DAX Date Functions



- * =CALENDAR("1/1/2020","1/1/2021")
- = = YEAR(DateTable[Date])
- = = MONTH(DateTable[Date])
- =FORMAT(DateTable[Date],"MMM")
 - ❖ "M": returns month number
 - *"MM": returns the 2-digit month number
 - *"MMM": returns the first three characters of the month name
 - *"MMMM": returns the full length of the month name
- = = WEEKDAY(DateTable[Date],1)
- =WEEKDAY(DATEADD(DateTable[Date],1,DAY))
- =FORMAT(WEEKDAY(DateTable[Date]), "DDD")

DAX Date Functions



- ❖ TODAY
 - current date
- ❖ NOW
 - current datetime

DAX Date Functions



- **♦** DATEDIFF
 - **❖** SECOND
 - ❖ MINUTE
 - **♦** HOUR
 - DAY
 - ❖ WEEK
 - **⋄** MONTH
 - **QUARTER**
 - YEAR
 - first_date := MIN('BI DimDate'[GregorianDate])
 - first_date := FIRSTDATE('BI DimDate'[GregorianDate])
 - last_date := LASTDATE('BI DimDate'[GregorianDate])
 - days_since_first_date :=DATEDIFF(first_date, TODAY(), DAY)

DAX Periods



```
sum_debit:= SUM('BI FactAccountingVoucher'[Debit])
sum_debit_pre_year:= CALCULATE(
                               SUM('BI FactAccountingVoucher'[Debit]),
                                PARALLELPERIOD('BI DimDate'[GregorianDate], -1, YEAR)
YOY_Growth:=([sum_debit]-[sum_debit_pre_year])/ [sum_debit_pre_year]
YOY_Growth_New:= IF( [sum_debit_pre_year],
                       ([sum_debit]-[sum_debit_pre_year])/ [sum_debit_pre_year],
                       BLANK()
```

DAX Periods



OpeningMonth:=OPENINGBALANCEMONTH(

SUM('BI FactAccountingVoucher'[Debit]),

'BI DimDate'[GregorianDate])

OpeningYear:=OPENINGBALANCEYEAR(

SUM('BI FactAccountingVoucher'[Debit]),

'BI DimDate'[GregorianDate])

DAX Periods



ClosingMonth:=CLOSINGBALANCEMONTH(

SUM('BI FactAccountingVoucher'[Debit]),

'BI DimDate'[GregorianDate])

ClosingYear:=CLOSINGBALANCEYEAR(

SUM('BI FactAccountingVoucher'[Debit]),

'BI DimDate'[GregorianDate])





YTDDebitBalance:=TOTALYTD(SUM('BI FactAccountingVoucher'[Debit]),
'BI DimDate'[GregorianDate])