



**Tajamul Khan**

# Power BI DAX Cheat Sheet



**@Tajamulkhan**



# Math

- **SUMX:** SUMX(Sales, Sales[Quantity] \* Sales[Price])
- **AVERAGE:** AVERAGE ('Table'[Column])
- **MIN:** MIN('Table'[Column])
- **MAX:** MAX('Table'[Column])
- **ROUND:** ROUND('Table'[Number], 2)
- **ABS:** ABS('Table'[Number])
- **EXP:** EXP('Table'[Exponent])
- **LOG:** LOG('Table'[Number], 10)



@Tajamulkhan



# Text

- **CONCATENATE:** CONCATENATE('Table' [Text1], 'Table' [Text2])
- **LEFT:** LEFT( 'Table' [Text] , 3)
- **RIGHT:** RIGHT( 'Table' [Text] , 5)
- **LEN:** LEN( 'Table' [Text] )
- **UPPER:** UPPER( 'Table' [Text] )
- **LOWER:** LOWER( 'Table' [Text] )
- **TRIM:** TRIM( 'Table' [Text] )
- **SEARCH:** SEARCH( "keyword" , 'Table' [Text] )
- **CONTAINSSTRING:** CONTAINSSTRING( 'Table' [Text] , "keyword" )
- **LEFT:** LEFT ( 'Table' [Text] , 3)
- **RIGHT:** RIGHT( 'Table' [Text] ,3)



# Date

- **TODAY:** TODAY()
- **NOW:** NOW()
- **YEAR:** YEAR('Table'[Date])
- **MONTH:** MONTH('Table'[Date])
- **DAY:** DAY ('Table'[Date])
- **DATEDIFF:** DATEDIFF('Table'[StartDate], 'Table'[EndDate], DAY)
- **EOMONTH:** EOMONTH('Table'[Date], 0)
- **FORMAT:** FORMAT('Table'[Date], "yyyy-mm-dd")



@Tajamulkhann



# Statistics

- **AVERAGEX:** AVERAGEX(  
    'Table', 'Table' [Column])
- **COUNT:** COUNT ( 'Table'  
    (Column)) |
- **COUNTA:** COUNTA (' Table'  
    [Column])
- **COUNTAX:** COUNTAX( 'Table',  
    'Table' [Column])
- **STDEV.P:** STDEV.P( 'Table'  
    [Column])
- **VAR.P:** VAR.P( 'Table'  
    (Column))



@Tajamulkhann



# Logical

- **IF:** IF( 'Table'[Column] > 10, "Yes", "No") |
- **AND:** AND('Table'[Column1] > 5, 'Table'[Column2] < 10)
- **OR:** OR('Table'[Column1] > 5, 'Table'[Column2] < 10)
- **NOT:** NOT('Table'[Flag]) |
- **SWITCH:** SWITCH('Table'[Category], "A", 1, "B", 2, 0)



@Tajamulkhan

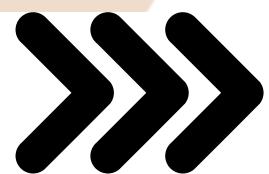


# Distribution

- **NORM. DIST:** NORM. DIST  
(1.96, 0, 1, TRUE)
- **NORM. INV:** NORM. INV (0.95,  
0, 1)
- **BINON. DIST:** BINOM. DIST(3,  
10, 0.5, FALSE)
- **POISSON. DIST:** POISSON.  
DIST(2, 5, FALSE)



@Tajamulkhan



# Finance

- **PV:** PV(0.05, 10, 1000, 0, 0)
- **FV:** FV(0.05, 10, -100, 0, 0)
- **MPV:** NPV(0.1, CashFlow1, CashFlow2, CashFlow3)
- **IRR:** IRR(CashFlows)
- **TOTALYTD:** TOTALYTD(SUM('Table'[Revenue]), 'Date'[Date])
- **CLOSINGBALANCEMONTH:**  
CLOSINGBALANCEMONTH ('Table'[Revenue], 'Date'[Date])
- **OPENINGBALANCEMONTH:**  
OPENINGBALANCEMONTH ('Table'[Revenue], 'Date'[Date])



@Tajamulkhan



# Ranking

- **RANKX:** RANKX( 'Table' ,  
'Table' [Sales] , , DESC)
- **TOPN:** TOPN(5 , 'Table' ,  
'Table'[Sales] , DESC)
- **RANK.EQ:** RANK.EQ( 'Table'  
[Sales] , 'Table' [Sales] ,  
DESC)



@Tajamulkhan



# Testing

- **T.TEST:** T. TEST('Group' [Data], 'Group2' [Data], 2, 1)
- **ANOVA:** ANOVA( 'Table' [Values], 'Table' [Category])
- **CHISQ. DIST:** CHISQ. DIST (3.84, 2, FALSE)
- **PERCENTILE.INC:** PERCENTILE. INC('Table' [Values], 0.75)
- **PERCENTILE.EXC:** PERCENTILE. EXC('Table' [Values], 0.75)
- **RANK. AVG:** RANK. AVG( 'Table' [Sales], 'Table' [Category], 1)
- **KEEPFILTERS:**  
KEEPFILTERS(CALCULATE (SUM('Table' [Sales]), 'Table' [Category] = "A"))



@Tajamulkhann

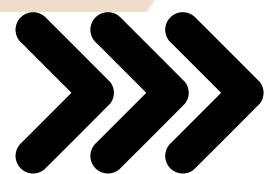


# Time Intelligence

- **TOTALYTD:** TOTALYTD(SUM ('Table' [Sales]), 'Date'[Date])
- **SAMEPERIODLASTYEAR:** CALCULATE(SUM('Table' [Sales]).  
SAMEPERIODLASTYEAR ('Date' [Date]))
- **YTD:** CALCULATE (SUM('Table' [Sales]),  
ALL('Date')), 'Date' [Date] <= MAX ('Date'  
[Date])) )
- **QUARTER:** QUARTER('Date' [Date])
- **MONTH:** MONTH 'Date' [Date])
- **WEEKDAY:** WEEKDAY ('Date' [Date], 2)
- **CALENDAR:** CALENDAR (DATE (2023, 1, 1),  
DATE(2023, 12, 31))
- **DATESBETWEEN:** DATESBETWEEN( 'Date' [Date],  
DATE(2022, 1, 1), DATE (2022, 12, 31))
- **TOTALMTD:** TOTALMTD(SUM( 'Table' [Sales]),  
'Date' [Date])
- **FIRSTDATE:** FIRSTDATE ('Date' [Date])
- **LASTDATE:** LASTDATE (( 'Date' [Date]))



@Tajamulkhan



# Table

- **VALUES:** VALUES('Table'[Column])
- **ALLSELECTED:** ALLSELECTED('Table')
- **ADDCOLUMNS:** ADDCOLUMNS ('Table', "Revenue", 'Table' [Quantity] \* 'Table' [Price])
- **SUMMARIZE:** SUMMARIZE ('Table', 'Table' [Category], "Total Sales", SUM( 'Table' [Sales]))
- **ROLLUP:** ROLLUP('Date', 'Date' [Year], 'Date' [Quarter], 'Date' [Month])
- **KEEPFILTERS:** KEEPFILTERS (CALCULATETABLE('Table', 'Table' [Column] > 100))
- **SELECTCOLUMNS:** SELECTCOLUMNS( 'Table', 'Table' [Column1], 'Table' [Column2])
- **SUMMARIZECOLUMNS:** SUMMARIZECOLUMNS( 'Table' [Column1], 'Table' [Column2], "Total Sales", SUM('Table'[Sales]))



@Tajamulkhan



# Parent-Child

- **PATH:** PATH('Table', 'Table' [ParentID], 'Table' [ID])
- **PATHITEM:** PATHITEM('Table' [Path], 1)
- **PATHLENGTH:** PATHLENGTH('Table' [Path])
- **ISFILTERED:** IF(ISFILTERED ('Table' [Column]), "Filtered", "Not Filtered")



@Tajamulkhann

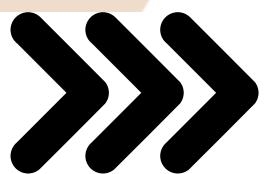


# Advanced

- **PREDICT:** PREDICT('Table', 'Table'[Value], FILTER('Table', 'Table'[Date] > DATE(2022, 1, 1)))
- **COVARIANCE.P:** COVARIANCE.P('Table1'[Values], 'Table2'[Values])
- **CORRELATION:** CORRELATION('Table1'[Values], 'Table2'[Values])
- **RANK.EQ:** RANK.EQ('Table'[Sales], 'Table'[Sales], DESC, 'Table'[Category])
- **PREDICT:** PREDICT('Table', 'Table'[Value], FILTER('Table', 'Table'[Date] > DATE(2022, 1, 1)))
- **COVARIANCE.P:** COVARIANCE.P('Table1'[Values], 'Table2'[Values])



@Tajamulkhann



# Information

- **ISBLANK:** IF ( ISBLANK ('Table' [Column]), "Blank", "Not Blank")
- **ISERROR:** IF(ISERROR(1/0), "Error", "No Error")
- **TYPEOF:** TYPEOF ('Table' [Column], INTEGER)



@Tajamulkhan



# Parameter

- **ISBLANK:** IF (ISBLANK ('Table' [Column]), "Blank", "Not Blank")
- **ISERROR:** IF(ISERROR(1/0), "Error", "No Error")
- **TYPEOF:** TYPEOF ('Table' [Column], INTEGER)



@Tajamulkhan



# Context

- **EARLIER:** CALCULATE (SUM('Table'[Sales]), 'Table'[Date] = EARLIER ('Table'[Date]) - 1)
- **FILTERS:** FILTERS( 'Table'[Category])
- **USERELATIONSHIP:**  
USERELATIONSHIP('Table1'[Column], 'Table2'[Column])



@Tajamulkhan



**Found  
Helpful ?**

---

**Repost**



**Follow for more!**