Power BI DAX Functions Cheat Sheet

Aggregation
Combine values into single result

Function	Description	Syntax
SUM	Adds all the numbers in a column.	= SUM([column])
AVERAGE	Calculate the mean of all the values in a column.	= AVERAGE([column])
MIN	Finds smallest value in a given column.	= MIN(table[column])
MAX	Finds largest value in a given column.	= MAX([column])
MINA	Finds smallest value including values & numbers represent as text.	= MINA([column])
MAXA	Finds largest value including values & numbers represent as text.	= MAXA([column])

Iterator
Perform calculations on each row of a table to solve specific problems

Function	Description	Syntax
SUMX	Fetches the sum of an expression evaluated for each row in a table.	= SUMX(Table, Expression)
AVERAGEX	Calculates the mean of a set of expression evaluated over a table.	= AVERAGEX(Table, Expression)
MINX	Finds smallest value that results from expression for each row of table	= MINX(Table, Expression, [Variant] - Optional)
MAXX	Finds largest value that results from expression for each row of table.	= MAXX(Table, Expression, [Variant] - Optional)
COUNTX	Counts number of rows of an expression evaluating non-blank value.	= COUNTX(Table, Expression)

Counting
Count the number of records or cells that meet specific criteria

Function	Description	Syntax
DISTINCTCOUNT	Counts the number of distinct values in a column.	= DISTINCTCOUNT([column])
COUNT	Count the items in a column even if repetitions are present.	= COUNT([column])
COUNTA	Counts the items in a column that is not empty.	= COUNTA([column])
COUNTROWS	Counts the number of rows in a table defined by expression.	= COUNTROWS(table)
COUNTBLANK	Counts the number of blank cells in a column.	= COUNTBLANK([column])

Math & Trig
Extract insights from data by performing calculations and manipulating data

Function	Description	Syntax
ABS	Returns the absolute value of a number.	= ABS([column])
CURRENCY	Returns the result as currency data type.	= CURRENCY([column])
DIVIDE	Performs division of 2 numbers, used to handle division by zero error.	= DIVIDE(Numerator, Denominator, [Result])
INT	Rounds a number down to the nearest integer.	= INT([column])
MROUND	Returns a number rounded to the desired multiple.	= MROUND([column], Multiple)
ROUND	Rounds a number to the specified number of digits.	= ROUND([column], NumberDigits)
QUOTIENT	Performs division, returns only integer of division result.	= QUOTIENT(Numerator, Denominator)
TRUNC	Removes any digits beyond specified number of decimal places.	= TRUNC([Column], NumberDdigits)

Statistical
Help users to gain deeper insights from the data

Function	Description	Syntax
MEDIAN	Returns the median of numbers in a column.	= MEDIAN([column])
STDEV.S	Returns the standard deviation of a sample population.	= STDEV.S([column])
STDEV.P	Returns the standard deviation of a entire population.	= STDEV.P([column])
VAR.S	Returns the variance of a sample population.	= VAR.S([column])
VAR.P	Returns the variance of a entire population.	= VAR.P([column])

Financial
Perform financial calculations, such as net present value and rate of return

Function	Description	Syntax
FV (Future Value)	Calculates investment future value based on constant interest rate. LoanAmount: Present value or lump-sum amount. Type: When payments due, 0 = end of period while 1 = beginning.	= FV(InterestRate/12, PaymentPeriods, PaymentAmount[, LoanAmount - Optional [, Type - Optional]])
PV (Present Value)	Calculates loan or investment present value based on constant interest rate. Principal&Interest: Future value or balance after making last payment. Type: When payments due, 0 = end of period while 1 = beginning.	= PV(InterestRate/12, PaymentPeriods, PaymentAmount[, Principal&Interest - Optional [, Type - Optional]])
IPMT (Payment)	Returns investment interest payment for a given period base on periodic, constant payments and interest rate. Period: When to calculate interest payment, must be between 1 & PP. LoanAmount: Present value or lump-sum amount. Principal&Interest: Future value or balance after making last payment. Type: When payments due, 0 = end of period while 1 = beginning.	= IPMT(InterestRate/12, Period, PaymentPeriods, LoanAmount[, Principal&Interest - Optional [, Type - Optional]])
PMT (Payment)	Calculates loan payment based on constant payments & interest rate. LoanAmount: Present value or lump-sum amount. Principal&Interest: Future value or balance after making last payment. Type: When payments due, 0 = end of period while 1 = beginning.	= PMT(InterestRate/12, PaymentPeriods, LoanAmount[, Principal&Interest - Optional [, Type - Optional]])
RATE	Calculates loan or investment present value based on constant interest rate. LoanAmount: Total amount that future payments is worth now. Principal&Interest: Future value or balance after making last payment. Type: When payments due, 0 = end of period while 1 = beginning. Guess: Rate it will be. If omited, it will be 10%.	= RATE(PaymentPeriods, PaymentAmount, LoanAmount[, Principal&Interest - Optional [, Type - Optional[, Guess - Optional]]])

Text
Format text values within table

Function	Description	Syntax
COMBINEVALUES	Joins two or more text strings into one text string.	= COMBINEVALUES(Delimiter, Expression1,...)
CONCATENATE	Joins two text strings into one text string.	= CONCATENATE(Text1, Text2)
CONCATENATEX	Concatenates result of an expression evaluated for each row.	= CONCATENATEX(Table, Expression, Delimiter)
FIND	Returns starting position of one text string within another text string. StartPosition: Charaqcter start to search. If omited, default = 1. NotFoundValue: 0 = not found, -1 = found, BLANK() = blank value	= FIND("FindText", [WithinTextColumn], [StartPosition] - Optional, [NotFoundValue])
FIXED	Rounds number to specified decimals & returns result as text. Decimals: if omit, it will be 2. NoCommas: if 0 or omited, it display commas.	= FIXED(NumberColumn, Decimals - Optional, NoCommas - Optional)
FORMAT	Converts a value to text according to the specified format.	= FORMAT(Value, Format, [LocalName])
LEFT / RIGHT	Extracts specfied number of characters from left/right of a text.	= LEFT/RIGHT("Text", NumberOfCharacters)
LEN	Returns the number of characters in a text string.	= LEN("Text"/[Column])
REPT	Repeats text based on given number of times.	= REPT("Text", NumberOfTimes)
REPLACE	Replace part of text string based on number of characters specify. StartPosition: Starts from 1, not 0. NumberofCharacters: If blank, entire NewText is inserted.	= REPLACE("OldText"/[Column], StartPosition, NumberofCharacters, "NewText")
SUBSTITUTE	Replaces existing text with new text in a text string. InstanceNumber: If omited, every instance of OldText is replaced.	= SUBSTITUTE("Text"/[Column], "OldText", "NewText", [InstanceNumber] - Optional)
TRIM	Removes all spaces except single spaces between words.	= TRIM("Text"/[Column])
UPPER / LOWER	Converts all letters in a text string to upper/lower case.	= UPPER/LOWER("Text"/[Column])

Date & Time
Create calculations based on dates and time

Function	Description	Syntax
CALENDAR	Returns table with single column, "Date", with contiguous set of dates	= CALENDAR(StartDate, EndDate)
DATE	Returns specified date in datetime format.	= DATE(year, month, day)
DATEIFF	Returns number of interval boundaries between 2 dates.	= DATEIFF(Date1, Date2, Interval)
DATEVALUE	Convert a date in text format to a date in datetime format.	= DATEVALUE("DateText")
DAY	Returns day of the month, a number from 1 to 31.	= DAY([Column])
MONTH	Returns month as a number from 1 (Jan) to 12 (Dec).	= MONTH([Column])
YEAR	Returns year of a date as 4 digit integer in the range of 1900-9999.	= YEAR([Column])
EDATE	Calculates dates based on months.	= EDATE([Column], NumberOfMonths)
EOMONTH	Returns date in datetime format of last day of month, before or after specified number of months.	= EOMONTH(StartDate, Month)
NETWORKDAYS	Returns number of whole workdays between 2 dates. Weekend: 1 = Default is 1 and 2, represent Sat & Sun. Holidays: PH table.	= NETWORKDAYS(StartDate, EndDate, [Weekend], [Holidays])
NOW	Displays current date and time in datetime format.	= NOW()
QUARTER	Returns the quarter as a number.	= QUARTER(Date)
TIME	Converts hours, minutes and seconds to time in datetime format.	= TIME(hour, minute, second)
WEEKDAY	Returns a number from 1 to 7 identifying day of the week of a date. ReturnType: 1 = week begins on Sunday, 2 = week begins on Monday	= WEEKDAY(Date, [ReturnType])
WEEKNUM	Returns week number for the given date and year. ReturnType: Default is 1	= WEEKNUM(Date, [ReturnType] - Optional)

Time Intelligence
Helps to analyze data over time periods to identify trends, track performance & make decisions

Function	Description	Syntax
DATEADD	Returns table with dates shifted by specified number of intervals from current context. Internal: year, quarter, month, day	= DATEADD([Column], NumberOfIntervals, Interval)
DATESBETWEEN	Returns table with dates between start and end date, inclusive.	= DATESBETWEEN([Column], StartDate, EndDate)
DATESINPERIOD	Generates table with dates from a specified start date for a specified number and type of intervals (DAY, MONTH, QUARTER or YEAR).	= DATESINPERIOD([Column], StartDate, NumberOfIntervals, Interval)
ENDOFMONTH	Returns last date of the month.	= ENDOFMONTH([Column])
ENDOFYEAR	Returns last date of the year. YearEndDate default is Dec 31.	= ENDOFYEAR([Column], [YearEndDate])
PARALLELPERIOD	Returns table with dates parallel to specified column, shifted by number of intervals (YEAR, QUARTER or MONTH).	= PARALLELPERIOD([Column], NumberOfIntervals, Interval)
SAMEPERIODLASTYEAR	Returns table with dates representing same period 1 year back from specified column in current context.	= SAMEPERIODLASTYEAR([Column])
PREVIOUSMONTH	Calculates first date to 1 day before end of previous month.	= PREVIOUSMONTH([Column])
PREVIOUSQUARTER	Calculates first date to 1 day before end of previous quarter.	= PREVIOUSQUARTER([Column])
PREVIOUSYEAR	Calculates period from last date to same date in previous year.	= PREVIOUSYEAR([Column], [YearEndDate])
TOTALMTD	Calculates total month-to-date across all dates.	= TOTALMTD(Expression, [Column], [Filter])
TOTALQTD	Calculates total quarter-to-date across all dates.	= TOTALQTD(Expression, [Column], [Filter])
TOTALYTD	Calculates total year-to-date across all dates. YearEndDate: Default is December 31	= TOTALYTD(Expression, [Column], [Filter], [YearEndDate])

Information
Provide insights into data and its contact without performing calculations

Function	Description	Syntax
COLUMNSTATISTICS	Returns a statistics for all column in all table in the model.	= COLUMNSTATISTICS()
HASONEVALUE	Returns TRUE if context has one distinct value, otherwise, FALSE.	= HASONEVALUE([Column])
ISBLANK	Checks blank value and returns TRUE or FALSE.	= ISBLANK(Value)
ISCROSSFILTERED	Returns TRUE when specified table/column is cross-filtered.	= ISCROSSFILTERED(Table or [Column])
ISERROR	Checks error value and returns TRUE or FALSE.	= ISERROR(Value)
ISFILTERED	Returns TRUE when specified table/column is directly filtered.	= ISFILTERED(Table or [Column])
ISINSCOPE	Returns true when specified column is the level of hierarchy.	= ISINSCOPE([Column])
SELECTEDMEASURENAME	Used by calculation items to reference measure in context by name.	= SELECTEDMEASURENAME()

Table Manipulation
Transform, reshapre and manipulate data to create dynamic insights

Function	Description	Syntax
ADDCOLUMNS	Adds calculated columns to given table or table expression.	= ADDCOLUMNS(Table, Name1, Expression1,...)
CROSSJOIN	Returns a table with Cartesian product of rows from specified tables.	= CROSSJOIN(Table1, Table2,...)
DISTINCT	Returns one-column table with distinct values from specified table.	= DISTINCT([Column])
EXCEPT	Returns rows of first table in expression which do not appear in second table.	= EXCEPT(LeftTable, RightTable)
GENERATESERIES	Returns single-column table with values from an arithmetic series.	= GENERATESERIES(StartValue, EndValue, [IncrementValue] - Optional)
GROUPBY	Similar to SUMMARIZE but does not implicitly CALCULATE for extension columns	= GROUPBY(Table, [Column1], [Column2], ... "Name1", [Expression1])
INTERSECT	Returns intersection of two tables, keeping duplicates rows.	= INTERSECT(LeftTable, RightTable)
NATURALINNERJOIN	Join two tables using inner join.	= NATURALINNERJOIN(LeftTable, RightTable)
NATURALLEFTOUTERJOIN	Performs a join on Left Table with Right Table.	= NATURALLEFTOUTERJOIN(LTable, RTable)
ROW	Returns a table with a single row containing values resulting from expressions given to each column.	= ROW (Name1, Expression1, ...)
SELECTCOLUMNS	Returns a table with selected columns from the table.	= SELECTCOLUMNS(Table, Name1, Expression1, Name2, Expression2, ...)
SUMMARIZE	Returns summary table for requested totals over a set of groups.	= SUMMARIZE(Table, [Column1], [Column2], ... "Name1", [Expression1])
TREATAS	Applies result of table expression as filters to columns from an unrelated table.	= TREATAS(Expression, [Column])
UNION	Combine tables vertically, preserving same rows.	= UNION(Table1, Table2)
VALUES	Returns one-column table with unique values from table/column	= VALUES(Table or [Column])

Filter
Isolate and analyze specific subsets of data based on conditions

Function	Description	Syntax
ALL	Clears filters and returns all data.	= ALL(Table, [Column1], [Column2])
ALLEXCEPT	Removes all filters in table except filters applied to specified column(s)	= ALLEXCEPT(Table, [Column1], [Column2])
ALLNOBLANKROW	Returns unique non-blank values from a related table.	= ALLNOBLANKROW(Table, [Column1], ...)
ALLSELECTED	Removes context filters, preserving other filters in current query.	= ALLSELECTED(Table, [Column1], [Column2])
CALCULATE	Evaluates an expression in a modified filter context.	= CALCULATE(Expression, Filter1, Filter2, ...)
FILTER	Returns table representing subset of another table or expression.	= FILTER(Table, FilterExpression)
LOOKUPVALUE	Retrieves value for a row metting specified criteria.	= LOOKUPVALUE(ResultColumn, SearchColumn, SearchValue, [AlternateResult])
ORDER BY	Sorts tables/columns based on specified expressions.	= ORDERBY(Table, Expression1, Order1, ...)
RANGE	Creating sequences of numbers that can be used in calculations.	= RANGE(StartValue, EndValue, [Step])
RANK	Ranks values within a set of values by using ASC or DESC.	= RANK(Table, Expression, [Order])
REMOVEFILTERS	Clear filters from the specified tables or columns.	= REMOVEFILTERS(Table,[Column1], ...)
SELECTEDVALUE	Returns single value if filtered, else returns alternate result.	= SELECTEDVALUE([Column1], [AlternateResult])

Logical
Evaluate expressions and return vlaues in TRUE or FALSE

Function	Description	Syntax
COALESCE	Returns first non-null value from list of expressions when dealing missing or null values in data.	= COALESCE(Expression1, Expression2, ...)
AND	Checks if both arguments are true.	= AND(Logic1, Logic2)
OR	Checks if either argument is true	= OR(Logic1, Logic2)
NOT	Changes FALSE to TRUE, or TRUE to FALSE.	= NOT(Logic)
IF	Checks condition and returns 1 value when it's TRUE, otherwise returns second value.	= IF(LogicalTest, ResultIfTrue, [ResultIfFalse])
IFERROR	Evaluates expression and returns specficied value if returns an error.	= IFERROR(Value, ValueIfError)
SWITCH	Returns result corresponding to first value that matches expression. If no value matches, returns default result.	= SWITCH(Expression, Value1, Result1, Value2, Result2, ... [Else])

Parent & Child
Illustrate parent/child hierarchies

Function	Description	Syntax
PATH	Retrieves complete hierarchy of an item, from root to current level.	= PATH([IDColumn], [ParentColum])
PATHCONTAINS	Checks if a specific item exists within a given hierarchy.	= PATHCONTAINS(Path, [Item])
PATHITEM	Extracts a specific level or item from a hierarchy path.	= PATHITEMS(Path, [Position], [Type])
PATHITEMREVERSE	Extracts specific level/item from hierarchy path from bottom up.	= PATHITEMREVERSE(Path, [Position], [Type])
PATHLENGTH	Determines the depth or level of an item in a hierarchy.	= PATHLENGTH(Path)

Relationship
Manage and manipulate relationship between tables in data models

Function	Description	Syntax
CROSSFILTER	Controls the direction of cross-filtering between two related columns. Mode: ONEWAY, BOTH, NONE)	= CROSSFILTER(Table1[Column], Table2[Column], [Mode])
RELATED	Retrieves related value from another table based on defined relationship.	= RELATED([Column])
RELATEDTABLE	Evaluates a table expression in a context modified by the given filters.	= RELATEDTABLE([Table])
USERELATIONSHIP	Specifies the relationship to be used in a specific calculation.	= USERELATIONSHIP(Table1[Column], Table2[Column])