

[Power BI Calculated Columns and Measures] [cheatsheet]

Basic Arithmetic

- Addition: `Total = [Revenue] + [Costs]`
- Subtraction: `Profit = [Revenue] - [Costs]`
- Multiplication: `Total Cost = [Quantity] * [Unit Price]`
- Division: `Profit Margin = DIVIDE([Profit], [Revenue], 0)`
- Exponentiation: `Squared Value = [Value] ^ 2`

Aggregation Functions

- Sum: `Total Sales = SUM(Sales[Amount])`
- Average: `Avg Sales = AVERAGE(Sales[Amount])`
- Count: `Customer Count = COUNT(Customers[CustomerID])`
- Distinct Count: `Unique Products = DISTINCTCOUNT(Products[ProductID])`
- Min: `Lowest Price = MIN(Products[Price])`
- Max: `Highest Price = MAX(Products[Price])`

Text Functions

- Concatenate: `Full Name = Employees[FirstName] & " " & Employees[LastName]`
- Left: `First Letter = LEFT(Products[ProductName], 1)`
- Right: `Last 3 Chars = RIGHT(Orders[OrderID], 3)`
- Len: `Name Length = LEN(Customers[CustomerName])`
- Upper: `Upper Case Name = UPPER(Products[ProductName])`
- Lower: `Lower Case Name = LOWER(Products[ProductName])`
- Proper: `Proper Case Name = PROPER(Customers[CustomerName])`
- Trim: `Trimmed Name = TRIM(Products[ProductName])`
- Substitute: `Fixed Phone = SUBSTITUTE(Customers[Phone], "-", "")`

Date and Time Functions

- Year: `Order Year = YEAR(Orders[OrderDate])`
- Month: `Order Month = MONTH(Orders[OrderDate])`
- Day: `Order Day = DAY(Orders[OrderDate])`
- Quarter: `Order Quarter = QUARTER(Orders[OrderDate])`
- Week Number: `Order Week = WEEKNUM(Orders[OrderDate])`
- Week Day: `Order Weekday = WEEKDAY(Orders[OrderDate])`

- Date Difference: Days Since Order = DATEDIFF(Orders[OrderDate], TODAY(), DAY)
- End of Month: Month End = EOMONTH(Orders[OrderDate], 0)
- Date Add: Due Date = DATEADD(Orders[OrderDate], 30, DAY)
- Is After: Is Recent = Orders[OrderDate] > DATE(2023, 1, 1)

Conditional Statements

- Simple IF: Status = IF(Products[Stock] > 0, "In Stock", "Out of Stock")
- Nested IF: Price Category = IF([Price] < 10, "Low", IF([Price] < 50, "Medium", "High"))
- SWITCH: Day Type = SWITCH(WEEKDAY(Orders[OrderDate]), 1, "Weekend", 7, "Weekend", "Weekday")
- AND: Is Valid = AND(Products[Price] > 0, Products[Stock] > 0)
- OR: Needs Attention = OR(Products[Price] = 0, Products[Stock] = 0)
- NOT: Is Not Valid = NOT([Is Valid])

Time Intelligence Functions

- Year-to-Date: YTD Sales = TOTALYTD(SUM(Sales[Amount]), 'Date'[Date])
- Quarter-to-Date: QTD Sales = TOTALQTD(SUM(Sales[Amount]), 'Date'[Date])
- Month-to-Date: MTD Sales = TOTALMTD(SUM(Sales[Amount]), 'Date'[Date])
- Previous Year: PY Sales = CALCULATE([Total Sales], SAMEPERIODLASTYEAR('Date'[Date]))
- Year-over-Year Growth: YoY Growth = DIVIDE([Total Sales] - [PY Sales], [PY Sales], 0)
- Moving Annual Total: MAT Sales = CALCULATE([Total Sales], DATESINPERIOD('Date'[Date], MAX('Date'[Date]), -1, YEAR))
- Previous Month: PM Sales = CALCULATE([Total Sales], PREVIOUSMONTH('Date'[Date]))
- Month-over-Month Growth: MoM Growth = DIVIDE([Total Sales] - [PM Sales], [PM Sales], 0)
- Rolling 3-Month Average: 3M Avg Sales = AVERAGEX(DATESINPERIOD('Date'[Date], MAX('Date'[Date]), -2, MONTH), [Total Sales])

Statistical Functions

- Median: Median Price = MEDIAN(Products[Price])
- Mode: Most Common Price = MODE(Products[Price])

- Percentile: 90th Percentile Price = PERCENTILE.EXC(Products[Price], 0.9)
- Standard Deviation: Price StDev = STDEV.S(Products[Price])
- Variance: Price Variance = VAR.S(Products[Price])
- Correlation: Price-Sales Correlation =
CALCULATE(CORRELATION(Products[Price], Sales[Amount]))

Ranking Functions

- Rank: Sales Rank = RANKX(ALL(Products), [Total Sales],, DESC)
- Dense Rank: Dense Sales Rank = RANKX(ALL(Products), [Total Sales],, DESC, Dense)
- Percent Rank: Sales Percentile = PERCENTRANK.EXC(ALL(Products[Total Sales]), [Total Sales])

Filter Functions

- All: Overall Avg Sales = AVERAGE(ALL(Sales[Amount]))
- AllExcept: Category Avg Sales = AVERAGE(ALLEXCEPT(Sales, Sales[Category]))
- Filter: High Value Sales = CALCULATE([Total Sales], FILTER(Sales, Sales[Amount] > 1000))
- TopN: Top 5 Products = CALCULATE([Total Sales], TOPN(5, ALL(Products), [Total Sales], DESC))

Iterator Functions

- SUMX: Total Revenue = SUMX(Sales, Sales[Quantity] * Sales[UnitPrice])
- AVERAGEX: Avg Order Value = AVERAGEX(Orders, RELATED(Sales[Amount]))
- MAXX: Highest Sales = MAXX(Sales, Sales[Amount])
- MINX: Lowest Sales = MINX(Sales, Sales[Amount])

Parent-Child Functions

- PATH: Employee Path = PATH(Employees[EmployeeID], Employees[ManagerID])
- PATHITEM: Manager Level 2 = PATHITEM([Employee Path], 2)
- PATHITEMREVERSE: Direct Manager = PATHITEMREVERSE([Employee Path], 1)
- PATHCONTAINS: Has Manager = PATHCONTAINS([Employee Path], Employees[ManagerID])

Error Handling

- IFERROR: Safe Division = IFERROR(DIVIDE([Revenue], [Costs]), 0)
- ISBLANK: Has Sales = NOT(ISBLANK([Total Sales]))
- ISEEMPTY: Has Orders = NOT(ISEEMPTY(Orders))
- HASONEVALUE: Single Product Selected = HASONEVALUE(Products[ProductName])

Text Analysis

- FIND: Contains A = IF(FIND("a", LOWER(Products[ProductName])), 1, -1) > 0, "Yes", "No")
- SEARCH: Position of Space = SEARCH(" ", Customers[CustomerName], 1, -1)
- UNICHAR: Star Rating = REPT(UNICHAR(9733), Products[Rating])
- UNICODE: First Char Code = UNICODE(LEFT(Products[ProductName], 1))

Advanced Calculations

- Running Total: Running Total Sales = CALCULATE(SUM(Sales[Amount]), FILTER(ALL('Date'), 'Date'[Date] <= MAX('Date'[Date])))
- Cumulative Percentage: Cumulative % = DIVIDE(CALCULATE([Total Sales], FILTER(ALL('Date'), 'Date'[Date] <= MAX('Date'[Date]))), [Overall Total Sales])
- Pareto Analysis: Pareto = DIVIDE(CALCULATE([Total Sales], FILTER(ALL(Products), Products[Total Sales] >= EARLIER(Products[Total Sales]))), [Overall Total Sales])
- Market Share: Market Share % = DIVIDE([Total Sales], CALCULATE([Total Sales], ALL(Products)))
- Contribution to Parent: Category Contribution % = DIVIDE([Total Sales], CALCULATE([Total Sales], ALL(Products[Category])))
- Year-over-Year Comparison: YoY Comparison = SWITCH(TRUE(), [YoY Growth] > 0.1, "High Growth", [YoY Growth] > 0, "Growth", [YoY Growth] = 0, "Flat", "Decline")

Window Functions

- Moving Average: 3-Day Moving Avg = AVERAGEX(DATESINPERIOD('Date'[Date], LASTDATE('Date'[Date]), -2, DAY), [Daily Sales])
- Cumulative Sum: Cumulative Sales = CALCULATE(SUM(Sales[Amount]), FILTER(ALL('Date'), 'Date'[Date] <= MAX('Date'[Date])))
- Percent of Running Total: % of Running Total = DIVIDE([Total Sales], [Cumulative Sales])
- Difference from Previous: Sales Diff = [Total Sales] - CALCULATE([Total Sales], PREVIOUSDAY('Date'[Date]))

- Rolling Year Comparison: $\text{Rolling Year Diff} = [\text{MAT Sales}] - \text{CALCULATE}([\text{MAT Sales}], \text{DATEADD}('Date'[\text{Date}], -1, \text{YEAR}))$

Advanced Time Intelligence

- Custom Year-to-Date: $\text{Custom YTD} = \text{CALCULATE}([\text{Total Sales}], \text{DATESYTD}('Date'[\text{Date}], "6-30"))$
- Fiscal Year Calculations: $\text{Fiscal YTD} = \text{CALCULATE}([\text{Total Sales}], \text{DATESYTD}('Date'[\text{Date}], "7-1"))$
- Semi-Annual Periods: $\text{Half Year} = \text{IF}(\text{MONTH}('Date'[\text{Date}]) \leq 6, "H1", "H2")$
- Week-over-Week Comparison: $\text{WoW Change} = [\text{Total Sales}] - \text{CALCULATE}([\text{Total Sales}], \text{DATEADD}('Date'[\text{Date}], -7, \text{DAY}))$
- Last N Periods: $\text{Last 3 Months Sales} = \text{CALCULATE}([\text{Total Sales}], \text{DATESINPERIOD}('Date'[\text{Date}], \text{MAX}('Date'[\text{Date}]), -2, \text{MONTH}))$

Data Categorization

- Sales Bracket: $\text{Sales Bracket} = \text{SWITCH}(\text{TRUE}(), [\text{Total Sales}] \geq 1000000, "Large", [\text{Total Sales}] \geq 100000, "Medium", "Small")$
- Age Group: $\text{Age Group} = \text{SWITCH}(\text{TRUE}(), \text{Customers}[\text{Age}] \geq 60, "Senior", \text{Customers}[\text{Age}] \geq 40, "Middle-aged", \text{Customers}[\text{Age}] \geq 20, "Young Adult", "Youth")$
- Product Performance: $\text{Performance Category} = \text{SWITCH}(\text{TRUE}(), [\text{YoY Growth}] > 0.2, "High Performer", [\text{YoY Growth}] > 0, "Growing", [\text{YoY Growth}] > -0.1, "Stable", "Declining")$

Data Quality Checks

- Completeness Check: $\text{Is Complete} = \text{AND}(\text{NOT}(\text{ISBLANK}([\text{Sales}])), \text{NOT}(\text{ISBLANK}([\text{Costs}])))$
- Range Check: $\text{Is Valid Price} = \text{AND}([\text{Price}] \geq 0, [\text{Price}] \leq 1000)$
- Format Check: $\text{Is Valid Email} = \text{SEARCH}("@", \text{Customers}[\text{Email}], 1, -1) > 0$
- Consistency Check: $\text{Is Consistent} = [\text{Sales}] \geq [\text{Costs}]$

KPI Calculations

- Gross Profit Margin: $\text{GPM} = \text{DIVIDE}([\text{Gross Profit}], [\text{Total Sales}], 0)$
- Customer Lifetime Value: $\text{CLV} = \text{DIVIDE}([\text{Total Sales}], \text{DISTINCTCOUNT}(\text{Customers}[\text{CustomerID}]))$
- Customer Acquisition Cost: $\text{CAC} = \text{DIVIDE}([\text{Marketing Spend}], [\text{New Customers}])$

- Return on Investment: $ROI = \text{DIVIDE}([\text{Net Profit}], [\text{Total Investment}], 0)$
- Debt-to-Equity Ratio: $D/E \text{ Ratio} = \text{DIVIDE}([\text{Total Liabilities}], [\text{Total Equity}], 0)$

Advanced Time Intelligence

- Sliding Window: $6M \text{ Sliding Window} = \text{CALCULATE}([\text{Total Sales}], \text{DATESINPERIOD}('Date'[\text{Date}], \text{MAX}('Date'[\text{Date}]), -5, \text{MONTH}))$
- Parallel Period: $\text{Parallel Period Sales} = \text{CALCULATE}([\text{Total Sales}], \text{PARALLELPERIOD}('Date'[\text{Date}], -1, \text{YEAR}))$
- Custom Period Comparison: $Q4 \text{ vs } Q2 = \text{DIVIDE}(\text{CALCULATE}([\text{Total Sales}], 'Date'[\text{QuarterNo}] = 4), \text{CALCULATE}([\text{Total Sales}], 'Date'[\text{QuarterNo}] = 2)) - 1$
- Rolling Forecast: $12M \text{ Forecast} = \text{CALCULATE}([\text{Total Sales}], \text{DATEADD}('Date'[\text{Date}], 1, \text{YEAR})) * (1 + [\text{YoY Growth}])$

Advanced Statistical Measures

- Z-Score: $\text{Sales Z-Score} = \text{DIVIDE}([\text{Total Sales}] - \text{AVERAGE}([\text{Total Sales}]), \text{STDEV.S}([\text{Total Sales}]))$
- Confidence Interval: $CI \text{ Upper} = \text{AVERAGE}([\text{Sales}]) + 1.96 * \text{STDEV.S}([\text{Sales}]) / \text{SQRT}(\text{COUNT}([\text{Sales}]))$
- Moving Correlation: $12M \text{ Rolling Correlation} = \text{CORRELATIONX}(\text{DATESINPERIOD}('Date'[\text{Date}], \text{MAX}('Date'[\text{Date}]), -11, \text{MONTH}), [\text{Sales}], [\text{Marketing Spend}])$
- Exponential Moving Average: $EMA = \text{CALCULATE}(0.2 * [\text{Sales}] + 0.8 * \text{CALCULATE}([EMA], \text{PREVIOUSDAY}('Date'[\text{Date}])))$

Advanced Ranking and Segmentation

- Percentile Bucketing: $\text{Sales Percentile Bucket} = \text{SWITCH}(\text{TRUE}(), [\text{Sales Percentile}] \leq 0.2, \text{"Bottom 20\%"}, [\text{Sales Percentile}] \leq 0.4, \text{"20-40\%"}, [\text{Sales Percentile}] \leq 0.6, \text{"40-60\%"}, [\text{Sales Percentile}] \leq 0.8, \text{"60-80\%"}, \text{"Top 20\%"})$
- Relative Ranking: $\text{Relative Sales Rank} = \text{RANKX}(\text{ALLSELECTED}(\text{Products}), [\text{Total Sales}], , \text{DESC})$
- ABC Analysis: $\text{ABC Category} = \text{SWITCH}(\text{TRUE}(), [\text{Cumulative Sales \%}] \leq 0.7, \text{"A"}, [\text{Cumulative Sales \%}] \leq 0.9, \text{"B"}, \text{"C"})$

Inter-table Calculations

- Average Basket Size: $\text{Avg Basket Size} = \text{DIVIDE}([\text{Total Sales}], \text{DIVIDE}(\text{COUNTROWS}(\text{Sales}), \text{DISTINCTCOUNT}(\text{Sales}[\text{OrderID}])))$
- Product Penetration: $\text{Product Penetration \%} = \text{DIVIDE}(\text{CALCULATE}(\text{DISTINCTCOUNT}(\text{Sales}[\text{CustomerID}]), \text{FILTER}(\text{Sales}, \text{Sales}[\text{ProductID}] = \text{EARLIER}(\text{Products}[\text{ProductID}])), \text{DISTINCTCOUNT}(\text{Sales}[\text{CustomerID}]))$
- Customer Segment Performance: $\text{Segment Performance} = \text{DIVIDE}(\text{CALCULATE}([\text{Total Sales}], \text{RELATEDTABLE}(\text{CustomerSegment})), \text{CALCULATE}([\text{Total Sales}], \text{ALL}(\text{CustomerSegment})))$

Advanced Conditional Formatting

- Dynamic Threshold: $\text{Performance Indicator} = \text{SWITCH}(\text{TRUE}(), [\text{Sales}] > [\text{Target}] * 1.1, 3, [\text{Sales}] > [\text{Target}], 2, [\text{Sales}] > [\text{Target}] * 0.9, 1, 0)$
- Gradient Scale: $\text{Temperature} = 1 - ([\text{Value}] - [\text{Min Value}]) / ([\text{Max Value}] - [\text{Min Value}])$

Complex Business Rules

- Tiered Discounting: $\text{Discount \%} = \text{SWITCH}(\text{TRUE}(), [\text{Order Quantity}] \geq 100, 0.15, [\text{Order Quantity}] \geq 50, 0.1, [\text{Order Quantity}] \geq 20, 0.05, 0)$
- Dynamic Pricing: $\text{Adjusted Price} = [\text{Base Price}] * (1 - [\text{Discount \%}]) * \text{IIF}([\text{Stock}] < 10, 1.1, 1)$
- Loyalty Points: $\text{Points Earned} = \text{ROUNDDOWN}([\text{Total Sales}] * \text{IIF}(\text{Customers}[\text{Tier}] = \text{"Gold"}, 0.05, \text{IIF}(\text{Customers}[\text{Tier}] = \text{"Silver"}, 0.03, 0.01)), 0)$

Text Analytics

- Sentiment Score: $\text{Sentiment} = \text{SWITCH}(\text{TRUE}(), \text{CONTAINSSTRING}([\text{Review}], \text{"excellent"}) || \text{CONTAINSSTRING}([\text{Review}], \text{"great"}), 2, \text{CONTAINSSTRING}([\text{Review}], \text{"good"}) || \text{CONTAINSSTRING}([\text{Review}], \text{"nice"}), 1, \text{CONTAINSSTRING}([\text{Review}], \text{"poor"}) || \text{CONTAINSSTRING}([\text{Review}], \text{"bad"}), -1, 0)$
- Word Count: $\text{Word Count} = \text{LEN}([\text{Text}]) - \text{LEN}(\text{SUBSTITUTE}([\text{Text}], " ", "")) + 1$

Financial Calculations

- Compound Annual Growth Rate: $CAGR = \text{POWER}(\text{DIVIDE}(\text{LASTNONBLANK}([Value], [Year]), \text{FIRSTNONBLANK}([Value], [Year])), 1 / ([Last Year] - [First Year] + 1)) - 1$
- Days Sales Outstanding: $DSO = \text{DIVIDE}([Accounts\ Receivable], [Total\ Sales]) * 365$
- Working Capital: $\text{Working Capital} = [Current\ Assets] - [Current\ Liabilities]$
- Debt Service Coverage Ratio: $DSCR = \text{DIVIDE}([EBITDA], [Total\ Debt\ Service])$

Forecasting and Predictive Measures

- Simple Linear Regression: $\text{Sales Forecast} = \text{AVERAGEX}(\text{Sales}, \text{Sales}[Amount]) + (\text{MAX}('Date'[DateKey]) - \text{AVERAGE}('Date'[DateKey])) * \text{DIVIDE}(\text{SUMX}(\text{Sales}, (\text{Sales}[Amount] - \text{AVERAGE}(\text{Sales}[Amount])) * ('Date'[DateKey] - \text{AVERAGE}('Date'[DateKey]))), \text{SUMX}(\text{Sales}, \text{POWER}('Date'[DateKey] - \text{AVERAGE}('Date'[DateKey]), 2)))$
- Seasonal Adjustment: $\text{Seasonally Adjusted Sales} = \text{DIVIDE}([Total\ Sales], \text{AVERAGE}(\text{CALCULATETABLE}(\text{VALUES}('Date'[MonthNo]))))$
- Holt-Winters Forecasting: $\text{HW Forecast} = [Level] + [Trend] * [Period] + [Seasonal\ Factor]$

Advanced Error Handling and Data Quality

- Multi-condition Error Check: $\text{Data Quality Flag} = \text{IF}(\text{OR}(\text{ISBLANK}([Sales]), [Sales] < 0, [Sales] > 1000000), \text{"Check Required"}, \text{"OK"})$
- Fuzzy Matching: $\text{Potential Duplicate} = \text{IF}(\text{MINX}(\text{FILTER}(\text{Customers}, \text{Customers}[CustomerID] <> \text{EARLIER}(\text{Customers}[CustomerID])), \text{SQRTSUMX}(\text{Customers}, \text{POWER}(\text{UNICODE}(\text{MID}(\text{Customers}[Name], [Char], 1)) - \text{UNICODE}(\text{MID}(\text{EARLIER}(\text{Customers}[Name]), [Char], 1)), 2))) < 5, \text{"Yes"}, \text{"No"})$

Dynamic Measures

- Measure Switch: $\text{Selected Measure} = \text{SWITCH}(\text{SELECTEDVALUE}(\text{MeasureSelector}[Measure]), \text{"Sales"}, [Total\ Sales], \text{"Profit"}, [Total\ Profit], \text{"Units"}, [Total\ Units], \text{BLANK}())$
- Dynamic Time Calculation: $\text{Dynamic Time Calc} = \text{CALCULATE}([Total\ Sales], \text{DATESINPERIOD}('Date'[Date], \text{MAX}('Date'[Date]), -[Selected\ Periods], [Selected\ Time\ Grain]))$

Geographic Calculations

- **Distance Calculation:** $\text{Distance} = \text{ACOS}(\text{SIN}(\text{RADIANS}(\text{Stores}[\text{Latitude}])) * \text{SIN}(\text{RADIANS}(\text{Customers}[\text{Latitude}])) + \text{COS}(\text{RADIANS}(\text{Stores}[\text{Latitude}])) * \text{COS}(\text{RADIANS}(\text{Customers}[\text{Latitude}])) * \text{COS}(\text{RADIANS}(\text{Stores}[\text{Longitude}] - \text{Customers}[\text{Longitude}])) * 6371$
- **Geohash:** $\text{Geohash} = \text{CONCATENATE}(\text{REPT}(\text{UNICHAR}(97 + \text{ROUNDDOWN}([\text{Latitude}] + 90) / 180 * 26, 0)), 3), \text{REPT}(\text{UNICHAR}(97 + \text{ROUNDDOWN}([\text{Longitude}] + 180) / 360 * 26, 0)), 3))$

Anomaly Detection

- **Outlier Flag:** $\text{Is Outlier} = \text{IF}(\text{ABS}([\text{Value}] - \text{AVERAGE}([\text{Value}])) / \text{STDEV.S}([\text{Value}])) > 3, \text{"Yes"}, \text{"No"})$
- **Anomaly Score:** $\text{Anomaly Score} = \text{ABS}([\text{Value}] - \text{AVERAGE}([\text{Value}])) / \text{STDEV.S}([\text{Value}]))$

Advanced Set Analysis

- **Market Basket Analysis:** $\text{Co-occurrence} = \text{DIVIDE}(\text{CALCULATE}(\text{DISTINCTCOUNT}(\text{Sales}[\text{OrderID}]), \text{FILTER}(\text{ALL}(\text{Products}), \text{Products}[\text{ProductID}] <> \text{EARLIER}(\text{Products}[\text{ProductID}])), \text{DISTINCTCOUNT}(\text{Sales}[\text{OrderID}]))$
- **Customer Overlap:** $\text{Shared Customers} = \text{DIVIDE}(\text{COUNTROWS}(\text{INTERSECT}(\text{CALCULATETABLE}(\text{VALUES}(\text{Customers}[\text{CustomerID}]), \text{Products}[\text{Category}] = \text{"A"}), \text{CALCULATETABLE}(\text{VALUES}(\text{Customers}[\text{CustomerID}]), \text{Products}[\text{Category}] = \text{"B"}))), \text{COUNTROWS}(\text{UNION}(\text{CALCULATETABLE}(\text{VALUES}(\text{Customers}[\text{CustomerID}]), \text{Products}[\text{Category}] = \text{"A"}), \text{CALCULATETABLE}(\text{VALUES}(\text{Customers}[\text{CustomerID}]), \text{Products}[\text{Category}] = \text{"B"}))))$