==
CATEGORY 4: DATE/TIME FUNCTIONS (10 Functions)
==

3️⃣1️⃣ TODAY() - CURRENT DATE

SYNTAX:
  TODAY()

PARAMETERS:
  • None

EXAMPLE FROM YOUR DATASET:

Days Since Order (Calculated Column):
  Days Old = INT(TODAY() - SalesData[OrderDate])

  Result: Calculates days between order date and today
    • Order from 2025-12-01 = 6 days old
    • Order from 2023-01-01 = 1036 days old

Days Since Customer Joined (Calculated Column):
  Customer Age Days = INT(TODAY() - CustomerMaster[JoinDate])

REAL USAGE:
  • Calculate age of orders
  • Days since registration
  • Dynamic time calculations

3️⃣2️⃣ DATE() - CREATE SPECIFIC DATE

SYNTAX:
  DATE(<year>, <month>, <day>)

PARAMETERS:
  • <year>: Year value (e.g., 2023)
  • <month>: Month value (1-12)
  • <day>: Day value (1-31)

EXAMPLE FROM YOUR DATASET:

Reference Date for Comparison (Calculated Measure):
  Reference Date = DATE(2025, 12, 31)

  Result: ₹12/31/2025

Year End 2025:
  Year End = DATE(2025, 12, 31)

REAL USAGE:
  • Create fixed dates
  • Set benchmarks
  • Period boundaries

3️⃣3️⃣ DATEDIFF() - DAYS BETWEEN DATES

SYNTAX:

```
  DATEDIFF(<date1>, <date2>, <interval>)

PARAMETERS:
  • <date1>, <date2>: Dates to compare
  • <interval>: Unit (DAY, MONTH, YEAR, QUARTER, WEEK, HOUR, MINUTE, SECOND)

EXAMPLE FROM YOUR DATASET:

Days Between Order and Today (Calculated Column):
  Days to Delivery = DATEDIFF(SalesData[OrderDate], TODAY(), DAY)

  Result: Number of days since order

Months Since Order (Calculated Column):
  Months Old = DATEDIFF(SalesData[OrderDate], TODAY(), MONTH)

  Result:
    • Order 2 months ago = 2
    • Order 5 months ago = 5

REAL USAGE:
  • Calculate delivery time
  • Measure period gaps
  • Age calculations
```

3️⃣4️⃣ EDATE() - ADD MONTHS TO DATE

```
SYNTAX:
  EDATE(<start_date>, <months>)

PARAMETERS:
  • <start_date>: Starting date
  • <months>: Number of months to add (negative to subtract)

EXAMPLE FROM YOUR DATASET:

Next Month From Order (Calculated Column):
  Next Month Review = EDATE(SalesData[OrderDate], 1)

  Result: Adds 1 month to order date
    • 2023-01-15 → 2023-02-15

3 Months from Order (Calculated Column):
  Review Date = EDATE(SalesData[OrderDate], 3)

12 Months Ago (Calculated Column):
  Year Ago Date = EDATE(TODAY(), -12)

REAL USAGE:
  • Calculate future dates
  • Warranty expiry
  • Review schedules
```

3️⃣5️⃣ EOMONTH() - END OF MONTH

```
SYNTAX:
  EOMONTH(<start_date>, <months>)
```

PARAMETERS:
  • <start_date>: Starting date
  • <months>: Months to offset (0 = same month, 1 = next month)

EXAMPLE FROM YOUR DATASET:

Month End for Order (Calculated Column):
  Month End = EOMONTH(SalesData[OrderDate], 0)

  Result: Last day of the month
    • 2023-01-15 → 2023-01-31
    • 2023-02-10 → 2023-02-28

Next Month End (Calculated Column):
  Next Month End = EOMONTH(SalesData[OrderDate], 1)

REAL USAGE:
  • Month-end reporting
  • Period calculations
  • Billing cycles

3️⃣6️⃣ YEAR() / MONTH() / DAY() - EXTRACT DATE PARTS

SYNTAX:
  YEAR(<date>)
  MONTH(<date>)
  DAY(<date>)

PARAMETERS:
  • <date>: Date to extract from

EXAMPLE FROM YOUR DATASET:

Extract Year (Calculated Column):
  Order Year = YEAR(SalesData[OrderDate])

  Result: 2023, 2024, 2025

Extract Month (Calculated Column):
  Order Month = MONTH(SalesData[OrderDate])

  Result: 1, 2, 3... 12

Extract Day (Calculated Column):
  Order Day = DAY(SalesData[OrderDate])

  Result: 1, 2, 3... 31

Combined: Year-Month (Calculated Column):
  Year Month = YEAR(SalesData[OrderDate]) & "-" & MONTH(SalesData[OrderDate])

  Result: "2023-01", "2023-02", etc.

REAL USAGE:
  • Group by year/month/day
  • Seasonal analysis
  • Date decomposition

3️⃣7️⃣ QUARTER() - GET QUARTER

SYNTAX:
  QUARTER(<date>)

PARAMETERS:
  • <date>: Date to extract quarter from

EXAMPLE FROM YOUR DATASET:

Order Quarter (Calculated Column):
  Order Quarter = QUARTER(SalesData[OrderDate])

  Result: 1, 2, 3, 4
    • Jan-Mar → Q1
    • Apr-Jun → Q2
    • Jul-Sep → Q3
    • Oct-Dec → Q4

Quarter Label (Calculated Column):
  Quarter Label = "Q" & QUARTER(SalesData[OrderDate])

  Result: Q1, Q2, Q3, Q4

REAL USAGE:
  • Quarterly reporting
  • Financial analysis
  • Trend grouping

3️⃣8️⃣ WEEKDAY() - DAY OF WEEK

SYNTAX:
  WEEKDAY(<date>, [return_type])

PARAMETERS:
  • <date>: Date to check
  • <return_type>: 1=Sunday-Saturday, 2=Monday-Sunday, 3=Monday=0

EXAMPLE FROM YOUR DATASET:

Day of Week Number (Calculated Column):
  Day Number = WEEKDAY(SalesData[OrderDate], 1)

  Result: 1-7 (1=Sunday, 7=Saturday)

Is Weekend (Calculated Column):
  Is Weekend = IF(WEEKDAY(SalesData[OrderDate]) > 5, "Weekend", "Weekday")

  Result: "Weekend" for Saturday/Sunday, "Weekday" otherwise

REAL USAGE:
  • Weekend vs weekday analysis
  • Day-based patterns
  • Weekend orders tracking

3️⃣9️⃣ DATESYTD() / DATESQTD() / DATESMTD() - TIME INTELLIGENCE

SYNTAX:

```
  DATESYTD(<dates>, [year_end_date])      – Year to date
  DATESQTD(<dates>, [quarter_end_date])   – Quarter to date
  DATESMTD(<dates>, [month_end_date])     – Month to date

PARAMETERS:
  • <dates>: Date column from Date Table
  • <year_end_date>: Year ending date (optional, default Dec 31)

EXAMPLE FROM YOUR DATASET:

Year-to-Date Sales (Calculated Measure):
  YTD Sales = CALCULATE(SUM(SalesData[FinalAmount]), DATESYTD(DateTable[Date]))

  Result: Sum of sales from Jan 1 to today
    • As of Dec 7, 2025: ₹12,500,000 (cumulative)

Quarter-to-Date Sales (Calculated Measure):
  QTD Sales = CALCULATE(SUM(SalesData[FinalAmount]), DATESQTD(DateTable[Date]))

  Result: Sum of sales from Q start to today

Month-to-Date Sales (Calculated Measure):
  MTD Sales = CALCULATE(SUM(SalesData[FinalAmount]), DATESMTD(DateTable[Date]))

  Result: Sum of sales from month 1st to today

REAL USAGE:
  • Track year-to-date progress
  • Monitor quarterly performance
  • Monthly targets vs actual
  • REQUIRES DATE TABLE!
```

4️⃣0️⃣ DATEADD() - ADD PERIODS TO DATE

```
SYNTAX:
  DATEADD(<dates>, <number>, <interval>)

PARAMETERS:
  • <dates>: Date column
  • <number>: Number to add (negative to subtract)
  • <interval>: Unit (DAY, MONTH, QUARTER, YEAR)

EXAMPLE FROM YOUR DATASET:

Same Month Last Year (Calculated Measure):
  Sales Last Year Same Month = CALCULATE(SUM(SalesData[FinalAmount]),
    DATEADD(DateTable[Date], -12, MONTH))

  Result: Sales from December last year
  Used for Year-over-Year comparison

Previous Quarter (Calculated Measure):
  Sales Last Quarter = CALCULATE(SUM(SalesData[FinalAmount]),
    DATEADD(DateTable[Date], -1, QUARTER))

Last 30 Days (Calculated Measure):
  Sales Last 30 Days = CALCULATE(SUM(SalesData[FinalAmount]),
    DATEADD(DateTable[Date], -30, DAY))

REAL USAGE:
  • Year-over-year comparisons
```

- Period-to-period analysis
- Historical trends
- REQUIRES DATE TABLE!