Date and Time DAX (Data Analysis Expressions

functions that help with calculations and transformations involving dates and times. These functions are commonly used for building time intelligence, aggregating data by specific periods, and creating dynamic date ranges.

Below is a detailed overview of **Date and Time DAX functions** in Power BI:

---

1. Date Creation Functions

These functions help you create date values.

- **`DATE(year, month, day)`**
  Creates a specific date.
  Example:
  `DATE(2024, 11, 19)` returns `19th November 2024`.

- **`TODAY()`**
  Returns the current date (without the time component).
  Example:
  If today's date is `19th November 2024`, `TODAY()` returns `19-Nov-2024`.

- **`NOW()`**
  Returns the current date and time.
  Example:
  If it's `19th November 2024, 3:00 PM`, `NOW()` returns `19-Nov-2024 15:00:00`.

---

2. Date Value Extraction Functions

These functions extract components like year, month, or day from a date.

- **`YEAR(date)`**
  Returns the year as an integer.
  Example:
  `YEAR(TODAY())` returns `2024`.

- **`MONTH(date)`**
  Returns the month as an integer (1 for January, 12 for December).
  Example:
  `MONTH(DATE(2024, 11, 19))` returns `11`.

- **`DAY(date)`**
  Returns the day of the month as an integer.

Example:
`DAY(DATE(2024, 11, 19))` returns `19`.

- **`WEEKDAY(date, [return_type])`**
  Returns the day of the week as an integer. By default, Sunday is `1` and Saturday is `7`.
  Example:
  `WEEKDAY(TODAY())` might return `3` (if today is Tuesday).

  Optional `return_type`:

  - `1`: Sunday as 1 (default).
  - `2`: Monday as 1.
  - `3`: Monday as 0.
- **`HOUR(datetime)`**
  Extracts the hour from a datetime value.
  Example:
  `HOUR(NOW())` returns `15` if the time is 3 PM.

- **`MINUTE(datetime)`**
  Extracts the minute from a datetime value.
  Example:
  `MINUTE(NOW())` returns `45` if the current time is 3:45 PM.

- **`SECOND(datetime)`**
  Extracts the second from a datetime value.
  Example:
  `SECOND(NOW())` returns `30` if the current time is 3:45:30 PM.

---

3. Date Difference Functions

These functions calculate differences between two dates or time periods.

- **`DATEDIFF(start_date, end_date, unit)`**
  Calculates the difference between two dates in specified units: `DAY`, `MONTH`, `QUARTER`, or `YEAR`.
  Example:
  `DATEDIFF(DATE(2023, 11, 19), DATE(2024, 11, 19), YEAR)` returns `1`.

- **`EOMONTH(start_date, months)`**
  Returns the last day of the month, offset by a specified number of months.
  Example:
  `EOMONTH(TODAY(), 1)` returns the last day of the next month.

---

4. Date Range Functions

These are helpful for creating dynamic date ranges.

- **`DATESBETWEEN(dates, start_date, end_date)`**
  Returns a table of dates between two specific dates.
  Example:
  `DATESBETWEEN('Date'[Date], DATE(2024, 1, 1), DATE(2024, 12, 31))` returns all dates in 2024.

- **`DATESYTD(dates, [year_end_date])`**
  Returns a table of dates from the start of the year to the specified date.
  Example:
  `DATESYTD('Date'[Date])` returns all dates from January 1 to today.

- **`DATESMTD(dates)`**
  Returns a table of dates from the start of the month to the specified date.
  Example:
  `DATESMTD('Date'[Date])` returns all dates from the 1st of the current month to today.

- **`DATESQTD(dates)`**
  Returns a table of dates from the start of the quarter to the specified date.
  Example:
  `DATESQTD('Date'[Date])` returns all dates from the start of the quarter.

---

5. Time Intelligence Functions

These functions are essential for analyzing data over time periods.

- **`SAMEPERIODLASTYEAR(dates)`**
  Returns a table of dates from the same period in the previous year.
  Example:
  Comparing November 2024 sales to November 2023 sales.

- **`PREVIOUSMONTH(dates)`**
  Returns the dates for the previous month.
  Example:
  `PREVIOUSMONTH('Date'[Date])` returns October 2024 dates if today is in November 2024.

- **`NEXTMONTH(dates)`**
  Returns the dates for the next month.
  Example:
  `NEXTMONTH('Date'[Date])` returns December 2024 dates if today is in November 2024.

- **`STARTOFYEAR(dates)`**
  Returns the first date in the year.
  Example:
  `STARTOFYEAR('Date'[Date])` returns `1st January 2024`.

- **`ENDOFYEAR(dates)`**
  Returns the last date in the year.
  Example:
  `ENDOFYEAR('Date'[Date])` returns `31st December 2024`.

- **`PARALLELPERIOD(dates, number, interval)`**
  Returns a table of dates shifted by the specified number of intervals (months, quarters, years).
  Example:
  `PARALLELPERIOD('Date'[Date], -1, YEAR)` returns all dates from the previous year.

---

6. Miscellaneous Date Functions

- **`TIME(hour, minute, second)`**
  Creates a time value based on the given components.
  Example:
  `TIME(14, 30, 0)` returns `2:30 PM`.

- **`WEEKNUM(date, [return_type])`**
  Returns the week number of the year. By default, the week starts on Sunday.
  Example:
  `WEEKNUM(TODAY())` might return `47` if today is in the 47th week of the year.

- **`FORMAT(date, format_string)`**
  Formats a date or time as text in a specific format.
  Example:
  `FORMAT(TODAY(), "MMM DD, YYYY")` returns `Nov 19, 2024`.

- **`ISBLANK(date)`**
  Checks whether the value is blank. Useful for handling null or missing dates.

Examples of Practical Usage

1. **Year-to-Date Sales**:

```
YTD Sales = CALCULATE(SUM(Sales[Amount]),
DATESYTD('Date'[Date]))
```

2. **Same Period Last Year**:

```
Last Year Sales = CALCULATE(SUM(Sales[Amount]),
SAMEPERIODLASTYEAR('Date'[Date]))
```

3. **Filter Last 30 Days**:

```
Last 30 Days =
CALCULATE(SUM(Sales[Amount]),
          FILTER('Date', 'Date'[Date] >= TODAY() - 30))
```

4. **Weekday Names**:

```
Day Name = FORMAT('Date'[Date], "dddd")
```

Key Takeaways

- **Date and Time DAX functions** are powerful for building dynamic reports, analyzing trends, and performing time-series analysis.
- Combining these functions with aggregations (like `SUM`, `AVERAGE`, etc.) and filters enables flexible and insightful analytics.
- Power BI's DAX functions align well with the **Date table**, so ensure you have a well-structured Date table in your data model for optimal usage.

Here's a table summarizing **Power BI Date and Time DAX Functions**, categorized by purpose and functionality:

Category	Function	Description	Example
Date Creation Functions	`DATE(year, month, day)`	Creates a specific date.	`DATE(2024, 11, 19)` → `19-Nov-2024`
	`TODAY()`	Returns the current date without the time component.	If today is `19-Nov-2024` → `19-Nov-2024`
	`NOW()`	Returns the current date and time.	If current time is `3:00 PM` → `19-Nov-2024 15:00:00`
Date Value Extraction	`YEAR(date)`	Extracts the year from a date.	`YEAR(TODAY())` → `2024`
	`MONTH(date)`	Extracts the month from a date.	`MONTH(DATE(2024, 11, 19))` - `11`
	`DAY(date)`	Extracts the day of the month from a date.	`DAY(DATE(2024, 11, 19))` → `19`
	`WEEKDAY(date, [return_type])`	Returns the weekday number of a date.	`WEEKDAY(TODAY())` → `3` (if today is Tuesday)
	`HOUR(datetime)`	Extracts the hour from a datetime value.	`HOUR(NOW())` → `15` (if time is 3:00 PM)
	`MINUTE(datetime)`	Extracts the minute from a datetime value.	`MINUTE(NOW())` → `45` (if time is 3:45 PM)
	`SECOND(datetime)`	Extracts the second from a datetime value.	`SECOND(NOW())` → `30` (if time is 3:45:30 PM)

Category	Function	Description	Example
Date Difference Functions	`DATEDIFF(start, end, unit)`	Calculates the difference between two dates in the specified unit (`DAY` , `MONTH` , etc.).	`DATEDIFF(DATE(2023, 11, 19) DATE(2024, 11, 19), YEAR)` → `1`
	`EOMONTH(start, months)`	Returns the last day of the month, offset by the specified number of months.	`EOMONTH(TODAY(), 1)` → `31-Dec-2024` (if today is in November 2024)
Date Range Functions	`DATESBETWEEN(dates, start, end)`	Returns dates between two specified dates.	`DATESBETWEEN('Date'[Date], DATE(2024, 1, 1), DATE(2024, 12, 31))`
	`DATESYTD(dates)`	Returns dates from the start of the year to the specified date.	`DATESYTD('Date'[Date])` → All dates YTD
	`DATESMTD(dates)`	Returns dates from the start of the month to the specified date.	`DATESMTD('Date'[Date])` → All dates MTD
	`DATESQTD(dates)`	Returns dates from the start of the quarter to the specified date.	`DATESQTD('Date'[Date])` → All dates QTD
Time Intelligence Functions	`SAMEPERIODLASTYEAR(dates)`	Returns dates for the same period in the previous year.	Comparing November 2024 to November 2023 sales
	`PREVIOUSMONTH(dates)`	Returns the dates for the previous month.	`PREVIOUSMONTH('Date'[Date])` → October 2024 dates
	`NEXTMONTH(dates)`	Returns the dates for the next month.	`NEXTMONTH('Date'[Date])` → December 2024 dates
	`STARTOFYEAR(dates)`	Returns the first date in the year.	`STARTOFYEAR('Date'[Date])` - `01-Jan-2024`
	`ENDOFYEAR(dates)`	Returns the last date in the year.	`ENDOFYEAR('Date'[Date])` → `31-Dec-2024`

Category	Function	Description	Example
	`PARALLELPERIOD(dates, num, unit)`	Returns dates shifted by the specified number of intervals (e.g., months, years).	`PARALLELPERIOD('Date'[Date] -1, YEAR)` → Previous year dates
Miscellaneous Functions	`TIME(hour, minute, second)`	Creates a time value from components.	`TIME(14, 30, 0)` → `2:30 PM`
	`WEEKNUM(date, [return_type])`	Returns the week number of the year.	`WEEKNUM(TODAY())` → `47` (if in 47th week of the year)
	`FORMAT(date, format_string)`	Formats a date or time as text.	`FORMAT(TODAY(), "MMM DD, YYYY")` → `Nov 19, 2024`
	`ISBLANK(value)`	Checks whether the value is blank (e.g., for null or missing dates).	`ISBLANK('Date'[Date])` → `TRUE` if blank

---

This table organizes DAX date functions based on their purpose, making it easy to understand their utility in Power BI.

```
In [ ]:
```