DAX Functions for Power BI: Information Functions

DAX Information functions analyze the input argument and produce a **True** or **False** output. The arguments of this function class are a cell or row from your Power BI table. The following functions fall under the **Information classification in DAX**:

1. **CONTAINS**: This function returns **true** or **false** depending on the values for referred columns. If all the values exist, then it returns true, otherwise false. It operates using the following syntax:

```
CONTAINS(<table>, <columnName>, <value>[, <columnName>, <value>]…)
```

2. **CUSTOMDATA**: This function reads the **connection string** and returns its **CustomData property** content. It operates using the following functions:

```
CUSTOMDATA()
```

3. **LOOKUPVALUE**: This function finds the **row** that meets the whole criteria provided by **search_columnName** and **search_value** and returns the value from **result_columnName**. It operates using the following syntax:

```
LOOKUPVALUE( <result_columnName>, <search_columnName>,  <search_value>[,
<search_columnName>, <search_value>]…)
```

4. **USERNAME**: This function returns your **domain** and **user name** that you have submitted to the system during connection setup. It operates using the following syntax:

```
USERNAME()
```

5. **ISBLANK**: This function returns **true** for **blank** and **false** for any other value after checking the input value. It operates using the following syntax:

```
ISBLANK(<value>)
```

The other important functions under **Information Functions** are as follows:

- ISERROR
- ISEVEN
- ISLOGICAL
- ISNUMBER
- ISODD
- ISTEXT

DAX Functions for Power BI: Logical Functions

DAX Logical Functions evaluate a certain input expression containing logical operators and provide a **True** or **False** output. The Logical functions work on evaluating a logical expression and are different from the Information functions which test the input argument by matching it to the stored data. The following functions fall under the **Logical classification in DAX**:

1. **AND**: This function returns **true** if both arguments **present** in the input expression are valid, else it returns **false**. It operates using the following syntax:

```
AND(<logical1>,<logical2>)
```

2. **NOT**: This function changes the value of the input expression to its opposite counterpart. This implies it changes from **true** to **false** and **vice-versa**. It operates using the following syntax:

```
  NOT(<logical>)
```

3. **OR**: This function returns **true** if **any** of the arguments present in the input expression are valid, else it returns **false**. It operates using the following syntax:

```
OR(<logical1>,<logical2>)
```

4. **SWITCH**: This function matches the input expression with a **list of values** and returns one of the various possible outcomes. It operates using the following syntax:

```
SWITCH(<expression>, <value>, <result>[, <value>, <result>]…[, <else>])
```

5. **IF**: This function checks the given condition and based on that returns a **true** or **false** condition provided the first argument is met. It operates using the following syntax:

```
IF(logical_test>,<value_if_true>, value_if_false)
```

The other important functions under **Logical Functions** are as follows:

- TRUE
- FALSE
- IFERROR
- IN

DAX Functions for Power BI: Mathematical & Trigonometric Functions

DAX Mathematical and Trigonometric functions allow you to perform basic mathematical operations with ease, These functions work similarly to their **Excel** counterparts. The following functions fall under the Mathematical & Trigonometric classification in DAX:

1. **ABS**: This function returns the **absolute value** of the given input number using the following syntax:

```
ABS(<number>)
```

2. **ACOS**: This function returns the **inverse cosine value** of a given input number. The angle that it returns is in radians ranging from **0 (zero) to pi**. It operates using the following syntax:

```
ACOS(number)
```

3) **CEILING**: This function returns the **nearest roundup value (integer)** of the given input using the following syntax:

```
CEILING(<number>, <significance>)
```

4) **COMBIN**: This function returns the number of **possible combinations** for the number of items given in the input. It operates using the following syntax:

```
COMBIN(number, number_chosen)
```

5) **CURRENCY**: This function **evaluates** the argument and returns the result as **currency data type**. It operates using the following syntax:

```
CURRENCY(<value>)
```

The other important functions under **Mathematical & Trigonometric** are as follows:

- ASIN
- ATAN
- COS

- DEGREES
- DIVIDE
- EVEN
- EXP
- FACT

DAX Functions for Power BI: Parent & Child Functions

DAX Parent and Child functions enable you to manage data that is constrained in a parent-child hierarchy. The following functions fall under the **Parent & Child classification in DAX**:

1. **PATH**: This function returns a **delimited text string** with the parent identifiers of all the current identifiers. It operates using the following syntax:

```
PATH(<ID_columnName>, <parent_columnName>)
```

2. **PATHITEM**: This function reads the **PATH function** and returns the item at the **specified position** from a string. It operates using the following syntax:

```
PATHITEM(<path>, <position>[, <type>])
```

The other important functions under **Parent & Child** are as follows:

- PATHITEMREVERSE
- PATHLENGTH
- PATHCONTAINS

DAX Functions for Power BI: Statistical Functions

DAX Statistical functions support operations related to **statistical formulas** including averages, approximations, and much more. The following functions fall under the **Statistical classification in DAX**:

1. **ADDCOLUMNS**: This function performs the **addition of calculated columns** of the given table. It operates using the following syntax:

```
ADDCOLUMNS(<table>, <name>, <expression>[, <name>, <expression>]…)
```

2. **AVERAGE**: This function calculates and returns the **arithmetic mean** of all the numbers of an input column using the following syntax:

```
    AVERAGE(<column>)
```

3. **AVERAGEA**: This function also returns the **arithmetic mean of column** values but also manages the text and non-numeric values. It operates on the following syntax:

```
AVERAGEA(<column>)
```

4. **BETA**.INV: This function returns the **inverse value of the beta** cumulative probability density function. It operates using the following syntax:

```
BETA.INV(probability,alpha,beta,[A],[B])
```

5. **CHISQ.INV**: This function returns the **inverse value of the left-tailed probability** from the chi-squared distribution. It operates using the following syntax:

```
CHISQ.INV(probability,deg_freedom)
```

The other important functions under **Statistical** are as follows:

- CONFIDENCE.NORM
- COUNT

- COUNTBLANK
- COUNTROWS
- CROSSJOIN
- DISTINCTCOUNT
- EXPON.DIST
- GENERATE
- GENERATEALL
- GEOMEAN
- GEOMEANX
- MAX
- MEDIAN
- MEDIANX
- MIN
- MINX

DAX Functions for Power BI: Text Functions

DAX Text functions allow you to work with tables and columns while using the String data type. Using these functions, you manipulate parts of a string, find certain text in strings, control the formats for dates, and do much more. The following functions fall under the Text classification in DAX:

1. **BLANK**: This function simply returns a **blank** and works using the following syntax:

```
BLANK()
```

2. **CODE**: This function returns a **numeric value(code)** corresponding to the first character in the input text string. IT operates using the following syntax:

```
CODE(text)
```

3. **CONCATENATE**: This function **joins two input text strings** and returns a single text string using the following syntax:

```
CONCATENATE(<text1>, <text2>)
```

4. **EXACT**: This function compares **two text strings** and if they are exactly the same, it returns a true value. It operates using the following syntax:

```
EXACT(<text1>,<text2>)
```

5. **FIND**: This function returns the **starting point of a text string** within another input text string. It operates using the following syntax:

```
FIND(<find_text>, <within_text>[, [<start_num>][, <NotFoundValue>]])
```

The other important functions under **Text** are as follows:

- COMBINEVALUES
- FORMAT
- LEFT
- MID
- REPLACE
- RIGHT
- SEARCH
- SUBSTITUTE
- TRIM

- UNICHAR

DAX Functions for Power BI: Other Functions

This category contains **DAX Functions** that can not be grouped in a specific class due to their unique characteristic properties. The following functions fall under the Other classification in DAX:

- DATATABLE
- ERROR
- EXCEPT
- GENERATESERIES
- GROUPBY
- INTERSECT
- ISEMPTY
- ISSELECTEDMEASURE
- NATURALINNERJOIN
- NATURALLEFTOUTERJOIN
- SELECTEDMEASUREFORMATSTRING
- SUMMARIZECOLUMNS
- UNION
- VAR