

# زبان DAX

## Data Analysis eXpression

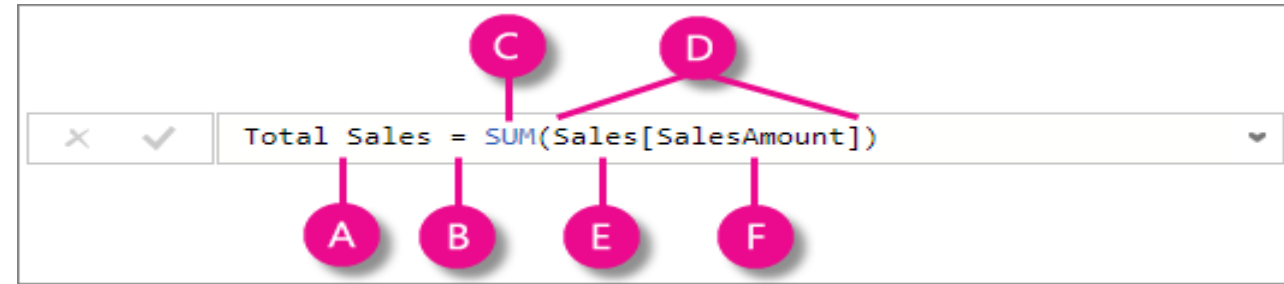
# DAX

- زبان DAX یک کتابخانه از توابع و اپراتورهایی است که می تواند برای ساخت فرمول و عبارات ترکیب شوند که در نهایت می تواند یک یا چندین مقدار را برگرداند.
- زبان DAX زبان برنامه نویسی نیست.
- DAX به شما کمک می کند اطلاعات جدیدی را از داده های موجود در مدل داده ای خود ایجاد کنید.
- فرمول های DAX شما را قادر به انجام مدل سازی داده ها، تجزیه و تحلیل داده ها و استفاده از نتایج برای گزارش گیری و تصمیم گیری می کند.

# DAX:

- Data Analysis Expressions (DAX) is a library of functions and operators that can be combined to build formulas and expressions in Power BI, Analysis Services, and Power Pivot in Excel data models.
- Once a physical table exists in a Power BI, DAX cannot add, change, or remove data from that table.
- Data can only be filtered or queried using DAX functions.
- DAX formulas are used in measures, calculated columns, calculated tables, and row-level security.
- DAX formulas include functions, operators, and values to perform advanced calculations and queries on data in related tables and columns in tabular data models.

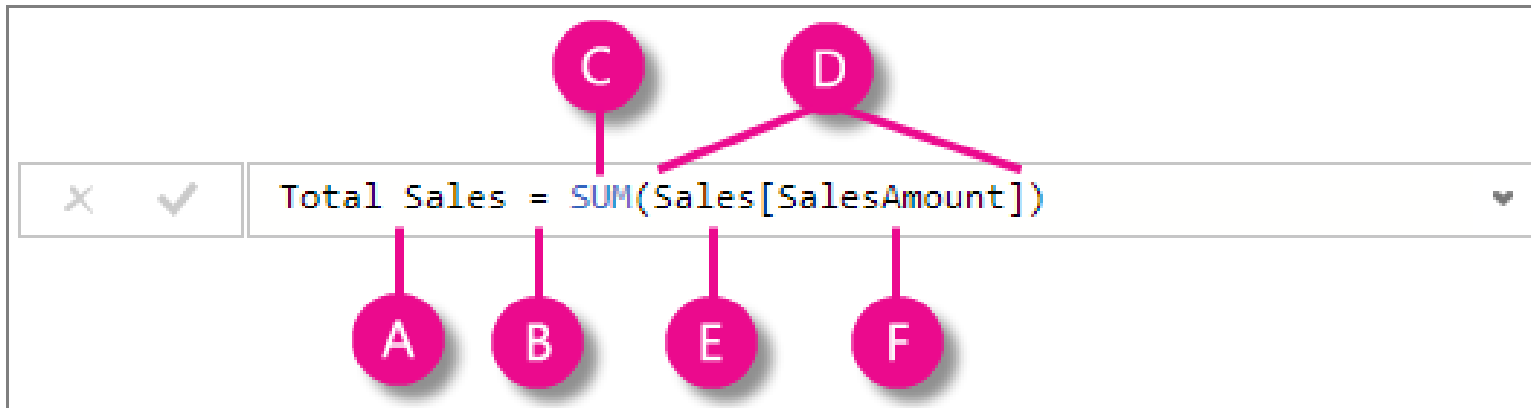
# Parts of DAX Formula



- **A.** The measure name, **Total Sales**.
- **B.** The equals sign operator (=), which indicates the beginning of the formula. When calculated, it will return a result.
- **C.** The DAX function **SUM**, which adds up all of the numbers in the **Sales[SalesAmount]** column. You'll learn more about functions later.
- **D.** Parenthesis **()**, which surround an expression that contains one or more arguments. All functions require at least one argument. An argument passes a value to a function.
- **E.** The referenced table, **Sales**.
- **F.** The referenced column, **[SalesAmount]**, in the Sales table. With this argument, the SUM function knows on which column to aggregate a SUM.

# فرمت زبان DAX

- A. عنوان شاخص ، قلم آماری (عنوان دلخواه)
- B. عملگر (=) نشانگر شروع فرمول محاسبه است.
- C. تابع Sum جمع ستونی فیلد SalesAmount می باشد
- D. نام جدول به همراه فیلد
- E. نام جدول
- F. ستون مورد نظر (ستون داخل براکت)



*For the measure named Total Sales, calculate (=) the SUM of values in the [SalesAmount] column in the Sales table.*

# نوشتن توضیحات (Comments) در فرمول:

```
Measure = DATEDIFF(  
    -- This is the first argument  
    TODAY() ,  
    // This is the second argument  
    DATE(2017,12,24),  
    /* This is the final argument */  
    DAY  
)
```

# استفاده از Quotation marks:

شی Object	مثال	توضیح
جدول	Sales	در عنوان جدول از کاراکتر Space یا کاراکترهای Special استفاده نشده است
جدول	'Canada Sales'	در عنوان جدول از کاراکتر Space یا کاراکترهای Special استفاده شده است برای این منظور نام جدول داخل تک کوتیشن قرار می گیرد
عنوان کامل ستون	Sales[Amount]	برای نوشتن کامل فیلد بهتر است نام جدول در ادامه داخل براکت فیلد ذکر شود
عنوان کامل ستون	'Canada Sales'[Amount]	

# Types of Operators:

- **Arithmetic operators**
- **Comparison operators**
- **Text concatenation operator**
- **Logical operators**



# Arithmetic operators

Arithmetic operator	Meaning	Example
+ (plus sign)	Addition	3+3
– (minus sign)	Subtraction or sign	3–1–1
* (asterisk)	Multiplication	3*3
/ (forward slash)	Division	3/3
^ (caret)	Exponentiation	16^4

# Comparison operators

Comparison operator	Meaning	Example
=	Equal to	[Region] = "USA"
==	Strict equal to	[Region] == "USA"
>	Greater than	[Sales Date] > "Jan 2009"
<	Less than	[Sales Date] < "Jan 1 2009"
>=	Greater than or equal to	[Amount] >= 20000
<=	Less than or equal to	[Amount] <= 100
<>	Not equal to	[Region] <> "USA"

All comparison operators except == treat BLANK as equal to number 0, empty string "", DATE(1899, 12, 30), or FALSE. As a result, [Column] = 0 will be true when the value of [Column] is either 0 or BLANK. In contrast, [Column] == 0 is true only when the value of [Column] is 0.

# Text concatenation operator

Text operator	Meaning	Example
& (ampersand)	Connects, or concatenates, two values to produce one continuous text value	[Region] & ", " & [City]

# Logical operators

Use logical operators (&&) and (||) to combine expressions to produce a single result.

Text operator	Meaning	Examples
&& (double ampersand)	Creates an AND condition between two expressions that each have a Boolean result. If both expressions return TRUE, the combination of the expressions also returns TRUE; otherwise the combination returns FALSE.	<code>(([Region] = "France") &amp;&amp; ([BikeBuyer] = "yes"))</code>
(double pipe symbol)	Creates an OR condition between two logical expressions. If either expression returns TRUE, the result is TRUE; only when both expressions are FALSE is the result FALSE.	<code>((([Region] = "France")    ([BikeBuyer] = "yes"))</code>
IN	Creates a logical OR condition between each row being compared to a table. Note: the table constructor syntax uses curly braces.	<code>'Product'[Color] IN { "Red", "Blue", "Black" }</code>

# Calculation order

An expression evaluates the operators and values in a specific order. All expressions always begin with an equal sign (=). The equal sign indicates that the succeeding characters constitute an expression.

Following the equal sign are the elements to be calculated (the operands), which are separated by calculation operators. Expressions are always read from left to right, but the order in which the elements are grouped can be controlled to some degree by using parentheses.


# Operator precedence

If you combine several operators in a single formula, the operations are ordered according to the following table. If the operators have equal precedence value, they are ordered from left to right. For example, if an expression contains both a multiplication and division operator, they are evaluated in the order that they appear in the expression, from left to right.

Operator	Description
$\wedge$	Exponentiation
$-$	Sign (as in $-1$ )
$*$ and $/$	Multiplication and division
$!$	NOT (unary operator)
$+$ and $-$	Addition and subtraction
$\&$	Connects two strings of text (concatenation)
$=, =, <, >, <=, >=, <>$	Comparison


# Example:

DAX

 Copy


$=5+2*3$

DAX

 Copy


$=(5+2)*3$

DAX

 Copy


$=(3 + 0.25)/(3 - 0.25)$

DAX

 Copy

$=-2^2$

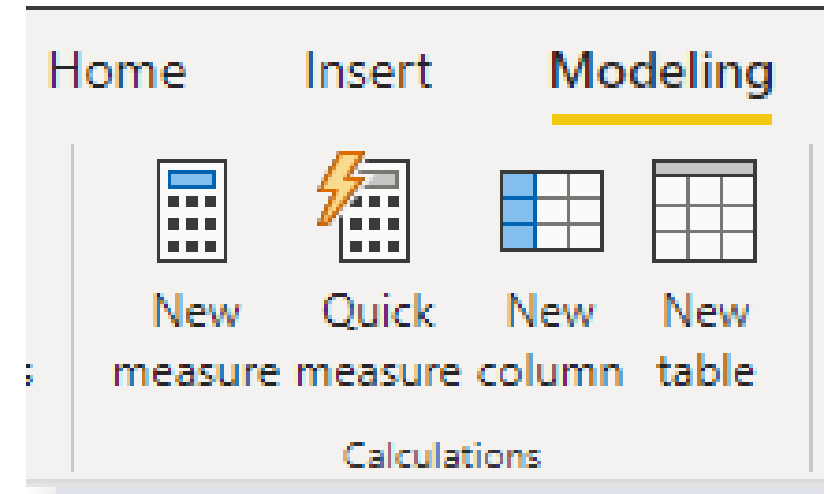
DAX

 Copy

$= (-2)^2$

# Calculation in DAX:

- Calculated columns
- Calculated measures
- Calculated tables

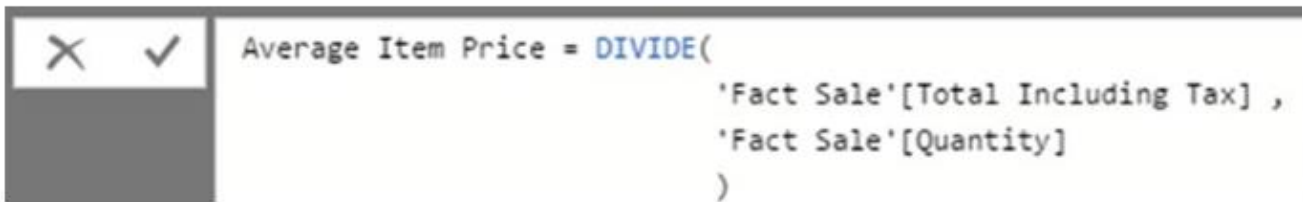




# ستون Calculated Column

- در صورتی که بخواهیم سطر به سطر محاسبات انجام پذیرد از Calculated Column استفاده می شود .
- محاسبه و ذخیره می شود. (Pre calculated & Store)
- داده ها بعد از فیلتر کاربر محاسبه مجدد نمی شوند (Not Recalculated)
- در زمانی که منبع اطلاعات رفرش (بارگذاری مجدد) شود محاسبات مجدد انجام خواهد شد.
- فضای RAM را اشغال می کند، از فضای دیسک هم استفاده می کند
- به عنوان فیلد ارتباطی بین جداول می توانید استفاده کنید.
- در حجم اطلاعات بالا کارایی داشبورد پایین می آید

# Calculated Column Example:



**Figure 1-4.** A calculated column for [Average Item Price]

Quantity	Total Including Tax	Average Item Price
90	1552.5	\$17.25
90	3001.5	\$33.35
90	3829.5	\$42.55
90	10246.5	\$113.85
90	3881.25	\$43.13
90	3881.25	\$43.13
90	2277	\$25.30
90	3001.5	\$33.35
90	4347	\$48.30
90	3415.5	\$37.95
90	2277	\$25.30
90	1552.5	\$17.25
90	10246.5	\$113.85

**Note** This formula works without the table name preceding each column name; however, it is highly recommended that whenever you reference a column, you always include the table name. This makes it much easier to differentiate between columns and measures when you are debugging longer DAX queries.

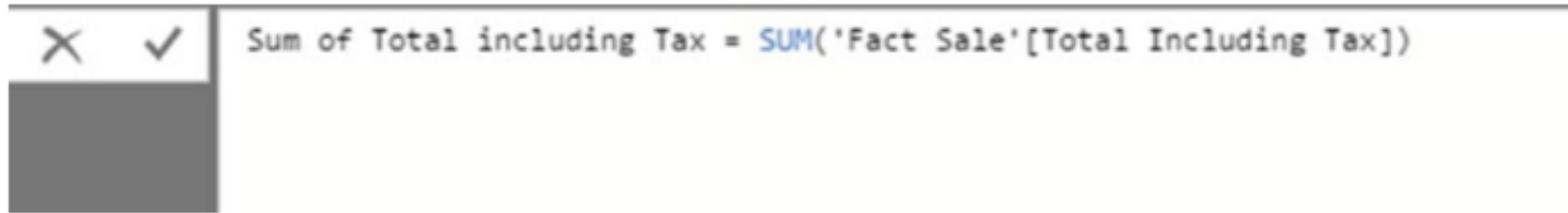
# Example:

Formula	Definition
<code>= TODAY()</code>	Inserts today's date in every row of a calculated column.
<code>= 3</code>	Inserts the value 3 in every row of a calculated column.
<code>= [Column1] + [Column2]</code>	Adds the values in the same row of [Column1] and [Column2] and puts the results in the calculated column of the same row.

# فیلد محاسباتی Measure

- در صورتی که بخواهیم محاسبات سطری یا مجموعه از سطرها را داشته باشیم می توانیم از Measure استفاده کنیم.
- در Ram ذخیره نمی شود.
- در صورتی که فیلتری انجام شود داده هایش محاسبه مجدد (recalculate) می شود.
- قابلیت استفاده مجدد در قلم آماری دیگر را نیز دارد.
- نمی توانید به عنوان فیلد ارتباطی بین جداول استفاده کنید.

# Calculated Measure Example:



**Figure 1-2.** DAX for the first calculated measure. This calculation uses the SUM function to return a single value anywhere the calculated measure is used in the report. When dragged and dropped to the report canvas using a visual with no other fields or filters, the value should show as \$198,043,493.45.



# مثال فیلد محاسباتی Measure

```
1 Sales[SalesAmount] = Sales[Quantity] * Sales[Unit Price]
2 Sales[TotalProductCost] = Sales[Quantity] * Sales[Unit Cost]
3 Sales[GrossMargin] = Sales[SalesAmount] - Sales[TotalProductCost]
4 Sales[Gross Margin %] := DIVIDE ( SUM ( Sales[GrossMargin] ), SUM (Sales[SalesAmount] ) )
```

```
1 Sales[Gross Margin %] :=
2 DIVIDE (
3     SUMX ( Sales, Sales[Quantity] * Sales[Unit Price] )
4     - SUMX ( Sales, Sales[Quantity] * Sales[Unit Cost] ),
5     SUMX ( Sales, Sales[Quantity] * Sales[Unit Price] )
6 )
```

```
1 Sales[Gross Margin %] :=
2 VAR SalesAmount = SUMX ( Sales, Sales[Quantity] * Sales[Unit Price] )
3 VAR TotalProductCost = SUMX ( Sales, Sales[Quantity] * Sales[Unit Cost] )
4 VAR GrossMargin = SalesAmount - TotalProductCost
5 RETURN DIVIDE ( GrossMargin, SalesAmount )
```

# جدول Table

- در صورتی که خروجی تابع یا اطلاعات بیش از یک فیلد باشد نوع خروجی اطلاعات باید جدول باشد .
- توابعی مثل Summarize، Group By و... نوع خروجی آنها جدول می باشد .

# Calculated Tables Example:



*Figure 1-7. Calculated table created using FILTER*



# DAX Function types:

- **Date and time functions**
- **Filter functions**
- **Financial functions**
- **Information functions**
- **Logical functions**
- **Math and Trig functions**
- **Other functions**
- **Parent and Child functions**
- **Relationship functions**
- **Statistical functions**
- **Table manipulation functions**
- **Text functions**
- **Time intelligence functions**

<https://docs.microsoft.com/en-us/dax/dax-function-reference>

# DAX Functions:

- [Logical functions](#) - These functions return information about values in an expression. For example, the TRUE function lets you know whether an expression that you are evaluating returns a TRUE value.
- [Text functions](#) - With these functions, you can return part of a string, search for text within a string, or concatenate string values. Additional functions are for controlling the formats for dates, times, and numbers.
- [Date and time functions](#) - These functions in DAX are similar to date and time functions in Microsoft Excel. However, DAX functions are based on the datetime data types used by Microsoft SQL Server.
- [Time intelligence functions](#) - These functions help you create calculations that use built-in knowledge about calendars and dates. By using time and date ranges in combination with aggregations or calculations, you can build meaningful comparisons across comparable time periods for sales, inventory, and so on.
- [Math and Trig functions](#) - Mathematical functions in DAX are similar to Excel's mathematical and trigonometric functions. However, there are some differences in the numeric data types used by DAX functions.
- [Statistical functions](#) - These functions perform aggregations. In addition to creating sums and averages, or finding minimum and maximum values, in DAX you can also filter a column before aggregating or create aggregations based on related tables.
- [Filter functions](#) - These functions help you return specific data types, look up values in related tables, and filter by related values. Lookup functions work by using tables and relationships between them. Filtering functions let you manipulate data context to create dynamic calculations.

# DAX Functions:

- [Information functions](#) - These functions look at a table or column provided as an argument to another function and tells you whether the value matches the expected type. For example, the ISERROR function returns TRUE if the value you reference contains an error.
- [Relationship functions](#) - These functions are for managing and utilizing relationships between tables. For example, you can specify a particular relationship to be used in a calculation.
- [Table manipulation functions](#) - These functions return a table or manipulate existing tables.
- [Other functions](#) - These functions perform unique actions that cannot be defined by any of the categories most other functions belong to.
- [Parent and Child functions](#) - These Data Analysis Expressions (DAX) functions help users manage data that is presented as a parent/child hierarchy in their data models.
- [Financial functions](#) - These functions are used in formulas that perform financial calculations, such as net present value and rate of return.

# توابع Logical Functions

## • توابع منطقی

ایجاد شرط ، استفاده از <<و>> و <<یا>> ،  
استفاده از Switch و ... در این دسته قرار  
دارد.

### ↳ DAX functions

#### New DAX functions

- › Date and time functions
- › Time-intelligence functions
- › Filter functions
- › Information functions
- › Logical functions
- › Math and trig functions
- › Other functions
- › Parent and child functions
- › Statistical functions
- › Text functions

# Logical Function

Function	Description
<a href="#"><u>AND</u></a>	Checks whether both arguments are TRUE, and returns TRUE if both arguments are TRUE.
<a href="#"><u>COALESC</u></a>	Returns the first expression that does not evaluate to BLANK.
<a href="#"><u>FALSE</u></a>	Returns the logical value FALSE.
<a href="#"><u>IF</u></a>	Checks a condition, and returns one value when it's TRUE, otherwise it returns a second value.
<a href="#"><u>IFERROR</u></a>	Evaluates an expression and returns a specified value if the expression returns an error
<a href="#"><u>NOT</u></a>	Changes FALSE to TRUE, or TRUE to FALSE.
<a href="#"><u>OR</u></a>	Checks whether one of the arguments is TRUE to return TRUE.
<a href="#"><u>SWITCH</u></a>	Evaluates an expression against a list of values and returns one of multiple possible result expressions.
<a href="#"><u>TRUE</u></a>	Returns the logical value TRUE.

# Logical Function

- IF :
- برای ایجاد شرط استفاده می شود.

DAX

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

DAX

```
=IF([Calls]<200,"low",IF([Calls]<300,"medium","high"))
```

# Logical Function

- یکی دیگر از دستورات شرطی که همانند یک لیست عمل می کند و مقداری که برابر است را بر می گرداند دستور Switch می باشد

DAX

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

DAX

```
=SWITCH([Month], 1, "January", 2, "February", 3, "March", 4, "April"  
    , 5, "May", 6, "June", 7, "July", 8, "August"  
    , 9, "September", 10, "October", 11, "November", 12, "December"  
    , "Unknown month number" )
```

# Logical Function

- AND : این تابع بررسی می کند که هر دو عبارت صحیح می باشند یا خیر و اگر هر دو آرگومان صحیح باشد True و در غیر اینصورت False برمی گرداند.

DAX

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

DAX

```
=IF(AND(10 > 9, -10 < -1), "All true", "One or more false")
```

```
USA Gold Medal Count:= SUMX(  
    Results,IF(AND([Country] = "USA",[Medal] = "Gold") = TRUE(),1,0)  
)
```



# Logical Function

- OR:

- این تابع بررسی می کند که آیا یکی از دو عبارت صحیح می باشند یا خیر و اگر هر یکی از آرگومان ها صحیح باشد True و در غیر اینصورت False برمی گرداند.

DAX

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

# Logical Function

- IN: این تابع عمل OR را در یک لیست بررسی می کند.

DAX

IN

DAX

```
Filtered Sales:=CALCULATE (  
    [Internet Total Sales], 'Product'[Color] IN { "Red", "Blue", "Black" }  
)
```

# Logical Function

- IFERROR: در صورتی که عبارت با خطا مواجه شود می توان شرطی را برای آن پیاده سازی کرد.

DAX





```
IFERROR(value, value_if_error)
```

DAX

```
=IFERROR(25/0,9999)
```

```
= IFERROR (5/0,"Div by zero") returns Div by zero.  
= IFERROR (5/1,"Div by zero") returns 5.
```

# مثال ۱ – ایجاد Indicator از طریق ICON

vehicle	Grade	Indicator
bicycle	11	
car	15	
carriage	8	
motor	12	
Total	46	

# توابع Text Functions

## • توابع متنی

این توابع محاسباتی بر روی رشته انجام می دهند ، عملیاتی مثل جمع کردن دو رشته ، واکشی مقداری از رشته و...

### ▼ DAX functions

#### ▼ New DAX functions

- Date and time functions
- Time-intelligence functions
- Filter functions
- Information functions
- Logical functions
- Math and trig functions
- Other functions
- Parent and child functions
- Statistical functions
- Text functions

# Text Function

Function	Description
<a href="#">COMBINEVALUES</a>	Joins two or more text strings into one text string.
<a href="#">CONCATENATE</a>	Joins two text strings into one text string.
<a href="#">CONCATENATEX</a>	Concatenates the result of an expression evaluated for each row in a table.
<a href="#">EXACT</a>	Compares two text strings and returns TRUE if they are exactly the same, FALSE otherwise.
<a href="#">FIND</a>	Returns the starting position of one text string within another text string.
<a href="#">FIXED</a>	Rounds a number to the specified number of decimals and returns the result as text.
<a href="#">FORMAT</a>	Converts a value to text according to the specified format.
<a href="#">LEFT</a>	Returns the specified number of characters from the start of a text string.
<a href="#">LEN</a>	Returns the number of characters in a text string.
<a href="#">LOWER</a>	Converts all letters in a text string to lowercase.
<a href="#">MID</a>	Returns a string of characters from the middle of a text string, given a starting position and length.
<a href="#">REPLACE</a>	REPLACE replaces part of a text string, based on the number of characters you specify, with a different text string.
<a href="#">REPT</a>	Repeats text a given number of times.
<a href="#">RIGHT</a>	RIGHT returns the last character or characters in a text string, based on the number of characters you specify.
<a href="#">SEARCH</a>	Returns the number of the character at which a specific character or text string is first found, reading left to right.
<a href="#">SUBSTITUTE</a>	Replaces existing text with new text in a text string.
<a href="#">TRIM</a>	Removes all spaces from text except for single spaces between words.
<a href="#">UNICHAR</a>	Returns the Unicode character referenced by the numeric value.
<a href="#">UNICODE</a>	Returns the numeric code corresponding to the first character of the text string.
<a href="#">UPPER</a>	Converts a text string to all uppercase letters.
<a href="#">VALUE</a>	Converts a text string that represents a number to a number.

# Text Function

- Concatenate: برای الحاق دو رشته می توان از تابع Concatenate استفاده کرد.
- در صورتی که بخواهیم سطر به سطر یک جدول محاسبات الحاق انجام بدهیم و بایک جداکننده آنها را بهم متصل کنیم می توانیم از ConcatenateX استفاده کنیم.

DAX

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

DAX

```
CONCATENATEX(<table>, <expression>, [delimiter])
```

# Text Function

- بررسی می کند که آیا دو رشته با هم برابر هستند یا خیر

DAX

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

- Find:  
در صورتی که رشته اول در رشته دوم وجود داشته باشد عدد اندیس مورد نظر را بر می گرداند

DAX

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

DAX

```
=FIND("BMX","line of BMX racing goods")
```



# Text Function

- Fixed: این تابع فرمت نمایشی اعداد را مشخص می کند و خروجی آن به صورت رشته می باشد

DAX

```
FIXED(<number>, <decimals>, <no_commas>)
```

DAX

```
=FIXED([PctCost],3,1)
```

FixedWithComma	FixedWithoutComma
334,251.453	334251.453
19,428,760.264	19428760.264
43,304.553	43304.553

# Text Function

- برای فرمت دهی اعداد ، تاریخ و زمان قابل استفاده می باشد خروجی این تابع رشته می باشد

DAX

```
FORMAT(<value>, <format_string>)
```

DAX

```
FORMAT( 12345.67, "General Number")  
FORMAT( 12345.67, "Currency")  
FORMAT( 12345.67, "Fixed")  
FORMAT( 12345.67, "Standard")  
FORMAT( 12345.67, "Percent")  
FORMAT( 12345.67, "Scientific")
```

General Number ▾	Currency ▾	Standard ▾	Percent ▾	Scientific ▾
12345.67	\$12,345.67	12,345.67	1234567.00%	1.23E+04

# Text Function

- Left, Len, Lower, Mid:

- تابع Left : از مقدار سمت چپ رشته بر اساس تعداد کاراکتر ورودی مقادیری را بر می گرداند.
- تابع Len : طول یک رشته را مشخص می کند
- تابع Lower : مقادیر رشته ای را به حروف کوچک تبدیل می کند.
- تابع Mid : از طریق این تابع می توانیم از هر مکانی در داخل رشته مقادیری را بر گردانیم.

# Text Function

- Replace, Rept, Right, Search:

- تابع Replace : عمل جایگزینی مقدار قبلی را با مقدار جدید از مکانی مشخص ، انجام می دهد.
- تابع Rept : عمل تکرار را انجام می دهد Rept مخفف Repeat می باشد
- تابع Right : از مقدار سمت راست رشته بر اساس تعداد کاراکتر ورودی مقادیری را بر می گرداند.
- تابع Search : بررسی دو رشته در داخل هم را انجام می دهد در صورتی که کاراکتر در رشته مورد نظر وجود نداشته باشد خطا بر می گرداند

# Text Function

- Substitute:

- این تابع شبیه به تابع Replace عمل جایگزینی را انجام می دهد.
- در صورتی که بخواهید متن خاصی را جایگزین کنید از تابع فوق استفاده کنید ، در صورتی که می خواهید در یک مکتب خاصی عمل جایگزینی را انجام دهید از Replace استفاده کنید.

DAX

```
SUBSTITUTE(<text>, <old_text>, <new_text>, <instance_num>)
```

DAX

```
=SUBSTITUTE([Product Code], "NW", "PA")
```

# Text Function

- Unichar: این تابع از طریق مقدار عددی که برایش وارد می شود شکل نمایشی Unicode را بر روی صفحه نمایش می دهد.

DAX
UNICHAR(number)

128526	128544	10026	128072	128073	128070	128071
😎	😞	🌟	👉	👉	👉	👉
😎	😞	🌟	👉	👉	👉	👉

## مثال ۲ – ایجاد Indicator از طریق Html Entity

Company	Rank	Stars
Donec Institute	1	★☆☆☆☆
Elit Nulla Facilisi LLP	3	★★★☆☆
Euismod Mauris Eu PC	2	★★☆☆☆
Massa Mauris Consulting	5	★★★★★
Risus Nulla Consulting	4	★★★★☆
<b>Total</b>	<b>15</b>	