



# ANÁLISE DE DADOS COM POWER BI

## Aula 07

- Iniciando a Linguagem DAX





# [ Visão Geral da Linguagem DAX ]

## O que é a linguagem DAX?

“A Linguagem DAX, é um acrônimo de **Data Analysis Expressions** ou **Expressos de Análises de Dados**, composta de uma coleção de funções, operadores e constantes **usadas para calcular e retornar um ou mais valores.**”

A linguagem DAX é utilizada no Power BI, Power Pivot no Excel e SQL Server Analysis Services Tabular. **Assim DAX é uma linguagem de expressões totalmente analíticas.**

# [ Tipos de Cálculos em DAX ]

A linguagem DAX possui 3 tipos de cálculos que podemos realizar, que são:



Medidas



Colunas



Tabelas



# [ Tipos de Cálculos em DAX ]



## Medidas (Contexto de Filtro ou Filter Context)

Medidas são fórmulas criadas especificamente para análise empregadas em relatórios e visualizações.

São cálculos agregadores que devem sempre retornar um valor único por contexto. Assim por padrão, as medidas são influenciadas pelos filtros aplicados ao modelo.

Um exemplo é quando utilizado uma função como SUM para calcular a totalização de uma coluna e seu resultado é uma medida.

Resumindo:

- Medidas estão no contexto de filtro.
- Contexto de filtro de total é diferente da linha.
- Os filtros internos e externos afetam o contexto de filtro de um valor.
- Contexto de filtro está diferentemente ligado as influencias que as tabelas fazem com as outras.
- A tabela dimensão propaga seus filtros para a fato (observar o modelo de relacionamento).

# [ Tipos de Cálculos em DAX ]



## Colunas Calculadas (Contexto de Linha ou Row Context)

Colunas são estruturas criadas em tabelas a qual devemos definir por uma expressão de cálculo, e os valores desta coluna são calculados linha por linha na tabela.

Assim uma coluna vai retornar um valor para cada linha da tabela onde está sendo calculado.

Exemplo é a criação de uma coluna calculada com os resultados de valores a serem inseridas no modelo de dados. No exemplo abaixo está a criação de uma coluna calculada a qual receberá o valor de desconto de 10% sobre o valor vendido.

Estrutura		Formatação	
	<input type="checkbox"/> <input checked="" type="checkbox"/>	1 Valor de Desconto = fVendas[Valor a Pagar]*0.1	
	Nota Fiscal	Data do Pedido	IDLoja IDProduto
	NFe-406690	segunda-feira, 4 de outubro de 2021	Loja 2 P40
	NFe-310840	terça-feira, 21 de fevereiro de 2023	Loja 2 P40

**:: LEMBRE-SE ::**

**Quando acrescentamos uma coluna calculada impactamos o desempenho de nosso arquivo PBIXs.**



# [ Tipos de Cálculos em DAX ]

## Colunas Calculadas

**Definição:** É uma coluna adicionada a uma tabela existente, onde cada linha da coluna é calculada com base em uma fórmula DAX (Data Analysis Expressions).

**Uso:** Ideal para cálculos que precisam ser feitos linha a linha, como concatenar valores de duas colunas ou calcular a diferença entre duas datas.

**Armazenamento:** Os valores são armazenados no modelo de dados, o que pode aumentar o tamanho do arquivo.

**Exemplo:** Criar uma coluna que mostra o valor da aplicação de desconto de 10% sobre a venda realizada de cada cliente.

## Medida

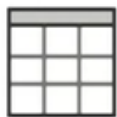
**Definição:** É um cálculo dinâmico que é avaliado no contexto do relatório, como filtros e seleções de usuário.

**Uso:** Ideal para cálculos agregados, como somas, médias ou contagens, que precisam ser recalculados com base no contexto do relatório.

**Armazenamento:** Os valores não são armazenados, mas calculados em tempo real, o que pode melhorar o desempenho.

**Exemplo:** Calcular a soma total das vendas

# [ Tipos de Cálculos em DAX ]



## Tabelas

São estruturas calculadas, que resultam em uma ou mais colunas, onde os dados são agregados e sumarizados em uma nova tabela.

Quanto ao tipo de tabela podem ser Tabela Física ou Tabela Virtual.

Uma **tabela é FÍSICA** quando é adicionada ao modelo podendo ser relacionada com outras tabelas.

Por outro lado uma **tabela é VIRTUAL** quando é criada em memória dentro de uma medida, e neste caso essa é parte intermediária de um cálculo.

# [ Tipos de Cálculos em DAX ]



## Tabelas

### Veja o exemplo a seguir:

A função CALCULATE avalia uma expressão em um contexto de filtro modificado. Ela permite aplicar filtros ou funções de modificação de filtro para alterar o contexto de uma avaliação. Aqui está um exemplo:

Suponha que temos uma tabela de vendas chamada "Tabela de Vendas" com as seguintes colunas:

- Vendas por Dia: Valor das vendas diárias.
- Data: Data da venda diária.

E com os dados acima foi solicitado que tenhamos de calcular a média de vendas para o período de 1º de janeiro de 2022 a 31 de março de 2022. Assim necessitamos de realizar os seguintes passos:

DAX

1ª medida a encontra → Média de Vendas = AVERAGE(Tabela de Vendas[Vendas por Dia])

2ª medida que gera uma tabela → Média de Vendas por Período = CALCULATE([Média de Vendas], Tabela de Vendas[Data] >= DATE(2022, 01, 01), Tabela de Vendas[Data] <= DATE(2022, 03, 31))

Neste exemplo, a medida Média de Vendas por Período calcula a média de vendas considerando apenas as datas dentro do período especificado





# [ Contexto Filter Context e Row Context do DAX ]

O contexto de linha (ROW CONTEXT) e o contexto de filtro (FILTER CONTEXT) são conceitos fundamentais na linguagem DAX (Data Analysis Expressions) do Power BI. Vamos explorar cada um deles:

## **FILTER CONTEXT (Contexto de Filtro) – Cálculo realizado nas colunas**

O FILTER CONTEXT é o ambiente no qual as fórmulas DAX são avaliadas, considerando os filtros aplicados aos dados antes que uma medida seja calculada. Ele afeta a maneira de como os dados são agregados e/ou escaneados.

Esses filtros podem vir de várias fontes, como: **segmentações**, **filtros visuais** ou filtros aplicados diretamente nas **fórmulas DAX**. O Filter Context é essencial para determinar quais dados serão considerados ao calcular uma medida.

### **Exemplo:**

Imagine que você tem uma tabela de vendas e deseja calcular o total de vendas para um ano específico. Se você aplicar um filtro para o ano de 2023, o conceito de Filter Context garantirá que apenas as vendas de 2023 sejam consideradas no cálculo.

Ou podemos

```
TotalVendas2023 = CALCULATE(SUM(Vendas[Valor]), Vendas[Ano] = 2023)
```



# [ Filter Context e Row Context do DAX ]

## ROW CONTEXT (Contexto de Linha) - Cálculo realizada em linha a linha

O ROW CONTEXT refere-se ao contexto de uma linha específica em uma tabela ou expressão DAX. Quando você escreve uma fórmula em uma coluna calculada, ela é avaliada para cada linha da tabela, criando um contexto de linha para cada uma delas.

O ROW CONTEXT, é criado automaticamente quando você usa colunas calculadas ou funções iterativas como SUMX, FILTER, AVERAGEX, ou com terminologia “x” ou CALCULATE, e permite que você acesse os valores das colunas da linha atual.

### **Exemplo:**

Imagine que você tem uma tabela de vendas com as seguintes colunas:

- Produto
- Quantidade
- Preço Unitário

Você quer criar uma nova coluna calculada que mostre o valor total de cada venda (Quantidade \* Preço Unitário). Para isso, você pode usar o DAX (Data Analysis Expressions) no Power BI.

Valor Total = Vendas[Quantidade] \* Vendas[Preço Unitário]

# [ Entendendo o Cálculo no DAX ]

A linguagem DAX em sua concepção consiste em  
Agregar e Filtrar os cálculos.



# [ Entendendo o Cálculo no DAX ]



## Vendas

Pedido	Cliente	Produtos	Vl. Total
1	João	Produto A	R\$ 1.000,00
1	João	Produto A	R\$ 3.000,00
2	João	Produto B	R\$ 2.000,00
3	Marcos	Produto B	R\$ 1.000,00
4	Marcos	Produto C	R\$ 3.000,00
4	Marcos	Produto D	R\$ 2.000,00
4	Marina	Produto A	R\$ 2.000,00
4	Sandra	Produto A	R\$ 1.000,00
5	Sandra	Produto A	R\$ 5.000,00
5	Sandra	Produto A	R\$ 6.000,00



**Total de Vendas = SUM(Vendas[Vl. Total])**

**= 26.000**

Este resultado é um valor agregado e que ainda não sofreu a influência de um filtro do modelo.

# [ Entendendo o Cálculo no DAX ]



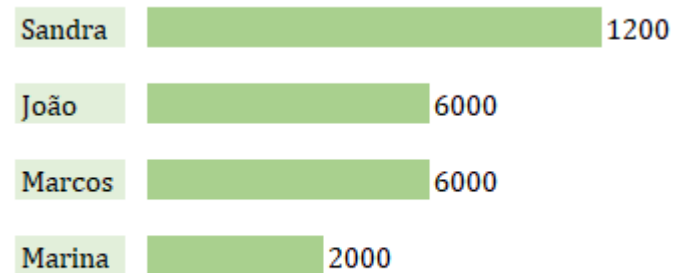
**Total de Vendas = SUM(Vendas[Vl. Total])**

**Aplicação da medida Total de Vendas no  
contexto de Filtro de Vendedores**

## Visual de Matriz

Cliente	Soma de Vendas
João	6.000
Marcos	6.000
Marina	2.000
Sandra	12.000
Total de Vendas	26.000

## Gráfico de Barras





# [ Entendendo o Cálculo no DAX ]

Quando arrastamos um campo para dentro de um visual que já possui uma medida o resultado é a realização de um **FILTRO**.

Neste caso: Sandra é um filtro de cliente, João é um filtro de cliente, Marcos é um filtro de cliente e Marina é um filtro de cliente.

Entretanto, os valores que surgem lateralmente de cada conteúdo é considerado como um Contexto. Assim, Contexto é um conjunto de dados que representa aquele filtro.

Cliente	Soma de Vendas
Sandra	12.000
João	6.000
Marcos	6.000
Marina	2.000
Total de Vendas	26.000

# [ Entendendo o Cálculo no DAX ]



Vendas



Total de Vendas =  
SUM(Vendas[VI. Total])

Tipo de Produtos =  
DISTINCTCOUNT(Vendas[Pro  
dutos])

Cliente	Soma de Vendas	Cliente	Soma de Vendas				Produtos Diferentes														
Sandra	12.000	Sandra	<table><tr><th>Pedido</th><th>Cliente</th><th>Produtos</th><th>VI. Total</th></tr><tr><td>4</td><td>Sandra</td><td>Produto A</td><td>R\$ 1.000,00</td></tr><tr><td>5</td><td>Sandra</td><td>Produto A</td><td>R\$ 5.000,00</td></tr><tr><td>5</td><td>Sandra</td><td>Produto A</td><td>R\$ 6.000,00</td></tr></table>	Pedido	Cliente	Produtos	VI. Total	4	Sandra	Produto A	R\$ 1.000,00	5	Sandra	Produto A	R\$ 5.000,00	5	Sandra	Produto A	R\$ 6.000,00	12.000	1
Pedido	Cliente	Produtos	VI. Total																		
4	Sandra	Produto A	R\$ 1.000,00																		
5	Sandra	Produto A	R\$ 5.000,00																		
5	Sandra	Produto A	R\$ 6.000,00																		
João	6.000	João	<table><tr><th>Pedido</th><th>Cliente</th><th>Produtos</th><th>VI. Total</th></tr><tr><td>1</td><td>João</td><td>Produto A</td><td>R\$ 1.000,00</td></tr><tr><td>1</td><td>João</td><td>Produto A</td><td>R\$ 3.000,00</td></tr><tr><td>2</td><td>João</td><td>Produto B</td><td>R\$ 2.000,00</td></tr></table>	Pedido	Cliente	Produtos	VI. Total	1	João	Produto A	R\$ 1.000,00	1	João	Produto A	R\$ 3.000,00	2	João	Produto B	R\$ 2.000,00	6.000	2
Pedido	Cliente	Produtos	VI. Total																		
1	João	Produto A	R\$ 1.000,00																		
1	João	Produto A	R\$ 3.000,00																		
2	João	Produto B	R\$ 2.000,00																		
Marcos	6.000	Marcos	<table><tr><th>Pedido</th><th>Cliente</th><th>Produtos</th><th>VI. Total</th></tr><tr><td>3</td><td>Marcos</td><td>Produto B</td><td>R\$ 1.000,00</td></tr><tr><td>4</td><td>Marcos</td><td>Produto C</td><td>R\$ 3.000,00</td></tr><tr><td>4</td><td>Marcos</td><td>Produto D</td><td>R\$ 2.000,00</td></tr></table>	Pedido	Cliente	Produtos	VI. Total	3	Marcos	Produto B	R\$ 1.000,00	4	Marcos	Produto C	R\$ 3.000,00	4	Marcos	Produto D	R\$ 2.000,00	6.000	3
Pedido	Cliente	Produtos	VI. Total																		
3	Marcos	Produto B	R\$ 1.000,00																		
4	Marcos	Produto C	R\$ 3.000,00																		
4	Marcos	Produto D	R\$ 2.000,00																		
Marina	2.000	Marina	<table><tr><th>Pedido</th><th>Cliente</th><th>Produtos</th><th>VI. Total</th></tr><tr><td>4</td><td>Marina</td><td>Produto A</td><td>R\$ 2.000,00</td></tr></table>	Pedido	Cliente	Produtos	VI. Total	4	Marina	Produto A	R\$ 2.000,00	2.000	1								
Pedido	Cliente	Produtos	VI. Total																		
4	Marina	Produto A	R\$ 2.000,00																		
Total de Vendas	26.000	Total de Vendas	TABELA VENDAS SEM O FILTRO DE CLIENTE	26.000	7																

# [ Operadores DAX ]

Temos quatro tipos diferentes de operadores de cálculo que são: aritmético, comparação, concatenação de texto e lógico.

Operador Aritmético	Significado	Exemplo
+ (sinal de adição)	Adição	3+3
- (sinal de subtração)	Subtração ou sinal	3-1-1
* (asterisco)	Multiplicação	3*3
/ (barra)	Divisão	03/mar
^ (sinal de interpolação)	Exponenciação	16^4

Operador de Comparação	Significado	Exemplo
=	Igual a	[Região] = "EUA"
==	Estrito igual a	[Região] == "EUA"
>	Maior que	[Data de vendas] > "Jan 2009"
<	Menor que	[Data de vendas] < "1º Jan 2009"
>=	Maior ou igual a	[Quantidade] >= 20000
<=	Menor ou igual a	[Quantidade] <= 100
<>	Diferente de	[Região] <> "EUA"

Operador de texto	Significado	Exemplo
& (e comercial)	Conecta ou concatena dois valores para produzir um valor de texto contínuo	[Region] & ", " & [City]

Operador de texto (lógico)	Significado	Exemplos
&& (E comercial duplo)	Cria uma condição AND entre duas expressões que têm um resultado booleano. Se ambas as expressões retornarem TRUE, a combinação das expressões também retornará TRUE; caso contrário, a combinação retornará FALSE.	(([Region] = "France") && ([BikeBuyer] = "yes"))
(barra vertical dupla)	Cria uma condição OR entre duas expressões lógicas. Se uma das expressões retornar TRUE, o resultado será TRUE; somente quando as duas expressões são FALSE o resultado é FALSE.	((([Region] = "France")    ([BikeBuyer] = "yes"))
IN	Cria uma condição OR lógica entre cada linha sendo comparada a uma tabela. Observação: a sintaxe do construtor de tabela usa chaves.	'Produto'[Cor] IN { "Vermelho", "Azul", "Preto" }





# [ Operadores e ordem de cálculo ]

## Operadores e ordem de precedência

Em alguns casos, a ordem na qual o cálculo é executado pode afetar o valor retornado. Portanto, é importante entender como o pedido é determinado e como você pode alterar a ordem para obter os resultados desejados.

## Ordem de cálculo

Uma expressão avalia os operadores e valores em uma ordem específica. Todas as expressões sempre começam com um sinal de igual (=). O sinal de igual indica que os caracteres seguintes constituem uma expressão.



# [ Categorias das funções DAX ]

As funções DAX (Data Analysis Expressions) no Power BI podem ser agrupadas em várias categorias, cada uma com um propósito específico. Aqui estão as principais categorias:

- **Funções de Agregação:** Calculam um valor (escalar), como contagem, soma, média, mínimo ou máximo para todas as linhas de uma coluna ou tabela, conforme definido pela expressão.
- **Funções de Data e Hora:** Semelhantes às funções de data e hora do Microsoft Excel, mas baseadas nos tipos de dados *datetime* usados pelo Microsoft SQL Server.
- **Funções de Filtro:** Ajudam a retornar tipos de dados específicos, pesquisar valores em tabelas relacionadas e filtrar o conteúdo por valores relacionados. Permitem manipular o contexto de dados para criar cálculos dinâmicos.
- **Funções Financeiras:** Usadas em cálculos financeiros, como valor líquido atual e taxa de retorno.
- **Funções de Informações:** Examinam uma tabela ou coluna e retornam se o valor corresponde ao tipo esperado.
- **Funções Lógicas:** Retornam informações sobre os valores em uma expressão.
- **Funções Matemáticas e Trigonométricas:** Semelhantes às funções do Excel, mas com diferenças nos tipos de dados numéricos usados.

Outras Funções:

Executam ações exclusivas que não se enquadram nas categorias anteriores.

- **Funções Pai e Filho:** Gerenciam dados apresentados como hierarquia pai/filho nos modelos de dados.
- **Funções de Relação:** Gerenciam e utilizam relações entre tabelas.
- **Funções Estatísticas:** Calculam valores relacionados a probabilidade e distribuições estatísticas.
- **Funções de Manipulação de Tabela:** Retornam ou manipulam tabelas existentes.
- **Funções de Texto:** Permitem retornar parte de uma cadeia de caracteres, pesquisar texto ou concatenar valores de cadeia de caracteres.

# [ Estrutura da medida DAX ]

Vamos começar com uma medida bem simples. Vamos fazer a soma de vendas (Valor das Vendas). Neste exemplo vamos somar os valores da coluna ValorTotal que está na tabela fDados.

```
Valor das Vendas = SUM( fDados[ValorTotal] )
```

Diagrama de numeração da fórmula:

- 1: Valor das Vendas
- 2: =
- 3: SUM
- 4: (
- 5: fDados
- 6: [ValorTotal]
- 7: )

Esta medida inclui os seguintes elementos de sintaxe:

- 1 – Nome da Medida, onde sempre é fornecido um nome que seja de fácil entendimento e único (sem repetição)
- 2 – Sinal de = que indica o início da Fórmula
- 3 – Função, existem várias funções que podemos usar.
- 4 – Abertura do parênteses
- 5 – Parâmetro referenciando a tabela fDados
- 6 – Parâmetro referenciando a coluna ValorTotal que está na tabela fDados
- 7 – Fechamento do parênteses: Toda função termina com parênteses.



# [ Comentário em estruturas DAX ]

Nas medidas DAX que usamos no Power BI podemos fazer comentários para podermos explicar alguma fórmula.

```
YoY Faturamento (Comentários) =  
IF(DISTINCTCOUNT(dCalendario[Ano])>1,0, -- Conta a coluna ano se há mais de 1 ano selecionado, caso VERDADEIRO, retorna zero (0)  
IF(ISBLANK([Faturamento LY]),0, // Verifica se há dados no ano anterior, caso estiver EM BRANCO, retorna zero (0)  
DIVIDE([Faturamento],[Faturamento LY]-1)) // Divide o faturamento pelo faturamento do ano anterior  
  
----- Comentários em Fórmulas DAX -----  
-- Teclas de Atalho --  
/*  
CTRL + ; ou CTRL + K + C = Comenta uma linha com ou nas selecionadas adicionando // (Para remover repita o atalho)  
ALT + SHIFT + A = Comentário para várias linhas selecionadas adicionando /* no início e */ no final. (Para remover repita o atalho)  
*/  
  
-----  
// de uma linha ( pode ser // ou -- )  
  
/*  
Comentário de  
Múltiplas linhas  
*/
```



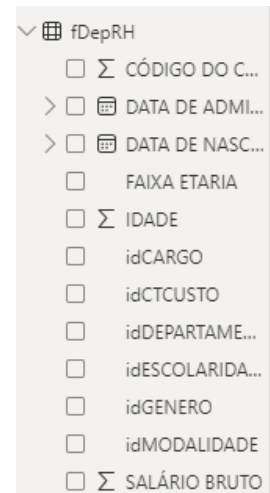
OPERATOR CLASSES

```
types.Operator):  
    error to the selected
```

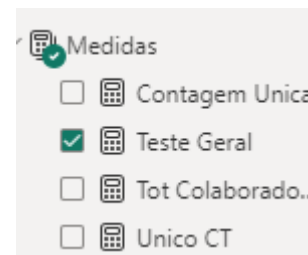
Temos dois tipos de medidas na Linguagem DAX que são as medidas Implícitas e as medidas Explícitas.

## MEDIDAS IMPLÍCITAS

São medidas criadas internamente e produzidas quando carregamos as tabelas tratadas na modelagem ou criadas através de uma inclusão de coluna calculada. Quando arrastamos uma coluna de uma tabela para o relatório, o Power BI cria uma agregação automaticamente.



**MEDIDAS EXPLÍCITAS:** São medidas que criamos com a linguagem DAX e gera um nome único. Por exemplo somar o valor da coluna ValorTotal. Valor da Vendas =SUM(fVendas[ValorTotal]).



# [ Medidas Implícita e Explícita no DAX ]

Temos dois tipos de medidas na Linguagem DAX que são as medidas Implícitas e as medidas Explícitas.

## Medidas Implícitas

**Definição:** São calculadas automaticamente pelo Power BI com base nas características dos dados. Você não precisa criar uma fórmula específica para elas.

**Exemplo:** Quando você arrasta um campo numérico para um visual, o Power BI pode automaticamente calcular a soma, média, mínimo, máximo, etc., desse campo.

**Uso:** São úteis para cálculos simples e rápidos, mas têm limitações em termos de flexibilidade e reutilização.

## Medidas Explícitas

**Definição:** São aquelas que você cria manualmente usando fórmulas DAX (Data Analysis Expressions). Você define explicitamente a fórmula e o nome da medida.

**Exemplo:** Criar uma medida chamada "Total de Vendas" usando a fórmula SUM(Tabela[Vendas]).

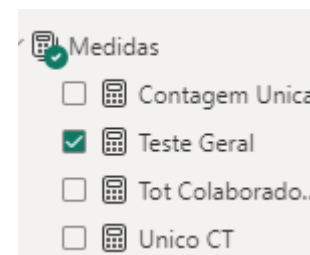
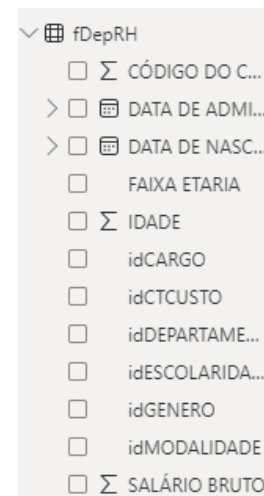
**Uso:** São mais flexíveis e poderosas, permitindo cálculos complexos e a aplicação de filtros dentro das medidas. Além disso, podem ser reutilizadas em diferentes visuais e relatórios.

## Vantagens das Medidas Explícitas

**Reutilização:** Podem ser usadas em múltiplos visuais e relatórios.

**Complexidade:** Permitem a criação de cálculos mais avançados.

**Filtros:** Podem incluir filtros diretamente na fórmula.





# [ Função CALENDARAUTO DAX ]

## CALENDARAUTO

Insere uma tabela calendário que compreende todas as datas do modelo de dados.

Processo: insira uma nova tabela no modelo de exibição de tabela, declare o nome e a função CALENDARAUTO() em seguida pressione enter. Podemos criar colunas com as formatações temporais a serem utilizadas ou declarar um arquivo txt com este contexto, como está demonstrado abaixo:

```
1 dCalendarioDAX = ADDCOLUMNS(  
2     CALENDARAUTO(),  
3     "Ano", FORMAT([Date],"YYYY"),  
4     "Mês", MONTH([Date]),  
5     "NomeMes", FORMAT([Date],"MMMM"),  
6     "Trimestre do Ano", VALUE(FORMAT([Date],"Q")),  
7     "Semana do Ano", WEEKDAY([Date],1),  
8     "Dia da Semana", FORMAT([Date],"dddd"),  
9     "Mês Abreviado", LEFT(FORMAT([Date],"MMMM"),3),  
10    "Mês/Ano", CONCATENATE(LEFT(FORMAT([Date],"MMMM"),3),"/"&YEAR([Date])),  
11    "Ano/Trimestre", CONCATENATE(FORMAT([Date],"YYYY")," - "&VALUE(FORMAT([Date],  
    "Q"))&"º Trimestre")  
12 )
```



# [ Função AVERAGE DAX ]

## AVERAGE

- Retorna o valor da MÉDIA (aritmética) de todos os números de uma coluna
- Estrutura da fórmula: AVERAGE (<coluna>) • Utilize a função AVERAGE em colunas numéricas. Retorna um número decimal como resultado, que é a média aritmética (conhecida também como média simples) da coluna.
- Ao calcular a média, lembre-se da diferença entre uma célula vazia e uma célula contendo o valor 0 (zero). Quando uma célula contiver 0, ela será adicionada à soma de números e a linha será contada entre o número de linhas usadas como o divisor. Porém, quando uma célula contiver um espaço em branco, a linha não será contada.





# [ Função MAX e MIN DAX ]

## MAX

- Pertence ao grupo de funções estatísticas. Esta função retorna o maior valor numérico de uma coluna.
- Estrutura da fórmula: MAX (<coluna>)
- Retorna um número decimal como resultado. Deve ser utilizada em colunas numéricas. Caso a coluna não contenha números a função retornará um espaço em branco. Células vazias, valores lógicos e texto são ignorados.

## MIN

- Pertence ao grupo de funções estatísticas. Ao contrário da função anterior, retorna o menor valor numérico de uma coluna.
- Estrutura da fórmula: MIN (<coluna>)
- Retorna um número decimal como resultado. Deve ser utilizada em colunas numéricas. Caso a coluna não contenha números a função retornará um espaço em branco. Células vazias, valores lógicos e texto são ignorados.



# [ Função COUNTROWS DAX ]

## COUNTROWS

- Conta o número de linhas na tabela especificada ou em uma tabela definida por uma expressão.
- Estrutura da fórmula: COUNTROWS (<tabela>)
- Esta fórmula conta todas as linhas de uma tabela, independentemente se há células ou colunas vazias. Diferentemente das duas anteriores que tratam a coluna e célula esta trata a tabela como um todo.
- A fórmula COUNTROWS é muito útil para tabelas em que cada linha representa um registro único, ou seja, se contarmos as linhas, teremos exatamente o resultado desejado.



# [ Função DISTINCTCOUNT DAX ]

## **DISTINCTCOUNT**

- Esta função conta o número de células diferentes em uma coluna de números.
- Estrutura da fórmula: DISTINCTCOUNT (<coluna>)
- Das funções DAX para contar esta é extremamente útil para identificar a quantidade de valores únicos. Numa coluna pode haver o mesmo valor repetindo 5, 10 ou mais vezes, ela contará este valor somente uma vez.
- É possível usar colunas que contêm qualquer tipo de dados. Quando a função não localizar nenhuma linha para contar, ela retornará BLANK, caso contrário, retornará a contagem de valores distintos.



# [ Leitura Recomendada ]

## **Visão geral de DAX**

<https://learn.microsoft.com/pt-br/dax/dax-overview>

## **Uso de funções iteradoras em DAX**

<https://learn.microsoft.com/pt-br/training/modules/dax-power-bi-iterator-functions/>

## **Referência de sintaxe DAX**

<https://learn.microsoft.com/pt-br/dax/dax-syntax-reference>

## **Referência de Funções DAX**

<https://learn.microsoft.com/pt-br/dax/dax-function-reference>

## **Operadores DAX**

<https://learn.microsoft.com/pt-br/dax/dax-operator-reference#operators-and-precedence-order>