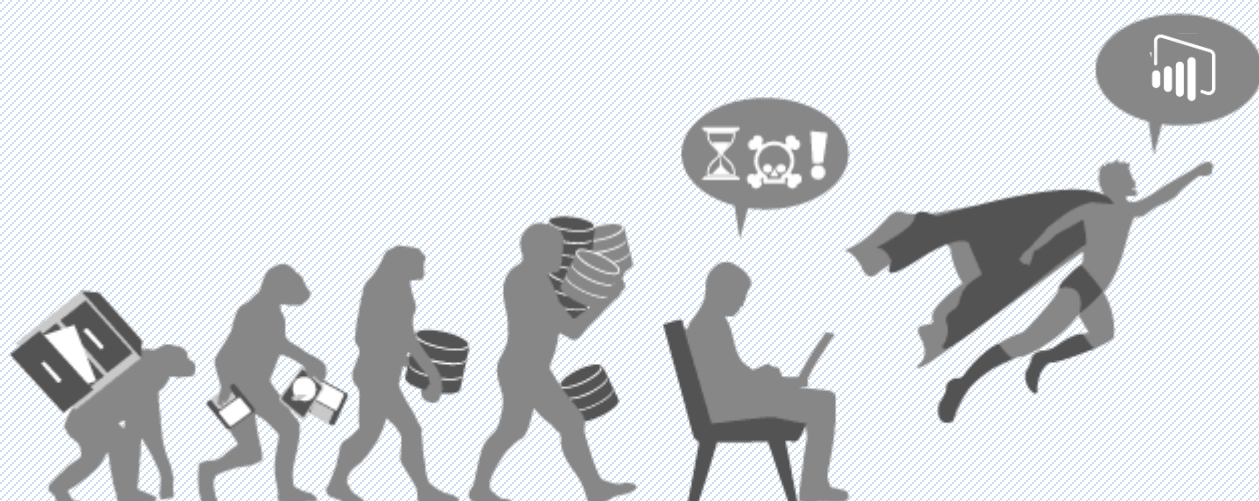
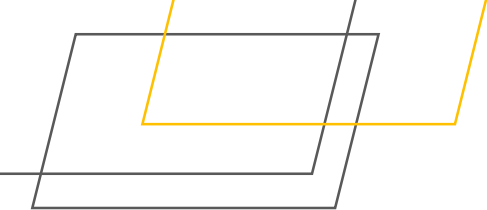


15 FUNÇÕES DAX PARA VOAR ALTO E FAZER A DIFERENÇA

Um guia das principais funções da linguagem DAX,
com exemplos práticos, descrições e explicações

Fernando Queiroz





_sobre a uaiSmart

A **uaiSmart** oferece uma abordagem diferenciada aos problemas relacionados à Inteligência de Negócios (BI) para qualquer tipo de empresa ou pessoa.

Oferecemos treinamentos, consultorias e soluções voltadas ao gerenciamento dos seus dados, com garantia de insights valiosos para tomadas de decisões assertivas.

Conheça mais em: uaismart.com





_sobre o eBook

Este eBook reúne as principais fórmulas DAX aplicadas ao Power BI de forma simples e prática.

A base de dados '*bdFinanceiro*' - que é utilizada nos exemplos - está disponível para download [clicando aqui](#).



Como aplicar esse eBook na prática?

Crie um novo arquivo no Power BI e conecte a base de dados '*bdFinanceiro*'. Depois é só criar as novas medidas utilizando as fórmulas dos exemplos.

Se estiver em dúvida sobre como conectar a base de dados ao seu relatório, confira um tutorial detalhado no nosso blog, [clicando aqui](#).

Os dados utilizados nos exemplos são fictícios e de propriedade da uaiSmart, devendo ser utilizados apenas para fins de aprendizagem.





15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

COUNTROWS

Descrição: Conta o número de linhas na tabela especificada ou em uma tabela definida por uma expressão.

Sintaxe: COUNTROWS(<tabela>)

Exemplo:

```
Total de Vendas = COUNTROWS (bdFinanceiro)
```

Valor retornado: 702

Explicação: Como cada registro de nossa base de dados equivale ao registro de uma venda, se contarmos a quantidade de linhas, estaremos contando a quantidade de vendas.

A fórmula COUNTROWS é muito útil para tabelas em que cada linha representa um registro único, ou seja, se contarmos as linhas, estaremos contando os registros.

É importante ressaltar que essa fórmula conta as linhas de toda a tabela e não de uma coluna específica.



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

COUNT

Descrição: Conta o número de células de uma coluna que contêm números.

Sintaxe: COUNT(<coluna>)

Exemplo:

```
Qtd de Operações =  
COUNT(bdFinanceiro[Unidades_Vendidas])
```

Valor retornado: 700

Explicação: A fórmula COUNT é muito parecida com a COUNTROWS, porém ela é mais usada com colunas que possuem mais de um tipo de valor. Ou seja, se uma mesma coluna tem campos de números e textos, essa função contará apenas a quantidade de células que estão preenchidas com números.

No nosso exemplo, o resultado é 700, pois temos duas células que estão sem preenchimento, ou seja, não são números.





15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

COUNTA

Descrição: Conta o número de células de uma coluna que não estão vazias. Ela conta não apenas as linhas contendo valores numéricos, mas também as linhas contendo valores não vazios, inclusive texto, datas e valores lógicos.

Sintaxe: COUNTA(<coluna>)

Exemplo:

```
Total de Registros =  
COUNTA (bdFinanceiro[País])
```

Valor retornado: 702

Explicação: Para finalizarmos as funções de Contagem, temos a função COUNTA. Ela basicamente conta todos os tipos de campo de uma coluna.

Diferente da função COUNT (que conta apenas números), a função COUNTA conta textos, datas, números, contanto que as células não estejam vazias.



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

MAX

Descrição: Retorna o maior valor numérico de uma coluna.

Sintaxe: MAX(<coluna>)

Exemplo:

```
Maior Preço de Venda =  
MAX(bdFinanceiro[Preço_Venda])
```

Valor retornado: 390

Explicação: A função MAX é utilizada quando se quer extrair o maior valor de uma coluna. No nosso exemplo, o maior valor para Preço de Venda.

Outras aplicações seria: Maior Preço de Custo, Maior Desconto, Maior Unidades Vendidas em uma única operação de Venda.

É possível apurar também o último dia disponível. Ou seja, se aplicarmos a função MAX na coluna Data, o valor retornado será a maior data da tabela.



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

MIN

Descrição: Retorna o menor valor numérico de uma coluna. Ignora valores lógicos e texto.

Sintaxe: MIN(<coluna>)

Exemplo:

```
Menor Qtd Vendida =  
MIN(bdFinanceiro[Unidades_Vendas])
```

Valor retornado: 200

Explicação: A função MIN é o oposto da função MAX. Ela retorna o menor valor de uma coluna.

No nosso exemplo, estamos buscando a menor quantidade vendida em uma operação.

Além do uso com números, essa função também pode ser utilizada para datas. Dessa forma, se precisar identificar a data mais antiga de uma coluna, basta aplicar a função MIN.





15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

SUM

Descrição: Adiciona todos os números de uma coluna.

Sintaxe: SUM(<coluna>)

Exemplo:

```
Quantidade Vendida =  
SUM(bdFinanceiro[Unidades_Vendas])
```

Valor retornado: 1.125.806

Explicação: Uma das fórmulas mais utilizadas em base de dados com valores é a função SUM. Ela soma todos os valores de uma coluna.

Então é possível totalizar as Quantidades Vendidas (como no exemplo), ou o Valor Total de Vendas, ou ainda o Valor Total de Desconto, entre outros.





15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

AVERAGE

Descrição: Retorna a média (aritmética) de todos os números de uma coluna.

Sintaxe: AVERAGE(<coluna>)

Exemplo:

```
Média do Preço de Custo =  
AVERAGE (bdFinanceiro[Preço_Custo])
```

Valor retornado: 168

Explicação: A fórmula AVERAGE calcula e retorna a média dos valores de uma coluna.

A aplicação dessa fórmula é bem útil para saber Preço Médio, Média de Vendas, Custo Médio dos Produtos, entre outras métricas.





15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

SUMX

Descrição: Retorna a soma de uma expressão avaliada para cada linha de uma tabela.

Sintaxe: SUMX(<tabela>, <expression>)

Exemplo:

```
Lucro = SUMX (bdFinanceiro;  
bdFinanceiro[Preço_Venda] -  
bdFinanceiro[Preço_Custo])
```

Valor retornado: 90.702

Explicação: A fórmula SUMX é um coringa na linguagem DAX. Supondo que você quer multiplicar/dividir uma coluna por outra, ou somar/subtrair uma por outra, ou ainda elevar uma coluna ao quadrado da outra, é só usar a função SUMX!

Ela recebe como parâmetro principal a <expressão>, na qual você informa qual operação deverá ser realizada em cada linha. Com isso, cria-se uma “coluna temporária” que é somada para retornar o resultado final.

No Exemplo, apuramos o Lucro subtraindo o Preço de Venda e o Preço de Custo em cada linha, e depois totalizando.



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

ALL

Descrição: Retorna todas as linhas de uma tabela ou todos os valores de uma coluna, ignorando qualquer filtro que possa ter sido aplicado. Esta função é útil para limpar filtros e criar cálculos em todas as linhas em uma tabela.

Sintaxe: ALL({<tabela> | <coluna>[, <coluna>[,...]]})

Exemplo:

```
% Represent Unid =  
SUMX (bdFinanceiro;bdFinanceiro[Unidades_V  
endidas]) / SUMX (ALL (bdFinanceiro);bdFinanç  
eiro[Unidades_Vendas])
```

Valor retornado: -

Explicação: A função ALL geralmente é utilizada dentro de um contexto, ou seja, dentro de uma fórmula. Ela faz com que a expressão calculada ignore todos os filtros aplicados no dashboard.

Isso significa que se você precisa calcular uma medida que não pode ser impactada pelos filtros disponíveis no relatório ou nos gráficos, é só usar a função ALL.

Geralmente usa-se para o cálculo de Representatividade ou valor Globais/Acumulados.



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

ALLEXCEPT

Descrição: Remove todos os filtros de contexto na tabela, exceto filtros que foram aplicados às colunas especificadas.

Sintaxe: ALLEXCEPT(<tabela>,<coluna>[,<coluna>[,...]])

Exemplo:

```
% Represent Unid 2 =  
SUMX (bdFinanceiro;bdFinanceiro[Unidades_V  
endidas]) / SUMX (ALLEXCEPT (bdFinanceiro;bdF  
inanceiro[País]);  
bdFinanceiro[Unidades_Vendidas])
```

Valor retornado: -

Explicação: A função ALLEXCEPT trabalha de forma similar à função ALL, ou seja, ela também ignora todos os filtros aplicados no relatório. A diferença é que com essa função é possível estabelecer colunas que podem ser filtradas e são tratadas como exceção.

No Exemplo, calculamos a representatividade das unidades vendidas, desconsiderando qualquer filtro, exceto o de País. Ou seja, filtros realizados na coluna "Segmento" não serão aplicados, mas filtros na coluna "País" afetarão os resultados.



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

FILTER

Descrição: Retorna uma tabela que representa um subconjunto de outra tabela ou expressão.

Sintaxe: FILTER(<tabela>,<filtro>)

Exemplo:

```
Vendas no Canadá =  
SUMX (FILTER (bdFinanceiro;  
bdFinanceiro[País]="Canadá");  
SUM (bdFinanceiro[Unidades_Vendidas]))
```

Valor retornado: 247.429

Explicação: A função FILTER é uma função que aplica filtros diretamente em uma tabela e retorna um valor.

Ela geralmente é usado dentro de uma outra fórmula (como SUM, SUMX, CALCULATE, etc).

No nosso Exemplo, estamos usando a FILTER para que a SUMX seja efetuada apenas quando o País for igual a "Canadá".



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

RELATEDTABLE

Descrição: Avalia uma expressão de tabela em um contexto modificado pelos filtros fornecidos.

Sintaxe: RELATEDTABLE(<tableName>)

Exemplo:

```
Vendas na Europa =  
SUMX (FILTER (RELATEDTABLE (Continentes);  
Continentes[Continente]="Europa"); [.Quant  
idade Vendida])
```

Valor retornado: 442.425

Explicação: Para aplicarmos essa função, importamos a tabela "Continentes" para buscamos as informações que precisamos.

Quando você precisa aplicar uma fórmula utilizando dados de uma outra tabela relacionada, é que se usa a RELATEDTABLE.

No nosso exemplo, precisamos buscar a informação de "Continente" que não está na tabela principal. Então usamos a função RELATEDTABLE para informar ao Power BI que vamos buscar uma informação de outra tabela.



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

RANKX

Descrição: Retorna a classificação de um número em uma lista de números para cada linha no argumento 'tabela'.

Sintaxe: RANKX(<tabela>, <expressao>)

Exemplo:

```
Ranking Vendas =  
RANKX (ALL (bdFinanceiro [Produto]); [.Quantidade Vendida])
```

Valor retornado: Ranking dos produtos vendidos

Explicação: A fórmula RANKX avalia uma expressão (que pode ser SUM, AVERAGE, COUNTROWS, etc) e retorna um ranking.

No Exemplo, estamos pedindo o ranking das Quantidades Vendidas na tabela bdFinanceiro.

Obs: aplicamos o filtro ALL para que o Ranking não seja alterado por filtros na coluna Produto.



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

CALCULATE

Descrição: Avalia uma expressão em um contexto que é modificado pelos filtros especificados.

Sintaxe: CALCULATE(<expressao>,<filtro1>,<filtro2>...)

Exemplo:

```
Quantidade Vendida Global =  
CALCULATE(SUM(bdFinanceiro[Unidades_Vendi  
das]); ALL(bdFinanceiro))
```

Valor retornado: 1.125.806

Explicação: A função CALCULATE é muito utilizada no Power BI, pois ela permite calcular diversas expressões (SUM, AVERAGE, COUNTROWS, MAX, MIN, entre outros) considerando filtros pré-definidos.

Ou seja, é possível estabelecer em qual contexto você quer que a expressão seja calculada.

No exemplo, estamos calculando a soma das Unidades Vendidas ignorando qualquer filtro na tabela bdFinanceiro. Então, aplicamos a operação matemática que queremos fazer (SUM) e o filtro que deve ser aplicado (ALL).

Assim, temos a quantidade global de vendas.



15 FÓRMULAS DAX QUE TODO ANALISTA DEVE SABER

IF

Descrição: Verifica se uma condição fornecida como o primeiro argumento foi atendida. Retorna um valor se a condição é Verdadeiro; retorna outro valor se a condição é Falso.

Sintaxe: IF(teste_logico, valor_verdadeiro, valor_se_falso)

Exemplo:

```
Top 3 Ranking = IF([.Ranking Vendas]<=3;  
[.Ranking Vendas];BLANK())
```

Valor retornado: -

Explicação: Se você já trabalhou com o Excel, sabe que a função SE é muito aplicada para validação de informações e criação de colunas auxiliares.

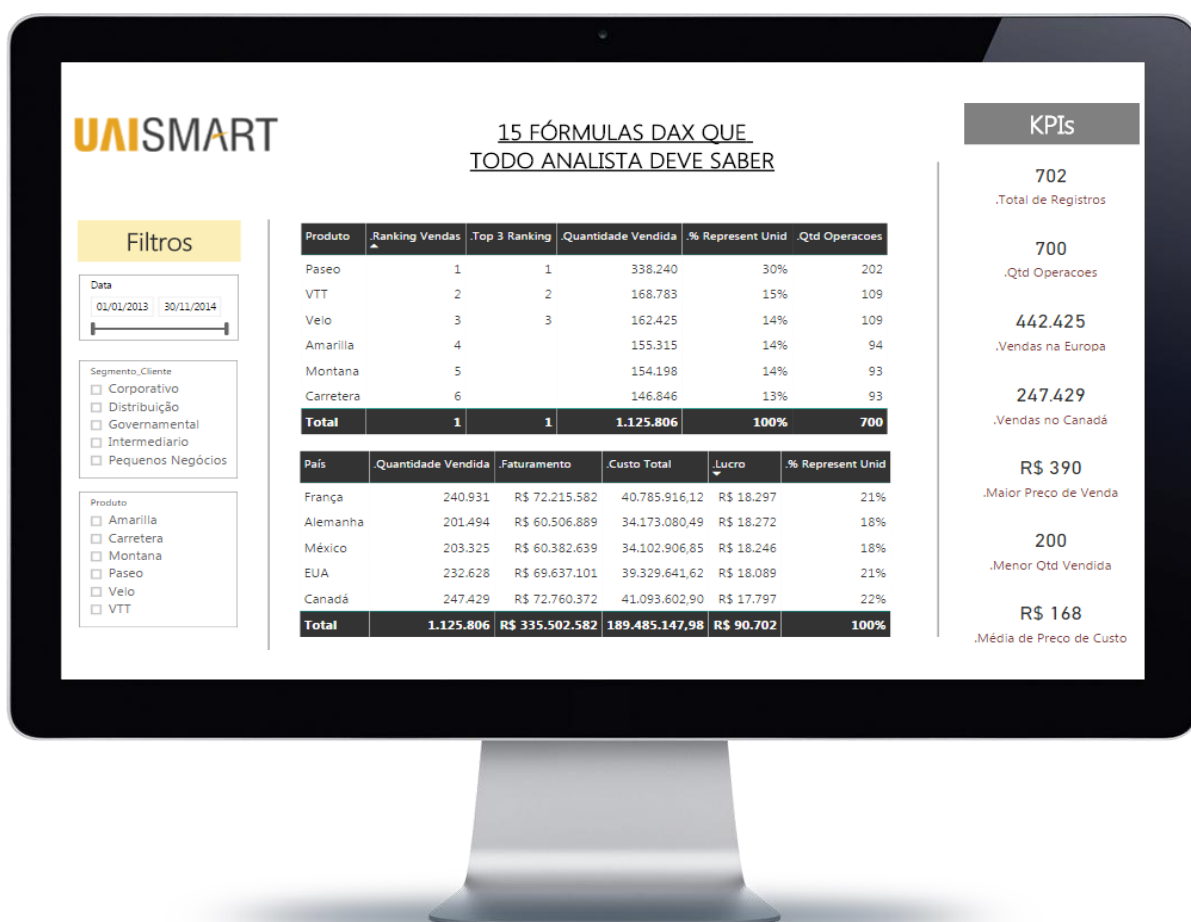
No Power BI, a função IF também tem esse objetivo. Ela realiza um teste lógico e retorna valores Verdadeiros/Falso.

No Exemplo, queremos extrair apenas os 3 primeiros itens do nosso Ranking de Vendas. Para isso, aplicamos a função IF para verificar "se o Ranking de Vendas for menor ou igual a 3, retornará o Ranking de Vendas, caso contrário, retornará uma célula em branco (Blank)".

E essas foram **as 15 fórmulas mais utilizadas** com a **linguagem DAX**. Com elas você conseguirá fazer boa parte dos seus trabalhos de **transformação de dados em insights!**

E para ter acesso ao **relatório** que construímos no **Power BI** utilizando essas fórmulas, é só [clique aqui](#) para baixar!

Acompanhe outros tutoriais e dicas pelo nosso [blog](#) e confira nosso portfólio de soluções para seu negócio!



UAI SMART

© 2017