# O Que é DAX?

A Linguagem DAX é usada para criar cálculos avançados e análises de dados no Power BI. Ela permite que você utilize funções semelhantes às do Excel, como soma, máximo, mínimo e contagem, porém, todas em inglês.

## Conceitos Fundamentais

DAX não é apenas sobre decorar funções; é crucial entender o conceito de contexto de avaliação, que inclui contexto de filtro e contexto de linha. Este conceito é a base para realizar cálculos avançados com poucas linhas de código.

## Funções Essenciais

Aprender DAX não significa decorar todas as funções existentes. Vamos focar nas funções essenciais que permitirão você realizar cálculos avançados. Um guia de funções, como o disponível em dax.guide, é um recurso valioso para consultar sempre que necessário.

## Exemplos de Funções em DAX

Vamos explorar funções de diferentes categorias:

1. \*\*Agregação\*\*:  
- `SUM`: Soma os valores de uma coluna.  
- `AVERAGEX`: Calcula a média de uma expressão avaliada em uma tabela.

2. \*\*Data e Hora\*\*:  
- `DATE`: Cria uma data a partir de ano, mês e dia.  
- `YEAR`, `MONTH`, `DAY`: Extrai partes específicas de uma data.

3. \*\*Texto\*\*:  
- `CONCATENATE`: Junta duas ou mais strings.  
- `SEARCH`, `FIND`: Localiza a posição de uma substring dentro de um texto.

4. \*\*Inteligência de Tempo\*\*:  
- `SAMEPERIODLASTYEAR`: Compara valores do mesmo período no ano anterior.  
- `DATESYTD`: Calcula o acumulado no ano até a data atual.

## Manipulação de Contexto

A função `CALCULATE` é uma das mais poderosas em DAX, permitindo modificar o contexto de filtro de uma expressão. Por exemplo:

CALCULATE(SUM(Vendas[Valor]), Vendas[Categoria] = 'Eletrônicos')

Aqui, calculamos a soma das vendas, mas filtramos apenas as vendas de eletrônicos.

## Power Query vs. DAX

Uma dúvida comum é sobre quando usar Power Query e quando usar DAX. Em geral:

- \*\*Power Query\*\*: Ideal para ETL (Extração, Transformação e Carga de dados). Use-o para ajustar e preparar seu modelo de dados.

- \*\*DAX\*\*: Use para criar medidas dinâmicas e análises avançadas. Evite usar DAX para tarefas de ETL.

## SQL e DAX

Se você trabalha com SQL, a ordem de processamento recomendada é:

1. \*\*SQL\*\*: Realize o tratamento inicial e a manipulação dos dados.

2. \*\*Power Query\*\*: Faça ajustes e transformação adicionais.

3. \*\*DAX\*\*: Utilize para criar medidas e análises avançadas.

## Importância de um Modelo Bem Estruturado

Um modelo de dados bem estruturado facilita o uso de DAX. Evite usar DAX para corrigir problemas de modelagem ou ETL. Faça essas correções no Power Query ou no SQL antes de passar para DAX.