

Data Science
Academy

Data Science
Academy

Formação Desenvolvedor Microsoft para Data Science e Inteligência Artificial

Power BI Avançado para Análise de Dados



Capítulo 3

Importação e Transformação de Dados com PowerQuery e Linguagem M - Parte 2





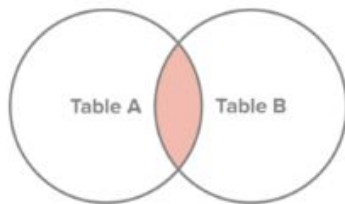
Conteúdo

- Transformações Append e Mesclagens
- Execuções de Scripts em Python e R
- Instalação do Microsoft SQL Server e SSMS
- Modos de Conectividade Import e Direct Query
- Programação utilizando Linguagem M

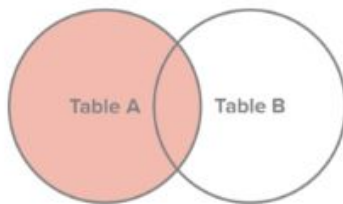


Joins (Mesclagem)

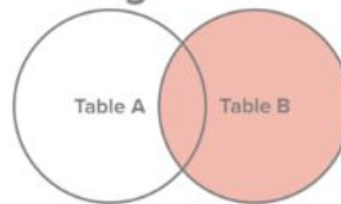
Inner Join



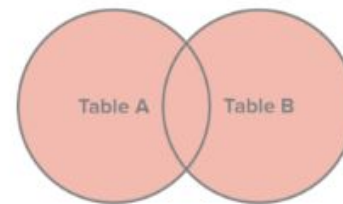
Left Join



Right Join



Full Join



<http://www.sql-join.com/sql-join-types>



Power Query

- Power Query é uma tecnologia de Conectividade e Preparação de Dados.
- Possibilita acesso a centenas de fontes de dados diferentes.
- Possui mais de 300 funcionalidades para limpeza e transformação de dados sem necessidade de utilização de código.
- Através do Custom Connectors SDK, possibilita que terceiros criem conectores para suas próprias fontes de dados.
- Operações realizadas através de interface gráfica do Power Query será automaticamente transformada em código da Linguagem M.



Modos de Conectividade

A screenshot of a software dialog box titled "SQL Server Database" with a close button (X) in the top right corner. The dialog contains the instruction "Import data from a SQL Server database." Below this, there are two text input fields: "Server" and "Database (optional)". Under the input fields, there are two radio button options: "Import" (which is selected) and "DirectQuery". A bright pink arrow points from the right side of the dialog towards the "Import" radio button. Below the radio buttons is a link labeled "Advanced options" with a right-pointing arrow. At the bottom right of the dialog are two buttons: "OK" (yellow) and "Cancel" (gray).

SQL Server Database

Import data from a SQL Server database.

Server

Database (optional)

☒ Import

☐ DirectQuery

Advanced options

OK Cancel





Modos de Conectividade: Import

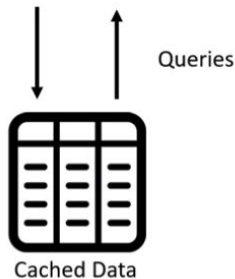
- Realiza a importação dos dados para o cache do PowerBI;
- Dados podem vir de diversas fontes;
- Recomendado quando os dados possuem tamanho menor que 1 Gb;
- Possível limitação na publicação de relatórios: 1 Gb no Power BI Serviço com uma conta Pro e 10 Gb no Power BI Premium;
- Possibilita a atualização dos dados;
- Processamento realizado localmente, em memória;
- Alta performance através do Query Engine;





Modos de Conectividade: Import

Import





Modos de Conectividade: Direct Query

- Dados não são importados para o Power BI;
- Dados vem de uma única fonte*
 - Modelos compostos;
- Performance depende da fonte de dados responsável pela consulta;
- Não suporta todas transformações possíveis e funções DAX como funções de Time Intelligence;
- Processamento realizado no Servidor de Banco de Dados;
- Fontes de dados com suporte a Direct Query no Power BI:

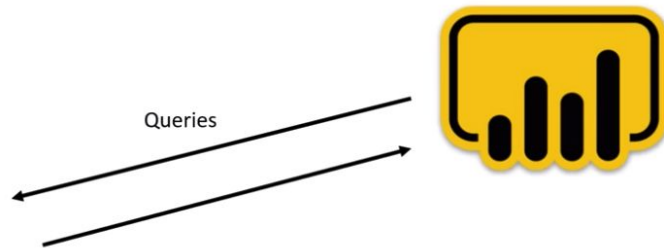
<https://docs.microsoft.com/pt-br/power-bi/desktop-directquery-data-sources>





Modos de Conectividade: Direct Query

DirectQuery



Guy in a Cube
Power BI Get Data: Import vs Direct Query vs Live

<https://www.youtube.com/watch?v=y4S-k34qmnw>





Import vs Direct Query

| Característica | Import | Direct Query |
|-----------------------------------|--------|--------------|
| Melhor Performance | V | |
| Melhor Experiência de Projeto | V | |
| Disponibilidade da Fonte de Dados | V | |
| Atualização dos Dados | | V |
| Tamanho da Fonte de Dados | | V |





Linguagem M





Power Query Formula Language ou Linguagem M

- Linguagem funcional sensível para o Power Query;
- Importante aprendermos “M” para não ficarmos limitados a interface gráfica do Power Query;
- Uma função realiza o mapeamento de um conjunto de dados de entrada para um conjunto de dados de saída;
- Classes de funções:

<https://docs.microsoft.com/en-us/powerquery-m/power-query-m-function-reference>





Linguagem M

- Bloco “**let**” onde ocorre as definições das funções e variáveis;
- Bloco “**in**” que é a saída da sua consulta;
- O corpo da função fica após o símbolo =>
- Declaração de tipos de dados dos parâmetros e de retorno da função são opcionais (declaração explícita X implícita);
- Parâmetros e tipos de retorno implícito serão do tipo “**any**”, sendo que todos os tipos de dados em “M” derivam do tipo “**any**”.





Parâmetros explícitos

```
let
    Multiplicacao = (x as number, y as number) as number => x * y,
    // Comentários podem ser realizados utilizando 2 barras
    CalcularMultiplicacao = Multiplicacao(3,5)
in
    CalcularMultiplicacao
```





Parâmetros implícitos

```
let
    Multiplicacao = (x, y) => x * y,
    CalcularMultiplicacao = Multiplicacao(3,5)
in
    CalcularMultiplicacao
```





Tipos Primitivos

| | |
|-----------|------------------|
| Number | 0, 5, -3, 4.5 |
| Text | "Power BI" |
| Date | 15/04/2019 |
| Date Time | 15/04/2019 14:00 |
| Logical | true, false |

Documentação:

<https://docs.microsoft.com/en-us/powerquery-m/expressions-values-and-let-expression>





Function Value

```
let
  Multiplicacao = (x as number, y as number) as number => x * y,
  CalcularMultiplicacao = Multiplicacao(3,5)
in
  CalcularMultiplicacao
```





List

- Sequência de valores;
- Definida com utilização de chaves;
- Zero-based index - o primeiro elemento da lista é acessado pelo índice 0;
- Uma lista pode ter valores de diferentes tipos;

```
{ 1, 15, 60, true, {20,30,40} }
```

Documentação:

<https://docs.microsoft.com/en-us/powerquery-m/expressions-values-and-let-expression>



Record

- Conjunto de campos, definidos por um par chave-valor;
- Definido com utilização de colchetes;

[

Nome = "João",

Curso = "Power BI",

Nota = 9.3

]

Documentação:

<https://docs.microsoft.com/en-us/powerquery-m/expressions-values-and-let-expression>



Table

- Conjunto de valores organizados em linhas e colunas

```
#table(  
  {"Código", "Nome", "Nota"},  
  {  
    {1, "Márcio", 8},  
    {2, "Simone", 9}  
  })
```

Documentação:

<https://docs.microsoft.com/en-us/powerquery-m/expressions-values-and-let-expression>



Linguagem M

- O corpo da função fica após o símbolo \Rightarrow
- Bloco “**let**” onde ocorre as definições das funções e variáveis;
- Bloco “**in**” que é a saída da sua consulta;
- Declaração de tipos de dados dos parâmetros e de retorno da função são opcionais (declaração explícita X implícita);
- Parâmetros e tipos de retorno implícito serão do tipo “**any**”, sendo que todos os tipos de dados em “M” derivam do tipo “**any**”.



Obrigado.

Tenha uma excelente jornada de aprendizagem!



Data Science
Academy