

# POWER BI

Domine conexão com a força M Language



Filipe Campanati

# Guia Prático

## Explorando e Dominando Conexões de Dados no Power BI

O Power BI é uma ferramenta poderosa de visualização de dados que permite a criação de relatórios e dashboards dinâmicos. Uma das suas funcionalidades mais importantes é a capacidade de se conectar a diversas fontes de dados e manipular esses dados através de fluxos eficientes. Neste guia, vamos explorar os principais fluxos de dados e conexões no Power BI, incluindo exemplos práticos de códigos e configurações.



# 01

## Conexão com Banco de Dados SQL

---

Configurando a Conexão, conectar-se a um banco de dados SQL é uma das formas mais comuns de importar dados para o Power BI.



# Configurando a Conexão



Abaixo estão os passos para configurar essa conexão:

1. Abra o Power BI **Desktop**.
2. Clique em "Obter Dados".
3. Selecione "Banco de dados SQL Server".
4. Insira o nome do servidor e o banco de dados.
5. Se necessário, insira as credenciais de acesso.

## Exemplo de Código M



```
1 let
2     Fonte = Sql.Database("NomeDoServidor", "NomeDoBancoDeDados"),
3     Consulta = Fonte{[Schema="dbo", Item="NomeDaTabela"]}[Data]
4 in
5     Consulta
```



# 02

## Conexão com um Arquivo Excel

---

O Excel é outra fonte comum de dados. Veja como conectar-se a um arquivo Excel



# Importando Dados de Excel



Conectar-se a um arquivo Excel no Power BI é um processo simples, que envolve alguns passos intuitivos. Aqui estão os passos detalhados:

1. **Abrir o Power BI Desktop:** Comece abrindo o Power BI Desktop no seu computador.
2. **Clique em "Obter Dados":** No painel inicial, clique em "Obter Dados" para ver uma lista de possíveis fontes de dados.
3. **Selecione "Excel":** Escolha a opção "Excel" da lista de fontes de dados.
4. **Navegue até o Arquivo:** Use o explorador de arquivos para localizar o arquivo Excel que deseja usar e selecione-o.
5. **Escolha as Tabelas ou Planilhas:** Depois de selecionar o arquivo, você verá uma lista de tabelas e planilhas disponíveis no arquivo. Escolha aquelas que você deseja importar para o Power BI.

## Exemplo de Código M

```
1 let
2     Fonte = Excel.Workbook(File.Contents("Caminho/Para/Seu/Arquivo.xlsx"), null, true),
3     Planilha1_Tabela = Fonte[Item="Planilha1",Kind="Sheet"] [Data],
4     ColunasPromovidas = Table.PromoteHeaders(Planilha1_Tabela, [PromoteAllScalars=true])
5 in
6     ColunasPromovidas
```

Neste exemplo:

- **Fonte:** Especifica o caminho do arquivo Excel.
- **Planilha1\_Tabela:** Refere-se à planilha específica no arquivo Excel.
- **ColunasPromovidas:** Promove a primeira linha da planilha a cabeçalhos de coluna, se ainda não estiver configurado.

# Configurações Adicionais



Depois de importar os dados, você pode precisar realizar configurações adicionais para preparar os dados para análise:

- **Renomear Colunas:** Certifique-se de que todas as colunas têm nomes significativos.
- **Remover Dados Nulos:** Limpe os dados para remover entradas nulas ou duplicadas.
- **Alterar Tipos de Dados:** Verifique se os tipos de dados estão corretos para cada coluna (texto, número, data, etc.).

## Exemplo de Código M

```
1 let
2     Fonte = Excel.Workbook(File.Contents("Caminho/Para/Seu/Arquivo.xlsx"), null, true),
3     Planilha1_Tabela = Fonte[Item="Planilha1",Kind="Sheet"][Data],
4     ColunasPromovidas = Table.PromoteHeaders(Planilha1_Tabela, [PromoteAllScalars=true]),
5     TiposAlterados = Table.TransformColumnTypes(ColunasPromovidas, {"Data", type date}, {"Valor", type number})
6 in
7     TiposAlterados
```



# 03

## Conexão com Serviços Online

---

Acessar dados armazenados no SharePoint





# Conectando ao SharePoint



O SharePoint é uma ferramenta amplamente utilizada para armazenar e compartilhar documentos dentro de organizações. Aqui está um guia detalhado sobre como fazer essa conexão:

1. Abrir o Power BI Desktop
2. Clique em "Obter Dados"
3. Selecione "SharePoint Folder": Na lista de fontes de dados, escolha "SharePoint Folder".
4. Insira a URL do SharePoint: Digite a URL do site SharePoint onde os dados estão armazenados.
5. Autentique-se: Se necessário, insira suas credenciais de acesso ao SharePoint.

## Exemplo de Código M

```
1 let
2     Fonte = SharePoint.Contents("https://SeuSite.sharepoint.com/sites/SeuSite"),
3     Pasta = Fonte[Name="Documentos"][Content],
4     Arquivo = Pasta[Name="SeuArquivo.xlsx"][Content],
5     Excel = Excel.Workbook(Arquivo),
6     Planilha = Excel[Item="Planilha1",Kind="Sheet"][Data],
7     Tabela = Table.PromoteHeaders(Planilha, [PromoteAllScalars=true])
8 in
9     Tabela
```

Neste exemplo:

- **Fonte:** Define a URL do site SharePoint.
- **Pasta:** Navega até a biblioteca de documentos específica.
- **Arquivo:** Seleciona o arquivo Excel dentro da biblioteca.
- **Excel:** Carrega o conteúdo do arquivo Excel.
- **Planilha:** Seleciona a planilha desejada no arquivo Excel.
- **Tabela:** Promove a primeira linha a cabeçalhos de coluna, se necessário.

# Configurações Adicionais



Após a importação, você pode precisar ajustar os dados para melhor visualização e análise:

- **Filtragem de Dados:** Remova linhas ou colunas desnecessárias.
- **Transformação de Dados:** Altere os tipos de dados e aplique fórmulas conforme necessário.
- **Atualizações Automáticas:** Configure a atualização automática para manter os dados atualizados.

## Exemplo de Código M

```
1  let
2    Fonte = SharePoint.Contents("https://SeuSite.sharepoint.com/sites/SeuSite"),
3    Pasta = Fonte[Name="Documentos"][Content],
4    Arquivo = Pasta[Name="SeuArquivo.xlsx"][Content],
5    Excel = Excel.Workbook(Arquivo),
6    Planilha = Excel[Item="Planilha1",Kind="Sheet"][Data],
7    Tabela = Table.PromoteHeaders(Planilha, [PromoteAllScalars=true]),
8    LinhasFiltradas = Table.SelectRows(Tabela, each [Status] = "Ativo"),
9    TiposAlterados = Table.TransformColumnTypes(LinhasFiltradas, {"Data", type date}, {"Valor", type number})
10 in
11    TiposAlterados
```



# 04

## Conexão com APIs Web

---

Conectar-se a uma API permite a integração de dados em tempo real.



# Importando Dados de uma API



Conectar-se a uma API permite a integração de dados em tempo real, trazendo dados dinâmicos diretamente para o Power BI. Esse tipo de conexão é especialmente útil para acessar dados que estão sempre mudando, como informações meteorológicas, financeiras ou de redes sociais. Aqui estão os passos para configurar uma conexão com uma API:

1. Clique em "Obter Dados".
2. **Selecione "API da Web"**: Na lista de fontes de dados, escolha "API da Web".
3. **Insira a URL da API**: Digite a URL da API que deseja usar.
4. **Autentique-se**: Se necessário, insira suas credenciais de acesso à API.

O código M abaixo exemplifica como importar dados de uma API JSON:

## Exemplo de Código M

```
1 let
2     Fonte = Json.Document(Web.Contents("https://api.exemplo.com/dados")),
3     Dados = Fonte[results],
4     Tabela = Table.FromList(Dados, Splitter.SplitByNothing(), null, null, ExtraValues.Error)
5 in
6     Tabela
```

Neste exemplo:

- **Fonte**: Define a URL da API.
- **Dados**: Extrai a lista de resultados da resposta JSON.
- **Tabela**: Converte a lista de dados em uma tabela para fácil manipulação no Power BI.



# Configurações Adicionais



Após importar os dados da API, você pode precisar realizar algumas configurações adicionais:

- **Filtragem de Dados:** Aplique filtros para mostrar apenas os dados relevantes.
- **Transformação de Dados:** Altere os tipos de dados e aplique fórmulas conforme necessário.
- **Atualizações Automáticas:** Configure a atualização automática para garantir que os dados estejam sempre atualizados.

## Exemplo de Código M

```
1 let
2     Fonte = Json.Document(Web.Contents("https://api.exemplo.com/dados")),
3     Dados = Fonte[results],
4     Tabela = Table.FromList(Dados, Splitter.SplitByNothing(), null, null, ExtraValues.Error),
5     Expandidos = Table.ExpandRecordColumn(Tabela, "Column1", {"SubCol1", "SubCol2"}, {"SubCol1", "SubCol2"}),
6     TiposAlterados = Table.TransformColumnTypes(Expandidos, {"SubCol1", type text}, {"SubCol2", type number})
7 in
8     TiposAlterados
```



# 05

# Transformação de Dados com Power Query

Transformar dados é essencial para preparar seus dados para análise.



# Exemplos de Transformações Comuns



O Power Query oferece uma ampla gama de ferramentas para transformar dados de maneira eficiente e intuitiva. A seguir, apresentamos alguns exemplos básicos de transformações comuns:

## Remover Colunas Desnecessárias:

Remover colunas que não são necessárias para a análise pode simplificar seu conjunto de dados e melhorar o desempenho do Power BI.

m

```
1 let
2     Fonte = Csv.Document(File.Contents("Caminho/Para/Seu/Arquivo.csv"), [Delimiter=",", Columns=5, Encoding=1252, QuoteStyle=QuoteStyle.None]),
3     RemoverColunas = Table.RemoveColumns(Fonte,{"Coluna1", "Coluna2"})
4 in
5     RemoverColunas
```

## Filtrar Linhas:

Filtrar linhas com base em condições específicas ajuda a focar apenas nos dados relevantes.

m

```
1 let
2     Fonte = Csv.Document(File.Contents("Caminho/Para/Seu/Arquivo.csv"), [Delimiter=",", Columns=5, Encoding=1252, QuoteStyle=QuoteStyle.None]),
3     LinhasFiltradas = Table.SelectRows(Fonte, each [Coluna3] > 100)
4 in
5     LinhasFiltradas
```

## Alterar Tipos de Dados:

Assegure-se de que cada coluna tem o tipo de dado correto para evitar erros na análise e na visualização.

m

```
1 let
2     Fonte = Csv.Document(File.Contents("Caminho/Para/Seu/Arquivo.csv"), [Delimiter=",", Columns=5, Encoding=1252, QuoteStyle=QuoteStyle.None]),
3     TiposAlterados = Table.TransformColumnTypes(Fonte, {"Coluna4", type number}, {"Coluna5", type date})
4 in
5     TiposAlterados
6
```

# Combinação de Dados



Combinar dados de diferentes fontes pode ser necessário para criar uma visão mais completa. O Power Query permite várias formas de combinar dados, como junções e mesclagens.

## Mesclar Consultas

Mesclar consultas permite combinar tabelas com base em colunas comuns.

m

```
1 let
2     Fonte1 = Csv.Document(File.Contents("Caminho/Para/Seu/PrimeiroArquivo.csv"), [Delimiter=";", Columns=5, Encoding=1252, QuoteStyle=QuoteStyle.None]),
3     Fonte2 = Csv.Document(File.Contents("Caminho/Para/Seu/SegundoArquivo.csv"), [Delimiter=";", Columns=5, Encoding=1252, QuoteStyle=QuoteStyle.None]),
4     Mesclado = Table.NestedJoin(Fonte1, {"Chave"}, Fonte2, {"Chave"}, "DadosCombinados", JoinKind.Inner),
5     Expandido = Table.ExpandTableColumn(Mesclado, "DadosCombinados", {"Coluna2", "Coluna3"}, {"Coluna2", "Coluna3"})
6 in
7     Expandido
```

## Anexar Consultas

Anexar consultas permite concatenar tabelas semelhantes para criar uma única tabela.

m

```
1 let
2     Fonte1 = Csv.Document(File.Contents("Caminho/Para/Seu/PrimeiroArquivo.csv"), [Delimiter=";", Columns=5, Encoding=1252, QuoteStyle=QuoteStyle.None]),
3     Fonte2 = Csv.Document(File.Contents("Caminho/Para/Seu/SegundoArquivo.csv"), [Delimiter=";", Columns=5, Encoding=1252, QuoteStyle=QuoteStyle.None]),
4     Anexado = Table.Combine({Fonte1, Fonte2})
5 in
6     Anexado
```

## Aplicação de Fórmulas Personalizadas

Criar colunas personalizadas usando fórmulas pode adicionar novos insights ao seu conjunto de dados.

m

```
1 let
2     Fonte = Csv.Document(File.Contents("Caminho/Para/Seu/Arquivo.csv"), [Delimiter=";", Columns=5, Encoding=1252, QuoteStyle=QuoteStyle.None]),
3     ColunaPersonalizada = Table.AddColumn(Fonte, "NovaColuna", each [Coluna4] * 2)
4 in
5     ColunaPersonalizada
```



# AGRADECIMENTOS

---



# Obrigado por ler até aqui



Este Ebook Guia foi criado com fins didáticos durante o bootcamp, utilizando IAs para coleta de informações. Todo o conteúdo foi revisado e diagramado por mim.

Este guia apresentou as principais conexões de dados e fluxos que você pode utilizar no Power BI, com exemplos práticos para facilitar a implementação. A capacidade de se conectar a diversas fontes de dados e transformar esses dados de maneira eficiente é o ponto inicial e mais importante para criar relatórios e dashboards.

