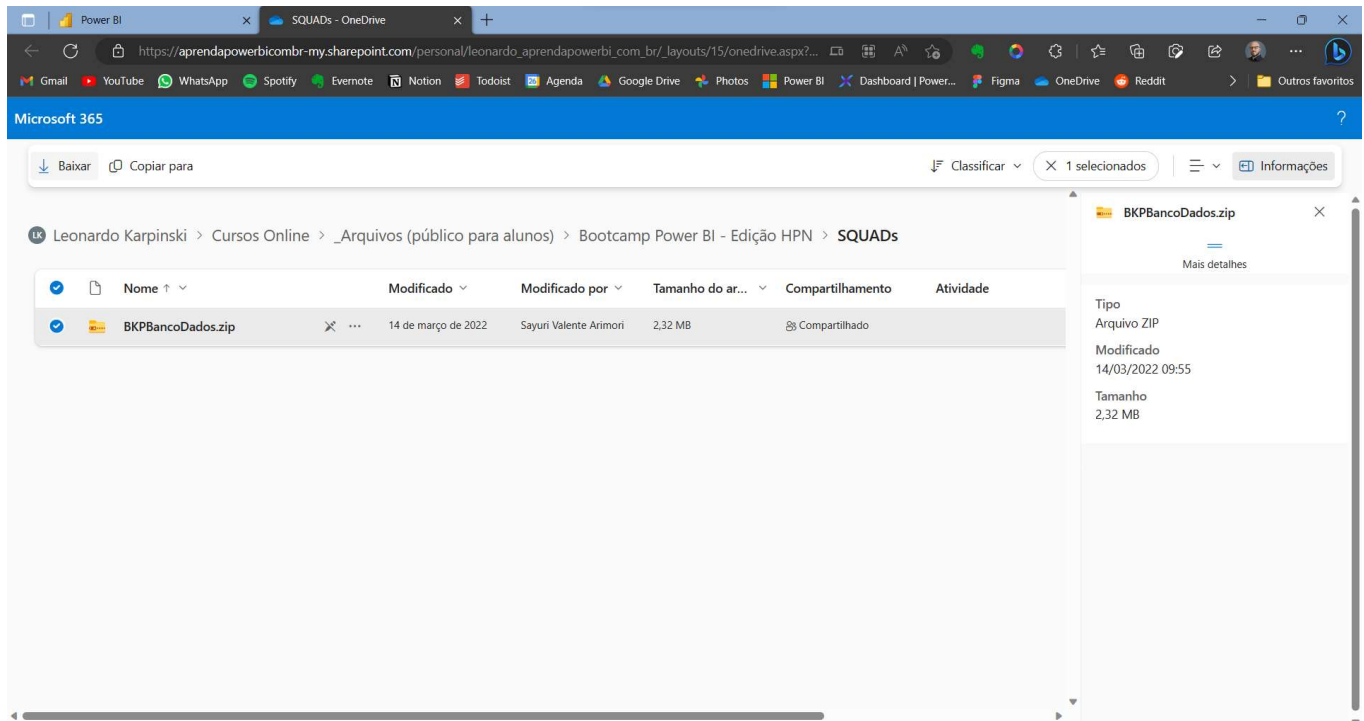


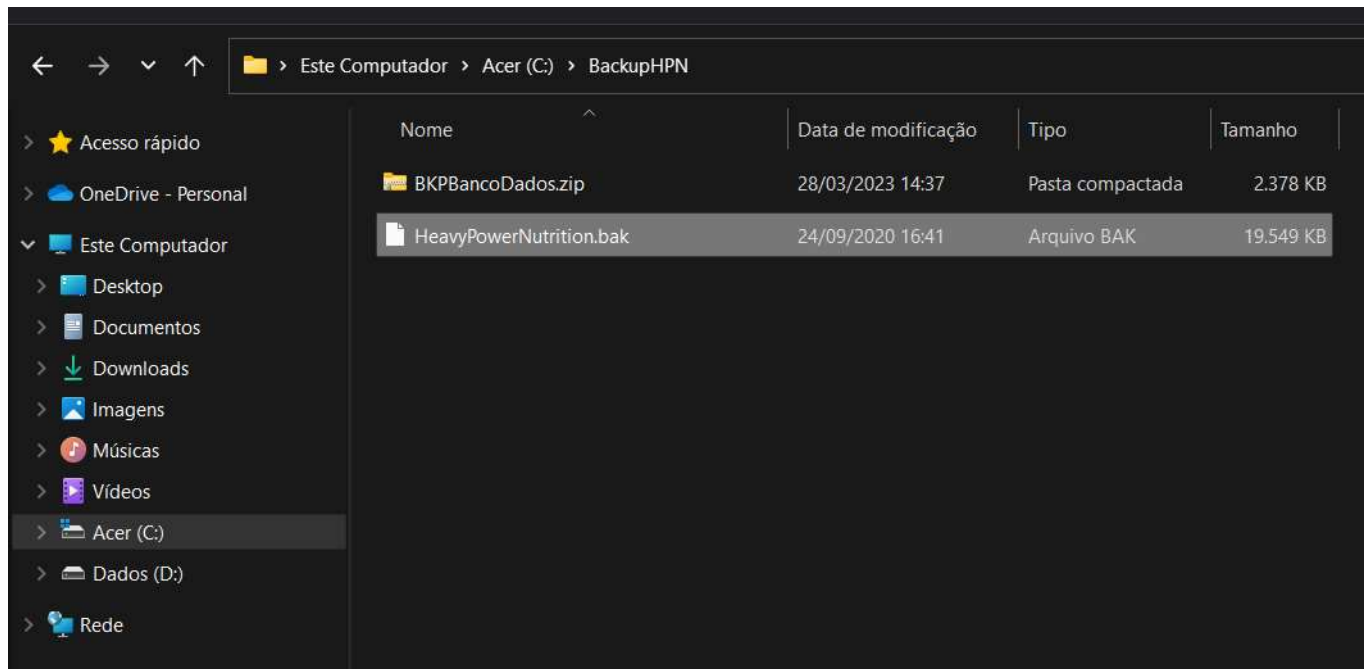
Procedimento

Download

Foi efetuado o download do backup do banco de dados zipado no arquivo "BKPBancoDados.zip" disponibilizado no [Sharepoint](#) .



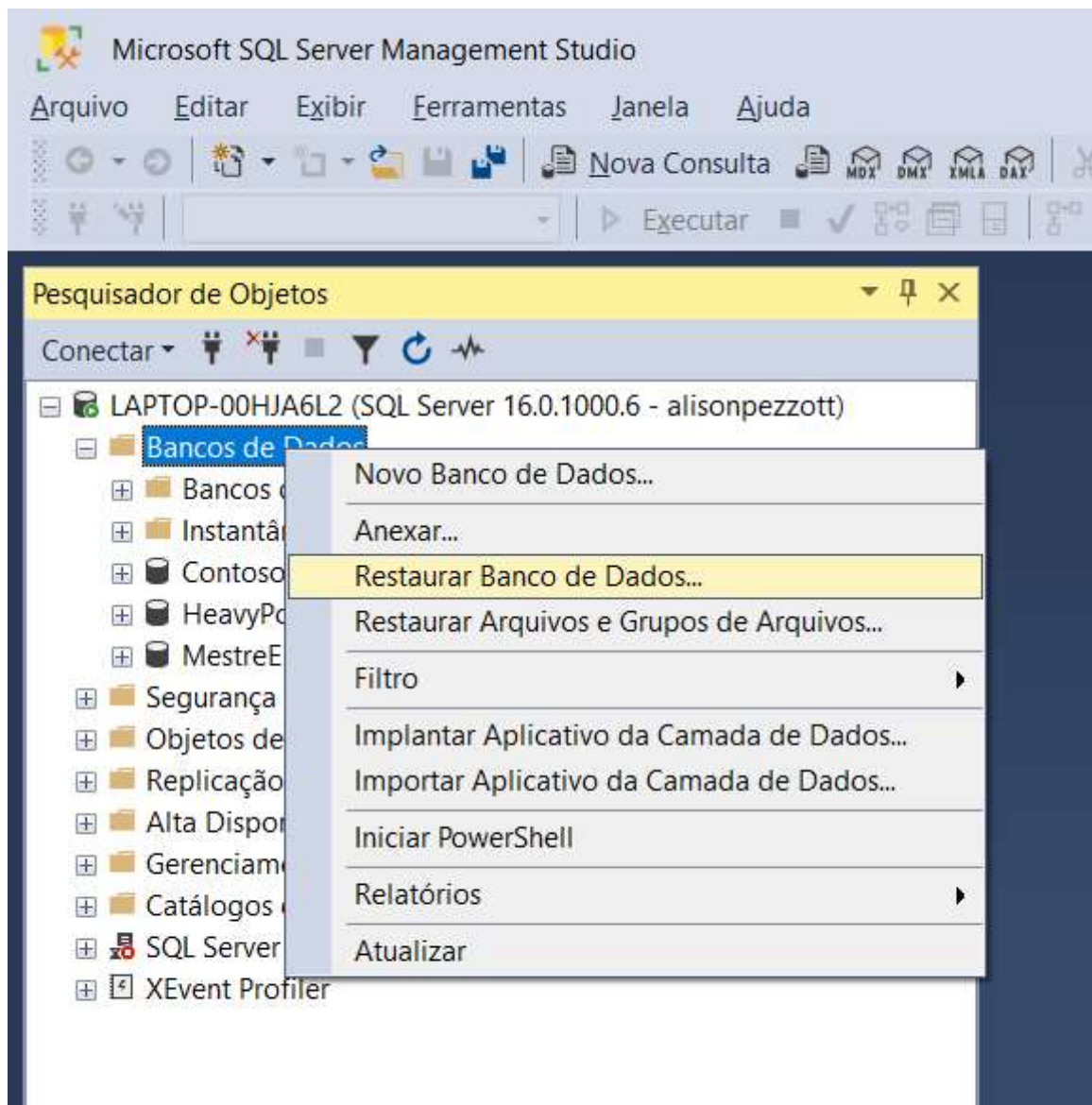
Após o download, foi efetuada a descompactação do arquivo zipado e o arquivo "HeavyPowerNutrition.bak" foi extraído.



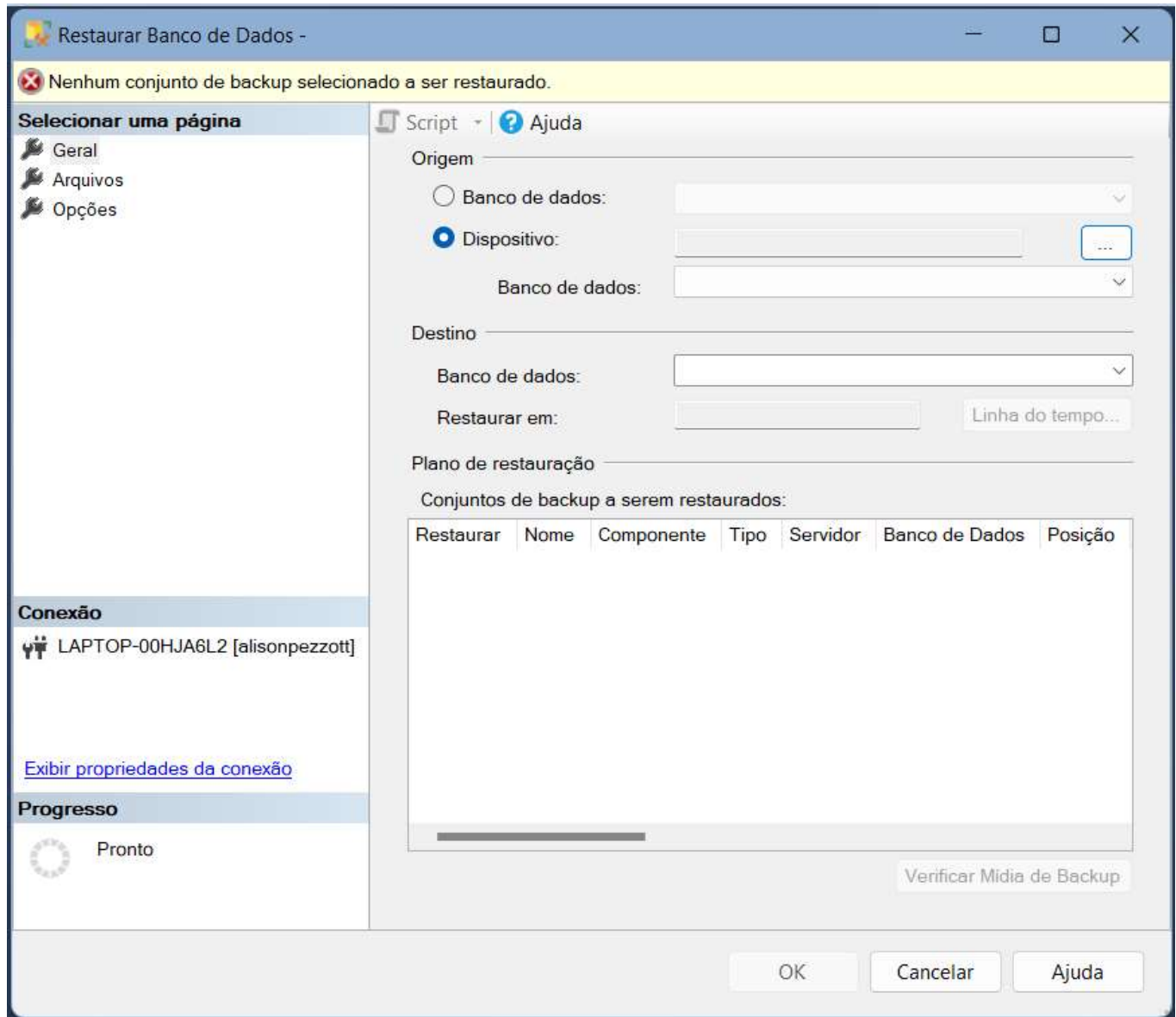
Banco de dados

Os passos a seguir são relacionados a etapa de restauração no banco de dados SQL Server.

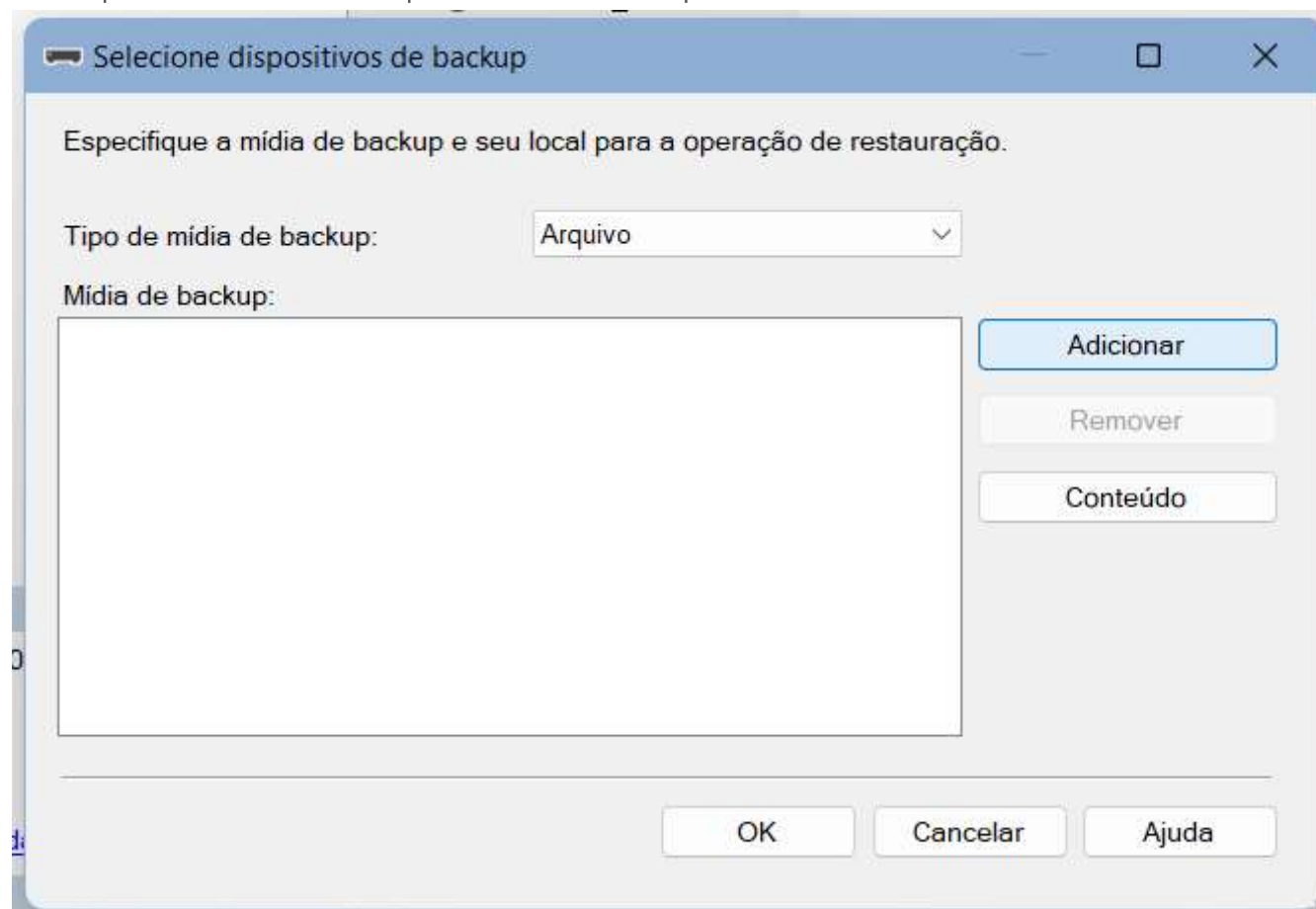
Utilizando o gerenciador do banco de dados **Microsoft SQL Server Managment Studio**", acessando no meu lateral foi clicado com o botão direito em Banco de Dados e depois Restaurar Banco de Dados



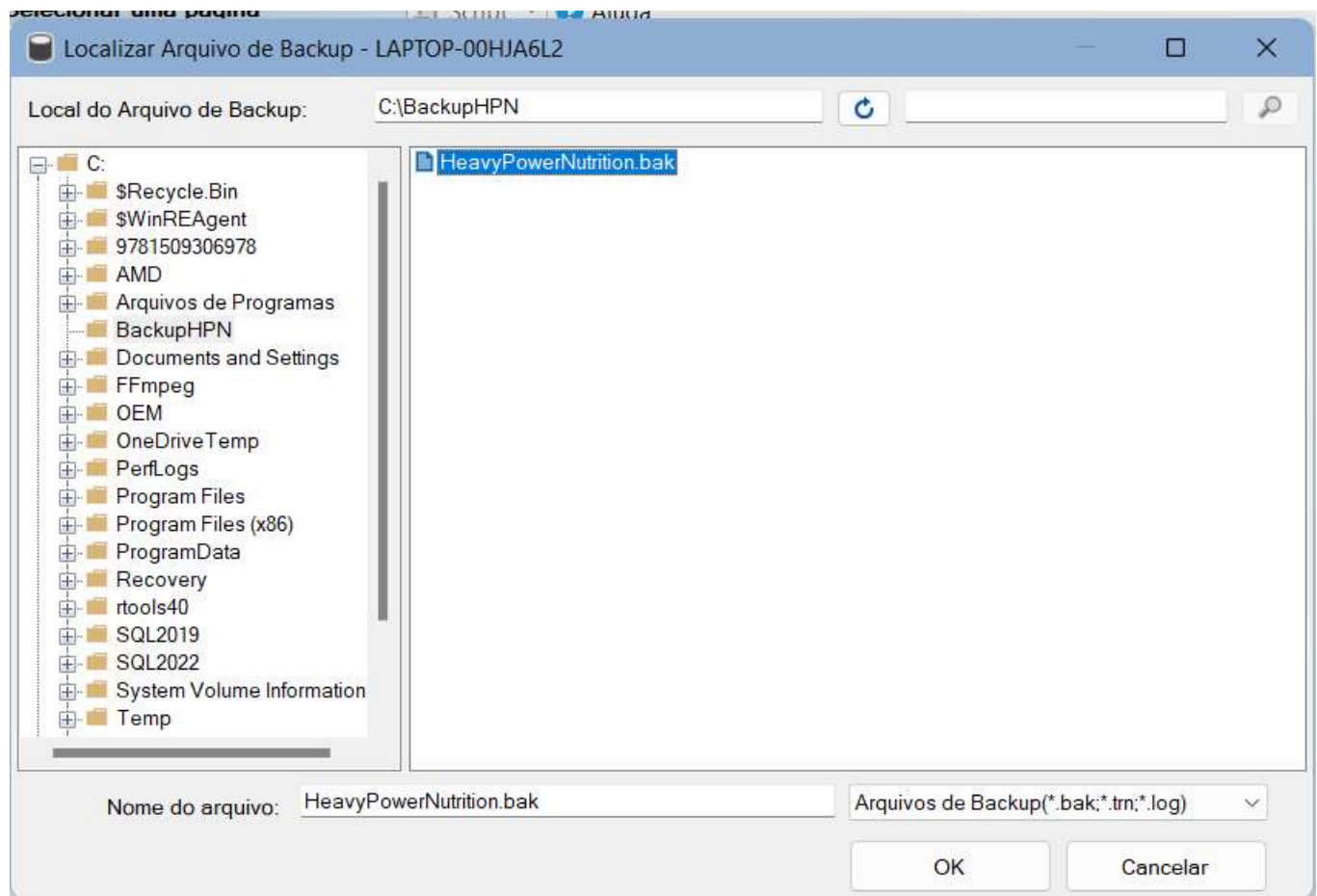
Na janela restaurar Bando de Dados , no campo Origem foi escolhida a opção Dispositivo: e depois foi clicado no botão ... que fica logo a direita.



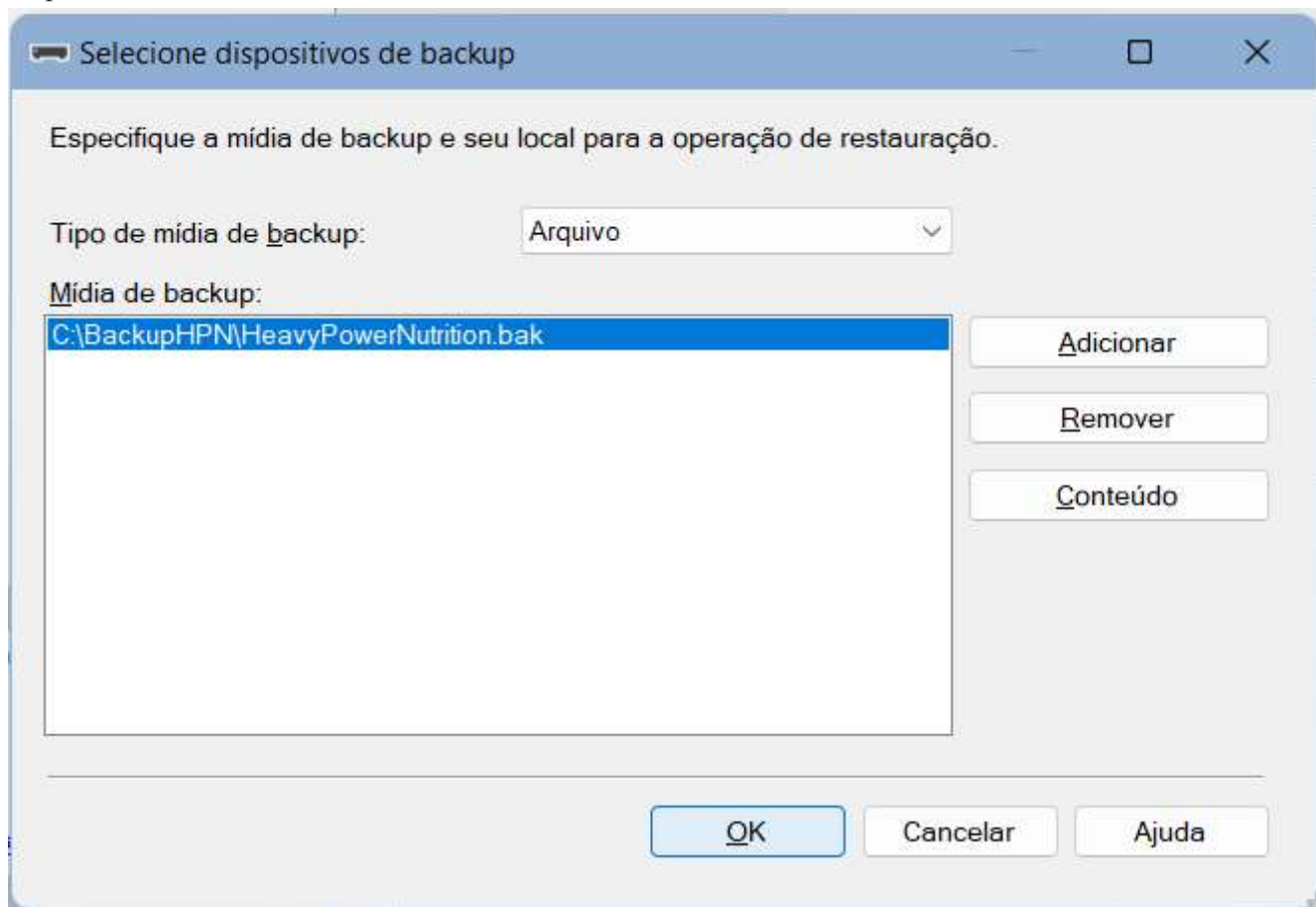
Em Tipo de mídia de backup foi selecionado arquivo e clicou-se em Adicionar.



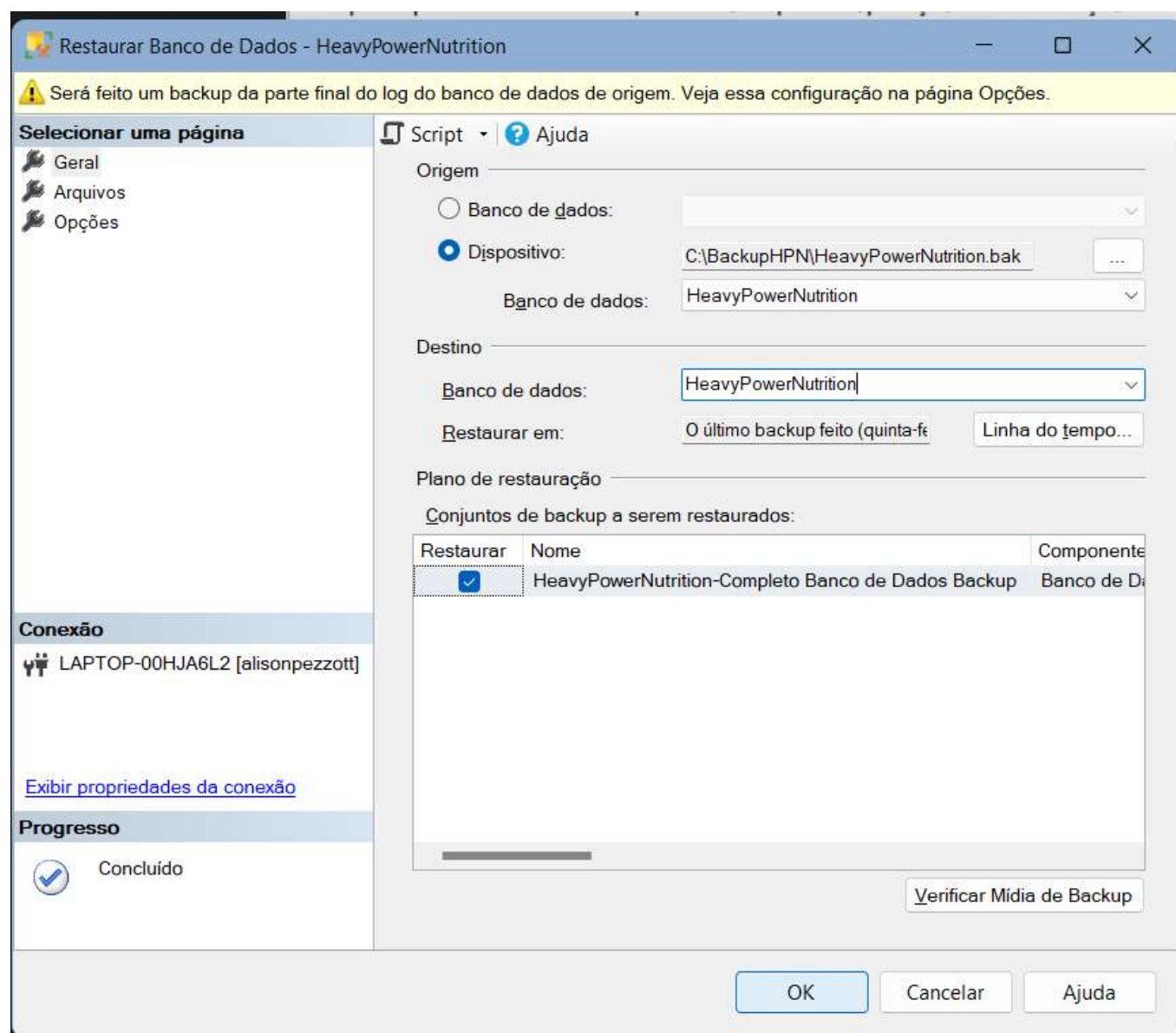
Foi informado o arquivo no diretório onde foi extraído o arquivo do backup e clicado em Ok.



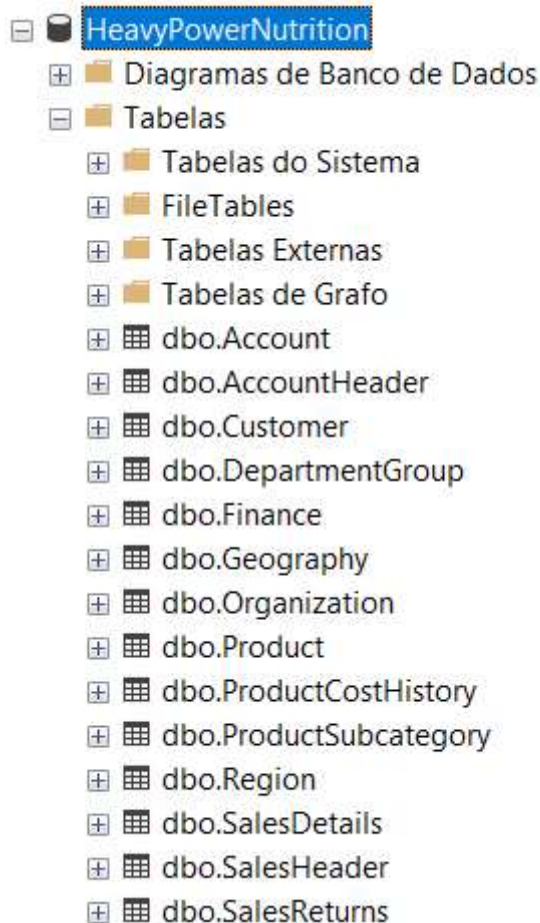
Depois foi clicado em Ok novamente



E finalmente Ok mais uma vez



O banco **HeavyPowerNutrition** foi restaurado com sucesso.



Análise exploratória e criação de views

Após a restauração dos dados foi iniciada uma análise exploratória dos dados a fim de:

Identificar quais as colunas das tabelas;

Identificar os relacionamentos;

Entender o melhor caminho para iniciar a modelagem dos dados para posterior análise.

SQLQuery3.sql - LA...alisonpezzott (64)

```

/***** Script do comando SelectTopRows de SSMS *****/
SELECT TOP (1000) [SalesHeaderKey]
,[SalesOrderNumber]
,[OrderDate]
,[DueDate]
,[ShipDate]
,[CustomerKey]
,[RegionKey]
,[DiscountAmount]
,[TotalAmount]
,[SalesAmount]
FROM [HeavyPowerNutrition].[dbo].[SalesHeader]

```

SalesHeaderKey	SalesOrderNumber	OrderDate	DueDate	ShipDate	CustomerKey	RegionKey	DiscountAmount	TotalAmount	SalesAmount
1	SO46648	2011-12-29	2012-01-10	2012-05-01	184	4	1794.08	57248.21	55464.13
2	SO45554	2011-08-29	2011-09-10	2011-05-09	187	4	427.42	25517.42	25090
3	SO44551	2011-05-01	2011-05-13	2011-08-05	385	4	364.48	41097.6	40733.12
4	SO45568	2011-09-29	2011-09-10	2011-05-09	3	4	356.46	15251.82	14895.36
5	SO53459	2013-02-28	2013-03-12	2013-07-03	582	4	1120.08	30487.97	29367.89
6	SO53607	2013-02-28	2013-03-12	2013-07-03	5	4	409.62	15369.76	14960.15
7	SO53589	2013-02-28	2013-03-12	2013-07-03	25	4	31.56	1101.28	1069.72
8	SO47687	2012-03-30	2012-04-11	2012-06-04	203	4	1281.1	47238.65	45957.55
9	SO57035	2013-04-30	2013-05-12	2013-07-05	77	4	106.16	2936.9	2830.74
10	SO71857	2013-11-29	2013-12-11	2013-06-12	624	4	862.24	28139.25	27277.01
11	SO48027	2012-04-30	2012-05-12	2012-07-05	492	4	10.18	314.46	304.28
12	SO45290	2011-08-01	2011-08-13	2011-08-08	384	4	1017.7	33469.9	32452.2
13	SO46377	2011-11-29	2011-12-11	2011-06-12	3	4	389.82	9853.78	9463.96
14	SO50747	2012-11-28	2012-12-10	2012-05-12	75	4	1241.56	36204.2	34962.64
15	SO67305	2013-09-29	2013-10-11	2013-06-10	599	4	3676.24	116730.56	113054.32
16	SO57142	2013-04-30	2013-05-12	2013-07-05	313	4	15.58	905.88	890.3
17	SO68795	2013-11-29	2013-12-11	2013-06-12	120	4	1069.19	29499.02	27549.02

100 % Mensagens

Resultados

Consulta executada com êxito.

LAPTOP-00HJA6L2 (16.0 RTM) alisonpezzott (64) HeavyPowerNutrition 00:00:00 1,000 linhas

Tabelas sendo examinadas

Após a análise, foi chegada à conclusão da necessidade de criação de *views* ou *exibições* que são tabelas virtuais criadas através da linguagem SQL, também conhecidas como *queries* para que fosse possível analisar os dados no Power BI mais tarde.

Para criar uma view basta ir no menu de navegação e clicar em **Nova Consulta** ou utilizar o atalho **Ctrl+N** e colar os scripts em SQL para criação das views conforme segue:

dimAccount

```

CREATE VIEW [dbo].[dimAccount] AS SELECT AccountKey
,Account
,AccountType
,AccountHeaderKey
,AccountSubheaderKey
,AccountSubheader
,Cast(SubheaderDetail AS int)SubheaderDetail
FROM Account;

```

dimAccountHeader

```

CREATE VIEW [dbo].[dimAccountHeader] AS SELECT AccountHeaderKey
,AccountHeader
,Cast(Detail AS int)Detail
FROM AccountHeader;

```

dimCustomers

```
CREATE VIEW [dbo].[dimCustomers] AS SELECT CustomerKey
      ,GeographyKey
      ,BusinessType
      ,Customer
      ,NumberEmployees
      ,AnnualRevenue
      ,YearOpened
FROM Customer;
```

dimDepartmetGroup

```
CREATE VIEW [dbo].[dimDepartmentGroup] AS SELECT * FROM DepartmentGroup
```

dimGeography

```
CREATE VIEW [dbo].[dimGeography]
AS
SELECT G.GeographyKey
      ,G.CityName
      ,G.StateCode
      ,G.StateName
      ,G.CountryCode
      ,G.CountryName
      ,G.RegionKey
      ,R.Region
      ,R.Continent
FROM GEOGRAPHY AS G
INNER JOIN Region AS R ON G.RegionKey = R.RegionKey;
```

dimOrganization

```
CREATE VIEW [dbo].[dimOrganization] AS
SELECT OrganizationKey
      ,Organization
      ,ParentOrganization
FROM Organization;
```

dimProducts

```
CREATE VIEW [dbo].[dimProducts] AS SELECT P.ProductKey
      ,P.ProductName
      ,P.Size
      ,P.Detail
```

```
,S.CategoryName  
,S.SubcategoryName  
FROM Product AS P  
INNER JOIN ProductSubcategory AS S ON P.ProductSubcategoryKey =  
S.ProductSubcategoryKey;
```

ftFinance

```
CREATE VIEW [dbo].[ftFinance] AS SELECT F.AccountKey  
, F.DepartmentGroupKey  
, F.OrganizationKey  
, F.Scenario  
, F.Date  
, (F.Amount * A.Sign)[Finance Amount]  
FROM Finance AS F  
INNER JOIN Account AS A ON F.AccountKey = A.AccountKey;
```

ftReturns

```
CREATE VIEW [dbo].[ftReturns] AS SELECT ReturnKey  
,ReturnDate  
,OrderDate  
,SalesOrderNumber  
,CustomerKey  
,ProductKey  
,ReturnQuantity  
,[UnitPrice]  
FROM SalesReturns;
```

ftSales

```
CREATE VIEW [dbo].[ftSales]  
AS  
SELECT U.SalesOrderNumber  
,U.OrderDate  
,U.DueDate  
,U.ShipDate  
,U.CustomerKey  
,U.ProductKey  
,U.OrderQuantity  
,U.UnitPrice  
,U.DiscountPerc  
,C.UnitCost  
FROM (  
    SELECT G.CountryCode
```

```

    ,T.SalesOrderNumber
    ,T.OrderDate
    ,T.DueDate
    ,T.ShipDate
    ,T.CustomerKey
    ,T.RegionKey
    ,T.GeographyKey
    ,T.ProductKey
    ,T.OrderQuantity
    ,T.UnitPrice
    ,T.DiscountPerc
FROM (
    SELECT H.SalesOrderNumber
        ,H.OrderDate
        ,H.DueDate
        ,H.ShipDate
        ,H.CustomerKey
        ,H.RegionKey
        ,C.GeographyKey
        ,D.ProductKey
        ,D.OrderQuantity
        ,D.UnitPrice
        ,(H.DiscountAmount / H.TotalAmount) DiscountPerc
    FROM SalesHeader AS H
    INNER JOIN SalesDetails AS D ON H.SalesHeaderKey = D.SalesHeaderKey
    INNER JOIN Customer AS C ON H.CustomerKey = C.CustomerKey
    ) AS T
    INNER JOIN GEOGRAPHY AS G ON T.GeographyKey = G.GeographyKey
    ) AS U
    INNER JOIN ProductCostHistory AS C ON U.CountryCode = C.CountryCode
    AND YEAR(U.OrderDate) = C.Year
    AND MONTH(U.OrderDate) = C.MonthNo
    AND U.ProductKey = C.ProductKey;

```

Como pode se ver a view ftSales foi a mais complexa, isto fez-se necessário para se encontrar o valor das vendas na menor granularidade que estava presente somente na tabela SalesDetails porém os campos relacionados aos períodos e ao clientes só estavam na tabela SalesHeader . Para encontrar o custo no produto que estava presente somente na tabela ProductCostHistory foi necessário encontrar em qual região ele foi vendido e como o caminho para se chegar a região era através no cliente também foi necessário o *join* com as tabelas Customer e Geography . Vide exemplo das 1000 primeiras linhas desta view na imagem abaixo.

Resultados Mensagens


	SalesOrderNumber	OrderDate	DueDate	ShipDate	CustomerKey	ProductKey	OrderQuantity	UnitPrice	DiscountPerc	UnitCost
1	SO43659	2010-12-29	2011-01-10	2011-05-01	676	220	4	20,1865	0,0264032885952381	6,057
2	SO43659	2010-12-29	2011-01-10	2011-05-01	676	223	2	5,1865	0,0264032885952381	1,557
3	SO43659	2010-12-29	2011-01-10	2011-05-01	676	346	2	2039,994	0,0264032885952381	574,914
4	SO43659	2010-12-29	2011-01-10	2011-05-01	676	349	1	2024,994	0,0264032885952381	572,532
5	SO43659	2010-12-29	2011-01-10	2011-05-01	676	350	3	2024,994	0,0264032885952381	580,017
6	SO43659	2010-12-29	2011-01-10	2011-05-01	676	347	1	2039,994	0,0264032885952381	582,249
7	SO43659	2010-12-29	2011-01-10	2011-05-01	676	229	3	28,8404	0,0264032885952381	8,655
8	SO43659	2010-12-29	2011-01-10	2011-05-01	676	235	1	28,8404	0,0264032885952381	8,655
9	SO43659	2010-12-29	2011-01-10	2011-05-01	676	218	6	5,7	0,0264032885952381	1,698
10	SO43659	2010-12-29	2011-01-10	2011-05-01	676	345	1	2039,994	0,0264032885952381	579,975
11	SO43659	2010-12-29	2011-01-10	2011-05-01	676	351	1	2024,994	0,0264032885952381	569,394
12	SO43659	2010-12-29	2011-01-10	2011-05-01	676	344	1	2039,994	0,0264032885952381	574,935
13	SO43660	2010-12-29	2011-01-10	2011-05-01	117	326	1	419,4589	0,0142476337647286	125,838
14	SO43660	2010-12-29	2011-01-10	2011-05-01	117	319	1	874,794	0,0142476337647286	262,44
15	SO43661	2010-12-29	2011-01-10	2011-05-01	442	215	5	20,1865	0,038188036110208	6,057
16	SO43661	2010-12-29	2011-01-10	2011-05-01	442	232	4	28,8404	0,038188036110208	8,646
17	SO43661	2010-12-29	2011-01-10	2011-05-01	442	200	1	890,76	0,028188036110208	243,021


Consulta executada com êxito. LAPTOP-00HJA6L2 (16.0 RTM) alisonpezzott (72) HeavyPowerNutrition 00:00:00 1.000 linhas


Fluxos de Dados no Power BI


Os fluxos de dados ou *dataflows* são a porta de entrada no serviço do Power BI Online, para onde vão os dados, a nuvem.


Primeiramente foi o criado o workspace `HPN` e dentro dele o fluxo de dados `DW_HPNI`.


 **Power BI** HPN



Página Inicial



Criar



Procurar



Hub de dados



Métricas


Aplicativos



Pipelines de implantação


Aprender



Workspaces



HPN

Workspaces

 Meu workspace

Todos

 **HPN**

 **+ Novo workspace**

Power BI HPN DW_HPND Avaliação: 39 dias restantes Pesquisar

Editar tabelas Adicionar tabelas Fechar

Tabelas Modelos de machine learning

NOME DA TABELA	TIPO DE TABELA	AÇÕES
dimAccountHeader	Custom	
dimDepartmentGroup	Custom	
dimGeography	Custom	
dimOrganization	Custom	
dimProducts	Custom	
dimCustomers	Custom	
ftSales	Custom	
ftReturns	Custom	
ftFinance	Custom	

De forma semelhante ao Power BI Desktop, os fluxos de dados possuem o mesmo método de leitura e tratamento. Basta clicar em obter dados > Bancos de dados do SQL Server e informar o nome do servidor, e as views. Observação: Como o banco de dados foi instalado localmente em uma das máquinas da equipe de desenvolvimento, antes de importar os dados para o fluxo de dados foi necessária a configuração do gateway *on premises* na mesma máquina onde instalado o banco de dados.

Power Query

Pesquisar (Alt + Q)

Página Inicial Transformações Adicionar coluna Exibir Ajuda

Obter dados - Nova consulta Inserir dados - Opções Gerenciar parâmetros - Parâmetros Atualizar - Consultas Propriedades - Editor avançado Gerenciar - Consultas Escolher colunas - Gerenciar colunas Remover colunas - Reduzir linhas Manter as linhas - Reduzir linhas Filtrar linhas - Reduzir linhas Classificar - Reduzir linhas Dividir coluna - Agrupar por Usar a primeira linha como cabeçalho - Substituir valores

Combinar Mapear para a entidade Insights da IA

Consultas [10]

Fonte: [Schema="dbo", Item="ftSales"] [Data]

	1. SalesOrderNumber	2. OrderDate	3. DueDate	4. ShipDate	5. CustomerKey	6. ProductKey	7. OrderQuantity	8. UnitPrice	9. DiscountPercentage
1	SO43659	29/12/2010	10/01/2011	01/05/2011	676	220	4	20.1865	0.021
2	SO43659	29/12/2010	10/01/2011	01/05/2011	676	223	2	5.1865	0.021
3	SO43659	29/12/2010	10/01/2011	01/05/2011	676	346	2	2039.994	0.021
4	SO43659	29/12/2010	10/01/2011	01/05/2011	676	349	1	2024.994	0.021
5	SO43659	29/12/2010	10/01/2011	01/05/2011	676	350	3	2024.994	0.021
6	SO43659	29/12/2010	10/01/2011	01/05/2011	676	347	1	2039.994	0.021
7	SO43659	29/12/2010	10/01/2011	01/05/2011	676	229	3	28.8404	0.021
8	SO43659	29/12/2010	10/01/2011	01/05/2011	676	235	1	28.8404	0.021
9	SO43659	29/12/2010	10/01/2011	01/05/2011	676	218	6	5.7	0.021
10	SO43659	29/12/2010	10/01/2011	01/05/2011	676	345	1	2039.994	0.021
11	SO43659	29/12/2010	10/01/2011	01/05/2011	676	351	1	2024.994	0.021
12	SO43659	29/12/2010	10/01/2011	01/05/2011	676	344	1	2039.994	0.021
13	SO43660	29/12/2010	10/01/2011	01/05/2011	117	326	1	419.4589	0.014
14	SO43660	29/12/2010	10/01/2011	01/05/2011	117	319	1	874.794	0.014
15	SO43661	29/12/2010	10/01/2011	01/05/2011	442	215	5	20.1865	0.038
16	SO43661	29/12/2010	10/01/2011	01/05/2011	442	232	4	28.8404	0.038
17	SO43661	29/12/2010	10/01/2011	01/05/2011	442	300	1	809.76	0.038
18									

Colunas: 10 Linhas: 99+

Configurações de consulta

Propriedades

Nome: ftSales

Tipo de entidade: Personalizar

Etapas aplicadas

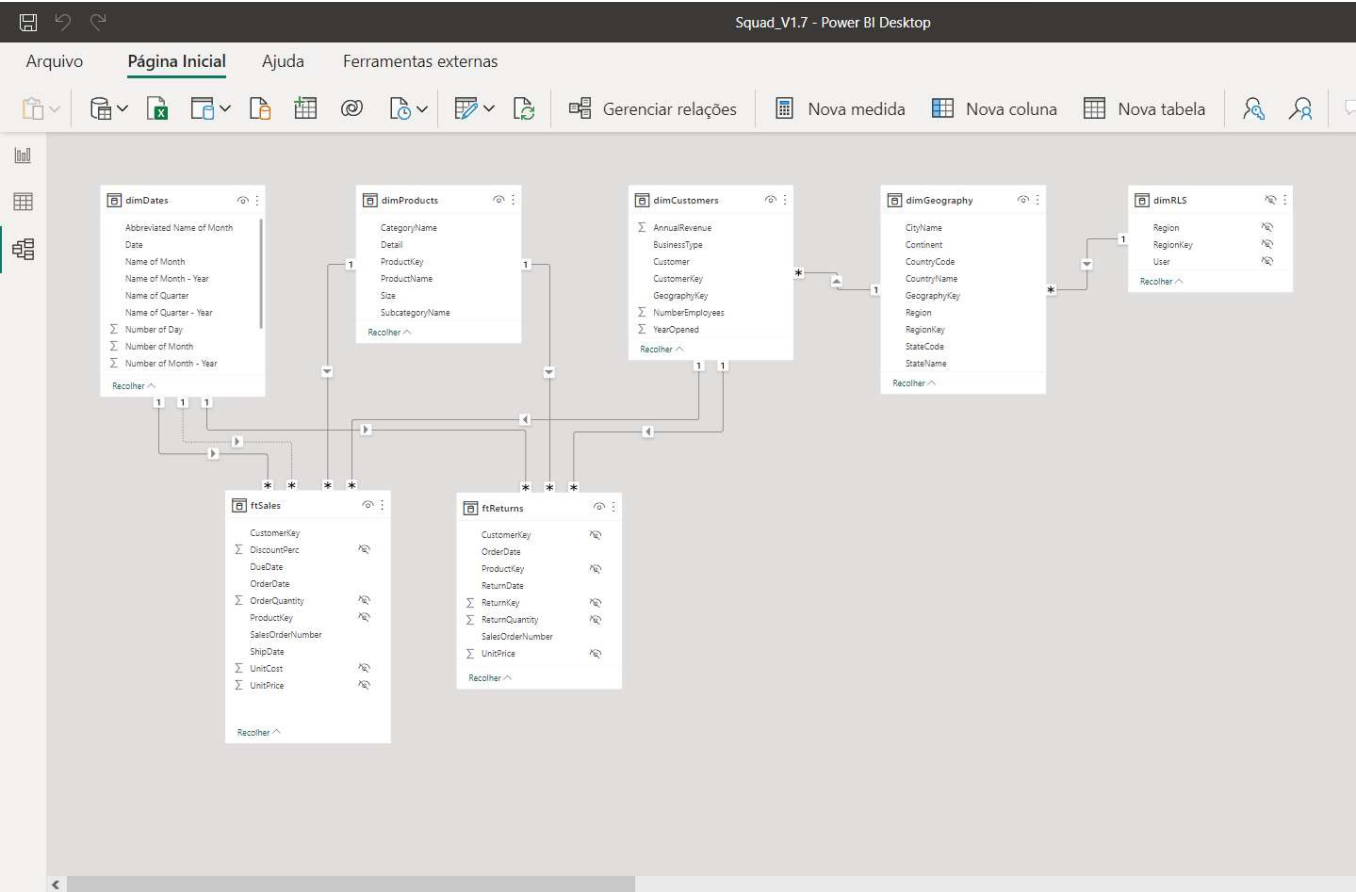
Fonte: dbo_ftSales

Como as views foram tratadas no banco de dados, no dataflow não foi necessária nenhuma transformação. Apenas informadas as views e foram carregadas para a nuvem.

Carga e Modelagem no Power BI Desktop

Na sequência, as tabelas foram importadas no Power Query do Power BI Desktop, onde também não houve transformações dos dados em relação aos que já estavam no dataflow.

Aqui também foi tomada a decisão de importar somente as tabelas do escopo solicitado que seria da área comercial. Portando as tabelas da área financeira seguiram somente até o dataflow. Vide na foto abaixo o modelo de dados já importado no Power BI Desktop bem como as suas chaves de relacionamento.



Repare que há uma tabela chamada dimRLS. Esta tabela foi criada pela equipe para cumprir o papel da segurança do nível de linha RLS (*row level security*) onde os funcionários verão os dados somente da sua região.

Region	RegionKey	User
Northwest	1	northwest@wganalytics.com.br
Northeast	2	northeast@wganalytics.com.br
Central	3	central@wganalytics.com.br
Southwest	4	southwest@wganalytics.com.br
Southeast	5	southeast@wganalytics.com.br
Canada	6	canada@wganalytics.com.br
France	7	france@wganalytics.com.br
Germany	8	germany@wganalytics.com.br
Brazil	9	brazil@wganalytics.com.br
United Kingdom	10	unitedkingdom@wganalytics.com.br

Através desta recurso, a função de segurança checa quem está acessando o relatório publicado e propaga a relação para mostrar somente os dados que forma restringidos.

Gerenciar as funções de segurança



Crie novas funções de segurança e use filtros para definir restrições de dados em nível de linha.

Funções

+ Novo

- Admin ...
- Region ...

Selecionar tabelas

- auxSpecialMo... ...
- Categorical Fie... ...
- dimCustomers ...
- dimDates ...
- dimGeography ...
- dimProducts ...
- dimRLS ...
- Field Parameters ...
- Field Paramete... ...
- ftReturns ...
- ftSales ...

Filtrar dados

Alternar para o editor padrão

[User] == USERNAME()

Salvar

Fechar

Tabelas e Medidas DAX

Neste tópico serão descritas as medidas e expressões de tabela que foram utilizadas nas análises.