

## Api – Cervejarias

Documentação Api: [Open Brewery DB](https://openbrewerydb.org/)

End point utilizado: <https://api.openbrewerydb.org/v1/breweries?>

Método: GET

Authorization: Não necessário

Foi utilizado paginação: Sim, “page”

Foi utilizado o controle de itens da pagina: Sim, “per\_page”

End point final: [https://api.openbrewerydb.org/v1/breweries?page=1&per\\_page=50](https://api.openbrewerydb.org/v1/breweries?page=1&per_page=50)

Observações.:

- Na estrutura de coleta do pipeline de dados, foi desenvolvido um Loop de paginação.
- No controle de itens da página, foi utilizado “50” como default;
- Na documentação não indica quais campos podem sofrer update com o tempo;
- Não foi passado ou encontrado na documentação, se há algum tipo de limitação de requisição por minuto.

## Base de dados

SGBD escolhido: SQL Server

Servidor: LocalHost

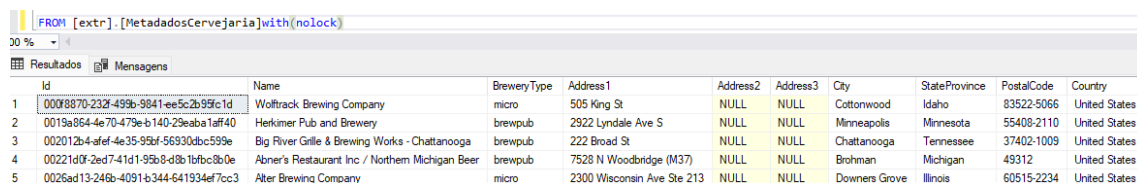
Banco de dados: DadosAmbev

Schemas:

- Extr - Bronze - Para o armazenamento dos dados brutos extraídos da Api;
- Transf - Prata - Criado para armazenar os dados transformados e higienizados;
- Rlt - Ouro - Criado para armazenamento dos dados agrupados e sumarizados pela visão desejada;
- Dim - Criado para possíveis dimensões do processo.

Tabelas:

- extr.MetadadosCervejaria - Tabela responsável por armazenar, os dados brutos coletados da Api, com ela, é possível criar dimensões em Data Warehouse ou permanecer em Data Lake;



|   | Id                                   | Name   | Brewery Type | Address1                   | Address2 | Address3 | City          | State/Province | Postal Code | Country       |
|---|--------------------------------------|--|--------------|----------------------------|----------|----------|---------------|----------------|-------------|---------------|
| 1 | 000f8870-232f-499b-9841-ee5c2b959c1d | Wolfftrack Brewing Company                     | micro        | 505 King St                | NULL     | NULL     | Cottonwood    | Idaho          | 83522-5066  | United States |
| 2 | 0019a864-4e70-479e-b140-29eaba1aff40 | Herkimer Pub and Brewery                       | brewpub      | 2922 Lyndale Ave S         | NULL     | NULL     | Minneapolis   | Minnesota      | 55408-2110  | United States |
| 3 | 002012b4-afe4-4e35-99bf-56930dbc599e | Big River Grille & Brewing Works - Chattanooga | brewpub      | 222 Broad St               | NULL     | NULL     | Chattanooga   | Tennessee      | 37402-1009  | United States |
| 4 | 00221d0f-2ad7-41d1-95b0-d0b1bfbc8b0e | Abner's Restaurant Inc / Nothem Michigan Beer  | brewpub      | 7528 N Woodbridge (M37)    | NULL     | NULL     | Broham        | Michigan       | 49312       | United States |
| 5 | 0026ad13-246b-4091-b344-641934ef7cc3 | Alter Brewing Company                          | micro        | 2300 Wisconsin Ave Ste 213 | NULL     | NULL     | Downers Grove | Illinois       | 60515-2234  | United States |

- transf.MetadadosCervejaria - Armazena os dados processados e higienizados;

FROM [transf].[MetadadosCervejaria] with(nolock)

|   | BreweryType | Address1                   | Address2     | Address3     | City          | StateProvince | Country       | Longitude | Latitude | State     | Street                     | QtdCervejaria | DataInsercao            |
|---|-------------|----------------------------|--------------|--------------|---------------|---------------|---------------|-----------|----------|-----------|----------------------------|---------------|-------------------------|
| 1 | micro       | 505 King St                | Desconhecido | Desconhecido | Cottonwood    | Idaho         | United States | -116.3497 | 46.04895 | Idaho     | 505 King St                | 1             | 2025-07-09 18:19:54.637 |
| 2 | brewpub     | 2922 Lyndale Ave S         | Desconhecido | Desconhecido | Minneapolis   | Minnesota     | United States | -93.28838 | 44.94921 | Minnesota | 2922 Lyndale Ave S         | 1             | 2025-07-09 18:19:54.637 |
| 3 | brewpub     | 222 Broad St               | Desconhecido | Desconhecido | Chattanooga   | Tennessee     | United States | -85.31071 | 35.05387 | Tennessee | 222 Broad St               | 1             | 2025-07-09 18:19:54.637 |
| 4 | brewpub     | 7528 N Woodbridge (M37)    | Desconhecido | Desconhecido | Broham        | Michigan      | United States | NULL      | NULL     | Michigan  | 7528 N Woodbridge (M37)    | 1             | 2025-07-09 18:19:54.637 |
| 5 | micro       | 2300 Wisconsin Ave Ste 213 | Desconhecido | Desconhecido | Downers Grove | Illinois      | United States | NULL      | NULL     | Illinois  | 2300 Wisconsin Ave Ste 213 | 1             | 2025-07-09 18:19:54.637 |

- rlt.AgregadoCervejaria - Responsável por armazenar os dados agrupados e sumarizados;

FROM [rlt].[AgregadoCervejaria] with(nolock)

|   | Id | BreweryType | City         | StateProvince  | Country       | State          | QtdCervejaria | DataInsercao            |
|---|----|-------------|--------------|----------------|---------------|----------------|---------------|-------------------------|
| 1 | 1  | micro       | Andrews      | North Carolina | United States | North Carolina | 1             | 2025-07-09 18:19:55.493 |
| 2 | 2  | micro       | Hudson       | New York       | United States | New York       | 2             | 2025-07-09 18:19:55.493 |
| 3 | 3  | brewpub     | North Wales  | Pennsylvania   | United States | Pennsylvania   | 1             | 2025-07-09 18:19:55.493 |
| 4 | 4  | brewpub     | Collegeville | Pennsylvania   | United States | Pennsylvania   | 1             | 2025-07-09 18:19:55.493 |
| 5 | 5  | micro       | Falls Church | Virginia       | United States | Virginia       | 1             | 2025-07-09 18:19:55.493 |

- dim.LogStatus - Armazena o status da execução por etapa do processo;

FROM [dim].[LogStatus] with(nolock)

|   | Id | Schema | Tabela              | Status  | DataInsercao            |
|---|----|--------|---------------------|---------|-------------------------|
| 1 | 1  | extr   | MetadadosCervejaria | Sucesso | 2025-07-09 18:19:53.947 |
| 2 | 2  | transf | MetadadosCervejaria | Sucesso | 2025-07-09 18:19:54.790 |
| 3 | 3  | rlt    | AgregadoCervejaria  | Sucesso | 2025-07-09 18:19:55.603 |
| 4 | 4  | extr   | MetadadosCervejaria | Sucesso | 2025-07-09 18:47:53.430 |
| 5 | 5  | transf | MetadadosCervejaria | Sucesso | 2025-07-09 18:47:53.973 |

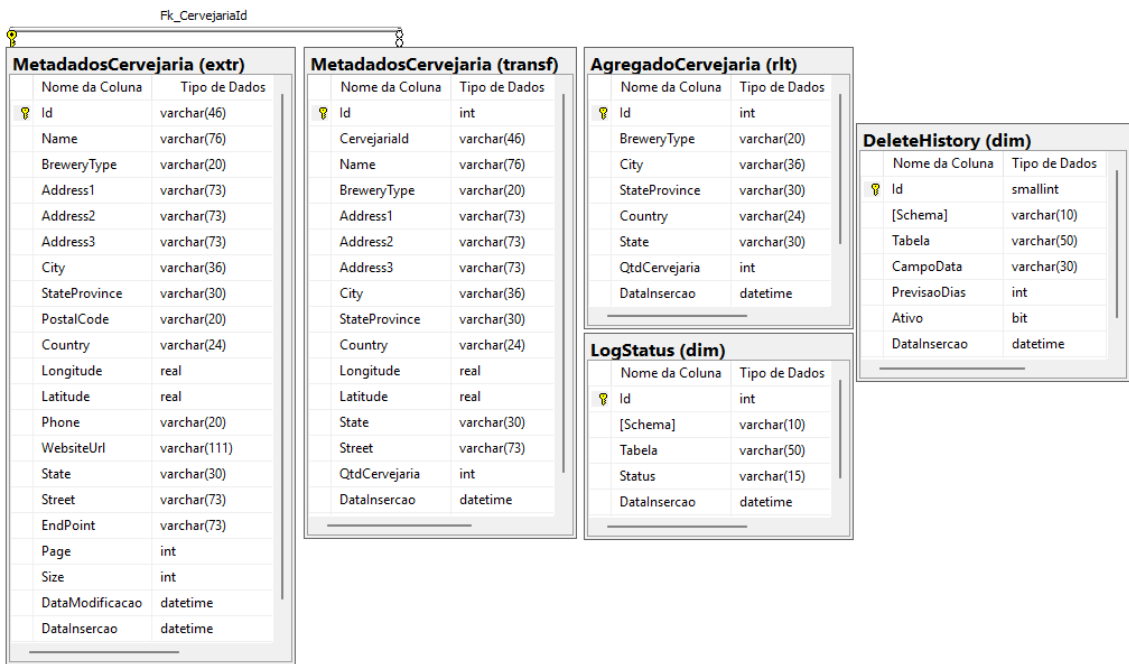
- dim.DeleteHistory - Com a definição do tempo que dado será armazenado no banco de dados, é feito o cadastro do nome da tabela onde os dados serão deletados, o campo de avaliação de tempo e o tempo em dias que o dado deve ser armazenado.

FROM [dim].[DeleteHistory] with(nolock)

|   | Id | Schema | Tabela              | CampoData    | PrevisaoDias | Ativo | DataInsercao            |
|---|----|--------|---------------------|--------------|--------------|-------|-------------------------|
| 1 | 1  | extr   | MetadadosCervejaria | DataInsercao | 180          | 1     | 2025-07-08 07:49:36.563 |
| 2 | 2  | transf | MetadadosCervejaria | DataInsercao | 180          | 1     | 2025-07-08 07:49:36.563 |

Observação.: Todas tabelas contem a coluna “DataInsercao” que referência a data e hora que o dado foi inserido na tabela, essa coluna é configurada na hora da criação “DataInsercao datetime default getdate()”.

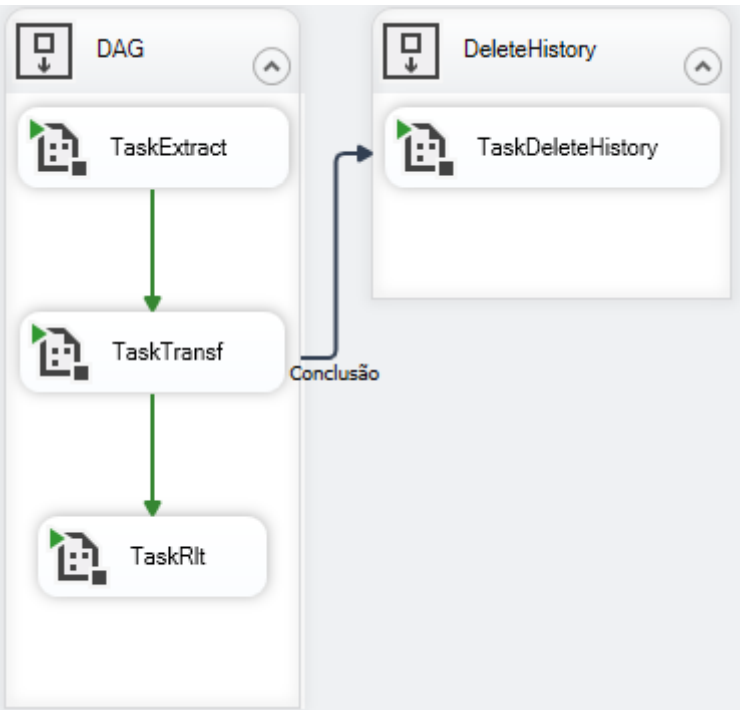
# Diagrama de tabelas



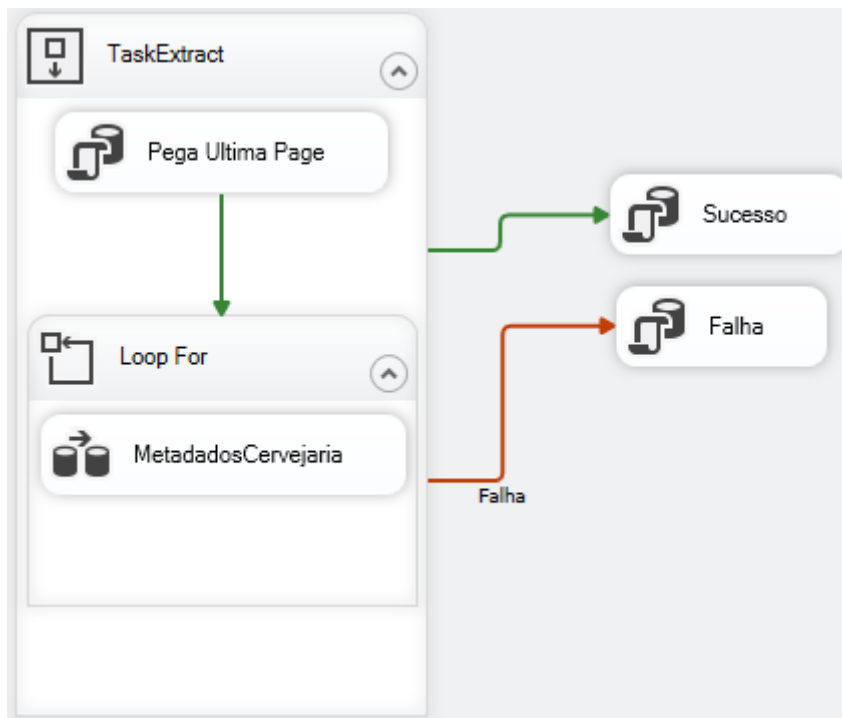
# Pipeline de dados

Ferramenta escolhida: Integration Services (SSIS)

Fluxo de execução:



- **TaskExtract** - Responsável por acessar a api em Loop decrescente por paginação e armazenar os dados na tabela extração.



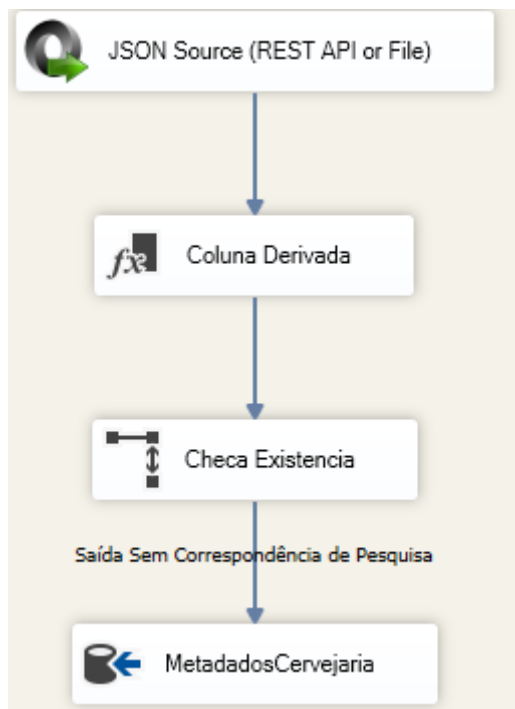
Variáveis criadas:

Var\_rest string == <https://api.openbrewerydb.org>  
 Var\_page int == 1 default  
 Var\_size int == 50 default  
 Var\_max\_page int == 0 default  
 Var\_endpoint string == @[User::Var\_rest]+"/v1/breweries?"+"page="+ (DT\_STR, 10, 1252) @[User::Var\_page]+"&per\_page="+ (DT\_STR, 10, 1252) @[User::Var\_size]

Configuração do Loop de paginação:

|                  |                            |
|------------------|----------------------------|
| EvalExpression   | @Var_page <= @Var_max_page |
| AssignExpression | @Var_page = @Var_page + 1  |

Ingestão e inserção dos dados:



JSON Source (REST API or File) - Acessa Api utilizando a variavel “Var\_endpoint” com o método GET;

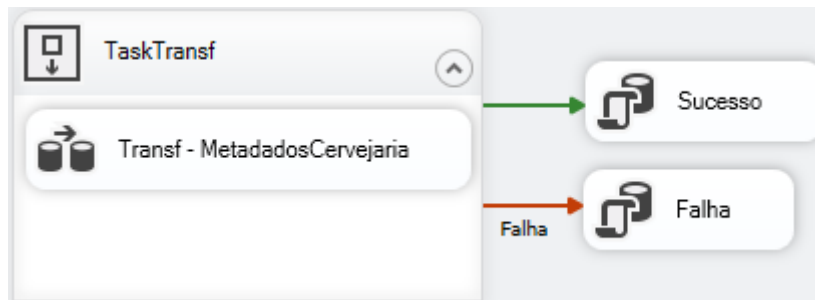
Coluna Derivada - Cria as colunas “EndPoint== Var\_endpoint”, “Page== Var\_page”, “Size== Var\_size” que podem servir como controle de log da Api;

| Derived Column Name | Derived Column      | Expression                            | Data Type                                | Length |
|---------------------|---------------------|---------------------------------------|--|--------|
| EndPoint            | <add as new column> | (DT_STR,73,1252)@[User::Var_endpoint] | cadeia de caracteres [DT_STR]            | 73     |
| Page                | <add as new column> | @[User::Var_page]                     | inteiro assinado de quatro bytes [DT_I4] |        |
| Size                | <add as new column> | @[User::Var_size]                     | inteiro assinado de quatro bytes [DT_I4] |        |

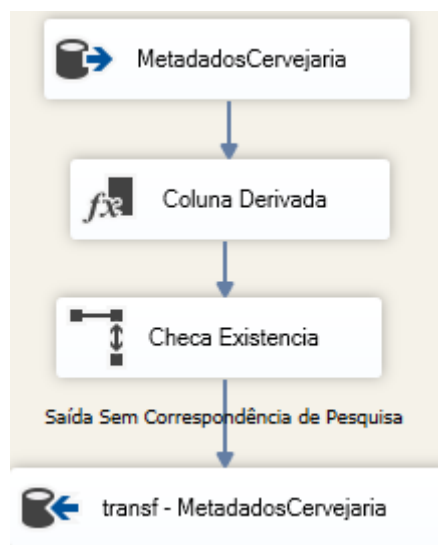
Checa Existência - No ato da criação da tabela, o campo “Id” foi adicionado como primary key, com isso conseguimos garantir integridade dos dados checando, checando se o “Id” já existe na tabela com isso nenhum dado é inserido, caso contrário, é feito o insert.

Destino - Após a checagem dos dados, é feito a inserção na tabela extr.MetadadosCervejaria.

- **TaskTransf** - Responsável por coletar os dados da tabela “extr.MetadadosCervejaria” e disponibilizar na “transf.MetadadosCervejaria”.



Ingestão e inserção dos dados:



MetadadosCervejaria - Responsável por pegar os dados da tabela extração de forma analítica;

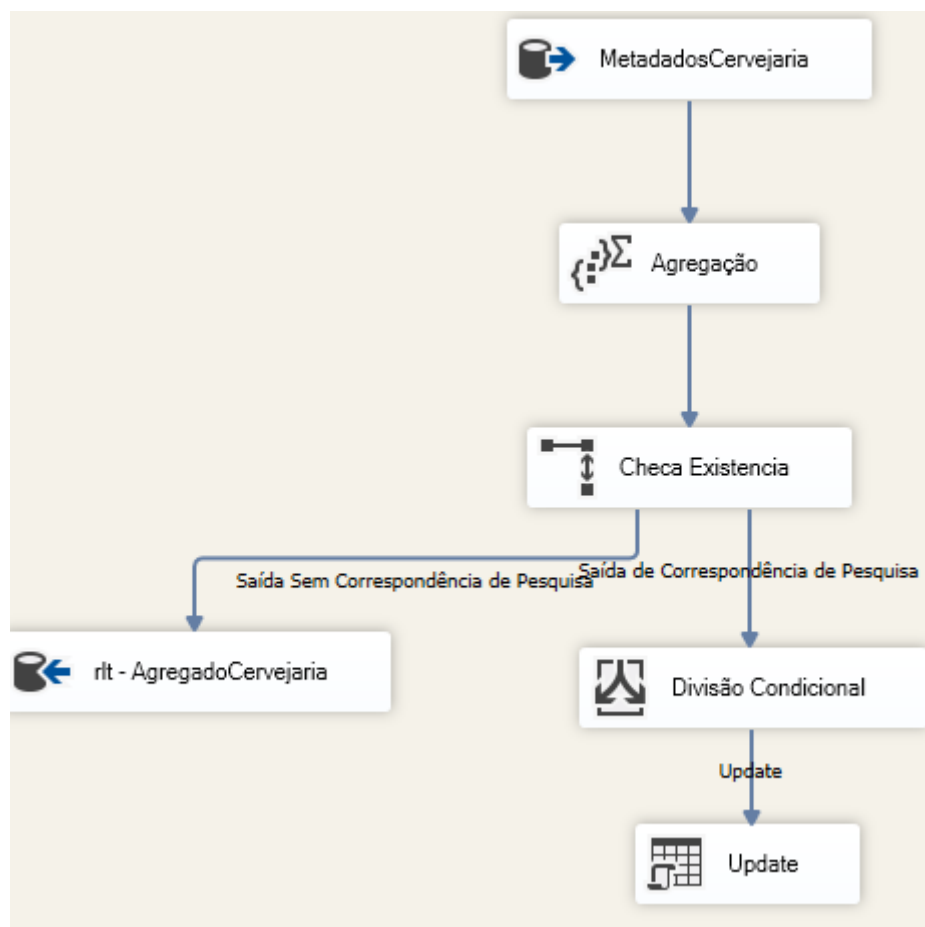
Coluna Derivada - Cria a coluna “QtdCervejaria == 1” e agrupa valores de NULL para “Desconhecido” nas colunas “Name”, “BreweryType”, “Address1”, “Address2”, “Address3”, “City”, “StateProvince”, “Country”, “State” e “Street”;

| Derived Column Name | Derived Column             | Expression                                | Data Type                                | Length |
|---------------------|----------------------------|---|--|--------|
| QtdCervejaria       | <add as new column>        | 1   | inteiro assinado de quatro bytes [DT_I4] |        |
| Name                | Substituir 'Name'          | REPLACENULL(Name,"Desconhecido")          | cadeia de caracteres [DT_STR]            | 76     |
| BreweryType         | Substituir 'BreweryType'   | REPLACENULL(BreweryType,"Desconhecido")   | cadeia de caracteres [DT_STR]            | 20     |
| Address1            | Substituir 'Address1'      | REPLACENULL(Address1,"Desconhecido")      | cadeia de caracteres [DT_STR]            | 73     |
| Address2            | Substituir 'Address2'      | REPLACENULL(Address2,"Desconhecido")      | cadeia de caracteres [DT_STR]            | 73     |
| Address3            | Substituir 'Address3'      | REPLACENULL(Address3,"Desconhecido")      | cadeia de caracteres [DT_STR]            | 73     |
| City                | Substituir 'City'          | REPLACENULL(City,"Desconhecido")          | cadeia de caracteres [DT_STR]            | 36     |
| StateProvince       | Substituir 'StateProvince' | REPLACENULL(StateProvince,"Desconhecido") | cadeia de caracteres [DT_STR]            | 30     |
| Country             | Substituir 'Country'       | REPLACENULL(Country,"Desconhecido")       | cadeia de caracteres [DT_STR]            | 24     |
| State               | Substituir 'State'         | REPLACENULL(State,"Desconhecido")         | cadeia de caracteres [DT_STR]            | 30     |
| Street              | Substituir 'Street'        | REPLACENULL(Street,"Desconhecido")        | cadeia de caracteres [DT_STR]            | 73     |

Checa Existência – Com base na coluna “Id” da tabela “extr.MetadadosCervejaria” é verificado a existência na tabela destino “[transf].[MetadadosCervejaria]” a coluna “CervejariaId”, caso não exista, o dado é inserido, se não é realizado o insert dos dados.

Destino - Após a checagem dos dados, é feito a inserção na tabela “[transf].[MetadadosCervejaria]”.

- **TaskRlt** - Com a garantia de integridade nas tasks anteriores, é feito o agrupamento das informações pelas colunas “BreweryType”, “City”, “StateProvince”, “Country”, “State” e sumarizado por “QtyCervejaria”.



Agregação - Agrupa as colunas "BreweryType", "City", "StateProvince", "Country", "State" e sumarizado por "QtdCervejaria".

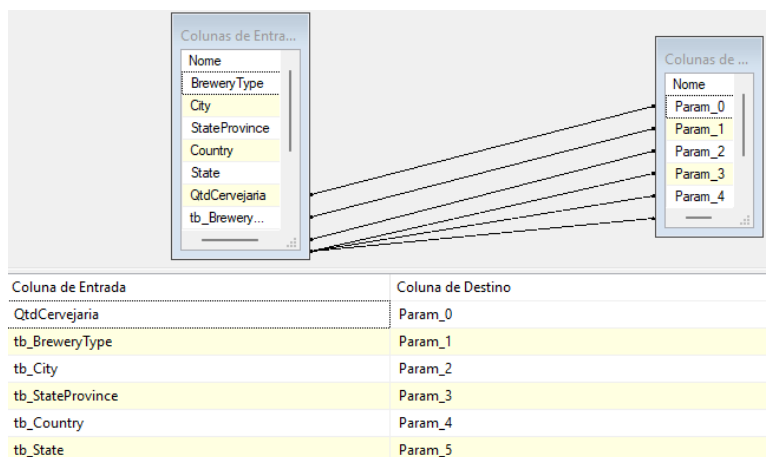
| Input Column  | Output Alias  | Operation   |
|---------------|---------------|-------------|
| BreweryType   | BreweryType   | Agrupar por |
| City          | City          | Agrupar por |
| StateProvince | StateProvince | Agrupar por |
| Country       | Country       | Agrupar por |
| State         | State         | Agrupar por |
| QtdCervejaria | QtdCervejaria | Soma        |

Checa Existência - Checa a existência com base nas colunas "BreweryType", "City", "StateProvince", "Country" e "State", caso o dado não exista, o dado é inserido, caso contrario é verificado se houve alteração no volume de "QtdCervejaria", se sim, é feito update na mesma na tabela "rlt.AgregadoCervejaria".

Divisão Condicional - Se dado já existir, é comparado o dado sumarizado da origem com a destino, para as linhas que houverem alteração, é feito update no campo "QtdCervejaria" na tabela "rlt.AgregadoCervejaria".

| Order | Output Name | Condition                         |
|-------|-------------|-----------------------------------|
| 1     | Update      | tb_QtdCervejaria != QtdCervejaria |

Update – Realiza update para as linhas de dados que tiverem alteração no campo "QtdCervejaria" na tabela "rlt.AgregadoCervejaria".



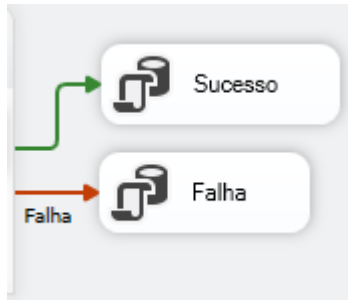
```
update AgregadoCervejaria
set AgregadoCervejaria.[QtdCervejaria] = ?
```

```
from [rlt].[AgregadoCervejaria] with(nolock)
where
AgregadoCervejaria.[BreweryType] = ?
```



```
and AgregadoCervejaria.[City] = ?  
and AgregadoCervejaria.[StateProvince] = ?  
and AgregadoCervejaria.[Country] = ?  
and AgregadoCervejaria.[State] = ?
```

Observação.: O insert na tabela “dim.LogStatus” que é feito nas Taks “TaskExtract”, “TaskTransf” e “TaskRlt”, é padrão alterando somente o schema da tabela.



Exemplo Sucesso:

```
insert into dim.LogStatus  
select  
    'rlt'  
    , 'AgregadoCervejaria'  
    , 'Sucesso'  
    , getdate()
```

Exemplo Falha:

```
insert into dim.LogStatus  
select  
    'rlt'  
    , 'AgregadoCervejaria'  
    , 'Falha'  
    , getdate()
```

- **TaskDeleteHistory** - Criado para definição do ciclo final dos dados extraídos e transformados. Os dados agregados não serão deletados, pois, já estão resumidos.

Variáveis:

```
Var_campo_data string == 0 default
```

```
Var_lista_previsao int == 0 default  
Var_lista_tabela string == 0  
Var_lista_tabela_obj objeto == objeto  
Var_consulta string ==
```

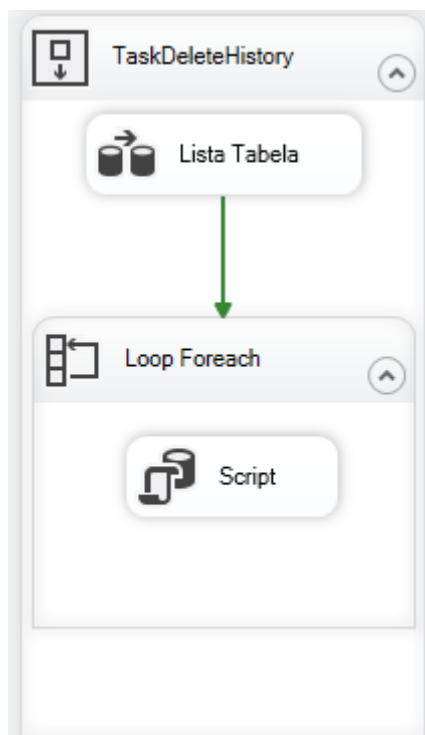
“DELETE

```
FROM " + @[User::Var_lista_tabela] +"  
WHERE " + @[User::Var_campo_data] + "<" +
```

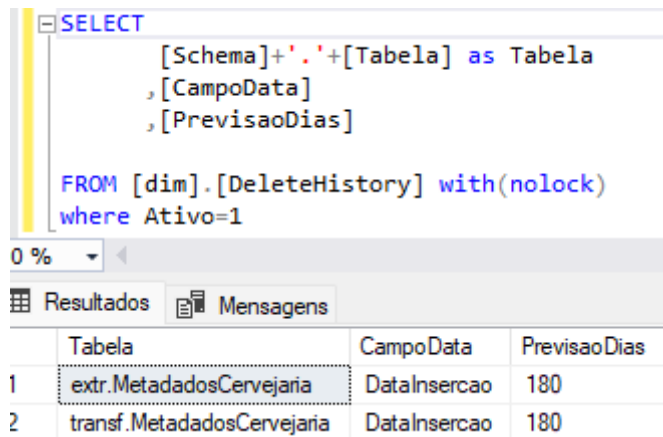
"(

SELECT

```
DATEADD(DAY,-"+ (DT_STR, 10, 1252)  
@[User::Var_lista_previsao]+",MAX(" + @[User::Var_campo_data] + ")) " +  
"FROM " + @[User::Var_lista_tabela] + " WITH(NOLOCK))"
```



Lista Tabela - seleciona os dados na tabela “dim.DeleteHistory” e armazena os campos nas variáveis criadas.



The screenshot shows a SQL query window with the following text:

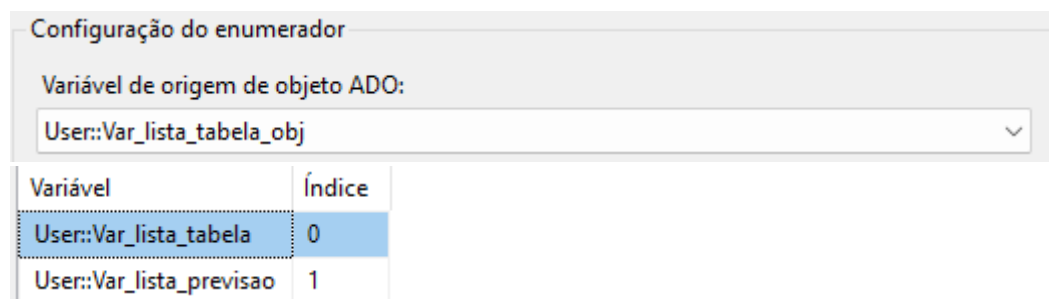
```
SELECT
    [Schema]+'.'+[Tabela] as Tabela
    ,[CampoData]
    ,[PrevisaoDias]

FROM [dim].[DeleteHistory] with(nolock)
where Ativo=1
```

Below the query, the 'Results' tab is active, displaying a table with 3 columns: Tabela, CampoData, and PrevisaoDias. The table contains 2 rows of data.

|   | Tabela                     | CampoData    | PrevisaoDias |
|---|----------------------------|--------------|--------------|
| 1 | extr.MetadadosCervejaria   | DataInsercao | 180          |
| 2 | transf.MetadadosCervejaria | DataInsercao | 180          |

Loop Foreach – Com base na quantidade de registros, é feito um Loop passando por tabela.



The screenshot shows the 'Configure Enumerator' dialog box. The 'ADO object source variable' is set to 'User::Var\_lista\_tabela\_obj'. Below this, a table lists the variables and their indices.

| Variável                 | Índice |
|--------------------------|--------|
| User::Var_lista_tabela   | 0      |
| User::Var_lista_previsao | 1      |

Esses dados são concatenados na variável “Var\_consulta” e por fim, é realizado o delete de acordo com tempo em dias estabelecido na tabela.