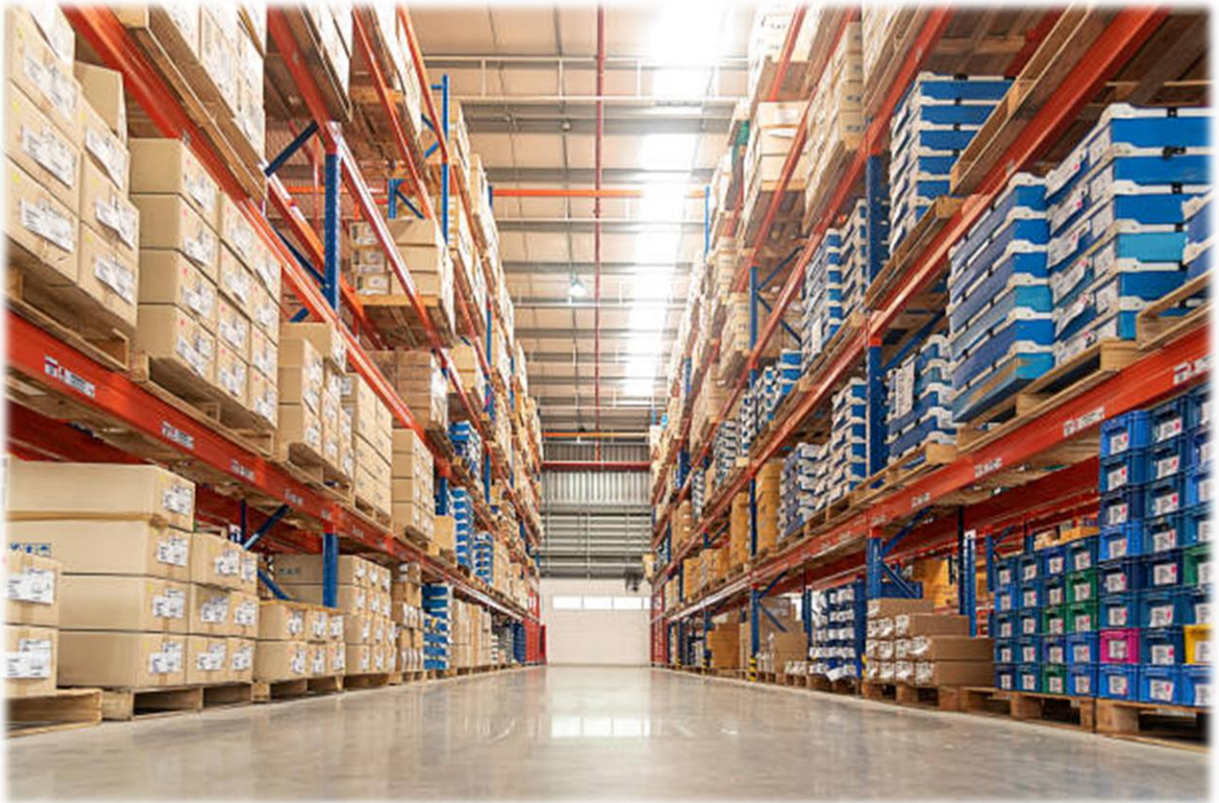


2023

## Banco de dados em um Centro de Distribuição



## MVP – Engenharia de dados

Kevin Sathler Rêgo Chagas

## 1. Introdução

Este MVP é sobre um Centro de Distribuição que deseja entender todo seu histórico de abastecimentos nas lojas do estado de São Paulo. Através de algumas informações existentes o Centro de Distribuição deseja carregar estes dados em um bucket, realizar todo trabalho de ETL (caso necessário) e por fim disponibilizá-los no Big Query para que as análises sejam realizadas.

Antes de iniciar a coleta de dados do projeto, foram realizadas duas atividades: A primeira delas foi entender o cenário que iria ser trabalhado, desenhando o principal objetivo do trabalho que seria realizado; A segunda atividade foi levantar algumas questões para serem respondidas através de análises no banco de dados.

## 2. Objetivo

Garantir um estoque ideal para que um Centro de Distribuição consiga abastecer corretamente as lojas presentes na cidade de São Paulo. Dessa forma, há dois principais objetivos:

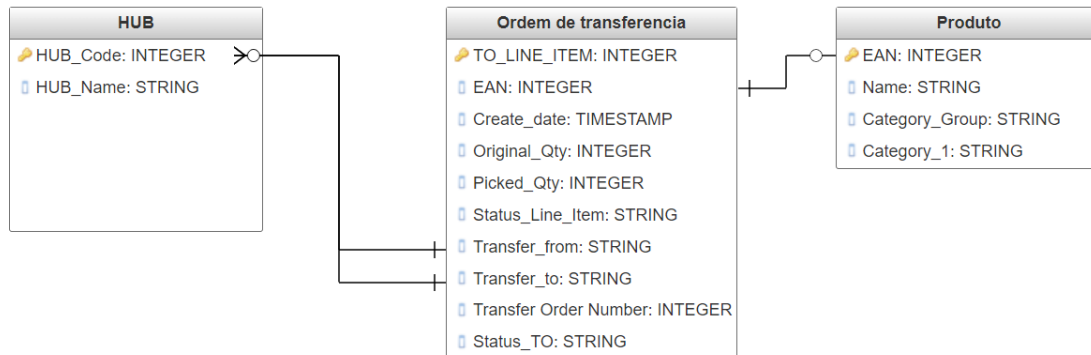
- Definir qual o estoque necessário em unidades para cada produto, baseado na média de saída + parâmetros de segurança (como o desvio padrão).
- Entender qual a categoria com maior necessidade de abastecimento e qual loja mais recebe os produtos desta categoria.

## 3. Perguntas a serem respondidas

- Qual a média necessária, em unidades, de estoque por dia, por produto?  
*Esta pergunta deve ser respondida entendendo a média vendida de cada produto por dia.*
- Qual o desvio padrão de cada produto, considerando quantas unidades são abastecidas, em um período de 60 dias  
*Esta pergunta será respondida com o objetivo de definir uma margem de segurança para o estoque, de forma a considerar as possíveis variações de demanda.*
- Qual a média de vendas por dia, por produto, mais o desvio padrão?
- Baseado no estoque + desvio padrão qual deveria ser o nº de unidades para se deixar em estoque de forma que tenhamos 2 dias de estoque disponível?
- Qual categoria mais abastecida nas lojas?
- Qual loja teve maior vendas dessa categoria?
- Por dia, qual o percentual de unidades não abastecidas em relação ao total de unidades solicitadas para reabastecimento?

## 4. Diagrama

Após definir qual o objetivo do projeto e as perguntas a serem respondidas através de análises, foi desenhado o diagrama da base de dados, para ter uma visão de quais serão os relacionamentos entre tabelas, quais serão os atributos e entender se os dados serão suficientes para responder aos questionamentos citados anteriormente.



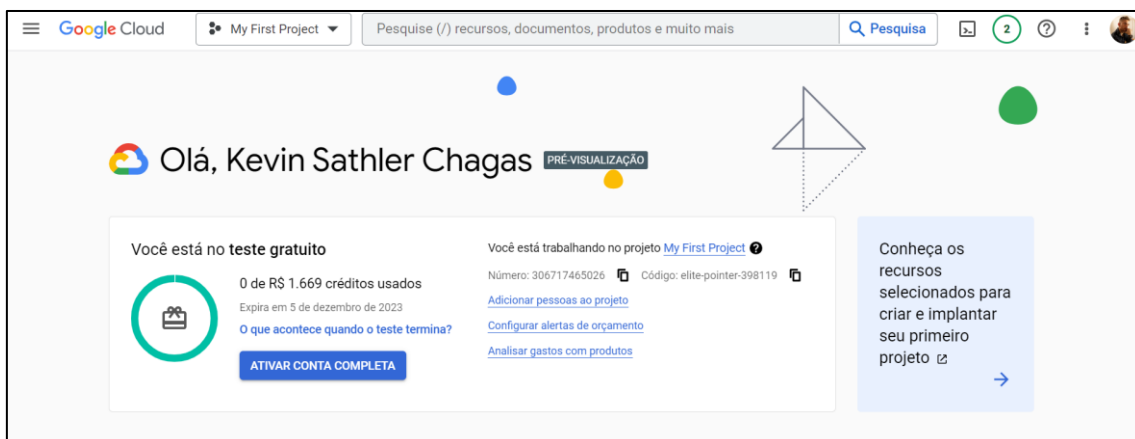
## 5. Metadados

1. **Tabela HUB:** Esta tabela representa todas as lojas que são diariamente abastecidas.
  - a. **HUB\_Code:** O código padrão que representa a loja. Este código é composto por 3 letras e, logo em seguida, 3 dígitos.
  - b. **HUB\_Name:** Uma string com nome da loja, em São Paulo.
2. **Tabela Produto:** Esta tabela representa todas as lojas que são diariamente abastecidas.
  - a. **EAN:** É o código de barras do produto, este é a chave primária da tabela produto. Cada produto tem um EAN representado por uma string.
  - b. **Name:** Uma string, com o nome do produto.
  - c. **Category\_Group:** Uma string que representa a categoria do grupo que o produto faz parte, este grupo é algo abrangente como por exemplo “Mercearia”, “Líquidos” ou “Saneantes”.
  - d. **Category\_1:** Uma string que representa a categoria detalhada do produto, é um subgrupo dentro da categoria. Por exemplo: “Mercearia” pode conter diversas subcategorias como “grãos”, “Molhos”, etc..
3. **Tabela Ordem de transferência:** Esta tabela contém o registro histórico de abastecimento do Centro de Distribuição para as lojas com informações relevantes para análises e rastreabilidade.
  - a. **TO\_Line\_Item:** Esta é a primary key desta tabela. Dentro de uma ordem de transferência (um pedido de abastecimento do CD para a loja) cada item deste pedido (Reconhecido pelo EAN) tem um TO\_Line\_Item como identificador. O TO\_Line\_Item deve um número inteiro maior que zero.

- b. **Transfer\_Order\_Number:** Este campo tem um número inteiro não negativo que representa a ordem de abastecimento. Cada ordem de abastecimento pode ter diversos Produtos e, conseqüentemente, diversos TO\_Line\_Item.
- c. **EAN:** Foreign Key da tabela “Produto”, representado por uma string. Este campo é composto por números, porém pode iniciar com “0”, dessa forma é necessário que este atributo seja uma string pois se fosse integer este não conseguiria apresentar o “0” ao início do número.
- d. **Create\_Date:** A data de criação da ordem de abastecimento. A data deve ser menor que a data atual.
- e. **Original\_Qty:** A quantidade solicitada para ser abastecida do CD para a loja. Este atributo deve sempre ser um número inteiro maior que zero.
- f. **Picked\_Qty:** A quantidade que realmente foi coletada do CD para abastecer a loja, este campo pode ser menor ou igual ao campo “Original\_Qty” mas nunca menor que zero.
- g. **Status\_Line\_Item:** Um campo que contém um texto identificador do status do produto dentro da ordem de transferência. Este identificador dirá se o produto foi “Separado”, “Não separado” ou “Produto indisponível”.
- h. **Transfer\_From:** Este campo irá conter o “HUB\_Code” como Foreign Key. É o código do CD ou loja que está transferindo os produtos para abastecer alguma loja.
- i. **Transfer\_To:** Este campo irá conter o “HUB\_Code” como Foreign Key. É o código da loja que está recebendo os produtos para serem abastecidos.
- j. **Status\_TO:** Um campo que contém um texto identificador do status da ordem de transferência. Este identificador dirá se a ordem está “Aberta para separação”, “Fechada”, “Abastecida”, “Cancelada”, etc..

## 6. Iniciando: Cadastro e registro no Google Cloud

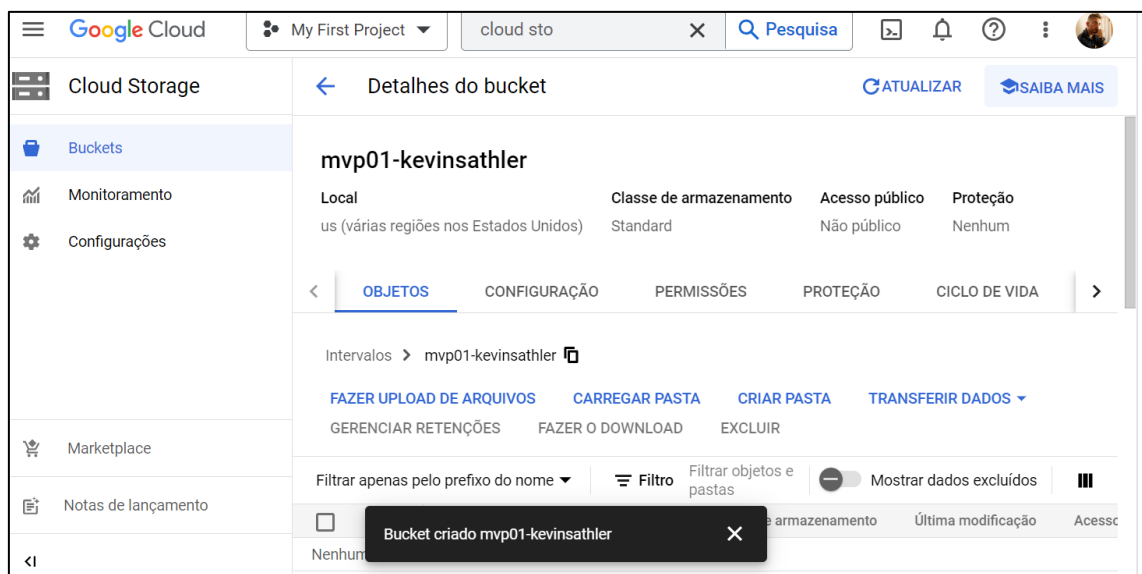
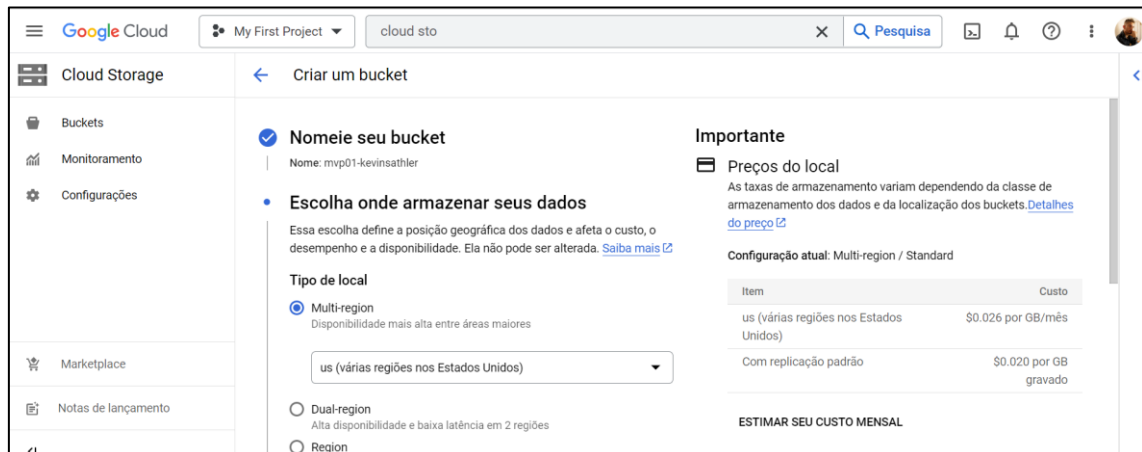
Para iniciar o trabalho, foi criada uma conta “free trial” no Google Cloud de forma que seja possível realizar todo o processo.



## 7. Criação do bucket

Buckets são os recipientes que armazenam os dados, de forma que tudo que seja armazenado em um Cloud Storage precisa estar em um bucket. Estes buckets são usados para armazenar e controlar o acesso aos dados.

Dessa forma, na imagem abaixo é possível verificar a criação do bucket feita com o nome “mvp01-kevinsathler”



My First Project
X
Pesquisa
>
4
?
:
K

---

Agora é possível capturar mais rapidamente as mudanças nas fontes do Google Cloud Storage e da Amazon S3 usando transferências orientadas a eventos. Assim, você consegue agir com base nos seus dados quase em tempo real. Para começar, crie um job de transferência com um fluxo de eventos do Pub/Sub ou do AWS SQS configurado para enviar notificações de eventos quando objetos forem criados ou atualizados.

[CRIAR JOB DE TRANSFERÊNCIA](#)
  
  
[SAIBA MAIS >](#)

Cloud Storage e as páginas de observabilidade do bucket. Com a tecnologia das Operações do Cloud, esses painéis podem ser personalizados para projetos específicos

[FAZER O TESTE AGORA](#)

	Filtro	Filtrar intervalos				
<input type="checkbox"/>	Nome ↓	Criado em	Tipo de local	Local	Classe de ar	
<input type="checkbox"/>	mvp-kevinsathler-temp	28 de set. de 2023 13:47:44	Multi-region	us	Standard	⋮
<input type="checkbox"/>	mvp-kevinsathler	28 de set. de 2023 13:39:40	Multi-region	us	Standard	⋮

Com os buckets criados, foi feito o upload dos arquivos em CSV neste bucket.

Google Cloud

My First Project

buck

Pesquisa

4

?

⋮

K

Detalhes do bucket

ATUALIZAR

SAIBA MAIS

mvp-kevinsathler

Local

Classe de armazenamento

Acesso público

Proteção

us (várias regiões nos Estados Unidos)

Standard

Não público

Nenhum

OBJETOS

CONFIGURAÇÃO

PERMISSÕES

PROTEÇÃO

CICLO DE VIDA

OBSERVABILIDADE

RE

Intervalos > mvp-kevinsathler

FAZER UPLOAD DE ARQUIVOS

CARREGAR PASTA

CRIAR PASTA

TRANSFERIR DADOS

GERENCIAR RETENÇÕES

FAZER O DOWNLOAD










EXCLUIR

Filtrar apenas pelo prefixo do nome

Filtro

Filtrar objetos e pastas

Mostrar dados excluídos

<input type="checkbox"/>	Nome	Tamanho	Tipo	Criado	Classe de armazenamento	
<input type="checkbox"/>	 <a href="#">HUB Info_CSV.csv</a>	1,3 KB	text/csv	28 de set. de 2023 13:50:29	Standard	 
<input type="checkbox"/>	 <a href="#">ProductsCSV.csv</a>	385,8 KB	text/csv	28 de set. de 2023 13:50:54	Standard	 
<input type="checkbox"/>	 <a href="#">Transfer_Order_CSV.csv</a>	44 MB	text/csv	28 de set. de 2023 13:50:42	Standard	 

## 9. Criação do pipeline

A ferramenta utilizada para a criação do pipeline foi o Data Fusion.

**Definição:** “O Cloud Data Fusion é um serviço totalmente gerenciado de integração de dados corporativos com nuvem nativa. Ele pode ser usado para gerar e gerenciar pipelines de dados.

A interface da Web do Cloud Data Fusion permite criar soluções escalonáveis de integração de dados para limpar, preparar, combinar, transferir e transformar dados, sem precisar gerenciar a infraestrutura.” (<https://cloud.google.com/data-fusion/docs/concepts/overview?hl=pt-br>)

### 9.1. Vinculando os GCS

Para iniciar o pipeline, foi criado um “box” com o GCS para cada arquivo carregado posteriormente no bucket. Para a criação destes GCS foi considerada a primeira linha como cabeçalho, de forma que o output fosse congruente com o esperado.

The screenshot shows the 'GCS Properties' configuration window in the Cloud Data Fusion console. The 'Label' is 'Products'. Under the 'Connection' tab, 'Use Connection' is set to 'YES', and the 'Connection' is 'Cloud Storage Default'. Under the 'Basic' tab, the 'Reference Name' is 'mvp01-kevinasathier/ProductsCSV.csv', the 'Path' is 'gs://mvp01-kevinasathier/ProductsCSV.csv', and the 'Format' is 'csv'. On the right, the 'Output Schema' table lists the following fields:

Field	Type	Nullable	Default
Category_Group	string	*	
Category_Level_1	string	*	
JOKR_Product_Name	string	*	
Barcode	string	*	

The screenshot shows the 'GCS Properties' configuration window in the Cloud Data Fusion console for a connection named 'HUBs'. The 'Label' is 'HUBs'. Under the 'Connection' tab, 'Use Connection' is set to 'YES', and the 'Connection' is 'Cloud Storage Default'. Under the 'Basic' tab, the 'Reference Name' is 'mvp01-kevinasathier/HUBInfo\_CSV.csv', the 'Path' is 'gs://mvp01-kevinasathier/HUBInfo\_CSV.csv', and the 'Format' is 'csv'. On the right, the 'Output Schema' table lists the following fields:

Field	Type	Nullable	Default
Hub_Hub_Code	string	*	
Hub_Name	string	*	

Cloud Data Fusion | Studio

OPERATIONS | HUB | SYSTEM ADMIN

GCS Properties 8.22.2

Reads objects from a path in a Google Cloud Storage bucket.

Validate

Properties | Documentation

Label \*

Transfer\_order

Connection

Use Connection

YES

Connection \*

Cloud Storage Default

BROWSE CONNECTIONS

Basic

Reference Name \*

mvdp01-heavinsathier/Transfer\_Order\_CSV.csv

GETCODE

Path \*

gs://mvdp01-heavinsathier/Transfer\_Order\_CSV.csv

Format \*

csv

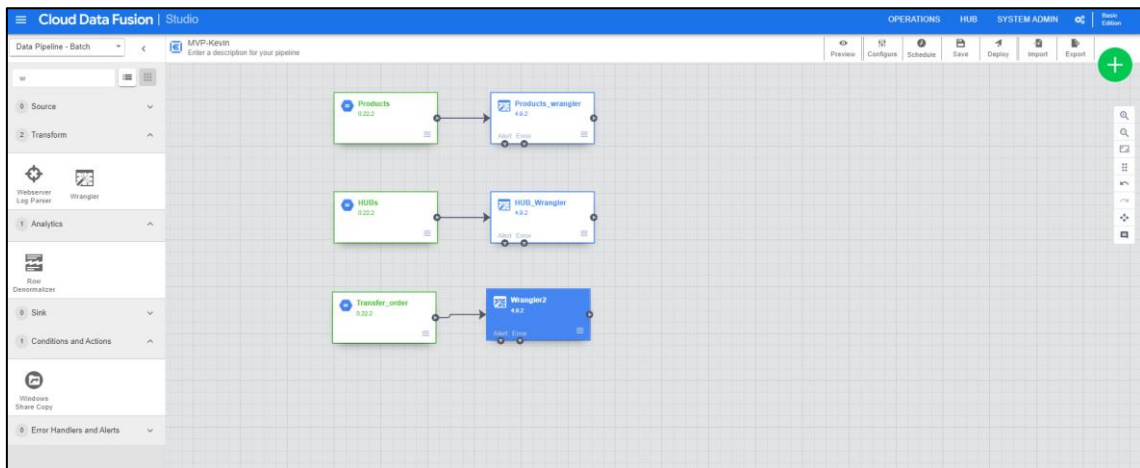
Output Schema

Actions

Transfer_Order_Number	int	+	+
TO_Line_Item_Number	int	+	+
Product_Barcode	long	+	+
Created_Date	date	+	+
Original_Quantity	int	+	+
Picked_Quantity	int	+	+
Picked_Date_Time	datetime	+	+
Status_Line_Item	string	+	+
Status_To	string	+	+
Transfer_From_Hub_Code	string	+	+
Transfer_To_	string	+	+

## 9.2. Wrangler

Cada um destes GCS passaram pelo processo de “Wrangler”. Durante este processo, o objetivo principal é que consigamos tratar a qualidade dos dados, fazendo filtros, limpeza de dados ou até mesmo a conversão de dados.





Durante o processo de “Wrangler”, alguns ajustes foram feitos. Na tabela “Produto” e na tabela “HUB” os únicos ajustes feitos foram uma alteração no nome das colunas, como apresentado abaixo:

Cloud Data Fusion | Studio

Wrangler Properties 4.9.2  
Wrangler - A interactive tool for data cleansing and transformation.

Properties Documentation

Input Schema

Field	Type	Actions
Category_Group	string	+ -
Category_Level_1	string	+ -
JOKR_Product_Name	string	+ -
Barcode	string	+ -

Label \*  
Products\_Wrangler

Input Selection and Prefilters

Input field name \*

Precondition Language  
☒ JEXL ☐ SQL

Precondition (JEXL)  
false

Directives

Recipe

```
1. rename Category_Group Grupo
2. rename Category_Level_1 Categoria
3. rename JOKR_Product_Name Nome
```

WRANGLE

User Defined Directives(UD) \*

Output Schema

Field	Type	Actions
Grupo	string	+ -
Categoria	string	+ -
Nome	string	+ -
Barcode	string	+ -

Cloud Data Fusion | Studio

Wrangler Properties 4.9.2  
Wrangler - A interactive tool for data cleansing and transformation.

Properties Documentation

Input Schema

Field	Type	Actions
Hub_Hub_Code	string	+ -
Hub_Name	string	+ -

Label \*  
HUB\_Wrangler

Input Selection and Prefilters

Input field name \*

Precondition Language  
☒ JEXL ☐ SQL

Precondition (JEXL)  
false

Directives

Recipe

```
1. rename Hub_Hub_Code Codigo_HUB
2. rename Hub_Name Nome_Hub
```

WRANGLE

User Defined Directives(UD) \*

Output Schema

Field	Type	Actions
Codigo_HUB	string	+ -
Nome_Hub	string	+ -

Na tabela “Transfer\_Order” algumas outras alterações foram realizadas:

- Foi percebida na coluna “Transfer\_to” a presença de “;” quando não deveria ter. Assim como apresentado nos metadados acima, as informações “Transfer\_to” são uma chave estrangeira da tabela “HUB” com um critério de ser composto por 3 letras e 3 números. Portanto os “;” foram localizados e removidos.
- Foi retirada uma coluna com o “Picked\_datetime” por dificuldades em realizar o “parse” da string para timestamp (DD/MM/YYYY HH:MM:SS). Ao dar um deploy no pipeline com esta coluna aparecia um erro o qual não consegui solucionar. Como esta coluna não irá afetar a análise que planejada realizar, decidi removê-la.
- Uma conversão da string “Created\_date” para timestamp no formato “DD/MM/YYYY”.

Wrangler Properties 4.0.2  
Wrangler - Interactive tool for data cleansing and transformation.

Properties Documentation

Input Schema

Transfer_Order_Numb	int	+	-
TO_Line_Item_Numb	int	+	-
Product_Barcode	string	+	-
Created_Date	string	+	-
Original_Quantity	int	+	-
Picked_Quantity	int	+	-
Picked_Date_Time	string	+	-
Status_Line_Item	string	+	-
Status_TO	string	+	-
Transfer_From_Hub_C	string	+	-
Transfer_To_	string	+	-

Label \*

Wrangler\_Transfer\_Order

Input Selection and Prefilters

Input field name \*

Precondition Language

☒ JEXL ☐ SQL

Precondition (JEXL)

false

Directives

Recipe

```

1 find-and-replace :Transfer_To_ s//;
2 drop :Picked_Date_Time
3 parse-as-simple-date :Created_Date MM/dd/yyyy
4 set-type :Product_Barcode string

```

Output Schema

Transfer_Order_Numb	int	+	-
TO_Line_Item_Numb	int	+	-
Product_Barcode	string	+	-
Created_Date	timestamp	+	-
Original_Quantity	int	+	-
Picked_Quantity	int	+	-
Status_Line_Item	string	+	-
Status_TO	string	+	-
Transfer_From_Hub_C	string	+	-
Transfer_To_	string	+	-

### 9.3. Joiner e big query

Após isso, com os dados dos GCS analisados de forma que a qualidade dos dados fosse garantida, foram criados dois “Joiner” para juntar as tabelas já tratadas.

Cloud Data Fusion | Studio

Joiner Properties 2.11.2  
Performs join operation on records from each input based on required inputs. If all the inputs are required inputs, inner join will be performed. Otherwise inner join will be performed on required inputs and records from non-required inputs will only be present if they match join criteria. If there are no required inp...

Properties Documentation

Input Schema

Transfer_Order_Number	int	+	-
TO_Line_Item_Number	int	+	-
Product_Barcode	string	+	-
Created_Date	timestamp	+	-
Original_Quantity	int	+	-
Picked_Quantity	int	+	-
Picked_Date_Time	datetime	+	-
Status_Line_Item	string	+	-
Status_TO	string	+	-
Transfer_From_Hub_Code	string	+	-
Transfer_To_	string	+	-

Transfer\_Order with Product

Basic

Fields \*

Transfer\_Order\_Wrangler

Products\_Wrangler

Join Type

Outer

Required Inputs

☒ Transfer\_Order\_Wrangler ☐ Products\_Wrangler

Join Condition Type

☒ Basic ☐ Advanced

Join Condition

Transfer\_Order\_Wrangler Product\_Barcode

Products\_Wrangler Barcode

Output Schema

Transfer_Order_Number	int	+	-
TO_Line_Item_Number	int	+	-
Product_Barcode	string	+	-
Created_Date	timestamp	+	-
Original_Quantity	int	+	-
Picked_Quantity	int	+	-
Picked_Date_Time	datetime	+	-
Status_Line_Item	string	+	-
Status_TO	string	+	-
Transfer_From_Hub_Code	string	+	-
Transfer_To_	string	+	-
Grupo	string	+	-
Categoria	string	+	-
Nome	string	+	-
Barcode	string	+	-

Cloud Data Fusion | Studio

Joiner Properties 2.11.2  
Performs join operation on records from each input based on required inputs. If all the inputs are required inputs, inner join will be performed. Otherwise inner join will be performed on required inputs and records from non-required inputs will only be present if they match join criteria. If there are no required inp...

Properties Documentation

Input Schema

Codigo_HUB	string	+	-
Nome_Hub	string	+	-

Transfer\_Order with Product

Basic

Fields \*

HUB\_Wrangler

Transfer\_Order with Product

Join Type

Outer

Required Inputs

☐ HUB\_Wrangler ☒ Transfer\_Order with Product

Join Condition Type

☒ Basic ☐ Advanced

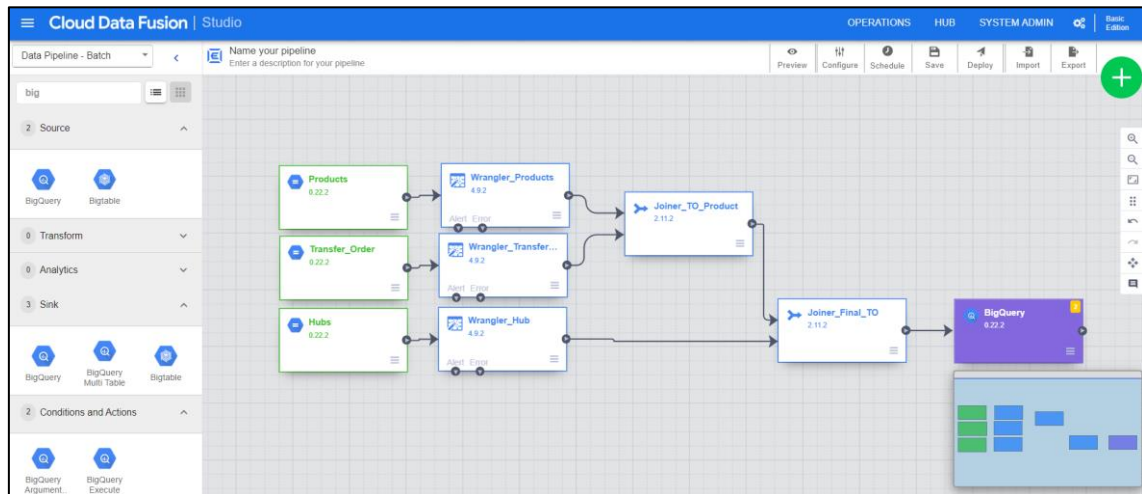
Join Condition

HUB\_Wrangler Codigo\_HUB

Output Schema

Nome_Hub	string	+	-
Transfer_Order_Number	int	+	-
TO_Line_Item_Number	int	+	-
Product_Barcode	string	+	-
Created_Date	timestamp	+	-
Original_Quantity	int	+	-
Picked_Quantity	int	+	-
Picked_Date_Time	datetime	+	-
Status_Line_Item	string	+	-
Status_TO	string	+	-
Transfer_From_Hub_Code	string	+	-
Transfer_To_	string	+	-
Grupo	string	+	-
Categoria	string	+	-
Nome	string	+	-
Barcode	string	+	-

O Join final, resultante destas junções foi chamado de “Joiner\_Final\_TO”. Este Join possui toda as informações das Transfer Order, junto ao nome e categorias do produto (proveniente da tabela Product) e o nome da loja (proveniente da tabela HUB).

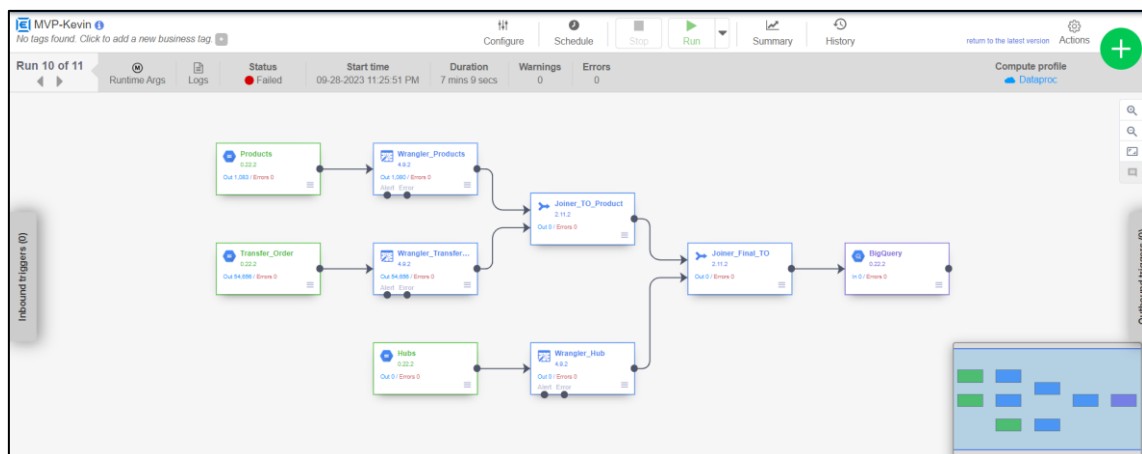


Como apresentado na imagem acima, o Join final foi lançado no Big Query para que possamos fazer todas as análises para chegar aos resultados planejados ao início do projeto.

## 10. Deploy, execução e erros:

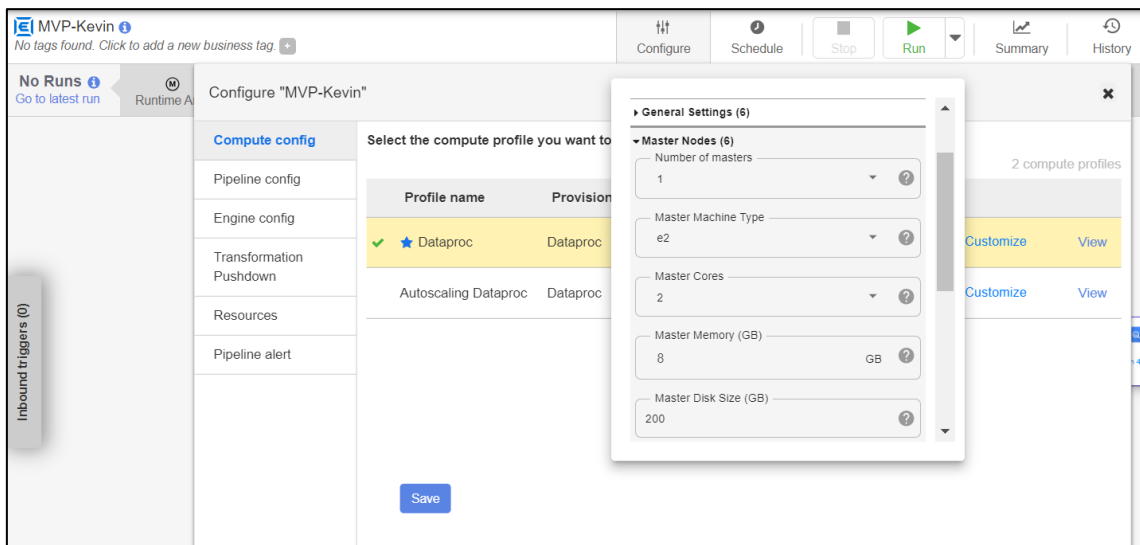
Após a conclusão do pipeline, foi feito o “Deploy” do mesmo e então o pipeline foi executado através do “Run”.

Conforme observado na imagem abaixo, diversas tentativas foram realizadas de executá-lo, porém sempre ocorria algum erro. Esta foi a maior dificuldade do projeto: compreender o erro e encontrar uma solução para ele.



Os principais erros encontrados, que foram solucionados conforme ia identificando-os, pude perceber:

- Um erro apontando que havia “Disk\_Total\_GB” quota insuficiente. Meu projeto estava exigindo 3000 GB enquanto havia disponibilizado para uso somente 2048 GB.
  - Para tratar este erro, foi necessário que acessasse as configurações do pipeline e customizasse o Dataproc.
  - O parâmetro de configuração “Master Node” estava exigindo 1000 GB de cada Node, exigindo um total de 3000 GB sendo que o disponível eram apenas 2048GB. Portanto para solucionar o problema, foi necessário apenas reduzir estes 1000 GB para 200 GB.

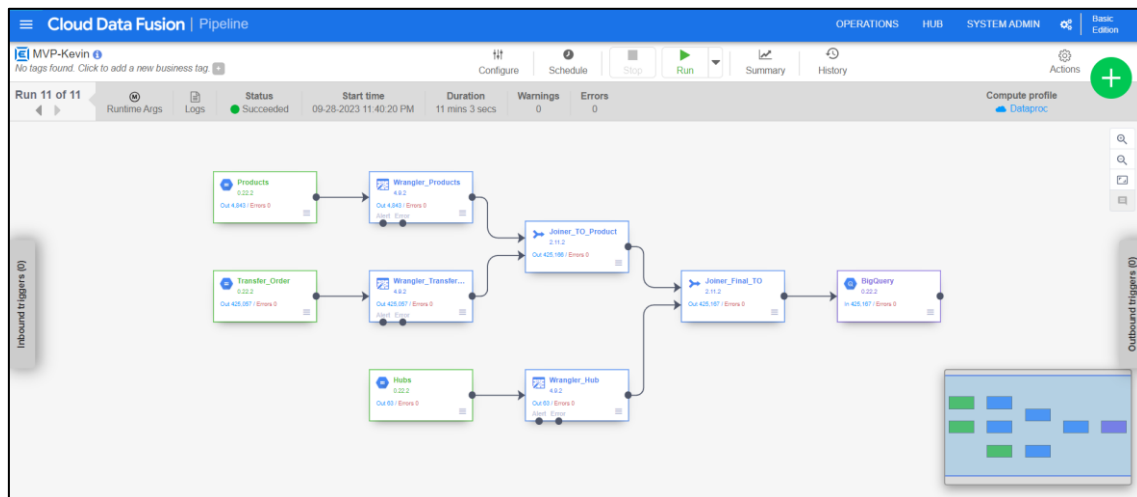


Além disso, outros dois problemas que dificultaram as execuções do pipeline foram:

- Incongruência entre características de dados durante etapas do pipeline.
  - Foram verificadas todas etapas do pipeline e qual classificação os dados estavam, dessa forma identifiquei que o “EAN” estava como “long” em um momento da pipeline e como “string” em outro momento. Isso estava resultando em um erro.
  - Para solucioná-lo revisei as etapas e garanti que a coluna estava sempre mantendo o mesmo formato dos passos anteriores, caso não houvesse feito nenhuma conversão utilizando o “Wrangler”

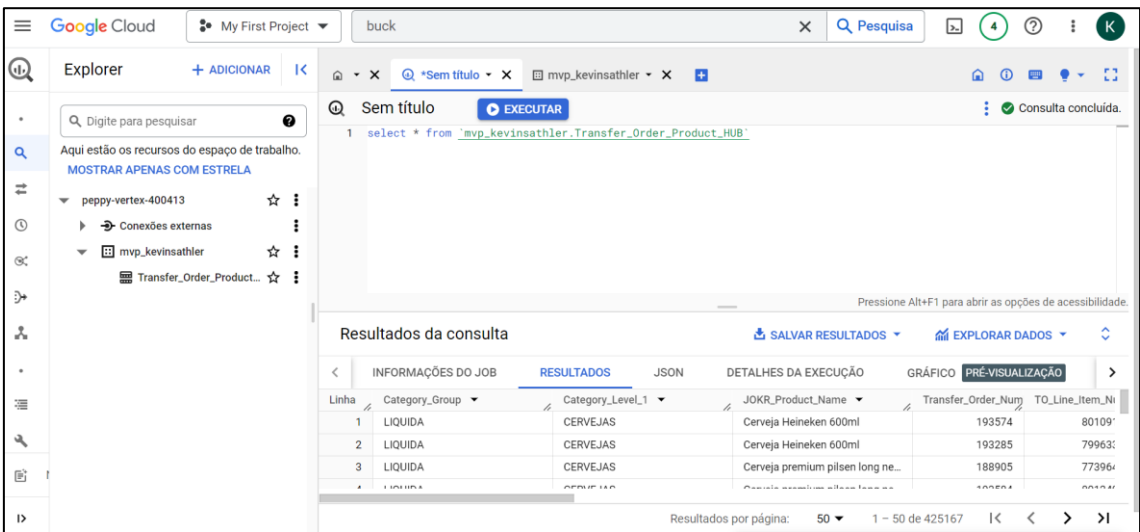
- Dificuldades em compreensão do parse de uma string como Datetime.
  - A coluna “Picked\_datetime” continha uma informação no formato string. Porém para utilizá-la corretamente havia feito um parse em seu formato durante o Wrangler para timestamp “DD/MM /YYYY HH:MM:SS”, porém esta coluna continuava dando problema constantemente.
  - Após diversas tentativas de parse e após utilizar vários formatos diferentes, o erro estava persistindo. Tentei converter esta coluna para string e utilizá-la mesmo assim, porém o erro aparecia com uma mensagem um pouco diferente.
  - Como esta informação não comprometeria as análises que gostaria de fazer, foi decidido removê-la utilizando o Wrangler.

Dessa forma, após 10 tentativas de executar o pipeline a tentativa de número 11 rodou normalmente!



## 11. Análise de dados:

Após o pipeline concluído, então os dados estavam prontos para serem analisados através do Big Query. Então a primeira consulta foi realizada e o retorno foi como o esperado.



Google Cloud My First Project buck Pesquisa

Explorer + ADICIONAR

Q Sem título EXECUTAR

```
1 select * from `mvp_kevinsathier.Transfer_Order_Product_HUB`
```

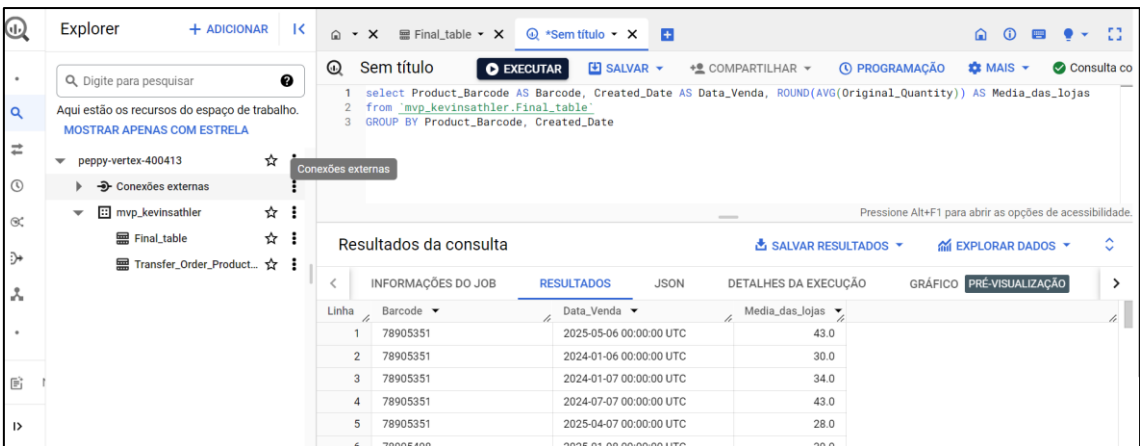
Resultados da consulta

Linha	Category_Group	Category_Level_1	JOKR_Product_Name	Transfer_Order_Num	TO_Line_Item_Ni
1	LIQUIDA	CERVEJAS	Cerveja Heineken 600ml	193574	80109
2	LIQUIDA	CERVEJAS	Cerveja Heineken 600ml	193285	79963
3	LIQUIDA	CERVEJAS	Cerveja premium pilsen long ne...	188905	77396
4	LIQUIDA	CERVEJAS	Cerveja premium pilsen long ne...	188905	77396

Resultados por página: 50 1 - 50 de 425167

Então, considerando que a base de dados estava pronta para ser utilizada, chegou o momento de responder as perguntas feitas ao início do projeto:

### 11.1. Qual a média necessária, em unidades, de estoque por dia, por produto?



Explorer + ADICIONAR

Q Sem título EXECUTAR

```
1 select Product_Barcode AS Barcode, Created_Date AS Data_Venda, ROUND(AVG(Original_Quantity)) AS Media_das_lojas
2 from `mvp_kevinsathier.Final_table`
3 GROUP BY Product_Barcode, Created_Date
```

Resultados da consulta

Linha	Barcode	Data_Venda	Media_das_lojas
1	78905351	2025-05-06 00:00:00 UTC	43.0
2	78905351	2024-01-06 00:00:00 UTC	30.0
3	78905351	2024-01-07 00:00:00 UTC	34.0
4	78905351	2024-07-07 00:00:00 UTC	43.0
5	78905351	2025-04-07 00:00:00 UTC	28.0

11.2 Quantas unidades de cada produto são vendidas por dia em cada loja?

Explorer

+ ADICIONAR

<

Digite para pesquisar

Aqui estão os recursos do espaço de trabalho.

MOSTRAR APENAS COM ESTRELA

▼

peppy-vertex-400413

☆

⋮

▶

Conexões externas

⋮

▼

mvp\_kevinsathler

☆

⋮

Final\_table

☆

⋮

Transfer\_Order\_Product...

☆

⋮

Final\_table

×

Q \*Sem título

+

EXECUTAR

SALVAR

COMPARTILHAR

PROGRAMAÇÃO

MAIS

Consulta co

```
1 select Product_Barcode AS Barcode, Created_Date AS Data_Venda, Transfer_To_ AS Loja, SUM(Original_Quantity) AS
2 Venda_por_dia
3 From `mvp_kevinsathler.Final_table`
4 GROUP BY Product_Barcode, Created_Date, Loja
5
6
7
```

Resultados da consulta

SALVAR RESULTADOS

EXPLORAR DADOS

<

INFORMAÇÕES DO JOB

RESULTADOS

JSON

DETALHES DA EXECUÇÃO

GRÁFICO

PRÉ-VISUALIZAÇÃO

>

Linha	Barcode	Data_Venda	Loja	Venda_por_dia
1	78905351	2025-05-06 00:00:00 UTC	SAO035	72
2	78905351	2024-01-06 00:00:00 UTC	SAO035	96
3	78905351	2024-01-07 00:00:00 UTC	SAO035	60
4	78905351	2024-07-07 00:00:00 UTC	SAO035	60
5	78905351	2025-04-07 00:00:00 UTC	SAO035	60

```
SELECT Product_Barcode AS Barcode, Created_Date AS Data_Venda,
ROUND(AVG(Original_Quantity)) AS Media_das_lojas
FROM `mvp_kevinsathler.Final_table`
GROUP BY Product_Barcode, Created_Date
```

11.3. Qual o desvio padrão de cada produto, considerando quantas unidades são vendidas por dia, em um período de 60 dias?

Google Cloud

My First Project

Pesquise (/) recursos, documentos, produtos e muito mais

Pesquisa

Explorer

+ ADICIONAR

<

Digite para pesquisar

Aqui estão os recursos do espaço de trabalho.

MOSTRAR APENAS COM ESTRELA

▼

peppy-vertex-400413

☆

⋮

▶

Conexões externas

⋮

▼

mvp\_kevinsathler

☆

⋮

Final\_table

☆

⋮

Transfer\_Order\_Product...

☆

⋮

Final\_table

×

Q \*Sem título

+

EXECUTAR

SALVAR

COMPARTILHAR

PROGRAMAÇÃO

MAIS

Consulta co

```
1 SELECT Product_Barcode AS Barcode, EXTRACT( Date FROM Created_Date) AS Data_Venda, ROUND(STDDEV
2 (Original_Quantity)) AS Desvio_Padrao
3 FROM `mvp_kevinsathler.Final_table`
4 GROUP BY Product_Barcode, Data_Venda
5 HAVING Data_Venda >= DATE_ADD(current_date(), interval -60 day)
6
7
```

Resultados da consulta

SALVAR RESULTADOS

EXPLORAR DADOS

<

INFORMAÇÕES DO JOB

RESULTADOS

JSON

DETALHES DA EXECUÇÃO

GRÁFICO

PRÉ-VISUALIZAÇÃO

>

Linha	Barcode	Data_Venda	Desvio_Padrao
1	78905351	2025-05-06	32.0
2	78905351	2024-01-06	28.0
3	78905351	2024-01-07	30.0
4	78905351	2024-07-07	29.0
5	78905351	2025-04-07	30.0

```
SELECT Product_Barcode AS Barcode, EXTRACT( Date FROM Created_Date) AS Data_Venda,
ROUND(STDDEV(Original_Quantity)) AS Desvio_Padrao
FROM `mvp_kevinsathler.Final_table`
GROUP BY Product_Barcode, Data_Venda
HAVING Data_Venda >= DATE_ADD(current_date(), interval -60 day)
```

#### 11.4. Qual a média necessária de armazenagem por dia, por produto, mais 50% do desvio padrão?

The screenshot shows the Google Cloud BigQuery interface. On the left, there's a sidebar with a search bar and a list of resources including 'peppy-vertex-400413', 'Conexões externas', 'mvp\_kevinsathler', 'Final\_table', and 'Transfer\_Order\_Product...'. The main area displays a SQL query titled 'Sem título' with the following code:

```
1 SELECT Product_Barcode AS Barcode, EXTRACT( Date FROM Created_Date) AS  
2 (STDDEV(Original_Quantity))) AS Saida_Media_com_Desvpad  
3 FROM `mvp_kevinsathler.Final_table`  
4 GROUP BY Product_Barcode, Data_Venda  
5 HAVING Data_Venda >= DATE_ADD(current_date(), interval -60 day)
```

Below the query, the 'Resultados da consulta' section shows a table with the following data:

Linha	Barcode	Data_Venda	Saida_Media_com_D
1	78905351	2025-05-06	548.0
2	78905351	2024-01-06	832.0
3	78905351	2024-01-07	870.0
4	78905351	2024-07-07	245.0
5	78905351	2025-04-07	978.0
6	78905498	2025-01-08	536.0

```
SELECT Product_Barcode AS Barcode, EXTRACT( Date FROM Created_Date) AS  
Data_Venda, ((SUM(Original_Quantity) + ROUND(STDDEV(Original_Quantity))) AS  
Saida_Media_com_Desvpad  
FROM `mvp_kevinsathler.Final_table`  
GROUP BY Product_Barcode  
HAVING Data_Venda >= DATE_ADD(current_date(), interval -60 day)
```

#### 11.5. Baseado no estoque + desvio padrão qual deveria ser o nº de unidades para se deixar em estoque de forma que tenhamos 2 dias de estoque disponível?

The screenshot shows the Google Cloud BigQuery interface. On the left, there's a sidebar with a search bar and a list of resources including 'peppy-vertex-400413', 'Conexões externas', 'mvp\_kevinsathler', 'Final\_table', and 'Transfer\_Order\_Product...'. The main area displays a SQL query titled 'Sem título' with the following code:

```
1 SELECT Product_Barcode AS Barcode, EXTRACT( Date FROM Created_Date) AS Data_Venda, ((SUM(Original_Quantity) + ROUND  
2 (STDDEV(Original_Quantity)))*2) AS Saida_Media_com_Desvpad  
3 FROM `mvp_kevinsathler.Final_table`  
4 GROUP BY Product_Barcode, Data_Venda  
5 HAVING Data_Venda >= DATE_ADD(current_date(), interval -60 day)
```

Below the query, the 'Resultados da consulta' section shows a table with the following data:

Linha	Barcode	Data_Venda	Saida_Media_com_D
1	78905351	2025-05-06	1096.0
2	78905351	2024-01-06	1664.0
3	78905351	2024-01-07	1740.0
4	78905351	2024-07-07	490.0
5	78905351	2025-04-07	1956.0
6	78905498	2025-01-08	1072.0

```
SELECT Product_Barcode AS Barcode, EXTRACT( Date FROM Created_Date) AS Data_Venda, ((SUM(Original_Quantity) + ROUND(STDDEV(Original_Quantity)))*2) AS  
Saida_Media_com_Desvpad  
FROM `mvp_kevinsathler.Final_table`  
GROUP BY Product_Barcode, Data_Venda  
HAVING Data_Venda >= DATE_ADD(current_date(), interval -60 day)
```



## 11.6. Quais são as 3 categorias mais vendidas?

The screenshot shows the Google Cloud BigQuery console. On the left, a sidebar displays the project structure with a search bar and a list of resources. The main area contains a SQL query editor with the following code:

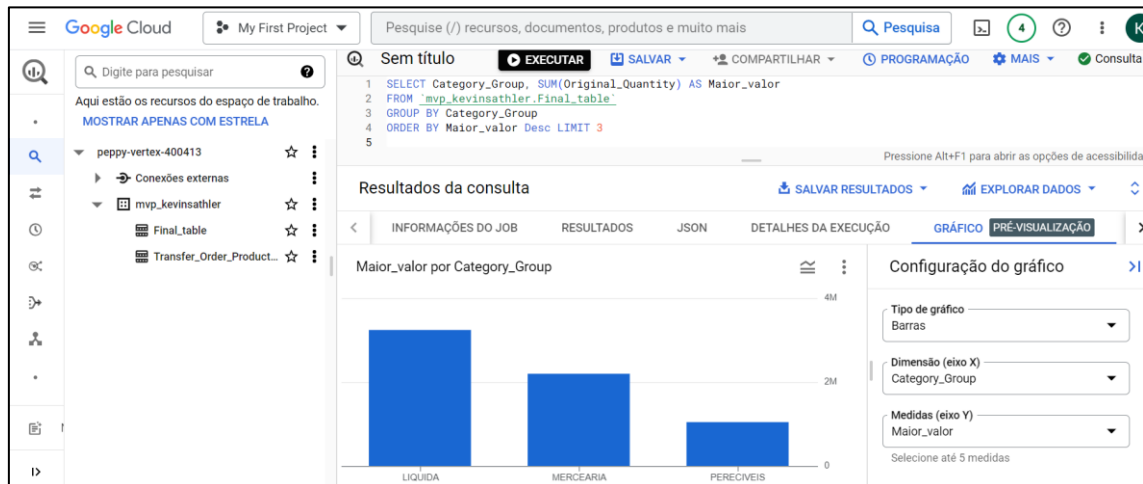
```
1 SELECT Category_Group, SUM(Original_Quantity) AS Maior_valor
2 FROM `mvp_kevinsathier.Final_table`
3 GROUP BY Category_Group
4 ORDER BY Maior_valor Desc LIMIT 3
```

Below the query editor, the 'Resultados da consulta' (Query Results) tab is active, showing a table with 3 rows and 2 columns:

Linha	Category_Group	Maior_valor
1	LIQUIDA	3245116
2	MERCEARIA	2198049
3	PERECIVEIS	1036756

```
SELECT Category_Group, SUM(Original_Quantity) AS Maior_valor
FROM `mvp_kevinsathier.Final_table`
GROUP BY Category_Group
ORDER BY Maior_valor Desc LIMIT 3
```

Como este output teve menos linhas, foi possível fazer a geração de um gráfico para análise visual das informações.



### 11.7. Qual loja teve maior vendas dessa categoria?

The screenshot shows the Google Cloud Console interface. On the left, the 'Explorer' pane displays the project structure, including 'Final\_table'. The main pane shows a SQL query titled 'Sem título' with the following code:

```
1 SELECT Hub_Name, SUM(Original_Quantity) AS Quantidade_Vendida
2 FROM `mvp_kevinsathler.Final_table`
3 WHERE Category_Group IN
4 (SELECT Category_Group
5 FROM `mvp_kevinsathler.Final_table`
6 GROUP BY Category_Group
7 ORDER BY SUM(Original_Quantity) Desc LIMIT 1
8 )
9 AND Hub_Name IS NOT NULL
10 GROUP BY Hub_Name
11 ORDER BY Quantidade_Vendida DESC LIMIT 1
12
13
```

Below the query, the 'Resultados da consulta' pane shows a table with the following data:

Linha	Hub_Name	Quantidade_Vendida
1	Vila Medeiros	139844

```
SELECT Hub_Name, SUM(Original_Quantity) AS Quantidade_Vendida
FROM `mvp_kevinsathler.Final_table`
WHERE Category_Group IN
(SELECT Category_Group
FROM `mvp_kevinsathler.Final_table`
GROUP BY Category_Group
ORDER BY SUM(Original_Quantity) Desc LIMIT 1
)
AND Hub_Name IS NOT NULL
GROUP BY Hub_Name
ORDER BY Quantidade_Vendida DESC LIMIT 1
```

### 11.8. Por dia, qual o percentual de unidades não abastecidas em relação ao total de unidades solicitadas para reabastecimento?

The screenshot shows the Google Cloud Console interface. On the left, the 'Explorer' pane displays the project structure, including 'Final\_table'. The main pane shows a SQL query titled 'Sem título' with the following code:

```
1 SELECT Created_Date, CONCAT(ROUND((((SUM(Original_Quantity)-
2 SUM(Picked_Quantity))/SUM(Original_Quantity))*100),2),',','%') AS Indisponiveis
3 FROM `mvp_kevinsathler.Final_table`
4 GROUP BY Created_Date
```

Below the query, the 'Resultados da consulta' pane shows a table with the following data:

Linha	Created_Date	Indisponiveis
1	2025-05-06 00:00:00 UTC	0.43%
2	2024-01-06 00:00:00 UTC	0.79%
3	2024-01-07 00:00:00 UTC	0.4%

```
SELECT Created_Date, CONCAT(ROUND((((SUM(Original_Quantity)-
SUM(Picked_Quantity))/SUM(Original_Quantity))*100),2),',','%') AS Indisponiveis
FROM `mvp_kevinsathler.Final_table`
GROUP BY Created_Date
```

## **12. Considerações finais:**

Após a realização deste MVP pude compreender a complexidade na criação de uma base de dados com qualidade de informação, de forma que as informações estejam saudáveis. Seja devido aos diversos parâmetros e configurações encontrados durante a criação do pipeline, ou seja, pela própria complexidade de algumas análises a serem realizadas.

Porém, a utilidade destes aprendizados já está sendo percebidas em minhas rotinas profissionais na criação de diversas queries que estão em tasks agendadas para serem atualizadas diariamente. Transformando-se assim em dashboards e auxiliando-me em decisões estratégicas.

Portanto, junto a conclusão deste trabalho gostaria de deixar meus sinceros agradecimentos e uma sensação de satisfação devido aos aprendizados agregados.