

# MVP - Engenharia de Dados - Leandra Mara da Silva

## Introdução

Como trabalho de conclusão da disciplina de Engenharia de Dados, foi solicitado:

- Construir um pipeline de dados **utilizando tecnologias na NUVEM**.
  - deve envolver busca, coleta, modelagem, carga e análise dos dados

Todas as seções seguintes, explicitam o contexto do trabalho, área de interesse, problemas que serão resolvidos e como cheguei aos resultados.

## 1. Objetivo

### Área de interesse

Finanças pessoais | controle de gastos | análise de gastos

### O problema

Extratos bancários exibem transações bancárias diárias de entrada e de saída, mas não nos auxiliam em análises de gastos mensais, muito menos anuais. Com o advento da digitalização dos bancos, podemos dizer que tornou-se mais fácil acompanhar os gastos pessoais, mas esses dados ainda são apresentados para serem analisados em um contexto histórico, ou seja, não respondem questões como: qual o mês de maior gasto no ano, qual a média de gasto mensal ou a média de gasto anual, por exemplo.

Agrava-se à necessidade de gerir gastos, quando o correntista utiliza **várias contas bancárias** no pagamento de contas, já que não existe uma opção de extrato que **integre as transações de múltiplos bancos**, ou seja, o correntista precisará acessar diferentes fontes de dados e realizar o agrupamento dos gastos nas múltiplas contas. **Note:** Talvez o [Open Finance](#), iniciativa do Banco Central do Brasil, viabilize recursos como esses que envolvam a integração de gastos e transações.

Ao solicitar um extrato bancário, o app do banco solicita a escolha do mês de interesse e o sistema envia para o email do correntista o extrato daquele mês. A solicitação de um longo período deve ser feita mês a mês e resulta na resposta do banco com N emails e N arquivos, cada um para um mês. Isso acontece no caso de extratos do **Nubank**, por exemplo.

Outros bancos, permitem que você selecione o período de interesse, o que faz com que o banco envie ao correntista um único arquivo com todas as transações realizadas naquele período. É como funciona o banco digital **Inter**, por exemplo.

*Diante do exposto, percebe-se que acompanhar métricas de gastos mensais ou anuais sem o auxílio de ferramentas de análise de dados pode ser dispendioso, pois demanda **integrar as transações dos inúmeros bancos** onde o correntista possui conta bancária. Cada banco gera seus arquivos de transações (extratos bancários), o que nos disponibiliza fontes de dados distintas e que podem ter padrões, nomenclaturas e tipos de dados completamente diferentes. Isso torna esses dados bastante interessantes para passar por ciclos de ETL.*

## A solução

Dado o problema exposto na seção anterior, vamos realizar o agrupamento das transações bancárias de saída apresentadas em extratos bancários de diferentes instituições bancárias. Nosso objetivo é facilitar a análise de gastos, algo que já era do meu interesse.

Como dito na seção anterior, os dados das transações bancárias a serem analisadas são de diferentes bancos, havendo nomenclaturas e tipos de dados distintos, o que demanda a necessidade de limpezas e transformações.

Os passos de transformação, padronização e unificação desses dados estão detalhados no notebook de codificação realizado no Databricks.

A solução foi realizada usando:

- [Databricks Community Edition](#), como ferramenta de análise de dados em nuvem
- SQL e Python, como linguagens de programação

## Perguntas do trabalho

Quando se trata de finanças pessoais e controle de gastos, existem várias perguntas de interesse, inclusive envolvendo categorização de gastos.

Para o escopo deste trabalho, nos limitaremos a responder as perguntas a seguir, considerando a integração de dados do Inter e Nubank.

Essas perguntas estão organizadas em gastos mensais, gastos anuais e ranking.

### GASTOS MENSAIS

1. Qual o total gasto mensalmente?
2. Qual a média de gasto mensal?

## GASTOS ANUAIS

3. Qual o total gasto anualmente?
4. Qual a média de gasto anual?

## RANKING

5. Qual o ranking (top 10) de meses de maior gasto em cada ano?

## Perguntas para o Futuro

Em um futuro próximo, queremos responder perguntas relacionadas a categorização de gastos. No contexto de gestão de custos, é bastante comum a criação de categorias, facilitando a tomada de ações de corte de gastos não essenciais.

### Futuro - item 1:

Fora do escopo deste trabalho, desejo classificar os tipos de gastos apresentados nas contas correntes permitindo a análise de gastos por categorias. Essas categorias ainda não foram escolhidas, muito menos associadas aos gastos. Obs.: Nesta etapa, não trataremos o detalhamento de gastos referente a faturas de cartão de crédito, considerando este como uma "caixa preta" no primeiro momento.

### Futuro - item 2:

Gostaria de classificar os tipos de gastos também nas faturas de cartões de crédito, ainda sem integrá-los aos gastos da conta corrente. Obs.: Note que os dados/transações de cartão de crédito não ficam especificados em extratos, mas em outras fontes de dados (faturas de cartão).

### Futuro - passo 3:

Gostaria de unificar gastos de diferentes contas correntes e de diferentes faturas de cartões de crédito de forma a fazer uma análise integrada dos gastos pessoais, agrupando os percentuais de gasto por cada categoria.

Nenhum destes passos estão contemplados no escopo do trabalho atual e foram deixados aqui apenas como registro para a elaboração de perguntas futuras.

## 2. Dados do trabalho (Datasets) | Coleta dos Dados

### Arquivos, URL para download e Linhagem

Consideramos no trabalho 2 arquivos .CSV

- 1 Extrato Bancário com transações do Banco Inter (2023-2024):
  - o mesmo foi extraído do app do banco e salvo em URL compartilhada no Google Drive:  
[https://drive.google.com/file/d/15yanx\\_Nr2Kmrk1xvvcjyBtt3CQbKPT9V/view?usp=sharing](https://drive.google.com/file/d/15yanx_Nr2Kmrk1xvvcjyBtt3CQbKPT9V/view?usp=sharing)
- 1 Extrato Bancário com transações Nubank (2023-2024)

- o mesmo foi extraído do app do banco e salvo em URL compartilhada no Google Drive:  
<https://drive.google.com/file/d/1ydYy3AlinzREm4v8fDS2hK3Uxi9SjGTs/view?usp=sharing>
  - como esse banco disponibiliza extratos mensais, por entender que não atrapalha o escopo do trabalho referente a construção da pipeline, os 24 arquivos foram integrados em 1.

**Linhagem:** como já citado, os extratos bancários foram solicitados no app de cada banco.

**Colunas nas tabelas "Raw" e tabelas resultantes:** as tabelas e respectivas colunas serão apresentadas na seção de "Modelo de dados".

## Etapas ou estrutura da solução

Criamos um pipeline de dados em nuvem seguindo o modelo de arquitetura e organização de dados conhecido como "**Arquitetura Medallion**". As camadas contempladas são **Bronze, Silver e Gold**. O objetivo é melhorar a estrutura e a qualidade dos dados de forma incremental e progressiva à medida que fluem de uma camada para outra (das tabelas de camadas Bronze ⇒ Prata ⇒ Ouro).

Como é possível ver no Databricks, vamos abordar o processo em etapas, começando pela **leitura dos arquivos CSV** e salvando as tabelas com os dados originais na camada **Bronze**, passando pelas **transformações e tratamentos de dados** necessários para armazenar os dados tratados na camada **Silver** e, finalmente, disponibilizando o modelo de análise em uma **tabela flat** (resultante da operação de UNION) contendo todas as transações devidamente tratadas e padronizadas na camada **Gold**.

Nesta tabela de análise resultante na Gold, criaremos as **consultas SQL** para responder às perguntas desejadas. No contexto descrito, apresentamos no notebook todas as etapas do escopo deste trabalho, onde foram contemplados os requisitos do trabalho: coleta, carga, modelagem, análise dos dados e autoavaliação.

Abaixo, apresentamos a estrutura (sumário) presente no Databricks e que utilizamos para alcançar a solução. Tal notebook que pode ser acessado na URL citada no leia-me do repositório git: <https://github.com/leandra-mara/mvp-data-science-e-analytics>. A solução está estruturada da seguinte forma:

### Leitura dos Arquivos CSV

Leitura do Arquivo .CSV do Banco Inter [ *Requisito: Carga de Dados* ]

Leitura do Arquivo .CSV do Banco Inter [ *Requisito: Carga de Dados* ]

### Camada BRONZE - Dados Brutos

Criando o banco de dados: BRONZE

## **BANCO INTER**

Criando tabela de transações: Inter - Bronze  
Olhando: Tabela com transações do Inter - Bronze  
Olhando: Tipos de dados do Inter - Bronze  
Olhando: Tipos de transação do Inter - Bronze  
Olhando: NULOS na tabela do Inter - Bronze

## **BANCO NUBANK**

Criando tabela de transações: Inter - Bronze  
Olhando: Tabela com transações do Inter - Bronze  
Olhando: Tipos de dados do Inter - Bronze  
Olhando: Tipos de transação do Inter - Bronze  
Olhando: NULOS na tabela do Inter - Bronze

## **Camada SILVER - Limpeza e Transformação**

Criando o banco de dados: SILVER

### **BANCO INTER**

TESTE DE FILTRO 1: Valores Negativos no Inter  
TESTE DE FILTRO 2: Descrição != Leandra no Inter  
TESTE DE FILTRO 3: Identificando Investimentos no Inter  
TESTE DE EXTRAÇÃO: Infs. de data no Inter  
TESTE: padronização das transações

TESTE dos tratamentos - Inter  
Criando tabela de transações: Inter - Silver  
Incluindo coluna com o banco - Silver  
Tabela Resultante no Silver - banco INTER

### **BANCO NUBANK**

TESTE DE FILTRO 1: Valores Negativos no Nubank  
TESTE DE FILTRO 2: Descrição != Leandra no Nubank  
TESTE DE FILTRO 3: Identificando Investimentos no Nubank  
TESTE DE EXTRAÇÃO: Infs. de data no Nubank  
TESTE DE EXTRAÇÃO: Split da descrição  
TESTE: Padronização nome da transação

SELECT DE filtragem geral no Nubank  
Criando tabela de transações: Nubank - Silver  
Incluindo coluna com o banco - Silver  
Tabela Resultante no Silver - banco Nubank

## **Camada GOLD**

Criando o banco de dados: GOLD

Conferindo antes da UNION: propriedades - Inter  
Conferindo antes da UNION: num. de registros - Inter  
Conferindo antes da UNION: propriedades - Nubank

Conferindo antes da UNION: num. de registros - Nubank  
TESTE: consulta de união entre as tabelas

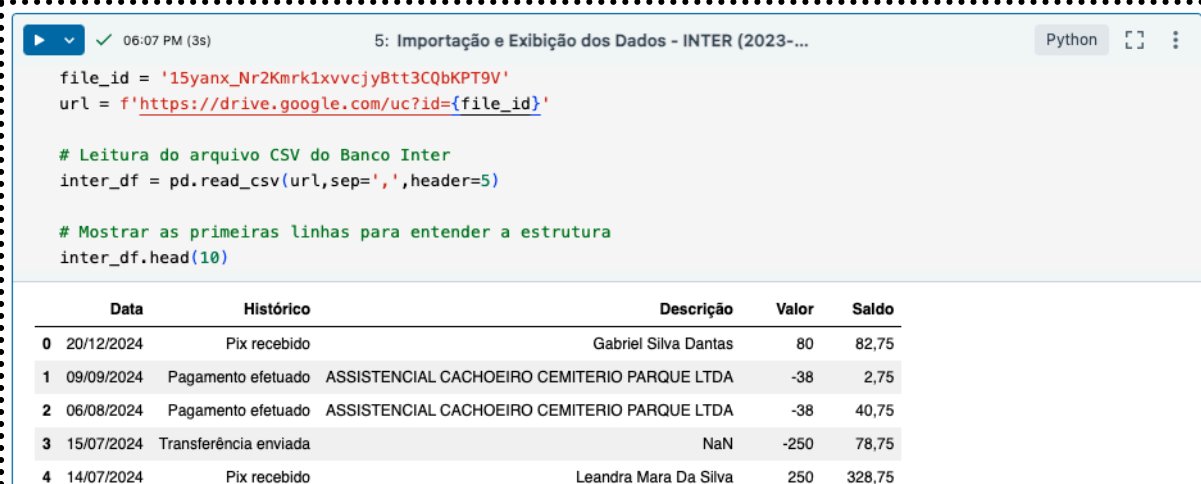
Criando tabela flat com transações filtradas e tratadas  
Olhando: Tabela Final - Gold

## Modelos, Catálogos e Carga de dados

### Dados de entrada (raw)

Segue imagens que apresentam uma amostra de como chegam os dados:

- No INTER (imagem abaixo):
  - a. **Data:** string; sem nulos; formato: 20/12/2024
  - b. **Histórico:** string; sem nulos
  - c. **Descrição:** string; com nulos
  - d. **Valor:** string; sem nulos; exemplos 80, -38, 2.324,52, - 2.324,52
  - e. **Saldo:** string; sem nulos; no formato do Valor



The screenshot shows a Jupyter Notebook interface with a title bar indicating the file is '5: Importação e Exibição dos Dados - INTER (2023-...'. The code cell contains the following Python code:

```
file_id = '15yanx_Nr2Kmrk1xvvcjyBtt3CQbKPT9V'  
url = f'https://drive.google.com/uc?id={file_id}'  
  
# Leitura do arquivo CSV do Banco Inter  
inter_df = pd.read_csv(url, sep=',', header=5)  
  
# Mostrar as primeiras linhas para entender a estrutura  
inter_df.head(10)
```

Below the code, the first 10 rows of the dataset are displayed in a table format:

	Data	Histórico	Descrição	Valor	Saldo
0	20/12/2024	Pix recebido	Gabriel Silva Dantas	80	82,75
1	09/09/2024	Pagamento efetuado	ASSISTENCIAL CACHOEIRO CEMITERIO PARQUE LTDA	-38	2,75
2	06/08/2024	Pagamento efetuado	ASSISTENCIAL CACHOEIRO CEMITERIO PARQUE LTDA	-38	40,75
3	15/07/2024	Transferência enviada	NaN	-250	78,75
4	14/07/2024	Pix recebido	Leandra Mara Da Silva	250	328,75

- No NUBANK (imagem abaixo):
  - a. **Data:** string; sem nulos; formato: 20/12/2024
  - b. **Valor:** string; sem nulos; exemplos: 80.00, -38.00, 2342.52, - 2342.52
  - c. **Identificador:** string; sem nulos; exemplos:  
66888d22-42fe-4318-868d-44eef7a6474f
  - d. **Descrição:** string; aceita nulos; exemplo: Transferência enviada pelo Pix - ENEL DISTRIBUICAO RIO - 33.050.071/0001-58 - ITAÚ UNIBANCO S.A. (0341) Agência: 911 Conta: 12768-6

```
06:18 PM (2s) 27: Importação e Exibição dos Dados - NUBANK (202... Python
```

```
file_id = '1ydYy3AIinzREm4v8fDS2hK3Uxi9SjGTs'
url = f'https://drive.google.com/uc?id={file_id}'

# Leitura do arquivo CSV do Banco Nubank
nubank_df = pd.read_csv(url, sep=',', header=0)

# Mostrar as primeiras linhas para entender a estrutura
nubank_df.head(10)
```

	Data	Valor	Identificador	Descrição
0	05/07/2024	281198.34	66888d22-42fe-4318-868d-44eef7a6474f	Transferência recebida pelo Pix - LEANDRA MARA...
1	06/07/2024	-600.00	66899882-751c-420f-b3b9-65689ca378f6	Transferência enviada pelo Pix - Benedita Sima...
2	08/07/2024	-706.00	668c4116-b560-4b0a-9e9b-818a0c62287a	Transferência enviada pelo Pix - Lafs Processa...
3	10/07/2024	-4000.00	668ed0c2-5350-4c85-b0da-e8593111c4c8	Transferência enviada pelo Pix - LEANDRA MARA ...
4	12/07/2024	-1340.15	66918b86-cd13-48a2-b2f4-35dca9a14648	Pagamento de boleto efetuado - SUL AMERICA COM...

## Qualidade dos dados


Os dados estão bem organizados e as diferenças existentes entre os mesmos e que poderiam dificultar as respostas são tratadas e explicadas na seção posterior quando detalhamos as tabelas geradas na camada silver.

Ainda assim, podemos aqui responder questões como:

- Os dados estão completos ou há campos nulos ou faltando? Sim, os dados estão completos. No databricks mostramos a análise de colunas com NULL e o único campo onde isso ocorre é onde o dado não é obrigatório. Por exemplo, não ter o nome de beneficiário em operações onde não existem beneficiários, como no caso de saque, por exemplo.
- Os dados seguem as regras esperadas? Sim. Não existem datas de transações realizadas no futuro e os valores das transações não contém informação com R\$0,00.
- Os dados refletem corretamente a realidade? Todos os dados em todas as colunas são coerentes e foram validados.

## 1. Tabelas da camada bronze

Na camada Bronze, os dados foram mantidos em sua forma bruta. O objetivo foi armazenar os dados como eles são (dados raw). Por meio de comparações, garantimos que os dados foram lidos corretamente.


 bronze


OverviewDetails

Filter tables


2 tables

Name

 inter\_raw\_table

 nubank\_raw\_table


bronze >

 inter\_raw\_table

OverviewSample DataDetailsHistory

About this table

Data source



Last updated

há 1 hora

Size

19.3KiB, 1 file


Description

Add description

Filter columns...

Column	Type
Data	string
Histórico	string
Descrição	string
Valor	string
Saldo	string


bronze >

 nubank\_raw\_table

OverviewSample DataDetailsHistory

About this table

Data source



Last updated

há 55 minutos

Size

17.6KiB, 1 file

Description

Add description

Filter columns...

Column	Type
Data	string
Valor	string
Identificador	string
Descrição	string

Tabela na Bronze: INTER_RAW_TABLE			
Coluna	Significado	Tipo	Valores   Categorias possíveis   Exemplo de formato
Data	coluna que representa a data de registro da	string	Exemplo: 20/12/2024



	transação; não podendo ser nulo		
Histórico	coluna que representa o tipo de transação englobando transações de entradas e saídas; não podendo ser nulo	string	<ol style="list-style-type: none"> <li>1. Transferência recebida</li> <li>2. Pix recebido devolvido</li> <li>3. Pix recebido</li> <li>4. Débito Renda Fixa</li> <li>5. Pagamento efetuado</li> <li>6. Compra no débito</li> <li>7. Saque</li> <li>8. Pagamento de Convênio</li> <li>9. Pagamento Tim</li> <li>10. Pagamento ENEL RJ</li> <li>11. Estorno</li> <li>12. Transferência enviada</li> <li>13. Pix enviado</li> <li>14. Eventos Renda Fixa</li> </ol>
Descrição	coluna que representa o beneficiário da transação; podendo ser nulo para transações que não demandam beneficiário (ex.: saque, etc).	string	Exemplo: PREDLINK REDE T LTDA EPP
Valor	coluna com o valor da transação, podendo ser positivo ou negativo, caso seja uma entrada ou saída da conta corrente; utiliza-se ponto para separação de milhares e vírgula como separação de decimal, quando os	string	Exemplos: <ul style="list-style-type: none"> <li>• 80</li> <li>• -38</li> <li>• 2.324,52</li> <li>• - 2.324,52</li> </ul>

	valores que representam centavos forem diferentes de zero; não podendo ser nulo		
Saldo	coluna com o valor acumulado somado ou subtraído a partir daquela transação; não podendo ser nulo  obs.: não teremos interesse neste dado	string	Mesmo formato do Valor

Tabela na Bronze: NUBANK_RAW_TABLE			
<u>Coluna</u>	<u>Significado</u>	<u>Tipo</u>	<u>Valores   Categorias possíveis   Exemplo de formato</u>
Data	coluna que representa a data de registro da transação; não podendo ser nulo	string	Exemplo: 20/12/2024
Valor	coluna com o valor da transação, podendo ser positivo ou negativo, caso seja uma entrada ou saída da conta corrente; utiliza-se ponto como separação de decimal; não podendo ser nulo	string	Exemplos: <ul style="list-style-type: none"> <li>• 80.00</li> <li>• -38.00</li> <li>• 2342.52</li> <li>• - 2342.52</li> </ul>
Identificador	coluna com uma sequencia de números e letras e que identificam a transação	string	66888d22-42fe-4318-868d-44eef7a6474f

Como pudemos ver nos dados brutos, os dados do Inter e do Nubank apresentam algumas colunas com nomes e conteúdos diferentes, formato de representação de Valor diferente, tipos de transação com nomenclaturas diferentes.

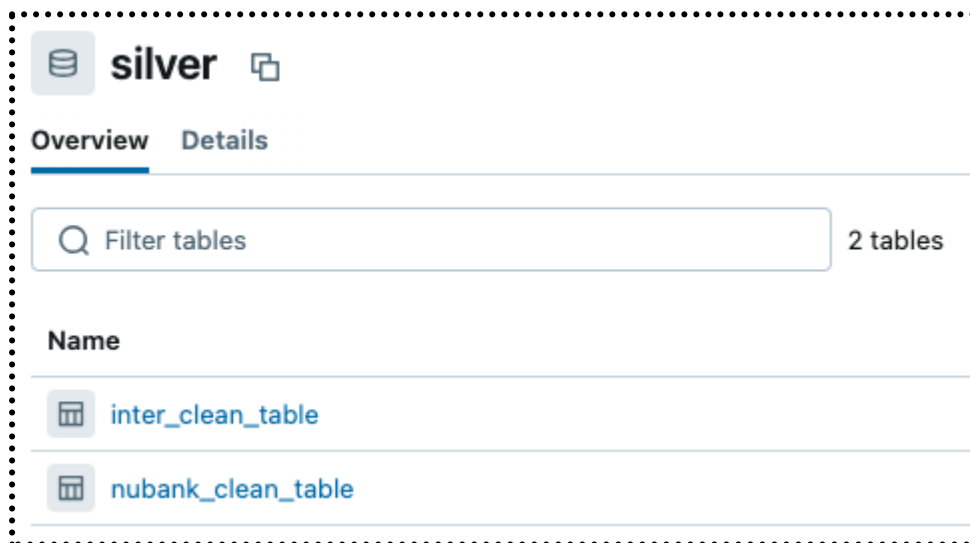
Além disso, enquanto o tipo de transação é apresentado na coluna “Histórico” no Inter, no Nubank, o tipo de transação fica armazenado junto de várias outras informações dentro do

campo “Descrição”. Este campo o sofrerá um SPLIT para desmembrar as informações contidas nele. Como parte dos tratamentos e filtragens, precisamos:

- recuperar apenas transações negativas (gastos), desconsiderando:
  - transações negativas que não são gastos
    - migração de quantias para o outro banco
    - saídas para investimentos (CDB, LCI, LCA, etc)
- padronizar o formato de representação do “Valor” transacionado para serem somados nos 2 bancos
- padronizar a nomenclatura dos tipos de transação (para análises futuras)
- inclusão de colunas de mês, ano e banco facilitando o agrupamento dos dados
- no Nubank: realizar o split da coluna “Descrição” para obtenção das informações de “Transação” e “Beneficiário”, deixando-os disponíveis para análises futuras.
- padronizar os nomes das colunas

## 2. Tabelas da camada silver

Na camada Silver, aplicamos as transformações necessárias para limpar os dados e prepará-los para as análises que respondem às perguntas de interesse. Segue como ficaram as tabelas resultantes nesta camada. Cada filtragem está documentada via comentários no notebook databricks.



A partir das diferenças relatadas na descrição dos dados na camada bronze, aqui as tabelas passaram a ter os mesmos nomes de colunas, os mesmos tipos de dados, mesma representação para Valor, mesmo conjunto e nomenclaturas para os tipos de transação e novas colunas contendo mês, ano e nome do banco. Segue o resultado das mesmas.

silver >	silver >
inter_clean_table	nubank_clean_table
Overview Sample Data Details History	Overview Sample Data Details History
About this table	About this table
Data source <a href="#">Delta</a>	Data source <a href="#">Delta</a>
Last updated há 55 minutos	Last updated há 48 minutos
Size 12.3KiB, 1 file	Size 6.2KiB, 1 file
Description	Description
Add description	Add description
Filter columns...	Filter columns...
Column Type	Column Type
Data date	Data date
Valor decimal(10,2)	Valor decimal(10,2)
Mes int	Mes int
Ano int	Ano int
Transacao string	Transacao string
Beneficiario string	Beneficiario string
Banco string	Banco string

Tabela na silver: inter_clean_table			
Coluna	Significado	Tipo	Valores   Categorias possíveis   Exemplo de formato
Banco	nome do banco de onde veio o registro da transação; não podendo ser nulo	string	Podendo ser apenas: - Inter
Data	coluna que representa a data de registro da transação; não podendo ser nulo	date	No formato: 2024-12-30
Valor	coluna com o valor da transação, contendo apenas informações de saída (valores negativos); utiliza-se ponto como separação de decimal; não	decimal (10,2)	No formato: -2123.68

	podendo ser nulo		
Mes	Mês da transação	int	Exemplo: 1 - representando o mês de janeiro   2 - representando o mês de fevereiro, etc
Ano	Ano da transação	int	Exemplo: 2024
Transacao	Tipo de transação, contendo apenas transações de saída	string	Podendo ser: <ul style="list-style-type: none"> <li>- Pagamento de boleto efetuado</li> <li>- Compra no débito</li> <li>- Pagamento de fatura</li> <li>- Saque</li> <li>- Transferência enviada</li> <li>- Pix enviado</li> </ul>
Beneficiario	Nome do beneficiário da transação, podendo ser nulo, nome de pessoa física ou pessoa jurídica (nome da empresa)	string	Exemplos: <ul style="list-style-type: none"> <li>- AMPLA ENERGIA E SERVICOS S A</li> <li>- Renato Magalhaes Junior</li> </ul>

Tabela na silver: nubank_clean_table			
<u>Coluna</u>	<u>Significado</u>	<u>Tipo</u>	<u>Valores   Categorias possíveis   Exemplo de formato</u>
Banco	nome do banco de onde veio o registro da transação; não podendo ser nulo	string	Podendo ser apenas: <ul style="list-style-type: none"> <li>- Nubank</li> </ul>

Data	coluna que representa a data de registro da transação; não podendo ser nulo	date	No formato: 2024-12-30
Valor	coluna com o valor da transação, contendo apenas informações de saída (valores negativos); utiliza-se ponto como separação de decimal; não podendo ser nulo	decimal (10,2)	No formato: -2123.68
Mes	Mês da transação	int	Exemplo: 1 - representando o mês de janeiro   2 - representando o mês de fevereiro, etc
Ano	Ano da transação	int	Exemplo: 2024
Transacao	Tipo de transação, contendo apenas transações de saída	string	Podendo ser: - Pagamento de boleto efetuado - Compra no débito - Pagamento de fatura - Saque - Transferência enviada - Pix enviado
Beneficiario	Nome do beneficiário da transação, podendo ser nulo, nome de pessoa física ou pessoa jurídica (nome da empresa)	string	Exemplos: - AMPLA ENERGIA E SERVICOS S A - Renato Magalhaes Junior

### 3. Tabela flat da camada gold

Na camada Gold, consolidamos os dados dos dois bancos possibilitando as análises requeridas. Por meio de uma operação de UNION (detalhada no notebook Databricks) integramos as transações dos dois bancos, garantindo que estamos tratando as mesmas colunas para ambos os bancos. Segue como ficou a tabela resultante:  
**gold.all\_banks\_table**

 **gold** 

Overview

Details



 Filter tables

1 table

Name

 all\_banks\_table

gold >

 **all\_banks\_table** 

Overview


Sample Data

Details

History

Description

Add description

 Filter columns...

Column	Type
Banco	string
Data	date
Valor	decimal(10,2)
Mes	int
Ano	int
Transacao	string
Beneficiario	string

Tabela na gold: all\_banks\_table



<u>Coluna</u>	<u>Significado</u>	<u>Tipo</u>	<u>Valores   Categorias possíveis   Exemplo de formato</u>
Banco	nome do banco de onde veio o registro da transação	string	Podendo ser: <ul style="list-style-type: none"> <li>- Inter</li> <li>- Nubank</li> </ul>
Data	coluna que representa a data de registro da transação; não podendo ser nulo	date	No formato: 2024-12-30
Valor	coluna com o valor da transação, contendo apenas informações de saída (valores negativos); utiliza-se ponto como separação de decimal; não podendo ser nulo	decimal (10,2)	No formato: -2123.68
Mes	Mês da transação	int	Exemplo: 1 - representando o mês de janeiro   2 - representando o mês de fevereiro, etc
Ano	Ano da transação	int	Exemplo: 2024
Transacao	Tipo de transação, contendo apenas transações de saída	string	Podendo ser: <ul style="list-style-type: none"> <li>- Pagamento de boleto efetuado</li> <li>- Compra no débito</li> <li>- Pagamento de fatura</li> <li>- Saque</li> <li>- Transferência enviada</li> <li>- Pix enviado</li> </ul>
Beneficiario	Nome do beneficiário da transação, podendo	string	Exemplos: <ul style="list-style-type: none"> <li>- AMPLA ENERGIA E</li> </ul>

	ser nulo, nome de pessoa física ou pessoa jurídica (nome da empresa)		SERVICOS S A - Renato Magalhaes Junior
--	--	--	---

## Resumo e Autoavaliação

O pipeline completo contemplou os seguintes passos:

1. **Leitura dos dados CSV** dos dois bancos (Inter e Nubank).
2. **Transformações na camada Silver:**
  - Filtragem de transações indesejadas com base em condições específicas.
  - Limpeza e reorganização dos dados.
3. **Consolidação e modelagem** na camada Gold, unindo os dados dos dois bancos em uma tabela consolidada.
4. **Consultas SQL** para responder às perguntas de análise, agrupadas nos tópicos de
  - Gastos mensais
  - Gastos anuais
  - Ranking de meses de maiores gastos

Foi utilizada uma arquitetura em camadas chamada “Medalhão” garantindo que os dados fossem estruturados de forma eficiente para responder às perguntas e facilitando a manutenção do pipeline.

Todas as perguntas foram respondidas por meio de consultas SQL, dado que essa foi uma das cadeiras desta sprint. Também era possível realizar várias das filtrações por meio dos dataframes spark ou pandas, mas demos prioridade à utilização do SQL.

Para trabalhos futuros, listamos vários tópicos que temos interesse em trabalhar e responder relacionados a categorização e classificação de gastos. Também a integração de gastos das contas correntes de diferentes bancos com os detalhamentos de faturas de cartão de crédito. Todos esses interesses de atuação futura estão descritos na seção “Objetivo > Perguntas para o Futuro”.

As discussões envolvendo cada uma das respostas que contemplavam o escopo deste trabalho, estão no notebook databricks.