📊 Case de QA e Análise de Dados – GA4 & BigQuery Simulation (MySQL Local)

Este projeto simula um cenário real de qualidade de dados e análise de conversão por canal utilizando dados exportados do Google Analytics 4 (GA4) para BigQuery, mas recriados em MySQL local para evitar custos de cloud.
O case foi estruturado para demonstrar habilidades de QA de dados, ETL, validação e análise de impacto da qualidade.

1️⃣ Estrutura do Projeto

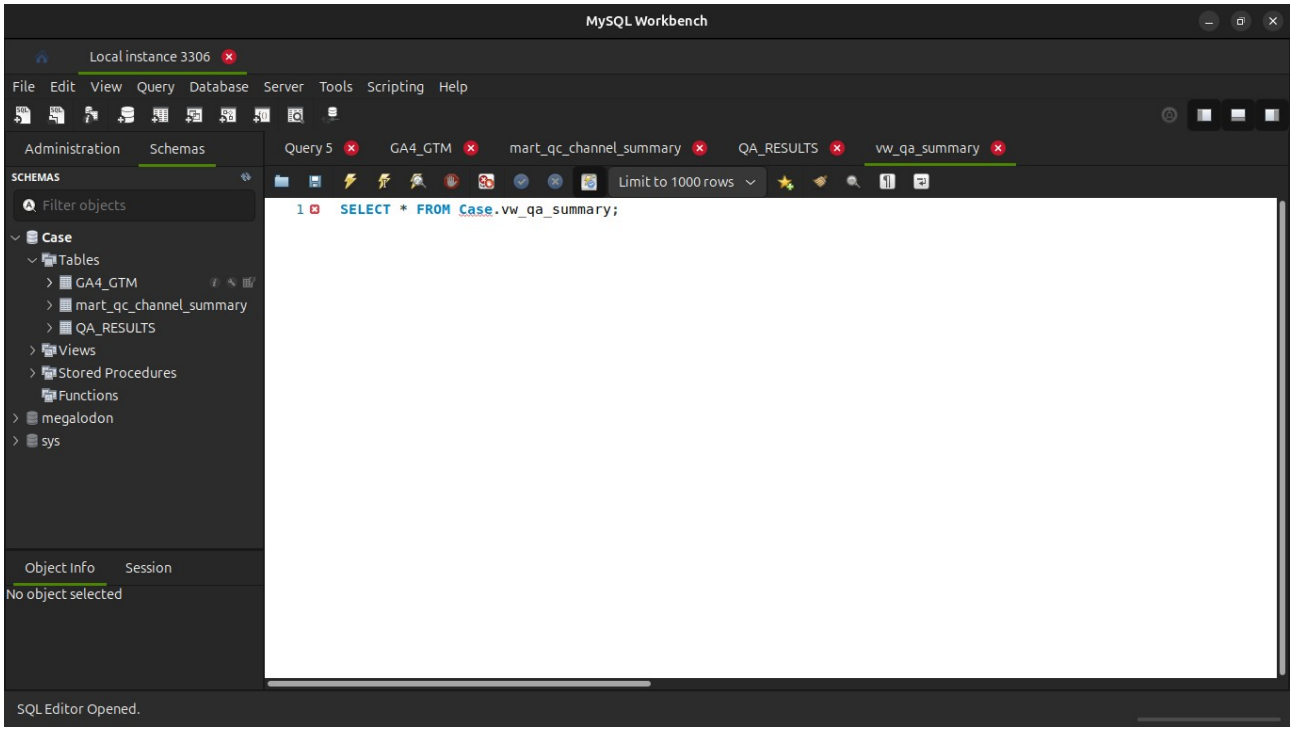
```
CASE/
├── CSV/ # Exportações CSV para uso no Power BI
├── SQL/ # Scripts SQL organizados por etapa
├── Power_BI/ # Relatórios e arquivos de conexão
├── venv/ # Ambiente virtual Python
├── .env # Variáveis de ambiente (credenciais MySQL)
├── populate_table.py # Script para popular dados simulados
└── README.md # Documentação do projeto
```

2️⃣ Etapas do Desenvolvimento

1. Criação do Schema e Tabela de Eventos

Arquivo: Cria_Schema_e_tabela_GA4_GTM.sql

- Cria o schema Case e a tabela GA4_GTM que receberá os eventos GA4 simulados.
- Estrutura compatível com colunas comuns no GA4 Export.

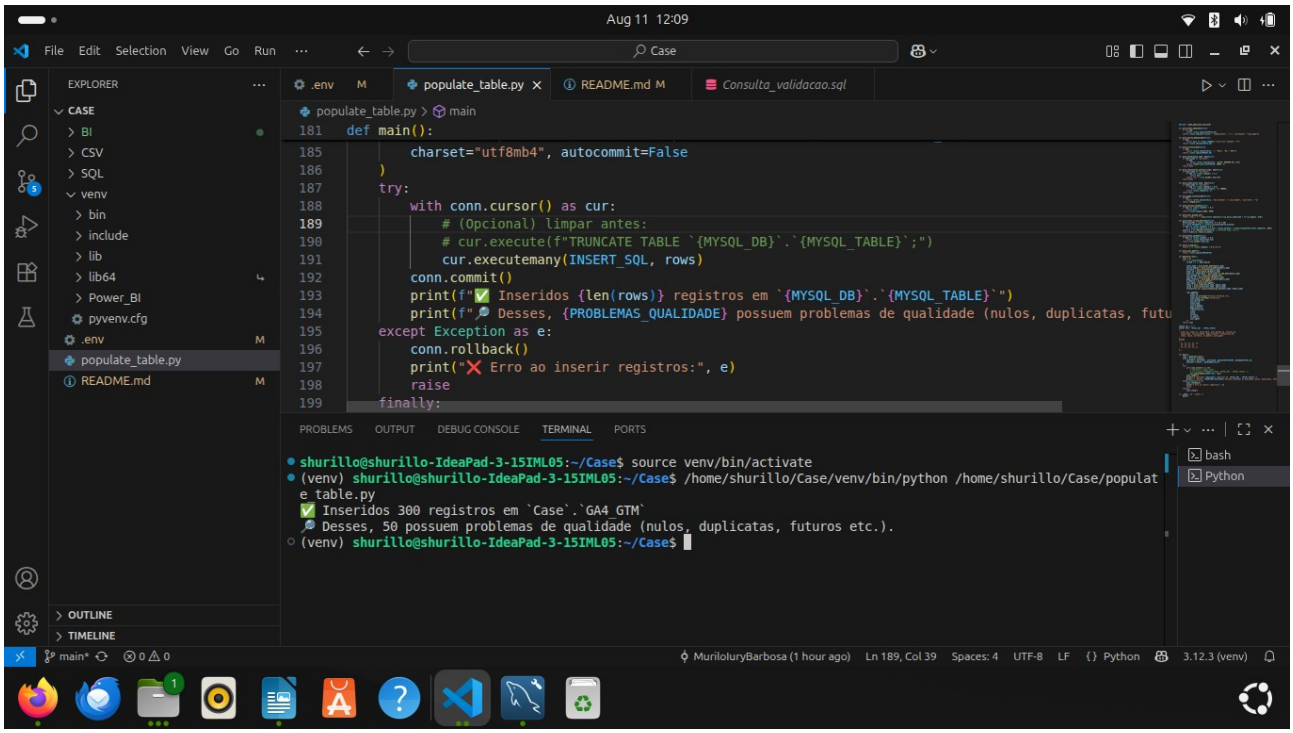


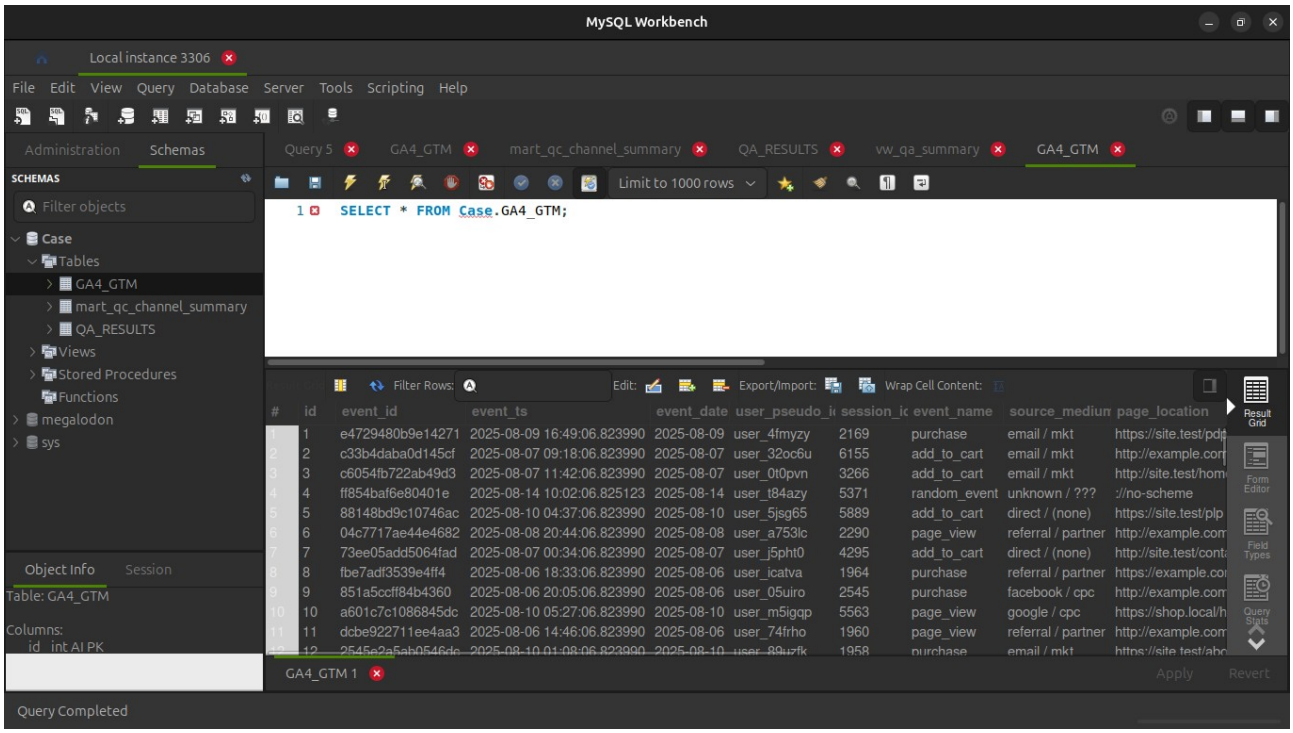
2. População de Dados com Simulação de Problemas de Qualidade

Arquivo: populate_table.py

- Script Python que insere 300 registros na tabela, sendo 50 com problemas de qualidade (nulos, duplicatas, datas futuras, valores inválidos etc.).

- Utiliza variáveis de ambiente no .env para conexão MySQL.
- Garante aleatoriedade controlada para reproduzibilidade.





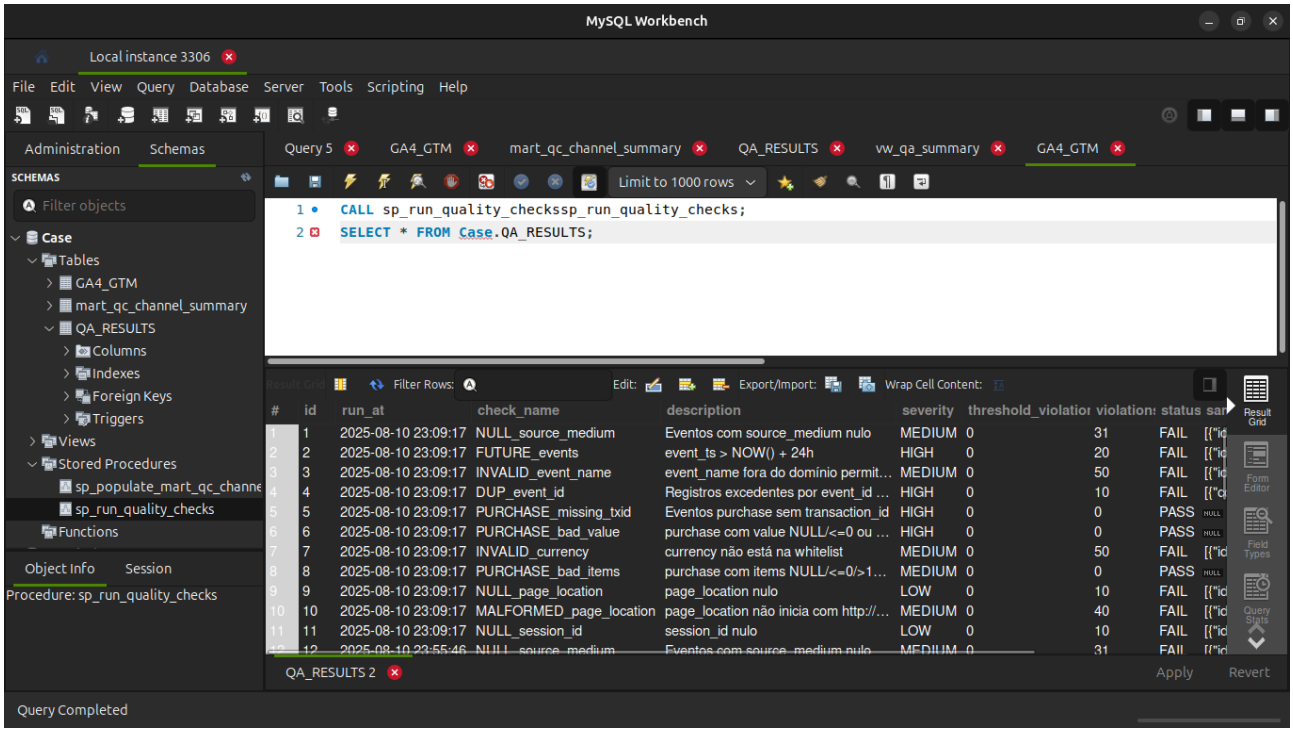
Problemas simulados:

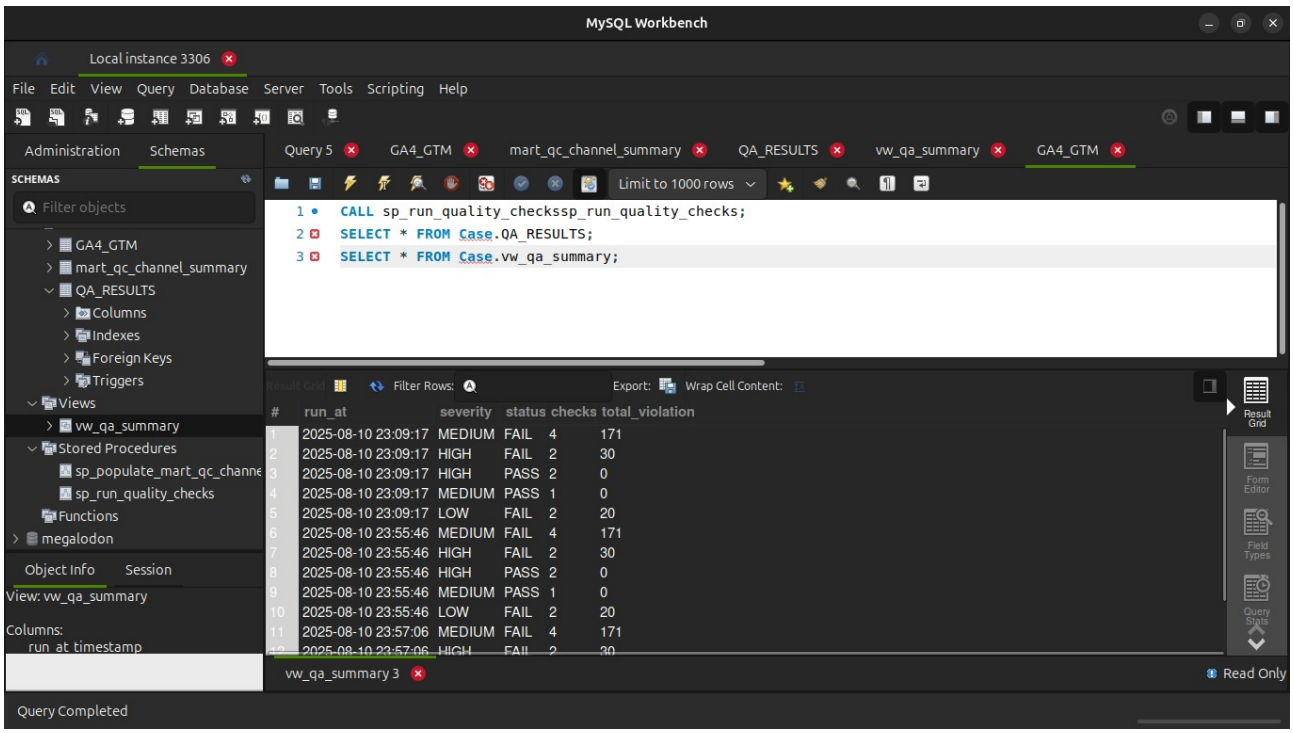
- source_medium nulo ou inválido
- event_id duplicado
- Datas futuras em event_ts
- event_name fora do domínio permitido
- page_location malformado
- Compras (purchase) sem transaction_id ou valores inválidos
- items negativos ou irreais
- session_id nulo

3. Validação de Qualidade dos Dados

Arquivo: Cria_Procedure_de_validacao_dedados.sql

- Procedure sp_run_quality_checks que:
- Executa regras de validação nos campos-chave.
- Registra resultados na tabela QA_RESULTS.
- View vw_qa_summary para resumo da qualidade dos dados.





Como executar:

CALL `Case`.`sp_run_quality_checks`();

-- Ver resultados detalhados
SELECT *
FROM `Case`.`QA_RESULTS`
WHERE run_at = (SELECT MAX(run_at) FROM `Case`.`QA_RESULTS`)
ORDER BY severity DESC, violations DESC;

-- Ver resumo
SELECT *
FROM `Case`.`vw_qa_summary`
WHERE run_at = (SELECT MAX(run_at) FROM `Case`.`QA_RESULTS`);

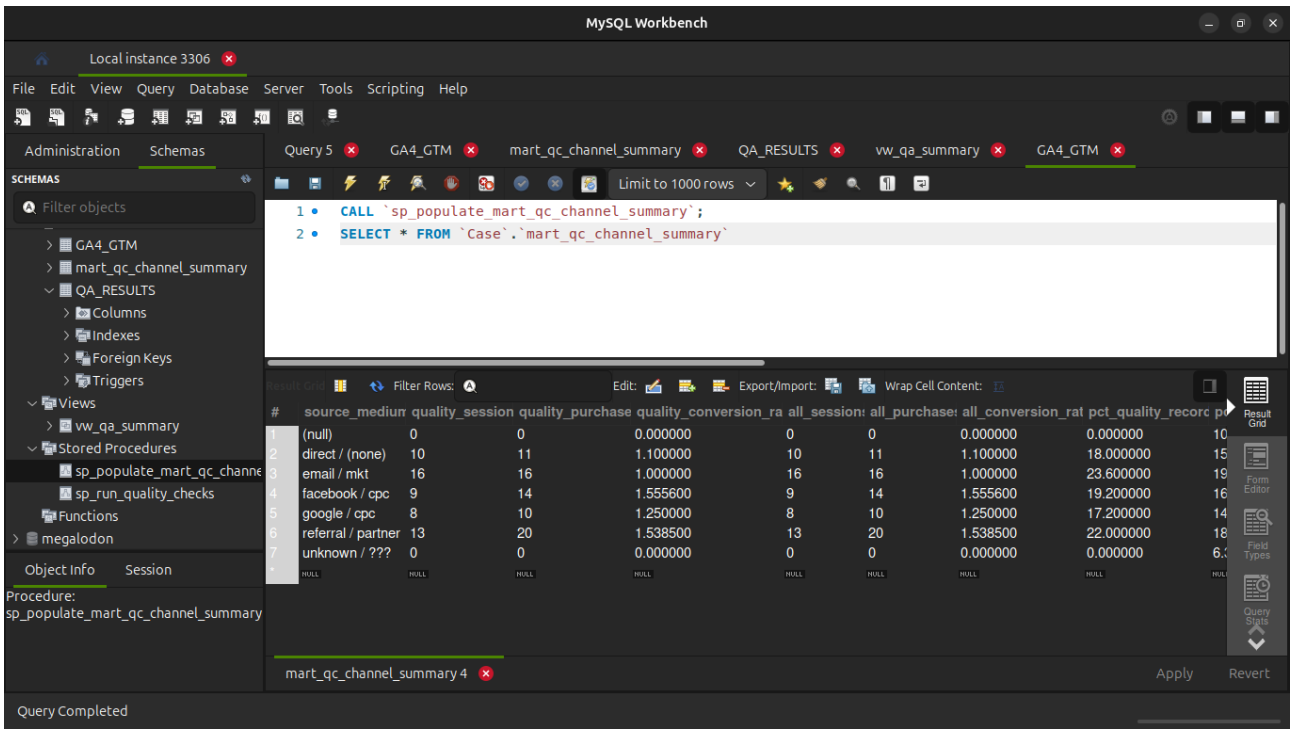
4. Cálculo de Conversão por Canal com e sem Qualidade

Arquivo: Cria_procedure_e_popula_calculos_conversao.sql

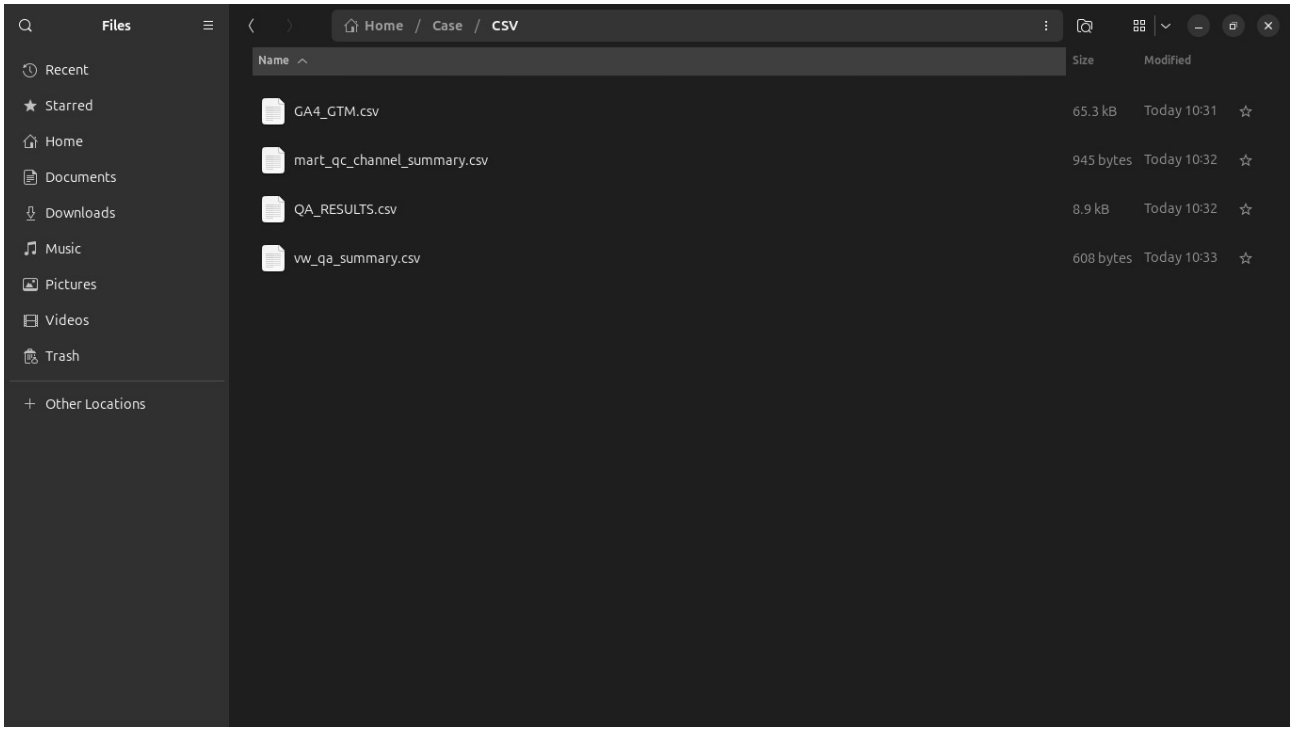
- Procedure sp_populate_mart_qc_channel_summary:
- Calcula taxa de conversão apenas com dados de qualidade assegurada.
- Calcula taxa de conversão com todos os dados (incluindo problemas).
- Mostra impacto (%) da falta de qualidade.
- Registra contagem de registros usados em cada cenário.

Como executar:
CALL `Case`.`sp_populate_mart_qc_channel_summary`();

SELECT *
FROM `Case`.`mart_qc_channel_summary`
ORDER BY all_sessions DESC, source_medium;



5. Exportação para Power BI (Passo extra)
    - Exportação do MySQL para CSV (pasta /CSV).
    - Desenvolvimento de dashboards no Power BI no Windows usando os arquivos exportados do Ubuntu.
    - Recentemente migrei de sistema operacional e passei a utilizar Ubuntu, e, por não ter aprendido ainda como instalar o Power BI no Ubuntu, decidi extrair para CSV e desenvolver no ambiente windows o dashboard. Porém num cenário comum o BI seria conectado diretamente no BigQuery do GCP.



3️⃣ Ferramentas Utilizadas

- ChatGPT Plus – Apoio na concepção e automação
- MySQL Workbench 8.0 – Modelagem e execução SQL
- Python 3.13 – Bibliotecas PyMySQL e python-dotenv
- Ubuntu – Ambiente principal de desenvolvimento
- Windows 11 – Ambiente para criação de dashboards no Power BI

📊 Desenvolvimento do BI no Power BI

Esta etapa teve como objetivo construir uma análise visual a partir dos dados simulados, permitindo identificar e mensurar o impacto de problemas de qualidade nos eventos registrados.

1️⃣ Conexão e Tratamento de Dados

- Conexão inicial realizada a partir de arquivos CSV, simulando uma conexão direta com o banco de dados.
- Conversão e tratamento dos tipos de dados de cada tabela para garantir consistência nas análises.
- Criação da tabela dCalendario, utilizada como dimensão de tempo para facilitar análises temporais.
- Criação de uma tabela chamada "Medidas" para organizar e centralizar todas as medidas DAX criadas.

2️⃣ Modelagem e Relacionamentos

- Estabelecimento dos relacionamentos entre as tabelas de fatos e a dCalendario, garantindo integridade na análise temporal.
- Relacionamento entre as tabelas de eventos (GA4_GTM) e a tabela de resultados de QA (QA_RESULTS) para cruzamento das informações de qualidade.

3️⃣ Medidas DAX Implementadas

- Quantidade de registros totais:

```
Qtd. Registros = DISTINCTCOUNT(GA4_GTM[id])
```

- Quantidade de registros sem qualidade:

```
Qtd. Registros bad = DISTINCTCOUNT(QA_RESULTS[id])
```

- Percentual de registros sem qualidade:

```
% s/ Qualidade = DIVIDE([Qtd. Registros bad], [Qtd. Registros])
```

4️⃣ Visualizações Criadas

- Visão Geral: Cartões com KPIs de volume total de registros, volume de registros com problema e percentual de dados sem qualidade.
- Análise Temporal: Gráficos lineares e de colunas para acompanhar a evolução da qualidade dos dados ao longo do tempo.
- Impacto por Canal/Fonte: Visual comparando o volume e percentual de dados com problema por source_medium.

Qtd. Registros	Qtd. Registros bad	% s/ Qualidade
300	33	11,00%

Qtd. de Registros por evento

Evento	Qtd. Registros
purchase	71
add_to_cart	65
begin_checkout	58
page_view	56
random_event	11
view_page	11
Total	**300**

Detalhamento de problemas de qualidade

Descrição inconsistência	Nível inconsistência	Status	Qtd. Registros bad	% s/ Qualidade
currency não está na whitelist	MEDIUM	FAIL	3	1,00%
event_name fora do domínio permitido	MEDIUM	FAIL	3	1,00%
event_ts > NOW() + 24h	HIGH	FAIL	3	1,00%
Eventos com source_medium nulo	MEDIUM	FAIL	3	1,00%
Eventos purchase sem transaction_id	HIGH	PASS	3	1,00%
page_location não inicia com http:// ou https://	MEDIUM	FAIL	3	1,00%
page_location nulo	LOW	FAIL	3	1,00%
Total			**33**	**11,00%**