# Desafio Técnico – Analytics Engineer Sênior

## 🎯 Análise de Retenção de Clientes com dbt + BigQuery + Looker Studio

Você foi convidado a participar de um desafio técnico que simula um problema comum em empresas digitais com alto volume de dados: analisar a retenção de clientes com base em dados brutos de múltiplas fontes.

O objetivo é avaliar sua capacidade de modelar dados com dbt, criar métricas de negócio relevantes e comunicar insights de maneira clara.

## 🧪 Descrição do Desafio

A empresa fictícia DataMall opera uma plataforma de e-commerce com milhares de transações mensais. Seu papel será preparar uma estrutura de dados analítica robusta e um dashboard de retenção, utilizando o stack moderno: dbt + BigQuery + Looker Studio.

Você trabalhará com datasets simulando Stripe (pagamentos), Shopify (pedidos) e Zendesk (suporte), fornecidos como arquivos CSV ou tabelas já carregadas.

## 🔧 Entregáveis Técnicos

\*\*Parte 1 – Modelagem de Dados com dbt\*\*

- Criar as camadas:

- staging: limpeza e padronização

- intermediate: construção de entidades (clientes, pedidos, pagamentos)

- marts: métricas e indicadores de negócio

- Criar ao menos:

- Uma tabela de cohorts de clientes por mês de aquisição

- Uma tabela de LTV (Lifetime Value) acumulado por cliente

- Uma tabela de churn (clientes inativos há mais de 30 dias)

\*\*Parte 2 – Visualização em Looker Studio ou Metabase\*\*

- Criar um dashboard com:

- Gráfico de retenção por cohort

- Tabela de clientes com LTV

- Indicadores de churn por faixa de tempo

- Permitir filtros por país, data de aquisição e status do cliente

## ✨ Extras que contam pontos

- Implementar dbt tests (unique, not\_null, relationships)

- Gerar documentação automática com dbt docs

- Escrever um README completo explicando:

- Stack utilizada

- Como rodar o projeto

- Lógica das transformações

- Utilizar .yml de documentação de modelos e métricas

## ⚠ Importante

Este desafio não exige deploy real nem conexão com APIs externas. Você pode mockar os dados ou usar amostras públicas.

Foque em:

- Clareza na modelagem

- Organização dos diretórios e código dbt

- Consistência nas métricas

- Boas práticas de versionamento e documentação

## 📦 Entrega esperada

- Repositório no GitHub com:

- Diretório models/ organizado

- Arquivos .sql e .yml do dbt

- README.md com explicações

- Link do dashboard publicado (Looker Studio, Metabase ou print com descrição)

## 🕒 Instruções Finais

Prazo para entrega: 3 dias corridos

Tempo estimado de execução: entre 6 a 8 horas

Nível de dificuldade: Sênior

Este teste tem como foco avaliar raciocínio analítico, clareza técnica e organização de dados — não a perfeição visual.

Boa sorte, e aproveite a oportunidade para mostrar seu domínio técnico! 🚀