## Case Técnico

Parabéns! 🎉 Você acaba de ser contratado(a) como **Engenheiro(a) de Dados** da empresa XYZ. No seu primeiro dia de trabalho, você já recebeu sua primeira missão: apoiar os times de Analytics e Comercial com uma análise estratégica de vendas.

A empresa possui um enorme volume de informações sobre vendas de bebidas em uma grande tabela de dados, mas até agora esse material está cru, sem tratamento adequado e sem visões analíticas que realmente tragam valor para o negócio.

O desafio é transformar esses dados em insights úteis que ajudem os times a entender melhor o comportamento dos clientes, tanto no mercado B2C quanto B2B.

O ambiente tecnológico em que você irá trabalhar é composto por Azure Databricks, Azure Storage Account, Power BI, entre outros. Leve essas informações em consideração ao estruturar seus dados, criar pipelines e desenvolver análises.

## Sua missão 🚀

Você recebeu dois conjuntos de dados:

1. Um **arquivo de vendas** (tabela com o histórico de compras dos clientes).
2. Um **arquivo de metas por marca** (definidas pelo time comercial).

Agora, precisa **usar Apache Spark (PySpark ou SQL)** para preparar, processar e analisar esses dados.

## Tarefas

**1. Processamento de dados no Apache Spark**

a. **Carregamento**

- Ler os conjuntos de dados recebidos em Spark DataFrames.

b. **Exploração e compreensão**

- Analisar a estrutura da base para entender colunas, tipos de dados e qualidade da informação.

c. **Limpeza e preparação**

- Tratar valores ausentes e realizar ajustes necessários para deixar os dados consistentes.

d. **Modelagem dimensional**

- A partir dos dados de vendas, crie uma **estrutura dimensional** (modelo estrela ou floco de neve), definindo fatos e dimensões de interesse.

- Construir uma visão mensal consolidada para cada cliente, incluindo:
  - **Volume médio dos últimos 3 meses**, contados a partir da última compra do cliente;
  - **Volume médio dos últimos 6 meses**, contados a partir da última compra do cliente;
  - **Share B2B nos últimos 3 meses**, contados a partir da última compra do cliente;
  - **Share B2B nos últimos 6 meses**, contados a partir da última compra do cliente.

📌 *Obs.: O Share B2B de cada cliente é calculado dividindo o faturamento mensal das linhas com* `b2b_status = true` *pelo faturamento mensal total daquele cliente.*

**2. Distribuição de metas por cliente**

Além da análise acima, você também vai precisar trabalhar com o **arquivo de metas por marca**. O desafio aqui é **quebrar as metas de cada marca para o nível de cliente**.

De forma didática, o cálculo funciona assim:

- Primeiro, identifique quanto cada cliente representa dentro das vendas de determinada marca em volume (exemplo: se a marca "X" vendeu 1.000 unidades no total e um cliente específico comprou 100, então esse cliente representa **10% das vendas da marca X**).
- Em seguida, multiplique esse percentual pela meta da marca (exemplo: se a meta da marca X for 50.000 unidades, esse cliente terá uma meta proporcional de **5.000 unidades**).
- Salve esse valor calculado em uma **nova coluna no DataFrame**.

Esse processo garante que as metas de marca sejam distribuídas de maneira justa e proporcional entre os clientes.

**3. Enriquecimento de dados via API**

Por fim, parte do seu desafio será **enriquecer os dados de clientes** com informações externas. Na base de vendas, você encontrará um campo de **CEP**. Sua missão será:

- Utilizar uma **API de consulta de CEP** para recuperar a cidade e o estado correspondentes;
- Incorporar essas informações ao seu DataFrame final.

Ao final, inclua **o(s) código(s) em Python** desenvolvido e o respectivo **modelo dimensional**, compactados em **um único arquivo .zip**.

## O que esperamos ver na sua entrega

- Uso adequado do **Apache Spark/PySpark**;
- Clareza no processo de **ETL (Extract, Transform, Load)**;
- Criação de um modelo dimensional consistente;
- Código bem estruturado e comentado;
- Boas práticas de engenharia de dados aplicadas no tratamento, agregação e enriquecimento dos dados;
- SUGESTÃO: Utilizar um workspace Databricks Free Edition para desenvolvimento dos códigos e o dbdiagram.io para a modelagem dimensional.