\*\*Case Study para Aprimorar Suas Habilidades em ETL com o Dataset de Vendas de Videogames\*\*

\*\*Objetivo:\*\*

Trabalhar com o processo ETL (Extract, Transform, Load) aplicado ao dataset de vendas de videogames, preparando os dados para análises de qualidade. Além disso, o storytelling vai ajudá-lo a visualizar as demandas práticas de limpeza e transformação de dados.

---

### \*\*História Contextual\*\*

Imagine que você é o analista de dados de uma grande empresa de mídia especializada em videogames. Seu chefe pediu para criar um relatório detalhado sobre os jogos mais vendidos globalmente. O objetivo é identificar padrões, como os gêneros de jogos preferidos em diferentes regiões, e determinar quais plataformas têm maior potencial para futuros lançamentos.

Para isso, você terá que construir um pipeline ETL que garanta a qualidade dos dados e forneça insights confiáveis. Eis o desafio:

---

### \*\*1. Extração de Dados\*\*

- \*\*Obtenção do Dataset:\*\* Baixe o dataset no Kaggle e importe-o para seu ambiente (por exemplo: Python com pandas).

Código exemplo:

```python

import pandas as pd

df = pd.read\_csv('videogamesales.csv')

print(df.head())

```

- \*\*Exploração Inicial:\*\* Examine as colunas disponíveis no dataset:

- Nome do jogo (`Name`)

- Plataforma (`Platform`)

- Ano de lançamento (`Year`)

- Gênero (`Genre`)

- Publisher

- Vendas por região (`NA\_Sales`, `EU\_Sales`, `JP\_Sales`, `Other\_Sales`)

- Vendas Globais (`Global\_Sales`)

Objetivo: Entender a estrutura e identificar problemas, como valores ausentes ou inconsistências.

---

### \*\*2. Limpeza e Pré-processamento\*\*

- \*\*Tratamento de Valores Ausentes:\*\*

Pergunte-se: quais colunas são críticas para análise? Por exemplo, `Year` é essencial para tendências históricas. Para valores ausentes:

- Preencha com valores padrões (ex: média para números).

- Ou exclua linhas onde dados ausentes causam distorções.

Código exemplo:

```python

df['Year'] = df['Year'].fillna(df['Year'].median())

```

- \*\*Detecção de Outliers:\*\*

Vendas absurdamente altas ou baixas podem indicar erros ou casos atípicos. Use gráficos como boxplots para identificar outliers em colunas de vendas.

Código exemplo:

```python

import matplotlib.pyplot as plt

df.boxplot(column=['Global\_Sales'])

plt.show()

```

- \*\*Correção de Tipos de Dados:\*\*

Certifique-se de que `Year` está como inteiro e colunas de vendas como float. Use `.astype()` para ajustar.

---

### \*\*3. Transformação de Dados\*\*

- \*\*Normalização:\*\*

Para comparações, normalize colunas como vendas por região, convertendo para escala entre 0 e 1.

Código exemplo:

```python

from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

df['Global\_Sales'] = scaler.fit\_transform(df[['Global\_Sales']])

```

- \*\*Codificação de Variáveis Categóricas:\*\*

Transforme colunas como `Genre` e `Platform` em valores numéricos para análise. Pode usar one-hot encoding:

```python

df = pd.get\_dummies(df, columns=['Genre', 'Platform'])

```

- \*\*Feature Engineering:\*\* (já existe, é a GLOBAL SALES, porém vou checar utilizando uma função de soma e comparação)

Crie uma nova coluna `Total\_Sales` somando todas as vendas regionais (`NA\_Sales`, `EU\_Sales`, `JP\_Sales`, `Other\_Sales`).

```python

df['Total\_Sales'] = df[['NA\_Sales', 'EU\_Sales', 'JP\_Sales', 'Other\_Sales']].sum(axis=1)

```

- \*\*Agregações:\*\*

Agrupe os dados por `Platform` e calcule vendas médias.

```python

platform\_sales = df.groupby('Platform')['Global\_Sales'].mean()

print(platform\_sales)

```

---

### \*\*4. Carga de Dados (Load)\*\*

- \*\*Preparação para Banco de Dados:\*\*

Certifique-se de que o dataset final está limpo e formatado para inserção no banco de dados.

- \*\*Configuração do SQLAlchemy:\*\*

Configure uma conexão com seu banco de dados SQL.

```python

from sqlalchemy import create\_engine

engine = create\_engine('sqlite:///videogames.db')

df.to\_sql('sales', engine, index=False)

```

- \*\*Carga no Banco de Dados:\*\*

Insira os dados no banco utilizando SQLAlchemy.

---

### \*\*5. Validação e Documentação\*\*

- \*\*Validação:\*\*

Confira se o pipeline está funcionando como esperado. Faça verificações rápidas nos dados carregados.

- \*\*Documentação:\*\*

Crie um documento explicando todas as etapas do ETL. Inclua justificativas para métodos de limpeza, transformação e carga.

- \*\*Análise Exploratória Básica:\*\*

Faça gráficos e estatísticas descritivas para checar qualidade dos dados e começar análises iniciais.

---

### \*\*Conclusão\*\*

Siga essas etapas para estruturar o case. Com isso, você terá uma visão clara de como conduzir o ETL, praticando limpeza e transformação de dados enquanto se prepara para análises aprofundadas. Se precisar de ajuda em algum passo, é só pedir! 🚀