ATIVIDADE 20:

MIGRAÇÃO E INTEGRAÇÃO DE BASE DE DADOS

Bruno Yaporandy

Etapas

Extract, Transform and Loading (ETL)

Extrair os dados de uma base em cloud

Foram extraídos datasets de um banco SQL e outro noSQL. (fornecidos pelo cliente)

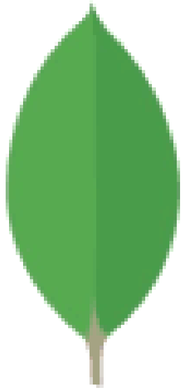
Tratamento, correções e padronizações

Realizado tratamentos e correções via Pandas.

Carregar os dados para uma base cloud

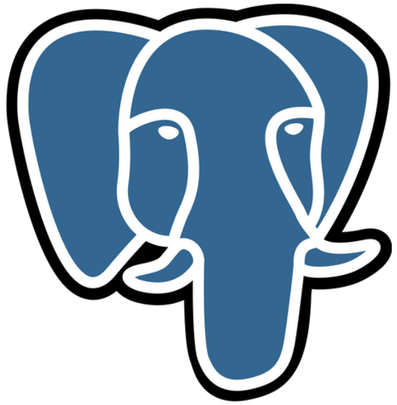
Com os dados prontos, foram enviados para um único banco noSQL(pedido do cliente).

Extraindo os datasets



Mongodb Atlas
banco noSQL

1º - Conexão com o Atlas,
2º - Transformação da coleção em DataFrame,
3º - Envio desse DataFrame para um bucket,
4º -Começa a etapa de transformação.



Postgres(GCP)
banco SQL

1º - Conexão com o Postgres,
2º - Transformei a tabela em DataFrame,
3º - Envio desse DataFrame para um bucket,
4º - Começa a etapa de transformação.

Extraindo dataset Mongodb Atlas

Definindo parâmetros de conexão

```
uri = 'mongodb+srv://<user>:
<senha>@cluster0.zef7j.mongod.....'
client = MongoClient( uri )
```

Transformando coleção em DataFrame

```
df_original_nosql = pd.DataFrame(collection.find())
```

Envio do DataFrame para o bucket

```
df.to_csv('gs://path/df-original-nosql.csv')
```

Extraindo dataset Postgres (GCP)

Definindo parâmetros de conexão

```
connector = psycopg2.connect( host="", database="atv20", user="", password="")
```

Transformando coleção em DataFrame

```
sql_select = "SELECT * FROM vendas;"
df_sql_original = pd.read_sql_query(sql_select, connector)
```

Envio do DataFrame para o bucket

```
df.to_csv('gs://path/df-sql-original.csv')
```

Tratamento

01 Mongodb Atlas banco noSQL

1º - Pré-analise,
2º - Correção de inconsistências,
3º - validação de dados com PANDERA,
4º -Começa a etapa de carregamento dos dados.

EXTRA

02 Postgres(GCP) banco SQL

1º - Pré-analise,
2º - Correção de inconsistências,
3º - validação de dados com PANDERA,
4º -Começa a etapa de carregamento dos dados.

Tratamento
banco noSQL

Pré-analise

```
df_nosql.dtypes
pd.value_counts(df_nosql['vendedor'])
df_nosql['vendedor'].isna().value_counts()
```

Correção de inconsistências

```
df_nosql['vendedor'] = df_nosql['vendedor'].fillna('NÃO FOI INFORMADO')
```

Validação de dados com PANDERA

```
schema.validate(df_nosql)
```

Tratamento
banco SQL

Pré-analise

```
df_sql.dtypes
pd.value_counts(df_sql['vendedor'])
df_sql['vendedor'].isna().value_counts()
```

Correção de inconsistências

```
df_sql['vendedor'] = df_sql['vendedor'].fillna('NÃO FOI INFORMADO')
```

Validação de dados com PANDERA

```
schema.validate(df_sql)
```

Carregamento

Mongodb Atlas

Ambos os DataFrames (SQL e noSQL) foram inseridos num banco de dados noSQL conforme solicitado pelo cliente.

Carregamento
para banco noSQL

Transformando em dicionário

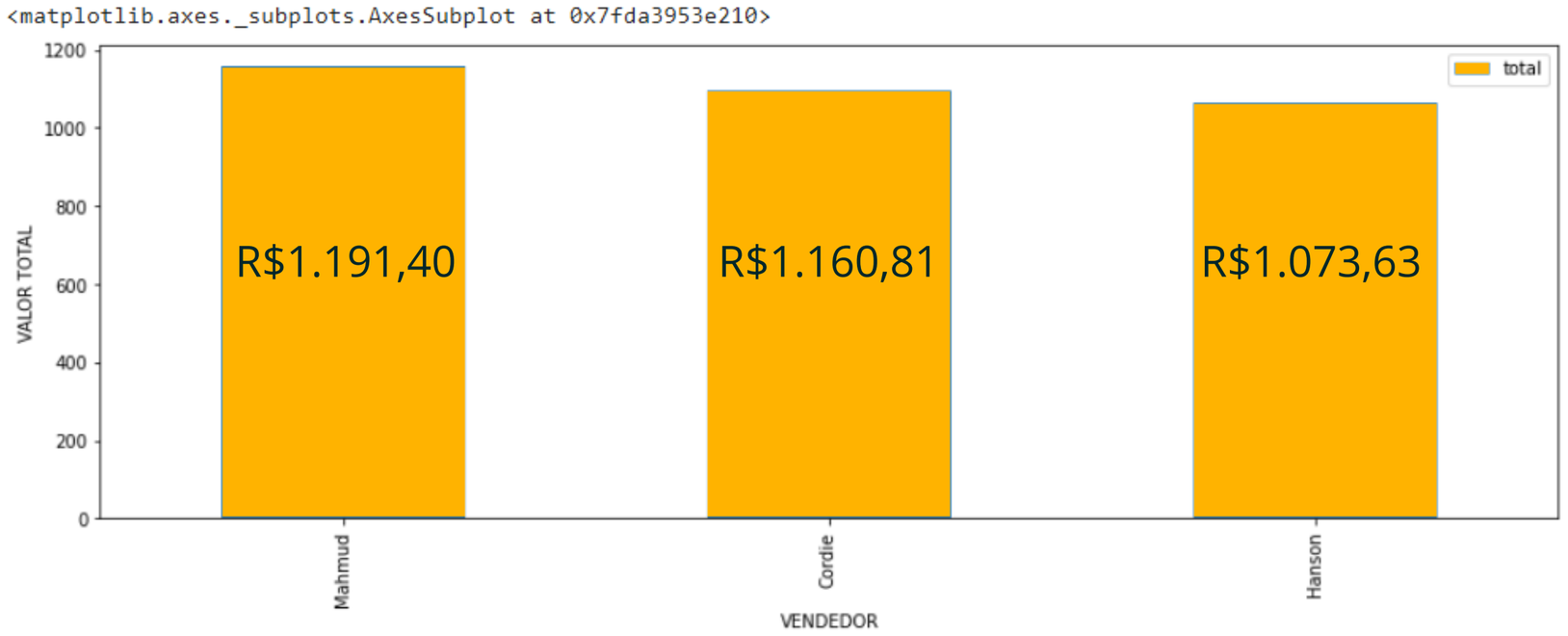
```
df_dicio = df.to_dict("records")
```

Inserindo coleção

```
collection.insert_many(df_dicio)
```

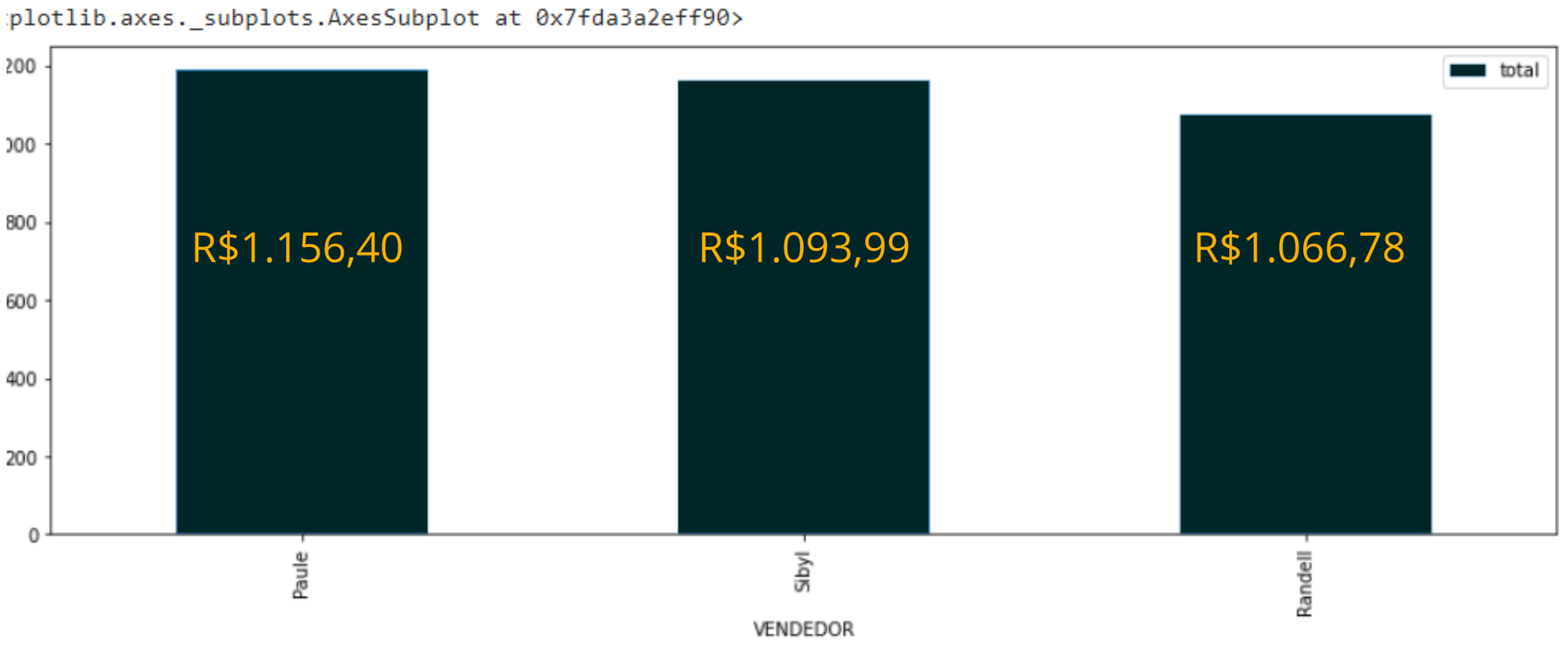
Extra: Insight

Maiores vendas das filiais



Extra: Insight

Maiores vendas da matriz



Obrigado,

Bruno Yaporandy

Imagens retiradas de:
https://miro.medium.com/max/780/1*Rmc568knYGLn7kJ3B97WUQ.png
https://upload.wikimedia.org/wikipedia/commons/thumb/2/29/Postgresql_elephant.svg/1200px-Postgresql_elephant.svg.png

Agradecimento aos colegas:
-Lucas David (Debates gerais sobre a atividade),
-Danilo Ferrari (Debates gerais sobre a atividade),
-Felipe Costa (Dica de resetar o Colab devido a um erro)