**Order Data analysis (Joins)**

import pandas as pd

from datetime import datetime

from pyspark.sql import SparkSession

from pyspark.sql.functions import col

categories\_df = pd.read\_csv('mavricbdhoct01233/user/categories.csv')

customers\_df = pd.read\_csv('mavricbdhoct01233/user/customers.csv')

departments\_df = pd.read\_csv('mavricbdhoct01233/user/departments.csv')

order\_items\_df = pd.read\_csv('mavricbdhoct01233/user/order\_items.csv')

orders\_df = pd.read\_csv('mavricbdhoct01233/user/orders.csv')

products\_df = pd.read\_csv('mavricbdhoct01233/user/products.csv')

1. **Problem Statement: What is the daily product revenue for CLOSED or COMPLETE orders?**

order\_status\_count = orders\_df['order\_status'].value\_counts()

closed\_orders\_df = orders\_df[orders\_df['order\_status'].isin(['COMPLETE', 'CLOSED'])]

**2. Load the required data in to DF like categories, customer,departments,order\_items,orders and products**

**3. Get the count for each order status**

daily\_revenue\_df = merged\_df.groupby(merged\_df['order\_date'].dt.date)['daily\_revenue'].sum().reset\_index()

**4. Filter only COMPLETE or CLOSED orders**

closed\_orders\_df['order\_date'] = pd.to\_datetime(closed\_orders\_df['order\_date'])

**5. Join the products , order\_items and orders tables and calculate daily product revenue**

spark = SparkSession.builder.appName("daily\_product\_revenue").getOrCreate()

spark\_daily\_revenue\_df = spark.createDataFrame(daily\_revenue\_df)

merged\_df = closed\_orders\_df.merge(order\_items\_df, how='inner', on='order\_id')

merged\_df = merged\_df.merge(products\_df, how='inner', left\_on='order\_item\_product\_id', right\_on='product\_id')

**6. Write the data in to the table Daily product revenue in Hive**

spark\_daily\_revenue\_df.write.mode('overwrite').saveAsTable('daily\_product\_revenue')