Hands on – Pyspark (03/09/2024)

from pyspark.sql import SparkSession

from pyspark.sql.functions import col

# Initialize SparkSession

spark = SparkSession.builder \

    .appName("Product Sales Analysis") \

    .getOrCreate()

# Sample data for products

products = [

    (1, "Laptop", "Electronics", 50000),

    (2, "Smartphone", "Electronics", 30000),

    (3, "Table", "Furniture", 15000),

    (4, "Chair", "Furniture", 5000),

    (5, "Headphones", "Electronics", 2000),

]

# Sample data for sales transactions

sales = [

    (1, 1, 2),

    (2, 2, 1),

    (3, 3, 3),

    (4, 1, 1),

    (5, 4, 5),

    (6, 2, 2),

    (7, 5, 10),

    (8, 3, 1),

]

# Define schema for DataFrames

product\_columns = ["ProductID", "ProductName", "Category", "Price"]

sales\_columns = ["SaleID", "ProductID", "Quantity"]

# Create DataFrames

product\_df = spark.createDataFrame(products, schema=product\_columns)

sales\_df = spark.createDataFrame(sales, schema=sales\_columns)

# Show the DataFrames

print("Products DataFrame:")

product\_df.show()

print("Sales DataFrame:")

sales\_df.show()

output:

Products DataFrame:

+---------+-----------+-----------+-----+

|ProductID|ProductName| Category|Price|

+---------+-----------+-----------+-----+

| 1| Laptop|Electronics|50000|

| 2| Smartphone|Electronics|30000|

| 3| Table| Furniture|15000|

| 4| Chair| Furniture| 5000|

| 5| Headphones|Electronics| 2000|

+---------+-----------+-----------+-----+

Sales DataFrame:

+------+---------+--------+

|SaleID|ProductID|Quantity|

+------+---------+--------+

| 1| 1| 2|

| 2| 2| 1|

| 3| 3| 3|

| 4| 1| 1|

| 5| 4| 5|

| 6| 2| 2|

| 7| 5| 10|

| 8| 3| 1|

+------+---------+--------+

#1. Join the DataFrames:

combined\_df = product\_df.join(sales\_df, on="ProductID")

print("Combined DataFrame:")

combined\_df.show()

Combined DataFrame:

+---------+-----------+-----------+-----+------+--------+

|ProductID|ProductName| Category|Price|SaleID|Quantity|

+---------+-----------+-----------+-----+------+--------+

| 1| Laptop|Electronics|50000| 1| 2|

| 1| Laptop|Electronics|50000| 4| 1|

| 2| Smartphone|Electronics|30000| 2| 1|

| 2| Smartphone|Electronics|30000| 6| 2|

| 3| Table| Furniture|15000| 3| 3|

| 3| Table| Furniture|15000| 8| 1|

| 4| Chair| Furniture| 5000| 5| 5|

| 5| Headphones|Electronics| 2000| 7| 10|

+---------+-----------+-----------+-----+------+--------+

#2. Calculate Total Sales Value:

from pyspark.sql.functions import expr

combined\_df = combined\_df.withColumn("TotalSalesValue", col("Price") \* col("Quantity"))

print("DataFrame with Total Sales Value:")

combined\_df.show()

DataFrame with Total Sales Value:

+---------+-----------+-----------+-----+------+--------+---------------+

|ProductID|ProductName| Category|Price|SaleID|Quantity|TotalSalesValue|

+---------+-----------+-----------+-----+------+--------+---------------+

| 1| Laptop|Electronics|50000| 1| 2| 100000|

| 1| Laptop|Electronics|50000| 4| 1| 50000|

| 2| Smartphone|Electronics|30000| 2| 1| 30000|

| 2| Smartphone|Electronics|30000| 6| 2| 60000|

| 3| Table| Furniture|15000| 3| 3| 45000|

| 3| Table| Furniture|15000| 8| 1| 15000|

| 4| Chair| Furniture| 5000| 5| 5| 25000|

| 5| Headphones|Electronics| 2000| 7| 10| 20000|

+---------+-----------+-----------+-----+------+--------+---------------+

#3. Find the Total Sales for Each Product Category:

category\_sales\_df = combined\_df.groupBy("Category").agg({"TotalSalesValue": "sum"}).withColumnRenamed("sum(TotalSalesValue)", "CategoryTotalSales")

print("Total Sales Value by Category:")

category\_sales\_df.show()

Total Sales Value by Category:

+-----------+------------------+

| Category|CategoryTotalSales|

+-----------+------------------+

|Electronics| 260000|

| Furniture| 85000|

+-----------+------------------+

#4. Identify the Top-Selling Product:

top\_selling\_product = combined\_df.groupBy("ProductName").agg({"TotalSalesValue": "sum"}).withColumnRenamed("sum(TotalSalesValue)", "TotalSalesValue").orderBy(col("TotalSalesValue").desc()).limit(1)

print("Top-Selling Product:")

top\_selling\_product.show()

Top-Selling Product:

+-----------+---------------+

|ProductName|TotalSalesValue|

+-----------+---------------+

| Laptop| 150000|

+-----------+---------------+

#5. Sort the Products by Total Sales Value:

sorted\_products\_df = combined\_df.groupBy("ProductName").agg({"TotalSalesValue": "sum"}).withColumnRenamed("sum(TotalSalesValue)", "TotalSalesValue").orderBy(col("TotalSalesValue").desc())

print("Products Sorted by Total Sales Value:")

sorted\_products\_df.show()

Products Sorted by Total Sales Value:

+-----------+---------------+

|ProductName|TotalSalesValue|

+-----------+---------------+

| Laptop| 150000|

| Smartphone| 90000|

| Table| 60000|

| Chair| 25000|

| Headphones| 20000|

+-----------+---------------+

#6. Count the Number of Sales for Each Product:

sales\_count\_df = combined\_df.groupBy("ProductName").count().withColumnRenamed("count", "NumberOfSales")

print("Number of Sales per Product:")

sales\_count\_df.show()

Number of Sales per Product:

+-----------+-------------+

|ProductName|NumberOfSales|

+-----------+-------------+

| Chair| 1|

| Laptop| 2|

| Table| 2|

| Smartphone| 2|

| Headphones| 1|

+-----------+-------------+

#7. Filter the Products with Total Sales Value Greater Than ₹50,000:

filtered\_products\_df = combined\_df.groupBy("ProductName").agg({"TotalSalesValue": "sum"}).withColumnRenamed("sum(TotalSalesValue)", "TotalSalesValue").filter(col("TotalSalesValue") > 50000)

print("Products with Total Sales Value Greater Than ₹50,000:")

filtered\_products\_df.show()

Products with Total Sales Value Greater Than ₹50,000:

+-----------+---------------+

|ProductName|TotalSalesValue|

+-----------+---------------+

| Laptop| 150000|

| Table| 60000|

| Smartphone| 90000|

+-----------+---------------+