Lab 2 — Basic Aggregation Transformation (Beginner)

Trainer Note: This lab introduces learners to simple aggregations in Spark using Databricks. The goal is to keep it extremely beginner-friendly—nothing advanced, only the basics like `groupBy`, `sum`, and `count`. This lab uses the dataset you uploaded (`orders_raw.csv`) into the same location as Lab 1.

Learning Objective

Learners will perform basic aggregations on a small orders dataset. They will load the CSV file, calculate total revenue, aggregate orders by customer, and create derived metrics.

Learning Outcomes

After completing this lab, learners will be able to: - Read an orders dataset stored in Volumes. - Understand the structure of transactional data. - Use `groupBy()` with basic aggregations (`sum`, `count`). - Create new calculated fields such as total order value. - Save aggregated output as a Delta table.

Dataset Description

Upload the file you downloaded earlier into the following location:

```
/Volumes/workspace/default/test/orders_raw.csv
```

The dataset contains: - **order_id** — Unique order identifier - **customer_id** — ID of the customer who placed the order - **product** — Product purchased - **quantity** — Quantity purchased - **price** — Unit price of the product

This dataset is intentionally simple so that learners can easily understand and visualize the transformation.

Step-by-Step Lab Instructions

Run each cell in your Databricks notebook and explain the changes observed.

Step 1 — Read the Orders Dataset

```
csv_path = '/Volumes/workspace/default/test/orders_raw.csv'

df_orders = spark.read.option('header', 'true').option('inferSchema',
'true').csv(csv_path)

display(df_orders)
df_orders.printSchema()
```

Trainer explanation: Using `inferSchema=true` is acceptable here because numeric types like quantity and price need to be automatically typed. This dataset is clean, so schema inference is safe.

Step 2 — Create a New Column: Total Order Value

Each order's value is calculated by multiplying quantity and price.

```
from pyspark.sql import functions as F

df_orders = df_orders.withColumn('order_value', F.col('quantity') *
F.col('price'))

display(df_orders)
```

Explain how derived columns are created using `withColumn`.

Step 3 — Aggregate Total Revenue Across All Orders

```
total_revenue = df_orders.agg(F.sum('order_value').alias('total_revenue'))

display(total_revenue)
```

Discuss the meaning of total revenue and how businesses use such KPIs.

Step 4 — Aggregate Data by Customer

This is the main objective of the lab—simple groupBy aggregation.

```
customer_agg = df_orders.groupBy('customer_id').agg(
    F.count('order_id').alias('total_orders'),
    F.sum('order_value').alias('total_spent'),
    F.avg('order_value').alias('avg_order_value')
)

display(customer_agg)
```

Learners will clearly see how each customer contributes to total sales.

Step 5 — Sort Aggregation Output for Better Readability

```
customer_agg_sorted = customer_agg.orderBy(F.desc('total_spent'))
display(customer_agg_sorted)
```

Explain why sorting helps in analytics.

Step 6 — Write Aggregated Data to the Silver Layer

```
silver_path = '/Volumes/workspace/default/test/customer_sales_agg'

customer_agg_sorted.write.format('delta').mode('overwrite').option('overwriteSchema',
'true').save(silver_path)
```

Explain why aggregated data is typically stored separately for reporting use cases.

Step 7 — Verify the Saved Silver Table

```
spark.read.format('delta').load(silver_path).show()
```

Emphasize good practice: always verify output after writing.

Post-Lab Practice

Encourage learners to try these small tasks: 1. Calculate the most frequently purchased product. 2. Find the highest priced order. 3. Add a derived column showing whether the order is “high value” (order_value > 500).

Trainer Closing Note

This lab completes the beginner introduction to aggregations. In the next lab, learners will explore more advanced grouping techniques, including multi-column grouping, window functions, and rollups.

End of Lab 2 — Basic Aggregation Transformation (Beginner)