
🚀 Capstone Project – *Retail Sales Analytics with PySpark*

Business Scenario

You are a **data engineer at a retail company**. The company sells products across multiple categories and tracks customer orders. Your task is to use **PySpark in Google Colab** to analyze sales, customers, and products, and generate insights.

Step 1: Setup in Google Colab

```
!pip install pyspark
from pyspark.sql import SparkSession

spark = SparkSession.builder.appName("Retail-Capstone").getOrCreate()
sc = spark.sparkContext
```

Step 2: Prepare Data

Customers Data

```
customers_data = [
    (1, "Rahul Sharma", "Bangalore", 28),
    (2, "Priya Singh", "Delhi", 32),
    (3, "Aman Kumar", "Hyderabad", 25),
    (4, "Sneha Reddy", "Chennai", 35),
    (5, "Arjun Mehta", "Mumbai", 30),
    (6, "Divya Nair", "Delhi", 29)
]
customers_cols = ["customer_id", "name", "city", "age"]
customers_df = spark.createDataFrame(customers_data, customers_cols)
```

Products Data

```
products_data = [
    (101, "Laptop", "Electronics", 55000),
    (102, "Mobile", "Electronics", 25000),
    (103, "Headphones", "Electronics", 3000),
    (104, "Chair", "Furniture", 5000),
    (105, "Book", "Stationery", 700),
    (106, "Shoes", "Fashion", 2500)
]
products_cols = ["product_id", "product_name", "category", "price"]
products_df = spark.createDataFrame(products_data, products_cols)
```

Orders Data

```
orders_data = [
    (1001, 1, 101, 1, "2024-01-10"),
    (1002, 2, 102, 2, "2024-01-12"),
    (1003, 1, 103, 3, "2024-02-05"),
    (1004, 3, 104, 1, "2024-02-08"),
    (1005, 5, 105, 5, "2024-03-01"),
    (1006, 6, 106, 2, "2024-03-15"),
    (1007, 7, 101, 1, "2024-03-20")  # Order with non-existent customer
]
orders_cols = ["order_id", "customer_id", "product_id", "quantity", "order_date"]
orders_df = spark.createDataFrame(orders_data, orders_cols)
```

Step 3: Capstone Tasks

Part A – RDD Basics

1. Convert a list of numbers `[10, 20, 30, 40, 50]` into an RDD.
    - Find sum, max, min, and average using RDD transformations & actions.
2. Create an RDD of sentences and perform **word count**.

Part B – DataFrame Operations

3. Show all customer names and their cities.
4. Find customers older than 30.
5. List all distinct product categories.
6. Find top 3 most expensive products.

Part C – Aggregations

7. Find the average age of customers per city.
8. Calculate total revenue generated from each product.
9. Find the most popular product (by total quantity sold).

Part D – Joins

10. Join customers with orders to list each customer’s purchases.
11. Join orders with products to get order details (product name + category).
12. Find customers who never placed an order.
13. Find products that were never ordered.

Part E – SQL Queries

14. Register all DataFrames (`customers` , `products` , `orders`) as temp views.
15. Query: Find the **top 2 cities by total revenue**.
16. Query: Find customers who spent more than 50,000 in total.
17. Query: Find the **best-selling product category** by revenue.

Part F – File I/O

18. Save the `orders_df` DataFrame into CSV format.
19. Load it back into a new DataFrame.

20. Save the `products_df` DataFrame into JSON format and reload it.

📊 Part G – Visualization (in Colab)

21. Convert PySpark DataFrame → Pandas (`toPandas()`).
22. Plot **Revenue by Category** as a bar chart.
23. Plot **Number of Orders per Month** as a line chart.
24. Plot **Revenue vs Quantity** as a scatter plot.
