
⚡ PySpark Master Task Set

🔍 *Covers: Ingestion, Transformation, Spark SQL, Aggregations, Joins, UDFs, and Storage*

📂 Data Preparation (Use This for All Tasks)

📄 **Dataset 1 – `customers.csv`**

```
CustomerID,Name,Email,City,SignupDate
101,Ali,ali@gmail.com,Mumbai,2022-05-10
102,Neha,neha@yahoo.com,Delhi,2023-01-15
103,Ravi,ravi@hotmail.com,Bangalore,2021-11-01
104,Sneha,sneha@outlook.com,Hyderabad,2020-07-22
105,Amit,amit@gmail.com,Chennai,2023-03-10
```

📄 **Dataset 2 – `orders.csv`**

```
OrderID,CustomerID,Product,Category,Quantity,Price,OrderDate
1,101,Laptop,Electronics,2,50000.0,2024-01-10
2,101,Mouse,Electronics,1,1200.0,2024-01-15
3,102,Tablet,Electronics,1,20000.0,2024-02-01
4,103,Bookshelf,Furniture,1,3500.0,2024-02-10
5,104,Mixer,Appliances,1,5000.0,2024-02-15
6,105,Notebook,Stationery,5,500.0,2024-03-01
7,102,Phone,Electronics,1,30000.0,2024-03-02
```

🚀 TASKS

🔹 1. Data Ingestion & Exploration

- Load both CSV files with schema inference.
- List all columns and data types.
- Count the total number of customers and orders.
- Show distinct cities.

🔹 2. DataFrame Transformations

- Add a column `TotalAmount = Price * Quantity`.
- Create a new column `OrderYear` from `OrderDate`.
- Filter orders with `TotalAmount > 10,000`.
- Drop the `Email` column from `customers`.

🔹 3. Handling Nulls & Conditionals

- Simulate a null in `City` and fill it with “Unknown”.
- Label customers as “Loyal” if SignupDate is before 2022, else “New”.
- Create `OrderType` column: "Low" if < ₹5,000, "High" if ≥ ₹5,000.

🔹 4. Joins & Aggregations

- Join customers and orders on `CustomerID`.

- Get total orders and revenue per city.
- Show top 3 customers by total spend.
- Count how many products each category has sold.

---

5. Spark SQL Tasks

- Create database `sales` and switch to it.
- Save both datasets as tables in the `sales` database.
- Write SQL to:
  - List all orders by customers from “Delhi”.
  - Find average order value in each category.
  - Create a view `monthly_orders` with month-wise total amount.

---

6. String & Date Functions

- Mask emails using regex (e.g., `a***@gmail.com`).
- Concatenate `Name` and `City` as “Name from City”.
- Use `datediff()` to calculate customer age in days.
- Extract month name from `OrderDate`.

---

7. UDFs and Complex Logic

- Write a UDF to tag customers:
  - “Gold” if spend > 50K, “Silver” if 10K–50K, “Bronze” if <10K.
- Write a UDF to shorten product names (first 3 letters + ...).

---

8. Parquet & Views

- Save the joined result as a Parquet file.
- Read it back and verify schema.
- Create and query a global temp view.
- Compare performance between CSV read and Parquet read.
