**Overview:**

You are tasked with analyzing a retail business's inventory and sales data. The organization operates across multiple locations and sells a variety of products to different customer demographics. The data is stored in multiple datasets covering customer information, product inventory, sales transactions, and store details. Your goal is to retrieve insights that can help the business improve its sales performance and inventory management.

**Problem Statement:**

**Q1.** Download the following datasets using the Kaggle API and load them into PySpark DataFrames:

* **customers.csv**: Contains customer details.
  + Download:

bash

Copy code

kaggle datasets download -d hamzabouissi/retail-dataset

Extract and use the file: customers.csv

* **orders.csv**: Contains order details.
  + Download:

bash

Copy code

kaggle datasets download -d hamzabouissi/retail-dataset

Extract and use the file: orders.csv

* **products.csv**: Contains product information.
  + Download:

bash

Copy code

kaggle datasets download -d hamzabouissi/retail-dataset

Extract and use the file: products.csv

* **order\_items.csv**: Contains detailed order line items.
  + Download:

bash

Copy code

kaggle datasets download -d hamzabouissi/retail-dataset

Extract and use the file: order\_items.csv

* **stores.csv**: Contains store information.
  + Download:

bash

Copy code

kaggle datasets download -d hamzabouissi/retail-dataset

Extract and use the file: stores.csv

Use the Kaggle API to download these datasets, and then load them into separate PySpark DataFrames for further processing.

**Q2.** Perform a join on all the DataFrames and create a new DataFrame called Retail\_Full\_DataFrame. Ensure no duplicate columns after the join:

* Join orders.csv with customers.csv using CustomerID.
* Join the result with order\_items.csv using OrderID.
* Join the result with products.csv using ProductID.
* Optionally, join the result with stores.csv using StoreID.

**Q3.** Convert the Order\_Date in orders.csv into DateType, print the schema, and show the top 5 records of the Order\_Date column.

**Q4.** Find the top 3 products that generated the highest total revenue.

* Total revenue = Quantity × UnitPrice

**Q5.** Create a new column Total\_Revenue in the Retail\_Full\_DataFrame that calculates the revenue for each transaction by multiplying the quantity and price of each order.

**Q6.** Find the customer who spent the most money across all transactions.

**Q7.** Using a window function, retrieve the total revenue generated by each store.

**Q8.** Count the number of unique customers who made purchases in January 2022 and how many of them returned to make at least one purchase in each subsequent month throughout the year.

**Q9.** Save the result of Q8 into a JSON file called customer\_retention.json.

### Download Instructions:

You can use the following Python code snippet to download the datasets using the Kaggle API:

python

Copy code

# Install Kaggle API if not already installed

!pip install kaggle

# Set up Kaggle credentials (Make sure to upload kaggle.json to your working directory)

!mkdir -p ~/.kaggle

!cp kaggle.json ~/.kaggle/

!chmod 600 ~/.kaggle/kaggle.json

# Download the retail dataset

!kaggle datasets download -d hamzabouissi/retail-dataset

# Unzip the dataset

!unzip retail-dataset.zip