Lab 1- Assignment

All students must take clear screenshots of each step, including starting the cluster, loading the dataset, running DataFrame operations, executing SQL queries, applying UDFs, and viewing the final results for every task and sub-task. At the end of the file, you must include the complete code for each task, and then upload everything in a PDF file.

You have four labs, and each lab is worth 2.5 credits. To pass, you need a total of 8 credits.

Task1: Retail Store Insights

Scenario: You're a data analyst at a large retail store. The store sells a variety of products, including books and fruits. The management wants insights into sales patterns, customer preferences, product popularity, and potential promotions.

Objective: Analyze the provided dataset to extract meaningful insights and present them to the management.

Instructions:

1. **Initialization:**
    - Set up your environment by initializing a Spark session. Name this session "RetailStoreInsights".
2. **Data Loading:**
    - Load the provided dataset into a DataFrame. This dataset will have information about various products, including their type (e.g., fruit or book) and price. Create DataFrame based on bellow data:

```
# Columns: product_name | category | price | quantity

data = [

  ('Ulysses', 'Book', 23.17, 16),

  ('Apple', 'Fruit', 2.34, 8),

  ('Pineapple', 'Fruit', 2.57, 1),

  ('Apple', 'Fruit', 2.43, 6),

  ('To Kill a Mockingbird', 'Book', 24.14, 19),

  ('To Kill a Mockingbird', 'Book', 11.18, 11),
  ('Watermelon', 'Fruit', 3.35, 15),
  ('Pride and Prejudice', 'Book', 24.99, 3),
  ('To Kill a Mockingbird', 'Book', 21.82, 17),
  ('Moby Dick', 'Book', 14.83, 20),
  ('Pride and Prejudice', 'Book', 5.03, 16),
  ('Jane Eyre', 'Book', 20.40, 8),
  ('Moby Dick', 'Book', 5.55, 20),
  ('Don Quixote', 'Book', 19.75, 17),
  ('Watermelon', 'Fruit', 2.31, 9),


  ('Hamlet', 'Book', 18.20, 12),
  ('Mango', 'Fruit', 4.10, 7),
  ('1984', 'Book', 16.75, 14),
  ('Strawberry', 'Fruit', 1.90, 25),
  ('War and Peace', 'Book', 22.50, 9),
  ('Orange', 'Fruit', 3.05, 13),
  ('The Great Gatsby', 'Book', 12.30, 10),
  ('Peach', 'Fruit', 2.80, 11),
  ('Grapes', 'Fruit', 2.60, 18),
  ('Pride and Prejudice', 'Book', 9.50, 5)
]
```

3.Data Exploration:

- Familiarize yourself with the dataset:
  - Display the first 10 rows to understand the structure and content.
  - Print the schema of the DataFrame to understand the data types and columns.

4.Data Analysis:

- Extract specific columns of interest using the select operation. For this task, focus on the product name and its price.

- Identify and display products that are priced above $2.
- Group the data by product type (e.g., fruit or book) and determine the count for each category.
- Calculate the average price of all products in the dataset.
- The store is considering a promotion where they offer a 10% discount on all products. Add a new column to the DataFrame that calculates the discounted price for each product.
- (Optional) If you find any columns that are not necessary for your analysis, you can drop them from the DataFrame.

17. **SQL Operations:**

- For more complex analysis, register the DataFrame as a temporary SQL table named "retail_sales".
- Using SQL, perform the following operations:
    - Determine the total number of products sold (including duplicates).
    - Calculate the total sales (sum of prices) for each product category (e.g., fruits vs. books).
- (Optional) Identify the products based on the frequency of their occurrence in the dataset. (To find out which products appear most often in the dataset.)

Task2

Scenario: You're a data engineering intern at a tech company. As part of your training, you're given a basic PySpark script that demonstrates the use of Python UDFs and Pandas UDFs. Your task is to modify the script by implementing a new transformation function.

Instructions:

1. **Initialization:**
    - Start a Spark session named "UDFTransformation".
2. **Using Python Native Function as UDF:**
    - Define a Python function that multiplies the input value by 3.
    - Convert this function into a Spark UDF.
    - Create a DataFrame with a series of numbers: [(4,), (5,), (6,)].

- Apply the UDF to the DataFrame to triple each number.
- Display the updated DataFrame.

3. **Using Pandas UDF:**
   - Define a Pandas UDF that subtracts 2 from each value in the input series.
   - Apply this UDF to the DataFrame from step 2.
   - Display the updated DataFrame with the subtracted values.
4. **Cleanup:**
   - Stop the Spark session.