Pandas Assignment

Objectives:

- Goal: Master real-world data analysis using Pandas — from basic operations to advanced insights — using a single dataset.
- Dataset: Sales_100k_Dataset.csv
- Skill Levels: Beginner → Intermediate → Advanced
- Note: Store all your answers in one master Jupyter Notebook or Colab.

Level 1 – Beginner (1–30)

Objective: Build your foundation by exploring and manipulating data.

1. Load the dataset and print the shape, column names, and data types.
2. Show the first 10 and last 10 rows of the dataset.
3. Convert OrderDate to datetime type.
4. Display the total number of unique customers.
5. Show all unique product categories and count them.
6. Filter all orders made in the city of "Los Angeles".
7. Select orders where quantity is more than 5.
8. Add a new column UnitTotal = Quantity * Price and compare with TotalAmount. Flag mismatches.
9. Create a new column DiscountApplied (True if UnitTotal > TotalAmount).

10.Drop the column DiscountApplied after saving it to a variable.

11. Filter all returned orders.
12. Show top 5 cities by number of orders.
13. What are the top 3 most sold products (by quantity)?
14. Rename CustomerName to CustName and revert it.

15.Check and count missing values in each column.

16. Fill missing values in City (if any) with "Unknown".

17. Remove duplicates based on OrderID.

18. Sort the data by TotalAmount in descending order.

19. Get the top 5 orders with the highest total amount.

20. Extract year and month from OrderDate.

21.Create a new column OrderMonth with full month names.

22. Show all orders from Q1 of any year (Jan–Mar).

23.Count number of orders for each region.

24.Display all customers who bought both "Phone" and "Laptop".

25.Convert all city names to uppercase and restore them back.

26. Find average price of each category.

27.Use groupby() to find total sales by Region.

28. Save only East region data to a CSV file.

29. Reset and set index using OrderID.

30.Display all orders placed on a weekend.

✓ Level 2 – Intermediate (31–60)

Objective: Apply groupings, merges, aggregations, reshaping, and logic.

31.Create a pivot table of TotalAmount by Region and Category.

32.Unpivot (melt) the pivot table from Task 31.

33.Use groupby() to get average Price by Product and Region.

34.Use .agg() to get min, max, and mean quantity for each category.

35.Create a lookup table for CustomerID and CustomerName and remove duplicates.

Create a lookup for Product and Category, remove duplicates.

37.Merge back the lookups with the main file (simulate master lookup join).

38.Compare the original and merged DataFrames – are all rows retained?

39.Calculate return rate % by region (total returned orders / total orders).

40.Create a column HighValue where

Level 3 – Advanced Pandas Tasks (61–90)

Objective: Perform deeper analytics, optimize performance, and simulate real-world business reporting use cases.

Dataset: Sales_100k_Dataset.csv

Data Transformations & Aggregations

51.Create a new column DiscountedPrice assuming a flat 10% discount on all products.

52.Use .rolling() to calculate a 7-day rolling average of TotalAmount per city.

53.Calculate the cumulative revenue for each Region sorted by OrderDate.

54. Rank customers by their total revenue generated (TotalAmount) in descending order.

55.Calculate the average order value per customer using groupby().

56. Generate a report showing total quantity sold per Category per Region.

57.Use .agg() to compute multiple stats (sum, mean, min, max) for TotalAmount per Region.

58.Create a summary DataFrame showing each customer's: total orders, total amount spent, and total products bought.

Time Series & Date Handling

59.Convert the OrderDate to datetime and extract new columns: Year, Month, and Weekday.

60.Create a monthly revenue trend using resample() on OrderDate.

61. Find the day of the week with the highest sales volume.

62.Compare monthly returned orders count vs. total orders.

63. Find peak sales month and peak returned month for each Region.

64. Identify cities that saw consistent monthly growth for 3+ months in a row.

65. Generate a YoY (Year-over-Year) growth percentage in TotalAmount for each Region

String & Conditional Logic

66. Extract only the first name from the CustomerName column.

77.Create a column HighValueOrder where TotalAmount > 5000.

78. Flag suspicious orders where Quantity > 10 and Price > 1000.

79. Normalize all text columns (lowercase, strip spaces, remove special chars if any).

80. Filter all products that contain the substring 'top' (case insensitive).

Advanced Features

81.Use pivot_table() to summarize TotalAmount by Region and Category.

82.Create a column to show each order's percentage of the total monthly revenue.

83.Use .transform() to normalize TotalAmount within each Region.

84. Perform one-hot encoding of the Region column.

85. Generate a heatmap-ready table using pivot_table() showing quantity sold by City and Product.

Merging and Lookup Files (Master File Creation)

86.Create a customer lookup DataFrame with unique CustomerID, CustomerName, and total orders. Save it.

87.Create a product lookup DataFrame with unique Product, Category, and average Price. Save it.

88.Merge the original Sales_100k_Dataset.csv with both the customer and product lookup DataFrames.

89. After merge, create a master file with these columns: OrderID, OrderDate, CustomerName, Product, Category, Price, Quantity, TotalAmount, Returned, and the AvgProductPrice and TotalCustomerOrders from lookup tables.

90. Analyze and visualize the top 5 customers and top 5 products contributing to revenue using the master file.