Scenario-Based Assignment: Python, NumPy, and Pandas

Scenario: E-commerce Customer & Sales Analysis

You are working as a data analyst at an e-commerce company. The company has provided you with two datasets:

1. customers.csv - Contains details about customers:
   - customer_id (int)
   - name (str)
   - email (str)
   - country (str)
   - signup_date (str, format: YYYY-MM-DD)

2. sales.csv - Contains purchase records:
   - order_id (int)
   - customer_id (int)
   - order_date (str, format: YYYY-MM-DD)
   - product (str)
   - category (str)
   - quantity (int)
   - price_per_unit (float)

Your job is to perform data cleaning, analysis, and transformation using Python, NumPy, and Pandas. Avoid visualization and focus on understanding and transforming the data.

Assignment Questions (From Easy to Advanced)

1. Load both CSV files into separate Pandas DataFrames.

2. Display the first 5 and last 5 rows of each DataFrame.

Scenario-Based Assignment: Python, NumPy, and Pandas

3. Show the column names, data types, and check for null values in both datasets.

4. Convert the date columns ('signup_date' and 'order_date') to datetime objects.

5. Calculate the total revenue for each order (quantity * price_per_unit) and create a new column 'total_amount'.

6. Merge the customers and sales datasets on 'customer_id'.

7. Find the top 5 customers who spent the most overall.

8. Count how many customers are from each country.

9. Calculate the average order value per customer.

10. Remove any duplicate records from both datasets.

11. Identify and handle any missing or invalid data (e.g., negative quantity or price).

12. Group the merged data by category and find:
- Total quantity sold per category
- Total revenue per category

13. Create a new column that extracts the year and month from the 'order_date' and analyze monthly sales.

14. Find customers who signed up in the last 6 months but haven't made any purchases.

15. Identify products that were sold less than 10 times in total (low performers).

Bonus:

16. Create a summary report DataFrame with the following per customer:

- Total orders
- Total items bought
- Total money spent
- Average order value

17. Use NumPy to perform any custom operation (e.g., apply discount rule using vectorized operations).