Experiment No: 1

Text File Analysis for Product Reviews

Objective:

To develop a Python script that reads multiple customer review files from a directory, extracts structured data using regular expressions, computes the average rating per product, identifies the top-rated products, and writes a detailed summary to a text file.

Task Description:

Consider a scenario where you are working as a data scientist for a large e-commerce company. Your team is responsible for analyzing customer feedback data, which is stored in multiple text files. Each text file contains customer reviews for different product categories. Your task is to write a Python script that performs the following operations:

Read the contents of all the text files in a given directory.

For each review, extract the following information:

- Customer ID (a 6-digit alphanumeric code)
- Product ID (a 10-digit alphanumeric code)
- Review date (in the format "YYYY-MM-DD")
- Review rating (an integer between 1 and 5)
- Review text (the actual feedback provided by the customer)

Calculate the average review rating for each product and store it in a dictionary where the product ID is the key and the average rating is the value.

Determine the top 3 products with the highest average review ratings.

Create a new text file named "summary.txt" and write the following information into it:

- The total number of reviews processed.
- The total number of valid reviews (reviews with all required information extracted successfully).
- The total number of invalid reviews (reviews with missing or incorrect information).
- The product ID and average rating of the top 3 products with the highest average ratings.

Your Python script should be robust, handling any potential errors or exceptions during the file handling process.

Additionally, you should implement efficient algorithms to handle large volumes of data without consuming excessive memory or processing time.

Write the Python script to achieve the above objectives and provide detailed comments explaining each step of your implementation.

Steps to Perform the Program:

1. Import required libraries:
    - os, re, defaultdict from collections
2. Set the directory path where text files containing reviews are stored.
3. Initialize data structures:
    - A dictionary to store product IDs and associated ratings.
    - Counters for total, valid, and invalid reviews.
4. Iterate through each .txt file in the directory:
    - Open and read the file contents.
    - For each review block, use regular expressions to extract:
        - Customer ID (6 alphanumeric)
        - Product ID (10 alphanumeric)
        - Review Date (YYYY-MM-DD)
        - Rating (1–5)
        - Review Text
5. Validate and store data:
    - If all fields are present and valid, store the rating under the product ID.
    - If any field is missing or malformed, count it as invalid.
6. Compute average ratings for each product:
    - Use sum and count of ratings to compute the average.
7. Identify top 3 products:
    - Sort products by average rating in descending order.
8. Write to summary.txt:
    - Total number of reviews
    - Valid and invalid review counts
    - Top 3 products with their average ratings
9. Handle all exceptions (e.g., missing file, read errors, regex mismatches) using try-except blocks.