Python programming Lab(23CP301P)

Name: Krishika Vansh   **Semester: V**

Roll No: 23BCP448   **Faculty: Mr. Davinder Singh**

Division: VII Batch G13

Branch: Computer Engineering

School of Technology

November 2025

LAB1:

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.

Code:

```python
import os

import re

from collections import defaultdict


directory = "reviews"


product_ratings = defaultdict(list)

total_reviews = 0

valid_reviews = 0

invalid_reviews = 0


customer_pattern = re.compile(r"[A-Za-z0-9]{6}")

product_pattern = re.compile(r"[A-Za-z0-9]{10}")

date_pattern = re.compile(r"\d{4}-\d{2}-\d{2}")

rating_pattern = re.compile(r"\b[1-5]\b")

review_text_pattern = re.compile(r"Review:\s*(.*)")


for filename in os.listdir(directory):

    if filename.endswith(".txt"):
```

```python
        filepath = os.path.join(directory, filename)

    try:

        with open(filepath, "r", encoding="utf-8") as file:

            content = file.read()

            reviews = content.strip().split("\n\n")


            for review in reviews:

                total_reviews += 1

                customer_id = customer_pattern.search(review)

                product_id = product_pattern.search(review)

                review_date = date_pattern.search(review)

                rating = rating_pattern.search(review)

                review_text = review_text_pattern.search(review)


                if all([customer_id, product_id, review_date, rating, review_text]):

                    valid_reviews += 1

                    product_ratings[product_id.group()].append(int(rating.group()))

                else:

                    invalid_reviews += 1


    except Exception as e:

        print(f"Error reading {filename}: {e}")


avg_ratings = {

  pid: round(sum(ratings) / len(ratings), 2)

  for pid, ratings in product_ratings.items()

}
```

```python
top_products = sorted(avg_ratings.items(), key=lambda x: x[1], reverse=True)[:3]


with open("summary.txt", "w") as summary:

    summary.write(f"Total reviews processed: {total_reviews}\n")

    summary.write(f"Valid reviews: {valid_reviews}\n")

    summary.write(f"Invalid reviews: {invalid_reviews}\n")

    summary.write("\nTop 3 Products:\n")

    for pid, avg in top_products:

        summary.write(f"{pid}: {avg}\n")


print("Summary saved to 'summary.txt'")
```
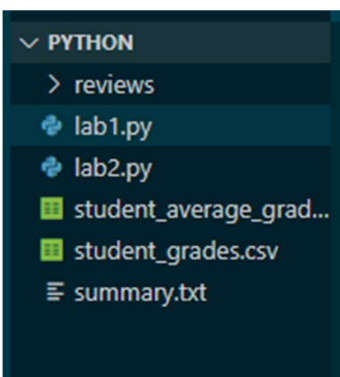
Output:

```
[Running] python -u "c:\Users\Admin\OneDrive\Desktop\python\lab1.py"
Summary saved to 'summary.txt'

[Done] exited with code=0 in 0.066 seconds
```

```
∨ PYTHON
  > reviews
  🐍 lab1.py
  🐍 lab2.py
  ▦ student_average_grad...
  ▦ student_grades.csv
  ≡ summary.txt
```

Summary.txt:

Total reviews processed: 6

Valid reviews: 6

Invalid reviews: 0

Top 3 Products:CustomerID: 3.83