

---

# Application-Based File Handling Exercises

---

(Real-time, corporate-grade tasks)

---

## 1. Student Certificate Batch Generator

Write a program that reads a list of student names from `students.txt` and generates a certificate file for each student automatically.

---

## 2. Expense Report Generator

Create a function that accepts multiple expense items (name, amount) and writes a formatted expense report into `report.txt`.

---

## 3. Attendance Log Writer

For each student name given by the user, append an entry to `attendance.log` with timestamp and status (Present/Absent).

---

## 4. Invoice Generator from CSV

Read a CSV file `orders.csv` containing columns: item, quantity, price.

Generate an invoice file that lists all items and the final total.

---

## 5. Automated Welcome Letter Creator

Write a function that takes a student's name and course, and generates a personalized welcome letter in a text file.

---

## 6. Error Log Application

Build a function `log_error(message)` that writes errors into `error.log` with timestamps.

Simulate errors by calling the function 5 times.

---

## 7. Product Catalog Formatter

Read a text file `products.txt` with product names and prices.

Generate a formatted catalog file `catalog.txt` with alignment.

Example format:

Product	Price
Laptop	55000
Mouse	800

---

## 8. Quiz File Generator

Write a function that takes a list of questions and writes them into `quiz.txt` with question numbers.

Include blank lines for answers.

---

## 9. Order Summary From User Input

Ask the user for item name, quantity, and price for 3 items.

Then generate a summary file `order_summary.txt` with totals.

---

## 10. System Log Analyzer

Read `system.log` and count:

- number of ERROR lines
- number of WARNING lines
- number of INFO lines

Write the summary into `log_summary.txt`.

---

## 11. Email Template Generator

Write a script that reads names from `names.txt` and generates email templates like:

Dear <name>,

Your training session starts tomorrow.

Regards,

Training Team

Each email should be saved into separate files.

---

## 12. Contact Exporter

Read a CSV file `contacts.csv` with name and phone columns.

Generate a formatted text file `contacts_export.txt` listing:

Name: Rahul | Phone: 9988776655

Name: Aisha | Phone: 9123456789

---