1-Create a program that manages a list of employees. There are two types of employees: FullTimeEmployee and PartTimeEmployee, both derived from a base class Employee. Each class should store the employee's name and calculate salary differently.

Requirements:

- Use inheritance and virtual methods for calculating salary.
- Read employee details from a file ("employees.txt").
- Write processed salary information to another file ("salaries.txt").
- Demonstrate polymorphism by storing a list of pointers to Employee and invoking salary calculation.

File Format Example (employees.txt**):**

FullTime John 4000

PartTime Alice 20 100

(Where FullTime is salary per month; PartTime is hours and rate per hour)

2-Model a set of geometric shapes (Circle, Rectangle, etc.) using a base class Shape with a virtual function area(). The program reads shape data from a file, computes areas, and writes the results to an output file.

Requirements:

- Use inheritance and virtual functions for area calculation.
- Read the shape type and dimensions from a file ("shapes.txt").
- Output the area of each shape to a file ("areas.txt").
- Use polymorphism to handle different derived classes uniformly.

File Format Example (shapes.txt**):**

Circle 3.5

Rectangle 4 6

3-Design a bank system with different account types (SavingsAccount, CheckingAccount) inheriting from a base Account class. Each has a virtual method applyInterestOrFees().

Requirements:

- Account details and transactions are read from a file ("accounts.txt").
- After processing, updated account information is written to a file ("updated_accounts.txt").
- Demonstrate polymorphism by processing all account types in a single collection.

File Format Example (accounts.txt**):**

Savings JohnDoe 4000

Checking Alice 500 2.5

4-Implement a library inventory system with different types of media (Book, DVD), both inheriting from a base class MediaItem. Each has a method getLateFee(int daysLate) that is overridden by the derived classes.

Requirements:

- Media items and late return information are read from "media.txt".
- The program calculates the late fee via polymorphism and writes results to "late_fees.txt".

File Format Example (media.txt**):**

Book Hamlet 5

DVD Matrix 3

5-Create a system for generating student transcripts. There are undergraduate and graduate students, derived from a common Student base. Both override a virtual function that calculates GPA differently.

Requirements:

- Read student records and course grades from a file ("students.txt").
- Calculate and write GPAs to "transcripts.txt".
- Store all students in a base class pointer array/list for unified processing using polymorphism.

File Format Example (students.txt**):**

Undergraduate John 4 3.5 3.0 4.0 2.5

Graduate Alice 3 4.0 3.7 3.9