9. Practice problems on classes and OOPs

1. Model a simple bank account. It must have:
    a. Class BankAccount with private data: owner (std::string) and balance (double).
    b. Two constructors with default values of (owner="", balance=0) and (owner, initialBalance).
    c. Public methods are
        i. deposit(double), withdraw(double) (refuse overdraft)
        ii. double getBalance() const;
        iii. const std::string& getOwner() const;

2. Write a program with a class that logs data to the screen
    a. Class Logger with function void log(const std::string&) const; This function just prints a message on the screen
    b. Class TimestampedLogger : public Logger that overrides log to print a fake timestamp, then calls Logger::log(msg) to print the message.
    c. Show both logs in main function.

3. Write a program to implement a base class Shape and two derived classes Rectangle and Circle with the following conditions:
    a. Abstract class Shape with pure virtual double area() const = 0; and virtual destructor.
    b. Rectangle (double w, double h) and Circle(double r) that override area() in Shape class.
    c. Write a function printArea(const Shape&) that prints the area.
    d. In main, create a Rectangle and Circle, call printArea on each.

4. Write a program that will generate a compile time error. The following aspects should be covered:
    a. Class Point with private variables x, y with getters
    b. Methods of Point class are:
        i. void move(double dx, double dy); that moves x and y by dx and dy
        ii. void print() const; that prints (x,y)
    c. Write a function outside the class called showPoint(const Point&) that calls print().
    d. Try calling move() from showPoint(). The program should not compile.

5. Write a program with the following features:
    a. Class Course with
        i. variable called name of type std::string and credits of type int,
        ii. A print() const; function that prints the variable values in the class
    b. Class Student with

    i. variable name and a single Course member.
    ii. Student() constructor which takes as input, the student name and a Course object, and stores the data in the class member variables with the constructor.
    iii. void print() const; function that prints both the student name and the course info (by calling print() belonging to Course class)

c. the main() function that creates a Student object with user defined values and calls the void print() const; function.