Practice set -6
Object-Oriented Programming

1. **Bank Account with Friend Function and Static Variable**
   Create a `Bank` class with:
   - Private: `accountNumber`, `balance`
   - Public: Constructor to initialize, `deposit()`, and `withdraw()`
   - Static variable: `interestRate`
   - Friend function: `applyInterest()` to apply interest to the balance
2. **Student Class with Constructor Overloading**
   Create a `Student` class with:
   - Private: `name`, `rollNumber`, `marks`
   - Public: Constructor overloading:
     - Default constructor
     - Constructor with `name` and `rollNumber`
     - Constructor with `name`, `rollNumber`, and `marks`
3. **Employee Class with Static Count**
   Create an `Employee` class with:
   - Private: `id`, `salary`
   - Public: Constructor to initialize values
   - Static variable: `employeeCount` (to track the number of employees)
4. **Rectangle Class with Friend Function**
   Create a `Rectangle` class with:
   - Private: `length`, `breadth`
   - Public: Constructor, `area()`
   - Friend function: `compareArea()` to compare areas of two rectangles
5. **Car Class with Destructor**
   Create a `Car` class with:
   - Private: `brand`, `price`

- Public: Constructor, Destructor (display a message when object is destroyed)

6. **Complex Number Class with Friend Function**
   Create a `Complex` class with:
   - Private: `real`, `imag`
   - Public: Constructor, `display()`
   - Friend function: `addComplex()` to add two complex numbers

7. **Counter Class with Static Variable**
   Create a `Counter` class with:
   - Private: `count`
   - Public: Constructor initializes `count`
   - Static variable: `totalCount` to track all counter objects

8. **Circle Class with Static Function**
   Create a `Circle` class with:
   - Private: `radius`
   - Public: Constructor, `area()`
   - Static function: `setPi()` to update value of π

9. **Vehicle Class with Inheritance and Constructor Overloading**
   Create a `Vehicle` base class and a `Car` derived class with:
   - Protected: `brand`
   - Public: Constructor overloading in `Car` for different car types

10. **Matrix Class with Friend Function for Addition**
    Create a `Matrix` class with:
    - Private: 2D array
    - Public: Constructor, `display()`
    - Friend function: `addMatrices()` to add two matrices