Assignments on Class, Constructors, Overloading

**1. Book Discount Calculator**

**Topics:** Class, Methods, Conditional Logic

**Problem Statement:**  
Create a class Book with:

* Title, Author (string)
* Price (double)
* IsBestseller (bool)

Create a method CalculateDiscountedPrice():

* If IsBestseller is true, apply a 10% discount.
* If price is above 500, apply a 5% additional discount.

**Tasks:**

* Create 2 books, one bestseller, one not.
* Print the original and discounted price using the method.

### 2. ****Rectangle - Validate and Compute****

**Topics:** Constructor Overloading, Input Validation

**Problem Statement:**  
Create a class Rectangle:

* Properties: Length, Width
* Two constructors:
  + Default: sets both to 1
  + Parameterized: assigns values only if > 0, else sets to 1 (validation)

Method: GetArea()  
Method: IsSquare() – returns true if length == width

**Tasks:**

* Create 2 rectangles (one square, one not)
* Use both constructors
* Display area and whether each rectangle is a square

### 3. ****Calculator with Method Overloading****

**Topics:** Method Overloading, Input Validation

**Problem Statement:**  
Create a class Calculator with overloaded Add methods:

* Add(int a, int b)
* Add(double a, double b)
* Add(int[] numbers) – sums all numbers

Each method should:

* Check if input values are valid (e.g., array not null)
* Log a warning if invalid input, return 0

**Tasks:**

* Call each method with valid and invalid inputs
* Print results and error messages

Adding two integers: 5 + 10

Result: 15

Adding two doubles: 3.5 + 4.7

Result: 8.2

Adding array of integers: 1 + 2 + 3 + 4 + 5

Result: 15

Invalid input: array is null or empty.

Result (should be 0): 0

Invalid input: array is null or empty.

Result (should be 0): 0

### 4. ****Student Grading System****

**Topics:** Constructor Overloading, Method with Logic

**Problem Statement:**  
Create a class Student:

* Name, Marks (int), Grade (char)

Constructors:

* One with only Name
* One with Name and Marks
* One with all three

Method: AssignGrade():

* If marks >= 90 → A
* 80-89 → B
* 70-79 → C
* Below 70 → D

DisplayInfo() should show name, marks, and grade

**Tasks:**

* Create 3 students using different constructors
* Call AssignGrade() then DisplayInfo() for each

### ****5. Bank Account Game****

**Topics:** Class, Constructor Overloading, Logic in Methods

**Concept:** Simulate a bank account with deposit/withdraw logic and validation.

**Game Idea:**

* Create class BankAccount having properties AccountHolder and Balance. Add appropriate Constructors. Add WithDraw(double amount) and Deposit(double amount) for the class
* Create 2 accounts with random starting balances
* Each takes turns to deposit/withdraw.
* Display winner with higher balance after 3 rounds.

Example

Round 1:

Account1 Withdraw or Deposit (w/d): w

Amount to Withdraw : 1000

Account2 Withdraw or Deposit (w/d): d

Amount to Deposit: 1500

Round 2:

Account1 Withdraw or Deposit (w/d): d

Amount to Deposit : 800

Account2 Withdraw or Deposit (w/d): W

Amount to Withdraw : 2000

Round 3:

Account1 Withdraw or Deposit (w/d): W

Amount to Withdraw : 2000

Account2 Withdraw or Deposit (w/d): D

Amount to Deposit: 3000