Assignment 05

First Project:

1-Define 3D Point Class and the basic Constructors (use chaining in constructors).

2-Override the ToString Function to produce this output:

Point3D P = new Point3D (10,10,10);

Console. WriteLine (P. ToString( ));

Output: “Point Coordinates: (10, 10, 10)”.

3- Read from the User the Coordinates for 2 points P1, P2 (Check the input using try Pares, Parse, Convert).

4-Try to use ==

If (P1 == P2) Does it work properly?

5-Define an array of points and sort this array based on X & Y coordinates.

6-Implement ICloneable interface to be able to clone the object.

To implement more than one interface.

class Point3D:IComparable ,ICloneable

Second Project:

Define Class Maths that has four methods: Add, Subtract, Multiply, and Divide, each of them takes two parameters. Call each method in Main ().

Modify the program so that you do not have to create an instance of class to call the four methods.

Third Project:

You are tasked with designing a system for an e-commerce platform that calculates discounts for different types of users and products. This system should utilize abstraction and include the following parts:

Part 1: Abstract Discount Class

1. **Create an abstract class Discount with:**

  - An abstract method CalculateDiscount(decimal price, int quantity) that returns the discount amount based on the original price and quantity.
  - A Name property to store the type of discount.

Part 2: Specific Discounts

2. **Implement the following concrete discount classes:**
    - **PercentageDiscount:**
        - Accepts a percentage (e.g., 10%).
        - Formula: Discount Amount=Price×Quantity×(Percentage/100)
    - **FlatDiscount:**
        - Accepts a fixed amount to be deducted (e.g., $50).
        - Formula: Discount Amount=Flat Amount×min(Quantity,1)
    - **BuyOneGetOneDiscount:**
        - Applies a 50% discount if the quantity is greater than 1.
        - Formula: Discount Amount=(Price/2)×(Quantity÷2)

Part 3: Discount Applicability

3. **Create an abstract class User with:**
    - A Name property to store the user name.
    - An abstract method GetDiscount() that returns a Discount object.
4. **Implement the following specific user types:**
    - RegularUser: Applies a PercentageDiscount of 5%.
    - PremiumUser: Applies a FlatDiscount of $100.
    - GuestUser: No discount is applied

Part 4: Integration

5. **Write a program that:**

    - Ask the user to input their type (Regular, Premium, or Guest).
    - Allows the user to input product details (price and quantity).
    - Calculates and displays the total discount and final price after applying the appropriate discount.