**Code Clinic Part 1**

**Question 1**

Write a class called **products** that holds data about an item in a retail store. The class should have the following member variables:

* **itemNumber** an int that holds the item’s item number.
* **quantity** an int for holding the quantity of the items on hand.
* **cost** A double for holding the wholesale per-unit cost of the item
* **totalCost** A double for holding the total inventory cost of the item (calculated as quantity times cost).

Please ensure you have a .h file for all the definitions and a .cpp file for class method implementation.

|  |  |
| --- | --- |
| **Member Functions** | **Description** |
| **Default Constructor** | Sets all the member variables to 0. |
| **Constructor #2** | Accepts an item’s number, cost, and quantity as arguments.  The function should copy these values to the appropriate  member variables and then call the setTotalCost function. |
| **setItemNumber** | Accepts an integer argument that is copied to the  itemNumber member variable. |
| **setQuantity** | Accepts an integer argument that is copied to the quantity member variable. |
| **setCost** | Accepts a double argument that is copied to the cost member variable. |
| **setTotalCost** | Calculates the total inventory cost for the item ( quantity times cost ) and stores the result in totalCost . |
| **getItemNumber** | Returns the value in itemNumber . |
| **getQuantity** | Returns the value in quantity . |
| **getCost** | Returns the value in cost . |
| **getTotalCost** | Returns the value in totalCost . |

**Question 2**

Demonstrate the class in a driver program.

*Input Validation: Do not accept negative values for item number, quantity, or cost.*

**Problem solving tip**

Before you look at the answers, try to take out a piece of paper and work out the code.

You can have a header file .h and also a .cpp file for class method definitions, and another .cpp file for testing this class. Total 3 files.

**Question 3**

Create another .cpp file called anotherTest.cpp. Create a five element array called inventory. This array should be able to store 5 objects created from the products class. You can add these information

|  |  |  |  |
| --- | --- | --- | --- |
| Item\_number | Quantity | Cost | Total Cost |
| 1 | 12 | 10 | 120 |
| 2 | 3 | 5 | 15 |
| 3 | 4 | 7 | 28 |
| 4 | 5 | 6 | 30 |
| 5 | 6 | 8 | 48 |

Create another method that can add up the total cost for all the five elements added up together

**Question 4**

Create a child class called saleItem. Create a new method called discount\_reduction. Reduce the price of the product based on the discount passed through the method argument. You can pass the entire product object in the saleItem default constructor and then reduce the price, this will be better.