Product Class

using System;

using System.Collections;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace project1

{

class Product

{

public int pCode;

public int qtyInStock;

public readonly double discountAllowed;

public static String brand;

public String productName;

double price;

public Product() { }

public Product(int pCode, String productName,int qtyInStock, int price,

double discountAllowed)

{

this.pCode = pCode;

this.qtyInStock = qtyInStock;

this.price = price;

this.discountAllowed = discountAllowed;

this.productName = productName;

}

//public void ProductToPurchase(string pname , int qty)

//{

//}

public void Calculatetotalamount(Product product, int qty)

{

double amount;

amount = (product.price \* qty) - product.discountAllowed;

double total\_amount = amount \* (50 / 100);

total\_amount = amount - total\_amount;

Printbill( product , qty , total\_amount);

}

public void Printbill(Product product, int qty , double total\_amount)

{

Console.WriteLine("product Name " + product.productName);

Console.WriteLine("Qty " + qty );

Console.WriteLine("Price " + product.price);

Console.WriteLine("Total Amount " + total\_amount );

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace project1

{

class program2

{

static List<Product> productlist;

static void Main()

{

// Collection Initializer

productlist = new List<Product>()

{

new Product(101, "Mouse", 90, 900, 100),

new Product(102, "Plotter", 10, 1900, 100),

new Product(103, "Printer", 10, 1900, 100),

new Product(104, "Scanner", 10, 1900, 200),

new Product(105, "Motherboard", 10, 1900,250)

};

Console.WriteLine("List of Items");

foreach(Product temp in productlist)

{

Console.WriteLine(temp.productName + "\t" + temp.qtyInStock); ;

}

int qty;

string pname;

Console.WriteLine("Enter product name");

pname = Console.ReadLine();

Console.WriteLine("Enter quantity");

qty = Convert.ToByte(Console.ReadLine());

Product productToPurchase=SearchProduct(pname, qty);

if(productToPurchase !=null)

{

Product product = new Product();

product.Calculatetotalamount(productToPurchase, qty);

}

else

{

Console.WriteLine("Product Not Found Or Qty is insuffficient");

}

}

static Product SearchProduct(string pname, int qty)

{

Product product = null;

foreach (Product temp in productlist)

{

if (temp.productName == pname)

{

product = temp;

if (qty <= temp.qtyInStock)

{

Console.WriteLine(temp.productName + "\t" + temp.qtyInStock); ;

}

else

{

product = null;

//Console.WriteLine("Insufficient Qty");

}

}

}

return product;

}

}

}