**Course Project – Sales Receipt - 2**

Tim Mastarone

Rasmussen University

Microsoft C# Programming

Instructor: Jim Barringer

Course Project – Module 02

April 14, 2024

The image below is a screen shot of my course project C# console app after a user enters information:

A screenshot of a computer

Description automatically generated

With the modifications the user is presented with a menu of products to choose from. The product names and prices are pulled from arrays.

**Below is my C# code from the SalesReceipt solution (Program.cs):**

class Program

{

//4% fixed tax rate

const double taxRate = .04;

static double[] items = new double[5];

static string[] productNames = { "Product 1", "Product 2", "Product 3", "Product 4", "Product 5" };

static double[] productPrices = { 10.0, 20.0, 30.0, 40.0, 50.0 };

static void Main(string[] args)

{

Customer customer = new();

Console.WriteLine("Enter your first name: ");

customer.FirstName = Console.ReadLine();

Console.WriteLine("Enter your last name: ");

customer.LastName = Console.ReadLine();

Console.WriteLine("Enter your phone number: ");

customer.Phone = Console.ReadLine();

Console.WriteLine("Enter your street address: ");

customer.StAddress = Console.ReadLine();

Console.WriteLine("Enter your email address: ");

customer.Email = Console.ReadLine();

// Display available products

Console.WriteLine("Available Products:");

for (int i = 0; i < productNames.Length; i++)

{

Console.WriteLine($"{i + 1}. {productNames[i]} - ${productPrices[i]}");

}

// Loop 5 times for the items the customer purchased

for (int i = 0; i < 5; i++)

{

Console.WriteLine($"Enter the index of item #{i + 1}:");

string? userInput = Console.ReadLine();

if (int.TryParse(userInput, out int itemIndex) && itemIndex >= 1 && itemIndex <= productNames.Length)

{

customer.ItemsTotal += productPrices[itemIndex - 1];

items[i] = productPrices[itemIndex - 1];

}

else

{

Console.WriteLine("Invalid input. Please enter a valid index.");

i--; // Decrement i to repeat the current iteration

}

}

double taxAmount;

taxAmount = CalculateTax(customer.ItemsTotal);

Console.WriteLine("Customer Information");

Console.WriteLine("========================");

Console.WriteLine("Name: " + customer.FirstName + ":" + customer.LastName);

Console.WriteLine("Phone: " + customer.Phone);

Console.WriteLine("Address: " + customer.StAddress);

Console.WriteLine("Email: " + customer.Email);

Console.WriteLine("");

Console.WriteLine("Sales Receipt");

Console.WriteLine("=============");

for (int i = 0; i < 5; i++)

{

Console.WriteLine("Item #" + i.ToString() + ": " + items[i]);

}

Console.WriteLine("Items Subtotal: $" + customer.ItemsTotal.ToString("F2"));

Console.WriteLine("Tax Amount: $" + taxAmount.ToString("F2"));

Console.WriteLine("Items Total Cost: $" + (customer.ItemsTotal + taxAmount).ToString("F2"));

}

static double CalculateTax(double subtotal)

{

double tax = 0;

tax = taxRate \* subtotal;

return tax;

}

}

public class Customer

{

public string? FirstName { get; set; }

public string? LastName { get; set; }

public string? Phone { get; set; }

public string? StAddress { get; set; }

public string? Email { get; set; }

public double ItemsTotal { get; set; }

}