Cognizant Digital Nurture 4.0 – DotNet FSE Deep Skilling

# Week 1 Exercises Report – Roll No: 22052918

## Exercise 2: E-commerce Platform Search Function

This C# program simulates a product search feature for an e-commerce platform. It allows searching products by keyword and lists the results sorted by price.

*using System;*

*using System.Collections.Generic;*

*using System.Linq;*

*class Product*

*{*

*public string Name { get; set; }*

*public string Category { get; set; }*

*public double Price { get; set; }*

*public Product(string name, string category, double price)*

*{*

*Name = name;*

*Category = category;*

*Price = price;*

*}*

*public override string ToString()*

*{*

*return $"Product: {Name}, Category: {Category}, Price: ₹{Price}";*

*}*

*}*

*class Program*

*{*

*static void Main()*

*{*

*List<Product> products = new List<Product>*

*{*

*new Product("Wireless Mouse", "Electronics", 699),*

*new Product("Bluetooth Speaker", "Electronics", 1299),*

*new Product("Running Shoes", "Footwear", 2499),*

*new Product("Cotton T-Shirt", "Apparel", 599),*

*new Product("Water Bottle", "Kitchen", 299)*

*};*

*Console.Write("Enter search keyword: ");*

*string keyword = Console.ReadLine().ToLower();*

*var results = products*

*.Where(p => p.Name.ToLower().Contains(keyword) || p.Category.ToLower().Contains(keyword))*

*.OrderBy(p => p.Price)*

*.ToList();*

*Console.WriteLine("\nSearch Results:");*

*if (results.Count == 0)*

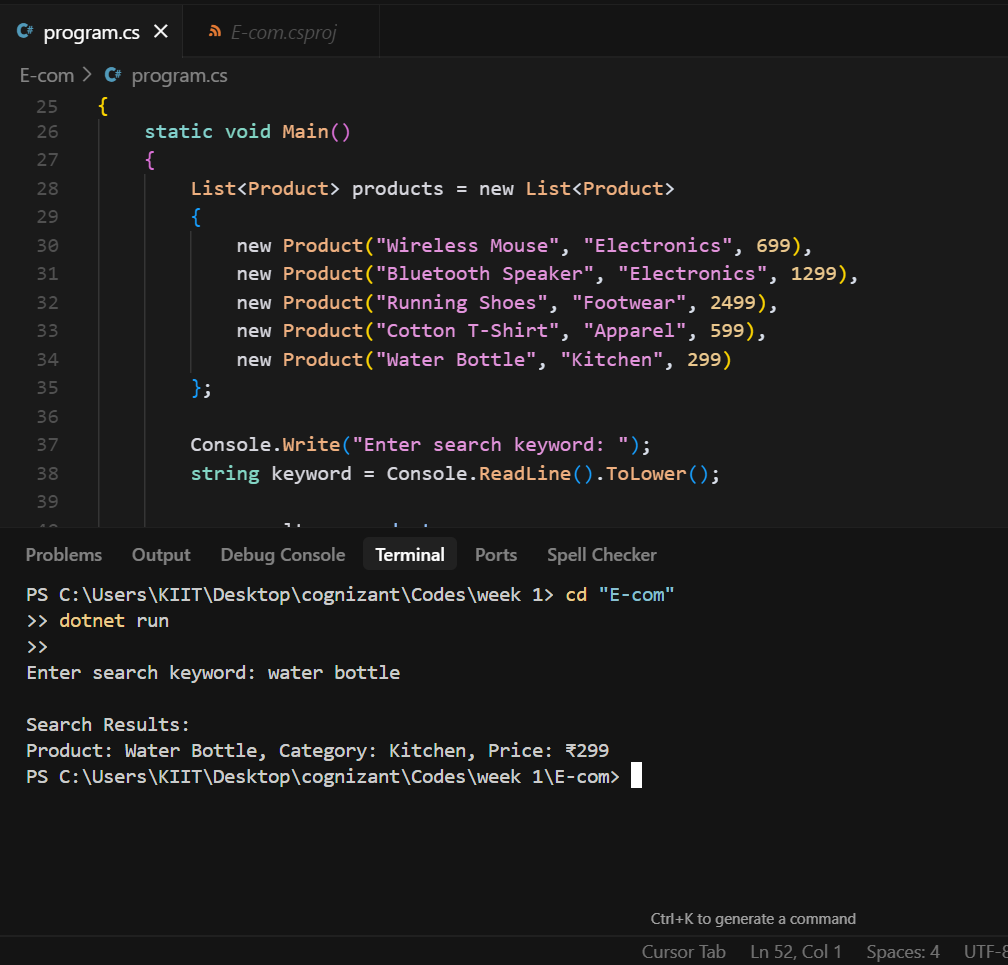
*Console.WriteLine("No products found.");*

*else*

*results.ForEach(p => Console.WriteLine(p));*

*}*

*}*



## Exercise 7: Financial Forecasting

This program demonstrates simple financial forecasting using a 3-month moving average on historical revenue data.

*using System;*

*using System.Collections.Generic;*

*using System.Linq;*

*class FinancialForecasting*

*{*

*static void Main()*

*{*

*List<double> revenues = new List<double> { 10000, 12000, 11000, 15000, 17000, 16000 };*

*int windowSize = 3;*

*if (revenues.Count < windowSize)*

*{*

*Console.WriteLine("Not enough data for moving average.");*

*return;*

*}*

*double sum = 0;*

*for (int i = revenues.Count - windowSize; i < revenues.Count; i++)*

*{*

*sum += revenues[i];*

*}*

*double forecast = sum / windowSize;*

*Console.WriteLine("Historical Revenue (last 6 months):");*

*revenues.ForEach(r => Console.WriteLine("₹" + r));*

*Console.WriteLine($"\nForecast for next month (using {windowSize}-month moving average): ₹{forecast:F2}");*

*}*

*}*

