Product.cs

public class Product

{

    public int ProductId { get; set; }

    public string ProductName { get; set; }

    public string Category { get; set; }

    public Product(int id, string name, string category)

    {

        ProductId = id;

        ProductName = name;

        Category = category;

    }

    public override string ToString()

    {

        return $"ID: {ProductId}, Name: {ProductName}, Category: {Category}";

    }

}

Program.cs

using System;

class Program

{

    static void Main(string[] args)

    {

        // Sample product data

        Product[] products = new Product[]

        {

            new Product(3, "Laptop", "Electronics"),

            new Product(1, "Notebook", "Stationery"),

            new Product(4, "Smartphone", "Electronics"),

            new Product(2, "Pen", "Stationery")

        };

        Console.WriteLine("🔎 Linear Search for ProductId = 2:");

        Product linearResult = LinearSearch(products, 2);

        Console.WriteLine(linearResult != null ? linearResult.ToString() : "Product not found.");

        // Sort by ProductId for binary search

        Array.Sort(products, (a, b) => a.ProductId.CompareTo(b.ProductId));

        Console.WriteLine("\n🔎 Binary Search for ProductId = 2:");

        Product binaryResult = BinarySearch(products, 2);

        Console.WriteLine(binaryResult != null ? binaryResult.ToString() : "Product not found.");

    }

    // Linear Search

    static Product LinearSearch(Product[] products, int targetId)

    {

        foreach (var product in products)

        {

            if (product.ProductId == targetId)

                return product;

        }

        return null;

    }

    // Binary Search (on sorted array)

    static Product BinarySearch(Product[] products, int targetId)

    {

        int low = 0, high = products.Length - 1;

        while (low <= high)

        {

            int mid = (low + high) / 2;

            if (products[mid].ProductId == targetId)

                return products[mid];

            else if (products[mid].ProductId < targetId)

                low = mid + 1;

            else

                high = mid - 1;

        }

        return null;

    }

}

Output

