

Exercise 1: A LINQ Query Application Filtering and Sorting a List of Products

Statement:

Develop a console application in C# that defines a Product class with properties such as Id, Name, Category, and Price. Create a list of products and use LINQ queries to filter and sort this list. First, filter the products by a specific category. Then, sort these filtered products by price in ascending order. Display the filtered and sorted list to the console.

Topics to Practice:

- Introduction to collections (List)
- Defining classes and objects
- LINQ queries on collections
- Filtering and sorting data

Terminal – FilteringSorting

LINQ Query Application Filtering and Sorting Products

Readin file from path: /Users/marcelogomesmarques/Projects/CSharp/FilteringSorting/FilteringSorting/Resources
Total records: 85
Average price = 863.57

Press any key to continue.

Terminal – FilteringSorting

Select the category by its number to report:

- 0 - None
- 1 - Monitor
- 2 - SamrtTv
- 3 - GameConsole
- 4 - GameController
- 5 - Handset
- 6 - HD
- 7 - GamingFurniture
- 8 - Games

9

Wrong numembar of category inputed! Press any key to try again!

Terminal – FilteringSorting

Select the category by its number to report:

- 0 - None
- 1 - Monitor
- 2 - SamrtTv
- 3 - GameConsole
- 4 - GameController
- 5 - Handset
- 6 - HD
- 7 - GamingFurniture
- 8 - Games

Wrong choice! Select a number from the list! Press any key to try again!

Products Report from Games

```
-----
Product: Doom 1 - Price: 15.99
Product: Doom 2 - Price: 15.99
Product: Gears of War 1 - Price: 19.99
Product: Gears of War 2 - Price: 29.99
Product: God Of War 1 - Price: 39.99
Product: Gears of War 3 - Price: 39.99
Product: God Of War 2 - Price: 49.99
Product: Gears of War 4 - Price: 49.99
Product: Canelone - Price: 53.99
Product: Gears of War 5 - Price: 59.99
Product: FiFa 10 - Price: 59.99
Product: FiFa 11 - Price: 69.99
Product: Doom 3 - Price: 69.99
Product: God Of War 3 - Price: 79.99
Product: FiFa 12 - Price: 79.99
Product: Forsa 1 - Price: 79.99
Product: FiFa 13 - Price: 89.99
Product: FiFa 14 - Price: 99.99
Product: Forsa 2 - Price: 99.99
Product: Hirizon 1 - Price: 99.99
Product: Forsa 3 - Price: 119.99
Product: God Of War 4 - Price: 159.99
Product: Gears of War 6 - Price: 159.99
Product: FiFa 15 - Price: 159.99
Product: Forsa 4 - Price: 159.99
Product: Doom Eternal - Price: 159.99
Product: Quacke - Price: 159.99
Product: Forsa 5 - Price: 199.99
Product: Hirizon 2 - Price: 199.99
Product: God Of War 5 - Price: 259.99
Product: Gears of War 6 Series S - Price: 259.99
Product: FiFa 16 - Price: 259.99
Product: Forsa 6 - Price: 259.99
Product: God Of War 5 Playstation - Price: 359.99
Product: FiFa 17 - Price: 359.99
Product: Forsa 6 series X - Price: 359.99
Product: Hirizon 2 Playstation 5 - Price: 399.99
```

Its average price = 140.69

Press 'n' to close the app, or press any other key to continue: █

Captura de Tela

Products Report from GameConsole

```
-----
Product: X-Box 360 - Price: 499.99
Product: Playstation 2 - Price: 599.99
Product: X-Box One - Price: 899.99
Product: X-Box One S - Price: 999.99
Product: Playstation 3 - Price: 1000.99
Product: X-Box Series S - Price: 1999.99
Product: Playstation 4 - Price: 1999.99
Product: X-Box Series S Pack Fifa - Price: 2099.99
Product: X-Box Series S Pack Doom and Forsa - Price: 2199.99
Product: X-Box Series S Pack Doom/Forsa and 3 months of Game Pass - Price: 2250.99
Product: Playstation 4 Pack God Of Wars - Price: 2259.99
Product: X-Box Series X - Price: 3999.99
Product: Playstation 4 Pro - Price: 3999.99
Product: X-Box Series X Pack Fifa - Price: 4099.99
Product: X-Box Series X Pack Doom and Forsa - Price: 4199.99
Product: Playstation 5 - Price: 4199.99
Product: X-Box Series SXPack Doom/Forsa and 3 months of Game Pass - Price: 4250.99
Product: Playstation 5 Pack God Of Wars - Price: 5199.99
```

Its average price = 2597.93

Press 'n' to close the app, or press any other key to continue: █

Exercise 2: Grouping and Aggregating Data from a Collection of Objects

Statement:

Create a console application that works with a collection of objects representing sales transactions. Each transaction should include properties such as TransactionId, ProductId, Quantity, and Price. Use LINQ to group these transactions by ProductId and then aggregate the data to calculate the total sales (i.e., the sum of Quantity * Price) for each product. Output the results to the console, displaying the ProductId and the corresponding total sales.

Topics to Practice:

- Working with collections (List)
- LINQ queries on collections
- Grouping and aggregating data
- Working with anonymous types for aggregation results

LINQ Transactions Grouping and Aggregations:

Readin file from path: /Users/marcelogomesmarques/Projects/CSharp/GroupingAggregating/GroupingAggregating/Resources
Total records: 64

Press any key to continue.

Terminal – GroupingAggregating

Sales Report:

```
-----  
TransactionId: A0003-12 - Total Sales: 9603.64  
TransactionId: A0008-12 - Total Sales: 9130.94  
TransactionId: A0004-12 - Total Sales: 4648.75  
TransactionId: A0005-12 - Total Sales: 3996.88  
TransactionId: A0007-12 - Total Sales: 2850.98  
TransactionId: A0001-12 - Total Sales: 2610.95  
TransactionId: A0006-12 - Total Sales: 2349.91  
TransactionId: A0002-12 - Total Sales: 1840.92
```

Average sales value = 37032.97

Average sales value by invoice item = 578.64

Press any key to close the app.