

Programação Visual

Trabalho de Laboratório nº 2

Objetivo	Programação avançada em C#: linguagem de interrogação LINQ.
-----------------	---

Programa	Analisar uma lista de produtos.
-----------------	---------------------------------

Regras	<p>Criar uma aplicação de consola.</p> <p>Implementar o código necessário e testar no fim de cada nível.</p> <p>Use as convenções de codificação adotadas para a linguagem C#.</p> <p>Na classe do programa não coloque atributos nem crie métodos para além do Main.</p> <p>Não é necessário obter dados do utilizador. Forneça os dados ao nível do código.</p>
---------------	--

Descrição

Nível 1

- Abra a solução **AnaliseProdutos.sln** que acompanha este enunciado. Na solução que abriu, dentro do método **Main** é obtida a lista de produtos que se pretende analisar. Os produtos são objetos da classe **Product** onde estão definidas propriedades implícitas para: a identificação do produto (**ProductId**), o nome do produto (**ProductName**), a categoria do produto (**Category**) o preço unitário (**UnitPrice**) e o número de unidades em Stock (**UnitsInStock**). No método **Main** é feita uma listagem de todos os produtos.
- Usando uma *query* LINQ liste todas as bebidas por ordem alfabética do seu nome.
- Liste agora todos os produtos cujo nome começa pela letra 'L'.
- Liste todas as bebidas cujo nome começa pela letra 'L'.

Nível 2

- Crie uma *query* LINQ para obter a lista de bebidas que custam entre 10 e 25.
- Dentro do método **Main**, acrescente à lista de produtos as seguintes bebidas:
 - 78, Heineken, Beverages, 15.4, 20
 - 79, Super Bock, Beverages, 10.5, 10
- Faça novamente a listagem reutilizando a *query* anterior (não volte a criá-la).
- Usando LINQ, obtenha o número de produto (**ProductId**) mais baixo das bebidas que existem na lista de produtos e escreva o nome da bebida e o seu número no ecrã.

Nível 3

- Pretende-se saber quanto dinheiro se tem em bebidas no stock. Neste sentido obtenha uma listagem com todas as bebidas mostrando em cada linha o nome da bebida, a quantidade em stock e o valor total deste stock (obtido multiplicando o número de unidades em stock pelo custo unitário). Neste caso utilize LINQ com projeção de dados, os objetos criados deverão ter as propriedades **Nome**, **Quantidade** e **ValorTotal**. No final da listagem apresente o total calculado a partir da soma dos diversos valores totais. Para a soma utilize o operador **Sum()** do LINQ.
- Usando novamente um dos operadores de agregação do LINQ, mostre a seguir a bebida com maior custo, com menor custo e o valor médio do preço das bebidas.

Programação Visual

Trabalho de Laboratório nº 2

Nível 4

- Usando LINQ com a sintaxe dos métodos de extensão e expressões lambda (quando for necessário) faça as seguintes apresentações:
 - A lista de produtos ordenada pelo nome (operador **OrderBy**)
 - A lista de categorias únicas (sem repetição) (operador **Distinct**)
 - Verificar se existem produtos com um custo menor que 5 (operador **Any**).
 - O número médio de unidades em stock.

Nível 5

- Usando LINQ com a sintaxe dos métodos de extensão:
 - Os primeiros 5 produtos ordenados pelo preço que custam menos de 100 (operador **Take**)
 - Os 5 primeiros produtos que aparecem a seguir aos do ponto anterior (operadores **Take** e **Skip**).
 - A lista de produtos ordenada pela categoria e depois pelo nome do produto (operadores **OrderBy** e **ThenBy**).

Desafio

- Mostre cada categoria seguida da lista de produtos incluídos na mesma omitindo o nome da categoria na listagem (utilize o operador **GroupBy**).

Notas

Para os identificadores siga as convenções adotadas pelo C#, nomeadamente:

- A notação **camelCase** para o nome das variáveis locais e identificadores privados.
- A notação **PascalCase** para os nomes públicos dos métodos e classes
- Não utilize o símbolo '_' nos identificadores
- Não use abreviaturas