

Exercício 1 – Cadastro de Produtos

O que era para fazer:

Neste exercício, deveríamos criar um programa em C# que permitisse ao usuário cadastrar 3 produtos. Cada produto deveria ter nome, preço e quantidade. Depois do cadastro, o programa deveria mostrar esses dados e calcular o valor total em estoque (preço × quantidade).

Como resolvemos:

1. Criamos uma classe chamada Produto com as propriedades Nome, Preço e Quantidade.
2. Dentro da classe, implementamos um método chamado ValorTotalEmEstoque() que retorna o resultado da multiplicação entre o preço e a quantidade.
3. No método Main(), declaramos um array de três posições para armazenar os objetos Produto.
4. Usamos um laço for para solicitar os dados dos produtos ao usuário.
5. Após o cadastro, utilizamos um laço foreach para exibir as informações de cada produto, incluindo o valor total em estoque.

Comandos e estruturas que utilizamos:

- class: para definir a estrutura do produto.
- array: para armazenar os produtos.
- Console.ReadLine(): para capturar a entrada do usuário.
- Convert.ToDouble() e Convert.ToInt32(): para conversão de string para número.
- for e foreach: para repetição durante o cadastro e exibição.
- Um método auxiliar: ValorTotalEmEstoque().

Por que escolhemos esse método:

Criamos uma classe para organizar melhor os dados e separar a lógica da apresentação. Como a quantidade de produtos era fixa, um array foi suficiente. Além disso, colocamos o cálculo do valor total dentro de um método para deixar o código mais limpo e fácil de manter.

Exercício 2 – Soma dos Números Pares

O que era para fazer:

Neste segundo exercício, o programa deveria pedir que o usuário digitasse 10 números inteiros. Em seguida, o programa deveria somar apenas os números pares e mostrar o resultado final.

Como resolvemos:

1. Declaramos um array de inteiros com 10 posições.
2. Utilizamos um laço for para capturar os 10 números digitados pelo usuário e armazená-los no array.
3. Em seguida, utilizamos um foreach para percorrer todos os elementos do array.
4. Para cada número, verificamos se era par usando a condição `numero % 2 == 0`.
5. Se o número fosse par, somávamos seu valor em uma variável chamada `somaPares`.
6. Por fim, exibimos o resultado da soma.

Comandos e estruturas que utilizamos:

- `int[]`: para armazenar os números digitados.
- `for`: para entrada dos dados.
- `foreach`: para percorrer os valores armazenados.
- `if`: para verificar se o número era par.
- `%`: operador de módulo, usado para verificar se o número é divisível por 2.

Por que escolhemos esse método:

Essa foi a forma mais direta e simples de resolver o problema. Como sabíamos que seriam exatamente 10 números, o uso de um array fixo foi suficiente. A verificação de números pares com `% 2 == 0` é um método comum e eficiente.