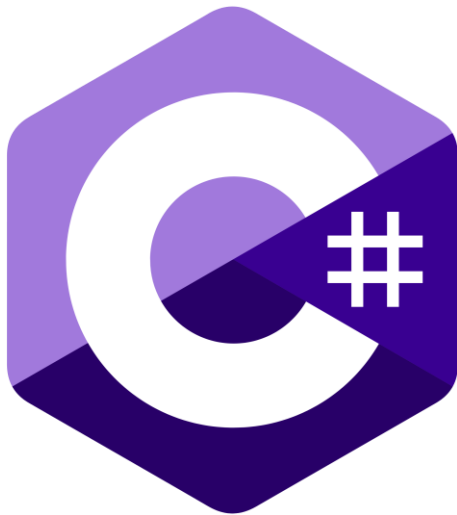


# C# Essencial



Coleções : Exercícios

# Exercícios

1- Crie um programa C# para armazenar os nomes de 10 frutas em um *array unidimensional* e a seguir realize as seguintes operações : ***Dados: Maça, Banana, Laranja, Uva, Manga, Pêra, Abacate, Mamão, Pêssego, Amora***

- a- Exiba os nomes das frutas no console e a quantidade de elementos do array usando os laços for e foreach
- b- Exiba os nomes da *segunda* e da *penúltima* fruta no console
- c- Altere o nome da *terceira* fruta para *Kiwi* e da *última* fruta para *Caqui* e exiba o nome de todas as frutas
- d- Ordene a coleção de frutas na ordem ascendente
- e- Exiba no console a coleção de nomes de frutas na ordem inversa

2- Crie um programa em C# que sirva para pesquisar *um array de valores inteiros*. Para fazer isso, siga estas etapas a seguir

- a-) Solicite via teclado o número de valores do array (*deve ser um número inteiro*)
- b-) Declare e inicialize um array de inteiros com a quantidade de números inteiros informados
- c-) Solicite ao usuário um valor de um número inteiro a procurar no array
- d-) Exiba se o número informado existe no array de inteiros ou não
- e-) Repita até que o texto 'fim' seja inserido via teclado

# Exercícios

3-Escreva um programa em C# que solicite ao usuário que informe o valor das notas, *do tipo float* , para 10 alunos divididos em 2 grupos de 5 alunos cada. Armazene as informações em um *array bidimensional* e a seguir exiba no console a média aritmética de cada grupo de alunos (grupo1 e grupo2).

4- Crie um programa C# que usa uma *ArrayList* para armazenar uma lista de objetos pessoas.

a-) Primeiro crie uma classe chamada **Pessoa.cs** com duas propriedades (**nome e idade**) e um método **Exibir()** para imprimir o nome e a idade;

b-) A seguir crie uma *ArrayList* e solicite a informação dos nomes e idades de 3 *pessoas* via teclado armazenando as informações na *ArrayList*.

c-) A seguir exiba no console a lista das pessoas com nome e idade

d-) Inclua mais *dois objetos* **Pessoa** na lista e exiba a nova lista de pessoas

e-) Remova o último elemento da coleção e exiba a lista completa

*Use seguintes dados iniciais:* **Ana, 22 - Diná, 21 - Maria, 19**

*Ao incluir os objetos pessoa use os dados :* **Jaime,20 - Tânia,18**

# Exercícios

5- Dada uma classe **Produto** contendo duas propriedades : **Nome (string)** e **Preco (decimal)**. Crie uma lista de objetos do tipo Produto usando os seguinte dados:

*Clips, R\$ 3,95 - Caneta, R\$ 5,99 - - Lápis, R\$ 4,15 - Estojo, R\$ 6,99 e Caderno, R\$ 7,55*

A seguir realize as seguintes operações na lista de objetos **Produto** criada :

1- Crie um método para exibir a relação de produtos , a soma total dos preços dos produtos, a média do preço dos produtos e a quantidade de produtos na lista

2- Inclua na lista o seguinte produto : *Mochila, R\$ 22,44* e exiba novamente a lista de produtos

3- Ordene a lista pelo *nome do produto* e exiba a lista ordenada

4- Obtenha e exiba no console os produtos com preço inferior a R\$ 5,00

5- Localize na lista o produto com nome **Estojo**

6- Crie um programa C# e defina um método chamado **ProcessaObjetos()** que possa receber um número *variáveis de argumentos* e que possa exibir os seus valores e também o tipo do objeto (*Use o método GetType() herdado de Object*) no console.

Passe os seguintes argumentos ao invocar o método : **ProcessaObjetos(1, "Maria", 3.45m, new Teste(), null)**

## Exercícios

7- Crie um **indexador** para uma classe **Aluno** para gerenciar uma *coleção de nomes de alunos* com no máximo *10* elementos.

Para testar a implementação atribua nomes e obtenha os valores atribuídos usando o indexador criado.