

XXX
NÃO USE COPY/PASTE
Dessa forma perde tempo e não aprenderá nada!
Só se aprende a programar programando, escrevendo código.
Digito código tal como lhe é pedido.
Use os snippets que já conhece
Construtores: ctor tab tab
Propriedades: prop tab tab
etc.
XXX

** Coleções 1 - Listas - List<T>

1A - Crie um projeto S051_TutorialColecoes_NA e guarde-o na pasta Semana 05 da pasta ProgVis2021

1B - Escreva no Main => Console.WriteLine("XXXXXXXXXXXXXXXXXXXXX Listas - List<T>")

1C - No Main crie uma lista de inteiros :>
List<int> listaDeInteiros = new List<int>();

1D - Prima o rato sobre o sinal vermelho à esquerda do código e escolha :>
insert using System.Collections.Generic;
para inserir o namespace que contém as classes de coleção genéricas

1E - Adicione os primeiros 20 inteiros (0 a 19) à lista usando um ciclo for :>
for (int i = 0; i < 20; i++)
listaDeInteiros.Add(i);

1F - Imprima o conteúdo da lista usando um ciclo foreach :>
foreach(int i in listaDeInteiros)
Console.WriteLine(i);

1G - O resultado é o que esperava?

1H - Some o conteúdo da lista usando um ciclo foreach :>
int soma = 0;
foreach (int i in listaDeInteiros)
soma += i;
Console.WriteLine(soma);

1I - O resultado é o que esperava?

1J - Adicione mais uma vez os primeiros 20 inteiros e veja o conteúdo da lista na consola escrevendo novamente o código d alínea 1F

NOTE que todos os algoritmos são apresentados pela ordem em que os inseriu e todos estão repetidos
(listas são coleções ordenadas e que aceitam elementos repetidos)

**** REVISÕES**

2 - Escreva no Main :>

```
Console.WriteLine("XXXXXXXXXXXXXXXXXXXXX CLASSES");
```

3A - Crie uma classe CartaoDoCidadao_NA com propriedades para o numero (String) e a data de validade do tipo DateTime.

3B - Crie um construtor com todos os parametros e um ToString que produza uma String no formato :>

123456789 - 2022/1

(NOTA: Use DataDeNascimento.Year e DataDeNascimento.Month para formatar a String retornada pelo ToString()).

4A - Teste criando dois objetos e imprimindo o seu estado na consola.

(NOTA: use new Date

4B - O resultado é o que esperava?

5 Escreva no Main:>

```
Console.WriteLine("XXXXXXXXXXXXXXXXXXXXX COMPOSIÇÃO");
```

6A - Crie uma classe Pessoa_NA com uma propriedade para o nome (String), outra para a data de nascimento (DateTime) e outra para um cartão do cidadão (CartaoDoCidadao_NA).

6B - Crie um construtor com todos os parametros e um ToString que produza uma String no formato :>

Ana Monteiro - 2002/3/4 - 222222222 - 2022/2.

(NOTA: Use DataDeNascimento.Year e DataDeNascimento.Month para formatar a String retornada pelo ToString()).

7 - Teste criando dois objetos e imprimindo o seu estado na consola.

NOTA: use o contrutor de DateTimee que aceita um ano, mês e dia :> new DateTime(2021, 1, 11)

O resultado é o que esperava?

8 Escreva no Main :>

```
Console.WriteLine("XXXXXXXXXXXXXXXXXXXXX HERANÇA");
```

9A - Crie uma classe Aluno_NA que herda de Pessoa_NA e tem uma propriedade para o numero de aluno (String).

9B - Crie o construtor e o ToString() que retorne uma string no formato :>

Lena Pereira - 2002/3/4 - 222222222 - 2022/2 CODAL: 20200622

9C - Teste! O resultado é o que esperava?

**** Coleções 2 - Listas - List<T>**

10 - Escreva no Main :>

```
Console.WriteLine("XXXXXXXXXXXXXXXXXXXXX COLEÇÕES => List<T>");
```

11 - No Main crie uma lista de Pessoa_NA =>
List<Pessoa_NA> pessoas = new List<Pessoa_NA>();

12 - Prima o rato sobre o sinal vermelho à esquerda do código e escolha :>
insert using System.Collections.Generic;
para inserir o namespace que contém as classes de coleção genéricas

13 - Adicione-lhe as 4 pessoas que criou usando pessoas.Add(<objeto>);

14A - Imprima o conteúdo da lista usando um foreach :>
foreach (Pessoa_NA p in pessoas)
Console.WriteLine(p);

14B - O resultado é o que esperava?

```
*****  
** Coleções 2 - Herança de Listas - : List<Pessoa_NA>  
*****
```

15 - Crie uma classe Pessoas_NA que herde de List < Pessoa_NA > :>
public class Pessoas_NA : List<Pessoa_NA>

16 - Crie um ToString que retorne uma String com todos os elementos da lista com o código :>

```
public override string ToString()  
{  
    String str = "";  
    foreach (Pessoas_NA p in this)  
        str += p + "\n";  
    return str;  
}
```

17 - Escreva no Main :>
Console.WriteLine("XXXXXXXXXXXXXXXXXXXXX COLEÇÕES QUE HERDAM DE List < T > ");

18A - Crie um objeto da classe Pessoas_NA, adicione-lhe as 4 pessoas que criou até ao momento e imprima na consola o conteúdo do objeto.

18B - O resultado é o que esperava?

```
*****  
** Consolidação de listas 1  
*****
```

19 - Escreva no Main :>
Console.WriteLine("XXXXXXXXXXXXXXXXXXXXX Consolidação de listas 1 =>
List<FiguraGeometrica_NA>");

20 - Crie uma classe Ponto_NA com propriedades para as coordenadas em x e em y, um construtor e um ToString que retorne uma string no formato => (10,20)

21 - Crie 4 pontos no main, dois deles com os valores dos outros dois e verifique o resultado na consola.

22 - Crie uma classe FiguraGeométrica_NA com uma propriedade do tipo

Ponto_NA, um construtor e um ToString.

23 - Crie 4 figuras geométricas, duas delas com os valores das outras duas e verifique o resultado na consola.

24 - Crie uma classe Retangulo_NA que herda de FiguraGeométrica_NA e tem uma propriedade para o comprimento do lado (int), um construtor e um ToString.

25 - Crie 4 retangulos no Main, dois deles com os valores dos outros dois e verifique o resultado na consola.

** Consolidação de herança de listas 1

26 - Repita os passo 15 a 18 para as classes dos pontos 20, 22 e 24 criando as respetivas listas: Pontos_NA, FigurasGeometricas_NA e Retangulos_NA

** Consolidação de listas 2

27A - Repita os passos 19 a 26 para uma classe Livraria_NA que é uma List<Livro_NA> sabendo que:

27B - Livro_NA é uma classe com 3 atributos: Titulo_NA (String), DataPrimeiraEdicao (DateTime), Autor_NA.

27B - Autor_NA é uma classe com 2 atributos: Nome_NA e LinguaOriginal_NA