

Usando expression variables

O C# 7 já permitia o uso de expressões contendo expression variables com o uso da palavra reservada `out`, mas ainda existiam algumas melhorias a serem feitas, que foram incluídas no C# 7.3.

No código de Lucas, logo abaixo, temos os 3 novos locais onde podemos fazer uso de expression variables e outro local onde isso não é possível causando um erro de compilação.

```
01 | public class TesteExpressionVariables : ClasseBase
02 | {
03 |     private int numeroDeString = int.TryParse("123", out int valor) ? valor : 1;
04 |
05 |     public TesteExpressionVariables() : base(out bool valor)
06 |     {
07 |         Console.WriteLine($"Valor de saída do construtor da classe base: {valor}")
08 |     }
09 |
10 |     public IEnumerable<(char, int)> ObterDigitosEmString(string texto)
11 |     {
12 |         return from caractere in texto
13 |             where int.TryParse(caractere.ToString(), out var numero) && numero
14 |             select (caractere, numero);
15 |     }
16 | }
17 |
18 | public class ClasseBase
19 | {
20 |     public ClasseBase(out bool valor) => valor = false;
21 | }
```

Quais são os 3 novos locais onde podemos fazer uso de expression variables?

Selecione 3 alternativas

- A** Inicializadores de campos, como na linha 3.
- B** Cláusulas em queries, como na linha 13.
- C** Inicializadores de construtores, como na linha 5.
- D** Construtores de classes, como na linha 20.
- E** Seletores em queries, como na linha 14.

