

第15期-弃元运算符

2022年5月19日 13:37

弃元介绍

从 C# 7.0 开始, C# 支持弃元, 是一种占位符变量, 弃元使代码意图更加明确。放弃表示代码永远不会使用变量。它们可以增强其可读性和可维护性。

元组和对象析构

```
static (string, double, int, int, int, int) Test()
{
    return ("", 0, 0, 0, 0, 0);
}

var (_, _, _, pop1, _, pop2) = Test();
```

模式匹配中的弃元运算符

```
public static decimal GetGroupTicketPrice(int visitorCount) => visitorCount
switch
{
    1 => 12.0m,
    2 => 20.0m,
    3 => 27.0m,
    4 => 32.0m,
    0 => 0.0m,
    _ => throw new ArgumentException(nameof(visitorCount)),
};
```

方法 OUT 参数占位符弃元

```
string str = "2018-05-01 14:57:32.8";

if (DateTime.TryParse(str, out _))
{
    Console.WriteLine($"{str}: valid");
}
```

独立弃元占位符

```
public static void Method(string arg)
{
    _ = arg ?? throw new ArgumentNullException(nameof(arg));
}

_ = Task.Run(() => Console.WriteLine(DateTime.Now));
```

下划线也是有效标识符

```
private static void ShowValue(int _)
{
    byte[] arr = { 0, 0, 1, 2 };
    _ = BitConverter.ToInt32(arr, 0);
    Console.WriteLine(_);
}
```