

## C#中的Finally: 保证执行还是存在意外? 解析与实例

作者: 微信公众号: 【架构师老卢】

12-20 18:33

~ 129



**概述:** 在C#中, 通常情况下`finally`块中的代码一定会执行。然而, 特殊情况如使用`Environment.Exit`或无限循环可能导致`finally`代码无法执行。开发者应谨慎处理可能中断程序正常流程的情况, 以确保程序的可靠性。

在C#中, `finally` 中的代码通常会执行, 但有一些特殊情况可能导致 `finally` 不被执行, 比如在执行 `Environment.Exit` 或者无限循环等情况下。

以下是一个详细实例, 演示 `finally` 中的代码是否会执行的情况:

```
1  using System;
2
3  class Program
4  {
5      static void Main()
6      {
7          try
8          {
9              // 模拟正常情况下的代码
10             Console.WriteLine("Try 模块执行");
11             DoSomething();
12         }
13         catch (Exception ex)
14         {
15             Console.WriteLine($"Catch 模块执行: {ex.Message}");
16         }
17         finally
18         {
19             Console.WriteLine("Finally 模块执行");
20         }
21
22         Console.WriteLine("程序结束");
23     }
24
25     static void DoSomething()
26     {
27         try
28         {
29             // 模拟异常情况
30             Console.WriteLine("DoSomething 执行");
31             throw new Exception("发生异常");
32         }
33         finally
34         {
35             Console.WriteLine("DoSomething Finally 模块执行");
36         }
37     }
38 }
```

在上述代码中，`DoSomething` 方法中发生了异常，但无论是否发生异常，`DoSomething Finally` 模块执行 都会被打印。然而，如果在 `try` 块或 `catch` 块中执行 `Environment.Exit` 或者进入无限循环，`finally` 块中的代码可能无法执行，因为程序被强制终止。

尽量避免在 `finally` 中放置可能导致程序无法正常退出的代码，以确保其可靠性。