

# 1 **System.Threading.ThreadStateException** 2 **Class**

3  
4

```
5 [ILASM]  
6 .class public serializable ThreadStateException extends  
7 System.SystemException  
  
8 [C#]  
9 public class ThreadStateException: SystemException
```

## 10 **Assembly Info:**

- 11 • *Name:* mscorlib
- 12 • *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- 13 • *Version:* 1.0.x.x
- 14 • *Attributes:*
  - 15 ○ CLSCompliantAttribute(true)

## 16 **Summary**

17

18 Represents errors that occur when a method is invoked on a  
19 **System.Threading.Thread** and the thread is in a  
20 **System.Threading.Thread.ThreadState** that is invalid for the  
21 method.

## 22 **Inherits From: System.SystemException**

23

24 **Library:** BCL

25

26 **Thread Safety:** All public static members of this type are safe for multithreaded  
27 operations. No instance members are guaranteed to be thread safe.

28

## 29 **Description**

30 Once a thread is created, it is in one or more states, as defined by  
31 **System.Threading.ThreadState**, until it terminates.  
32 **System.Threading.ThreadStateException** is thrown by methods  
33 that cannot perform the requested operation due to the current state  
34 of a thread. For example, calling **System.Threading.Thread.Start** on  
35 a thread that has terminated results in a  
36 **System.Threading.ThreadStateException** exception.

## 37 **Example**

38

39 The following example demonstrates an error that causes a  
40 **System.Threading.ThreadStateException** exception to be thrown.

```
1
2      [C#]

3      using System;
4      using System.Threading;
5      public class ThreadWork {
6          public static void DoWork() {
7              Console.WriteLine("Working thread...");
8          }
9      }

10
11     class ThreadStateTest{
12         public static void Main() {
13             ThreadStart myThreadDelegate = new
14 ThreadStart(ThreadWork.DoWork);
15             Thread myThread = new Thread(myThreadDelegate);
16             myThread.Start();
17             Thread.Sleep(0);
18             Console.WriteLine("In main. Attempting to restart
19 myThread.");
20             try {
21                 myThread.Start();
22             }
23             catch (ThreadStateException e) {
24                 Console.WriteLine("Caught: {0}", e.Message);
25             }
26
27         }
28     }
29

30     The output is
31
32     Working thread...
33
34
35     In main. Attempting to restart myThread.
36
37
38     Caught: Thread is running or terminated. Cannot restart.
39
```

40

# 1 ThreadStateException() Constructor

```
2 [ILASM]  
3 public rtspecialname specialname instance void .ctor()  
4 [C#]  
5 public ThreadStateException()
```

## 6 Summary

7 Constructs and initializes a new instance of the  
8 **System.Threading.ThreadStateException** class.

## 9 Description

10 This constructor initializes the  
11 **System.Threading.ThreadStateException.Message** property of  
12 the new instance to a system-supplied message that describes the  
13 error, such as "The requested operation cannot be performed on the  
14 thread due to its current state." This message takes into account the  
15 current system culture.

16  
17 The **System.Threading.ThreadStateException.InnerException**  
18 property of the new instance is initialized to **null**.

19

# 1 ThreadStateException(System.String)

## 2 Constructor

```
3 [ILASM]  
4 public rtspecialname specialname instance void .ctor(string  
5 message)  
  
6 [C#]  
7 public ThreadStateException(string message)
```

### 8 Summary

9 Constructs and initializes a new instance of the  
10 **System.Threading.ThreadStateException** class.

### 11 Parameters

12  
13

Parameter	Description
<i>message</i>	A <b>System.String</b> that describes the error. The content of <i>message</i> is intended to be understood by humans. The caller of this constructor is required to ensure that this string has been localized for the current system culture.

14

### 15 Description

16 This constructor initializes the  
17 **System.Threading.ThreadStateException.Message** property of  
18 the new instance using *message*. If *message* is **null**, the  
19 **System.Threading.ThreadStateException.Message** property is  
20 initialized to the system-supplied message provided by the constructor  
21 that takes no arguments.  
22

23 The **System.Threading.ThreadStateException.InnerException**  
24 property of the new instance is initialized to **null**.

25

# 1 ThreadStateException(System.String, 2 System.Exception) Constructor

```
3 [ILASM]  
4 public rtspecialname specialname instance void .ctor(string  
5 message, class System.Exception innerException)  
  
6 [C#]  
7 public ThreadStateException(string message, Exception  
8 innerException)
```

## 9 Summary

10 Constructs and initializes a new instance of the  
11 **System.Threading.ThreadStateException** class.

## 12 Parameters

13  
14

Parameter	Description
<i>message</i>	A <b>System.String</b> that describes the error. The content of <i>message</i> is intended to be understood by humans. The caller of this constructor is required to ensure that this string has been localized for the current system culture.
<i>innerException</i>	An instance of <b>System.Exception</b> that is the cause of the current exception. If <i>innerException</i> is non-null, then the current Exception was raised in a catch block handling <i>innerException</i> .

15  
16

## Description

17 This constructor initializes the  
18 **System.Threading.ThreadStateException.Message** property of  
19 the new instance using *message* and the  
20 **System.Threading.ThreadStateException.InnerException**  
21 property using *innerException*. If *message* is **null**, the  
22 **System.Threading.ThreadStateException.Message** property is  
23 initialized to the system-supplied message provided by the constructor  
24 that takes no arguments.

25  
26  
27

[Note: For more information on inner exceptions, see  
**System.Exception.InnerException**.]

28