

# System.Threading.ThreadStateException Class

```
[ILASM]
.class public serializable ThreadStateException extends
System.SystemException

[C#]
public class ThreadStateException: SystemException
```

## Assembly Info:

- *Name:* mscorlib
- *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- *Version:* 1.0.x.x
- *Attributes:*
  - CLSCompliantAttribute(true)

## Summary

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

## Inherits From: System.SystemException

## Library: BCL

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

## Description

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

## Example

The following example demonstrates an error that causes a **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

```

# 1 ThreadStateException() Constructor

```
2 [ILASM]  
3 public rtspecialname specialname instance void .ctor()  
4  
5 [C#]  
6 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

# ThreadStateException(System.String)

## Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(string
message)

[C#]
public ThreadStateException(string message)
```

### Summary

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

### Parameters

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.

### Description

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

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

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

```
[ILASM]
public rtspecialname specialname instance void .ctor(string
message, class System.Exception innerException)

[C#]
public ThreadStateException(string message, Exception
innerException)
```

## Summary

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

## Parameters

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> .

## Description

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

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