1 System.Threading.ThreadState Enum

```
[ILAsm]
.class public sealed serializable ThreadState extends System.Enum

[C#]
public enum ThreadState
```

6 **Assembly Info:**

7 - *Name:* mscorlib
8 - *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00]
9 - *Version:* 2.0.x.x
10 - *Attributes:*
11   - CLSCompliantAttribute(true)

12 **Type Attributes:**

13 - FlagsAttribute

14 **Summary**

15 Specifies the execution states of a `System.Threading.Thread`.

16 **Inherits From: System.Enum**
17
18 **Library:** BCL
19
20 **Description**

21 `System.Threading.ThreadState` defines the set of possible execution states for
22 threads. Once a thread is created, it is in one or more of these states until it terminates.
23 Not all combinations of `ThreadState` values are valid; for example, a thread cannot be
24 in both the `System.Threading.ThreadState.Stopped` and
25 `System.Threading.ThreadState.Unstarted` states.
26
27 The following table shows the actions that cause a thread to change state.

Action	ThreadState after Action
The thread is created	Unstarted
`System.Threading.Thread.Start` is invoked on the thread	Running
The thread calls `System.Threading.Thread.Sleep`	WaitSleepJoin

The thread calls `System.Threading.Monitor.Wait` to wait on an object	WaitSleepJoin
The thread calls `System.Threading.Thread.Join` to wait for another thread to terminate	WaitSleepJoin
The `System.Threading.ThreadStart` delegate methods finish executing	Stopped
Another thread requests the thread to `System.Threading.Thread.Abort`	AbortRequested
The thread accepts a `System.Threading.Thread.Abort` request	Aborted

1
2 In addition to the states noted above, there is also the
3 `System.Threading.ThreadState.Background` state, which indicates whether the thread
4 is running in the background or foreground.
5
6 The current state of a thread can be retrieved from the
7 `System.Threading.Thread.ThreadState` property, whose value is a combination of the
8 `System.Threading.ThreadState` values. Once a thread has reached the
9 `System.Threading.ThreadState.Stopped` state, it cannot change to any other state.

10

ThreadState.Aborted Field

```
[ILAsm]
.field public static literal valuetype System.Threading.ThreadState
Aborted = 0x100


[C#]
Aborted = 0x100
```

Summary

The thread represented by an instance of `System.Threading.Thread` has terminated as a result of a call to `System.Threading.Thread.Abort.` A thread in this state is also in the `System.Threading.ThreadState.Stopped` state.

ThreadState.AbortRequested Field

```
[ILAsm]
.field public static literal valuetype System.Threading.ThreadState
AbortRequested = 0x80


[C#]
AbortRequested = 0x80
```

Summary

The `System.Threading.Thread.Abort` method has been invoked on the thread, but the thread has not yet received the pending `System.Threading.ThreadAbortException` that will attempt to terminate it.

ThreadState.Background Field

```
[ILAsm]
.field public static literal valuetype System.Threading.ThreadState
Background = 0x4


[C#]
Background = 0x4
```

Summary

The thread represented by an instance of `System.Threading.Thread` is being executed as a background thread, as opposed to a foreground thread. [*Note:* This state is controlled by setting the `System.Threading.Thread.IsBackground` property.]

ThreadState.Running Field

```
[ILAsm]
.field public static literal valuetype System.Threading.ThreadState
Running = 0x0


[C#]
Running = 0x0
```

Summary

The thread represented by an instance of `System.Threading.Thread` has been started
and has not terminated.

To determine if a thread is running, check that its state does not include
`System.Threading.ThreadState.Unstarted` and does not include
`System.Threading.ThreadState.Stopped`.

ThreadState.Stopped Field

```
[ILAsm]
.field public static literal valuetype System.Threading.ThreadState
Stopped = 0x10


[C#]
Stopped = 0x10
```

Summary

The thread represented by an instance of `System.Threading.Thread` has terminated.

ThreadState.Unstarted Field

```
[ILAsm]
.field public static literal valuetype System.Threading.ThreadState
Unstarted = 0x8


[C#]
Unstarted = 0x8
```

Summary

The `System.Threading.Thread.Start` method has not been invoked on the thread.

ThreadState.WaitSleepJoin Field

```
[ILAsm]
.field public static literal valuetype System.Threading.ThreadState
WaitSleepJoin = 0x20


[C#]
WaitSleepJoin = 0x20
```

Summary

The thread represented by an instance of `System.Threading.Thread` is blocked as a
result of a call to `System.Threading.Monitor.Wait`, `System.Threading.Thread.Sleep`,
or `System.Threading.Thread.Join`.