

# System.Threading.SynchronizationLockException Class

```
[ILAsm]
.class public serializable SynchronizationLockException extends
System.SystemException

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

## Assembly Info:

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

## Summary

Represents the error that occurs when a method requires the caller to own the lock on a given `System.Threading.Monitor`, and the method is invoked by a caller that does not own that lock.

## 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

`System.Threading.SynchronizationLockException` is thrown if the `System.Threading.Monitor.Pulse`, `System.Threading.Monitor.PulseAll`, or `System.Threading.Monitor.Wait` methods are invoked for a specified object, but the caller does not own the lock on that object. `System.Threading.SynchronizationLockException` is also thrown when the `System.Threading.Monitor.Exit` method is called to release the lock on an object, but the caller does not own the lock on that object.

To acquire the lock on an object, use `System.Threading.Monitor.Enter`. To release the lock on an object, use `System.Threading.Monitor.Exit`.

# SynchronizationLockException() Constructor

```
[ILAsm]  
public rtspecialname specialname instance void .ctor()  
  
[C#]  
public SynchronizationLockException()
```

## Summary

Constructs and initializes a new instance of the `System.Threading.SynchronizationLockException` class.

## Description

This constructor initializes the `System.Threading.SynchronizationLockException.Message` property of the new instance to a system-supplied message that describes the error, such as "Synchronization method was invoked from an unsynchronized block of code." This message takes into account the current system culture.

The `System.Threading.SynchronizationLockException.InnerException` property is initialized to null.

# SynchronizationLockException(System.String) Constructor

```
[ILAsm]  
public rtspecialname specialname instance void .ctor(string message)  
  
[C#]  
public SynchronizationLockException(string message)
```

## Summary

Constructs and initializes a new instance of the `System.Threading.SynchronizationLockException` class.

## Parameters

| Parameter      | Description   |
|----------------|---|
| <i>message</i> | A <code>System.String</code> 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.SynchronizationLockException.Message` property of the new instance using *message*. If *message* is null, the `System.Threading.SynchronizationLockException.Message` property is initialized to the system-supplied message provided by the constructor that takes no arguments.

The `System.Threading.SynchronizationLockException.InnerException` property is initialized to null.

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

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

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

## Summary

Constructs and initializes a new instance of the `System.Threading.SynchronizationLockException` class.

## Parameters

| Parameter             | Description   |
|-----------------------|---|
| <i>message</i>        | A <code>System.String</code> 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 <code>System.Exception</code> that is the cause of the current exception. If <i>innerException</i> is non-null, then the current <code>Exception</code> was raised in a catch block handling <i>innerException</i> .                   |

## Description

This constructor initializes the `System.Threading.SynchronizationLockException.Message` property of the new instance using *message* and the `System.Threading.SynchronizationLockException.InnerException` property using *innerException*. If *message* is null, the `System.Threading.SynchronizationLockException.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`.]

