**Part 94 - Difference between Monitor and lock in C#**

In this video we will discuss, the Difference between Monitor class and lock.

Both **Monitor class**and **lock**provides a mechanism that synchronizes access to objects. lock is the shortcut for Monitor.Enter with try and finally.  
  
**This means that, the following code**

staticobject\_lock=newobject**();**

publicstaticvoidAddOneMillion**()**

**{**

for**(**inti=1**;**i<=1000000**;**i++**)**

**{**

lock**(**\_lock**)**

**{**

Total++**;**

**}**

**}**

**}**

**can be rewritten as shown below:**

staticobject\_lock=newobject**();**

publicstaticvoidAddOneMillion**()**

**{**

for**(**inti=1**;**i<=1000000**;**i++**)**

**{**

// Acquires the exclusive lock

Monitor.Enter**(**\_lock**);**

try

**{**

Total++**;**

**}**

finally

**{**

// Releases the exclusive lock

Monitor.Exit**(**\_lock**);**

**}**

**}**

**}**

**In C# 4, it is implement slightly differently as shown below**

staticobject\_lock=newobject**();**

publicstaticvoidAddOneMillion**()**

**{**

for**(**inti=1**;**i<=1000000**;**i++**)**

**{**

boollockTaken=false**;**

// Acquires the exclusive lock

Monitor.Enter**(**\_lock**,**reflockTaken**);**

try

**{**

Total++**;**

**}**

finally

**{**

// Releases the exclusive lock

if**(**lockTaken**)**

Monitor.Exit**(**\_lock**);**

**}**

**}**

**}**

**So, in short, lock is a shortcut and it's the option for the basic usage.** If you need more control to implement advanced multithreading solutions using TryEnter() Wait(), Pulse(), & PulseAll() methods, then the Monitor class is your option.