**Part 92 - Significance of Thread.Join and Thread.IsAlive functions**

In this video we will discuss, the significance of **Thread.Join**and **Thread.IsAlive**functions with an example.   
  
**Join**blocks the current thread and makes it wait until the thread on which Join method is invoked completes. Join method also has a overload where we can specify the timeout. If we don't specify the timeout the calling thread waits indefinitely, until the thread on which Join() is invoked completes. This overloaded Join(int millisecondsTimeout) method returns boolean. True if the thread has terminated otherwise false. Join is particularly useful when we need to wait and collect result from a thread execution or if we need to do some cleanup after the thread has completed.

**IsAlive returns boolean.**True if the thread is still executing otherwise false.

**Program code used in the demo:**

usingSystem**;**

usingSystem.Threading**;**

namespaceThreadingExample

**{**

classProgram

**{**

publicstaticvoidMain**()**

**{**

Console.WriteLine**(**"Main Started"**);**

ThreadT1=newThread**(**Program.Thread1Function**);**

T1.Start**();**

ThreadT2=newThread**(**Program.Thread2Function**);**

T2.Start**();**

if**(**T1.Join**(**1000**))**

**{**

Console.WriteLine**(**"Thread1Function completed"**);**

**}**

else

**{**

Console.WriteLine**(**"Thread1Function hot not completed in 1 second"**);**

**}**

T2.Join**();**

Console.WriteLine**(**"Thread2Function completed"**);**

for**(**inti=1**;**i<=10**;**i++**)**

**{**

if**(**T1.IsAlive**)**

**{**

Console.WriteLine**(**"Thread1Function is still doing it's work"**);**

Thread.Sleep**(**500**);**

**}**

else

**{**

Console.WriteLine**(**"Thread1Function Completed"**);**

break**;**

**}**

**}**

Console.WriteLine**(**"Main Completed"**);**

**}**

publicstaticvoidThread1Function**()**

**{**

Console.WriteLine**(**"Thread1Function started"**);**

Thread.Sleep**(**5000**);**

Console.WriteLine**(**"Thread1Function is about to return"**);**

**}**

publicstaticvoidThread2Function**()**

**{**

Console.WriteLine**(**"Thread2Function started"**);**

**}**

**}**

**}**