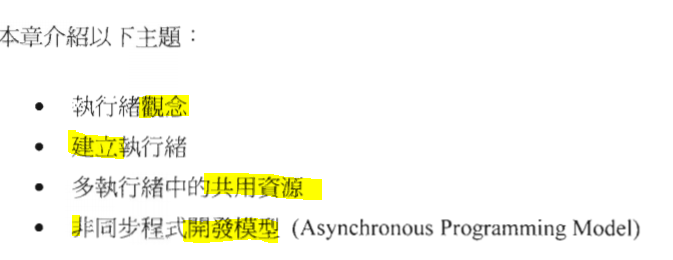
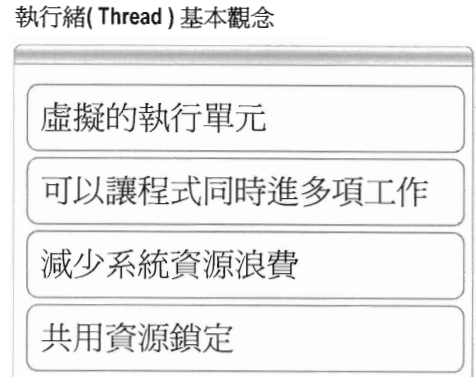
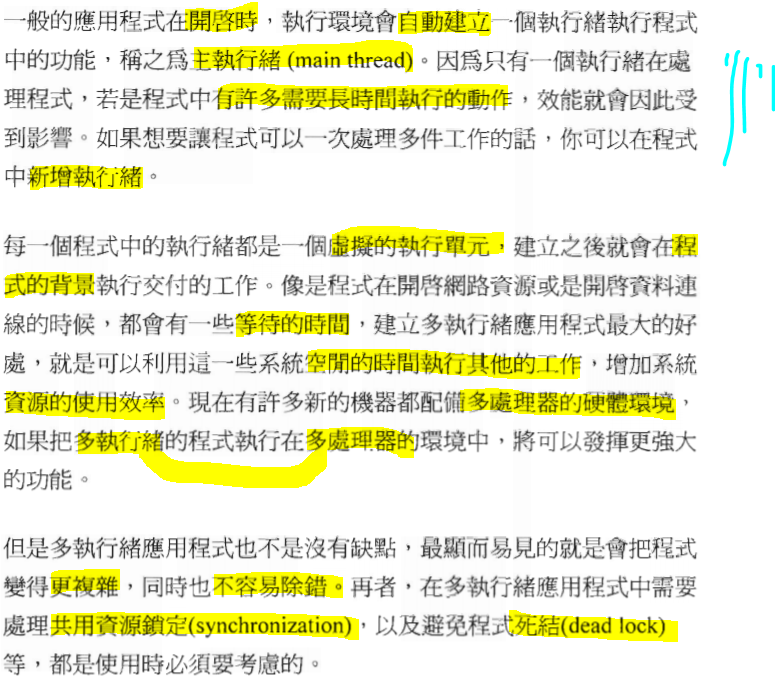
P 395

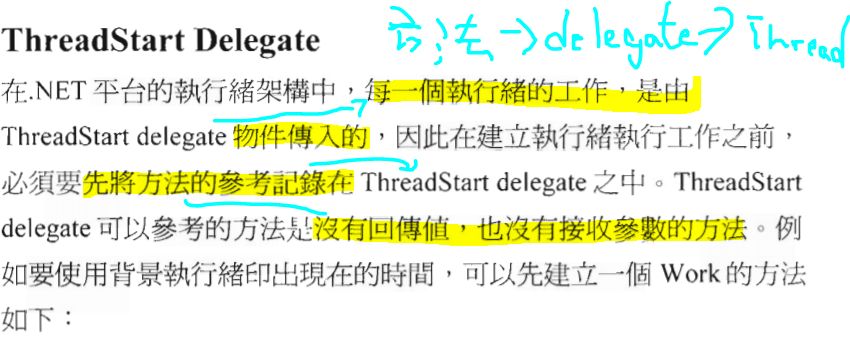




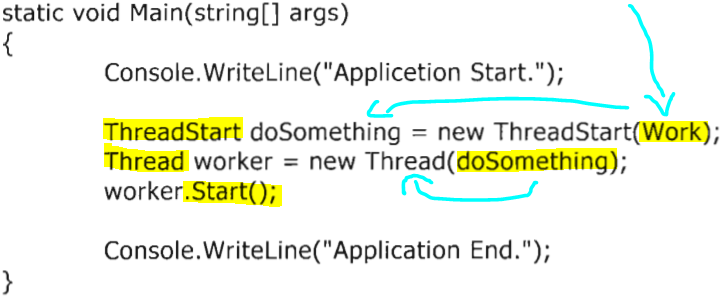


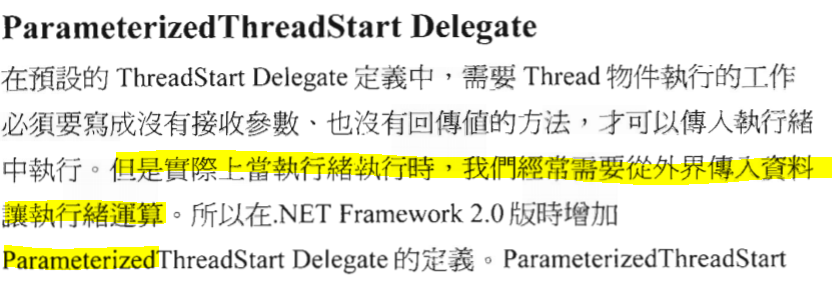
P 398 ThreadStart 和 ParameterizedThreadStart



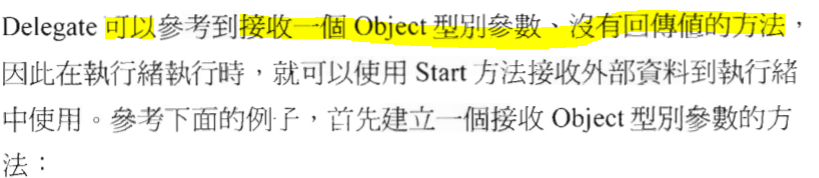


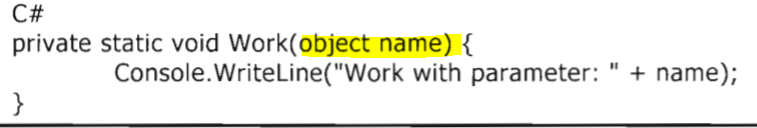


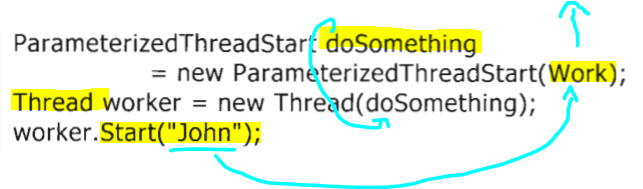




可接收參數的改良版 ( 只接收一個Object 沒有回傳 )



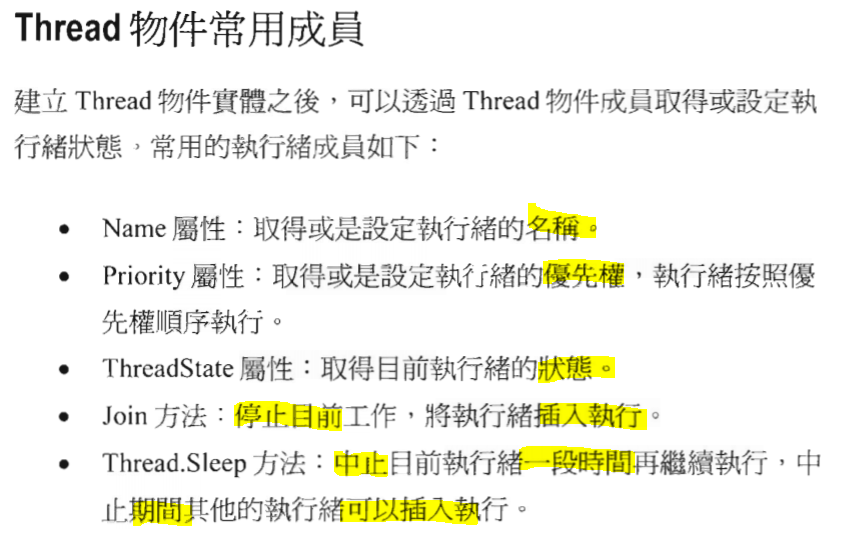




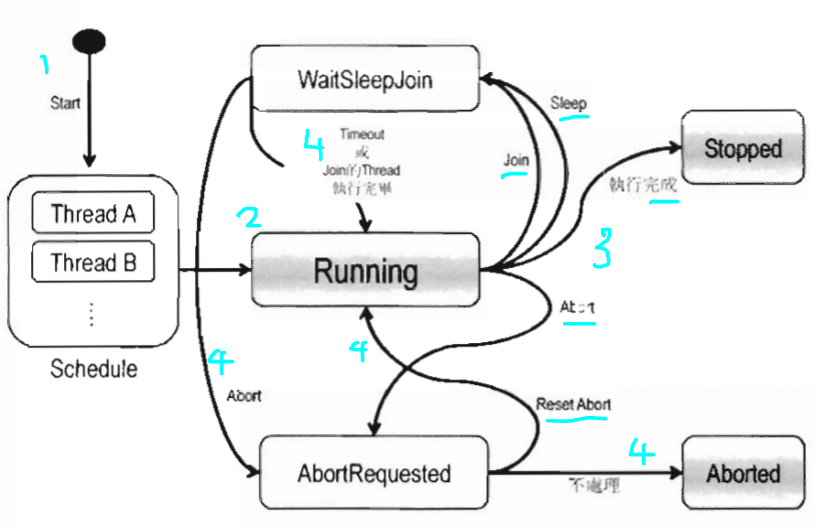
感覺可以實作看看，而且它一個thread可以加入多個方法嗎？( 順序 )

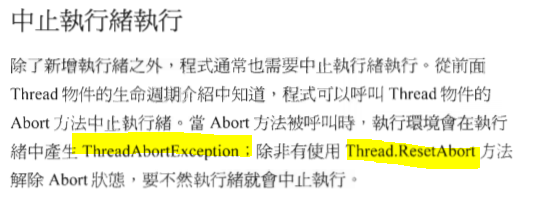
下面有結合這個內容的實作︰

P 401

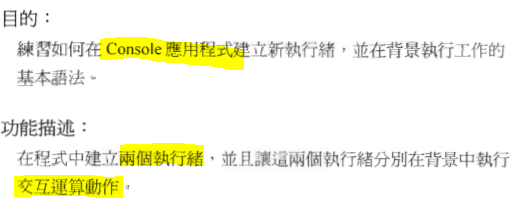


ThreadState 和 LifeCycle :





404 實作 note3\_ex2.Program



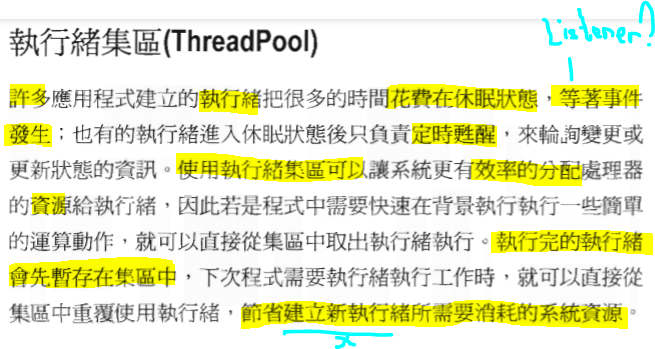
我看沒有「交互運算」的部份。

//它們是都跑 20 次，但是由不同的數字開始起跑。

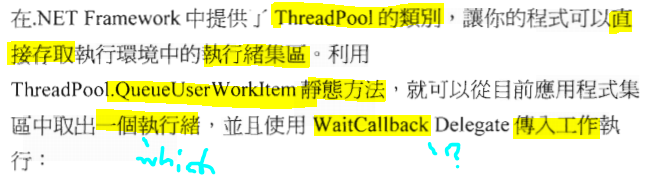
//它們一同起跑，所以每個Cycle都一起完成，一同前進…

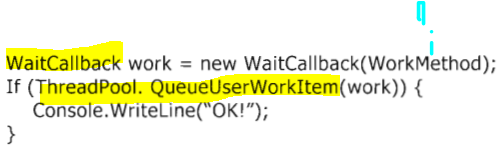
//但是它們並沒有交流。

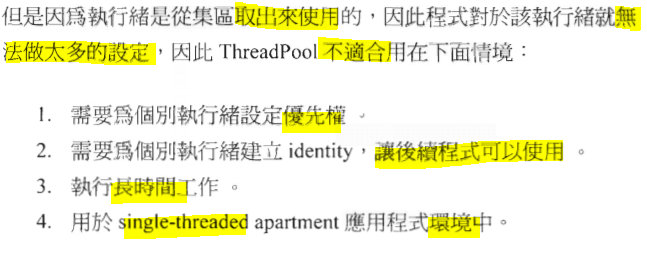
P 407 Thread Pool



所以它大概是個cache一樣的物件



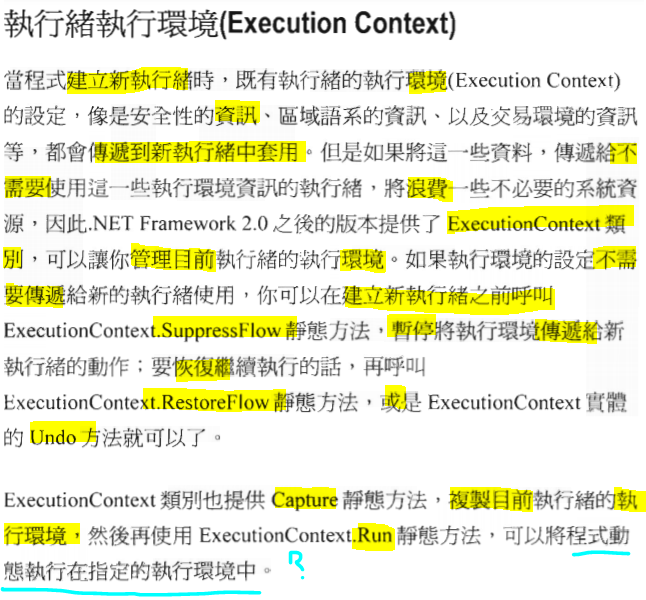




?? 我不明白 WorkMethod 是一個執行緒，還是一個方法

看上去是個執行緒，等等試試好了。下個主題後就有這個實作…

P 409 執行緒執行環境 ExecutionContext類別



環境資訊太多 >

控制環境 > 傳入 / 不傳入， 捕捉目前環境 / Run?是貼上的意思？

實作︰ note3\_ex2.Program

WaitCallback cb = new WaitCallback(Print);// 放入等待池

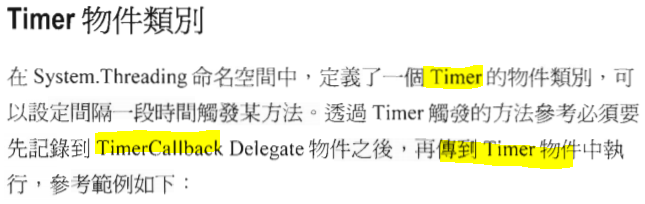
ThreadPool.QueueUserWorkItem(cb, 1); // 從池中取出並運作

ThreadPool.QueueUserWorkItem(cb, 15); // 再取一次

//和上面一個實作的對比

//它也像是開了兩個Thread，同時運行。

P 412 Timer



補充比較有用，PDF的實作什麼都看不出來。

https://docs.microsoft.com/zh-tw/dotnet/api/system.timers.timer?view=net-5.0

Main:----------

SetTimer();

// Timer開始，但頭一秒沒有事做，程式會先印︰

Console.WriteLine("\nPress the Enter key to exit the application...\n");

Console.WriteLine("The application started at {0:HH:mm:ss.fff}", DateTime.Now);

Console.ReadLine();

// 程式會在這裡一直等待輸入，期間Timer一直做自己的事…

aTimer.Stop(); // 當按下Enter，程式讀取了上一行，就會跑來「停止」timer

aTimer.Dispose(); // 一定要按停再解散？

Console.WriteLine("Terminating the application...");

---------------------------------------------------------------------

private static System.Timers.Timer aTimer;

private static void SetTimer()

{

// Create a timer with a two second interval.

aTimer = new System.Timers.Timer(2000);// 間隔 2 秒

// Hook up the Elapsed event for the timer.

aTimer.Elapsed += OnTimedEvent; // 當時間到了要做的事情

aTimer.AutoReset = true; // 自己循還

aTimer.Enabled = true;// 啟動

}

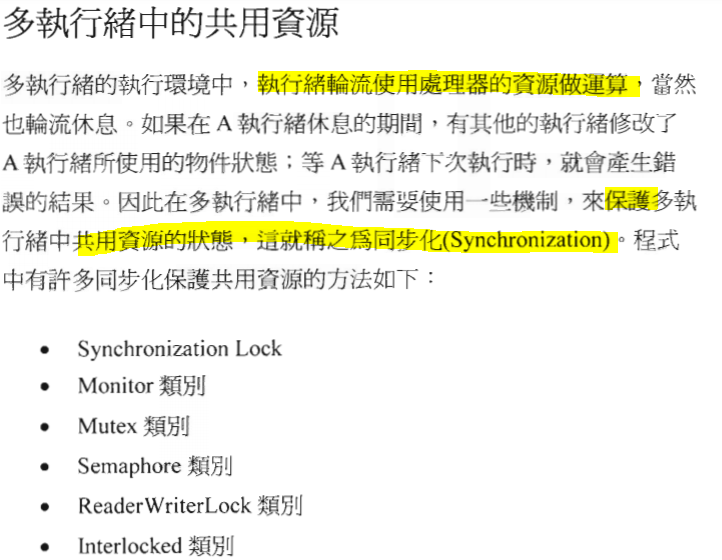
private static void OnTimedEvent(Object source, ElapsedEventArgs e)

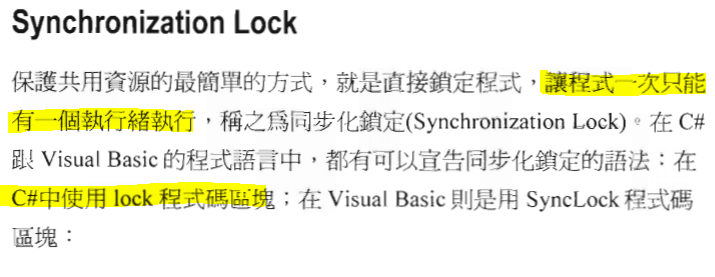
{

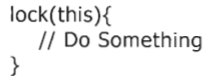
Console.WriteLine("The Elapsed event was raised at {0:HH:mm:ss.fff}",

e.SignalTime); }

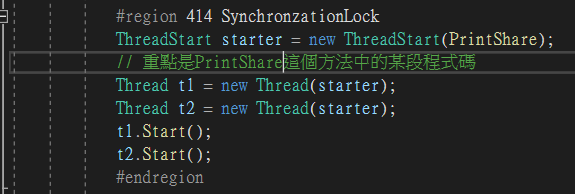
P 413 共用資源

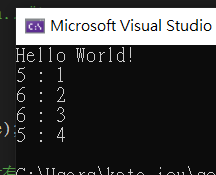


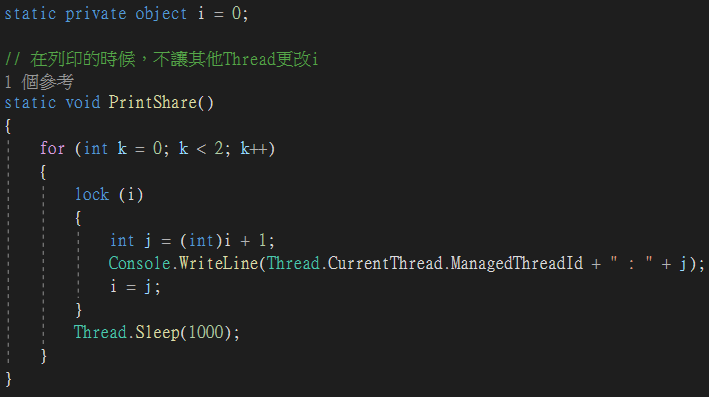


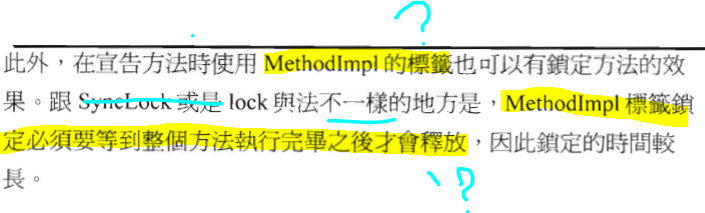


this 是一個屬性

實作︰

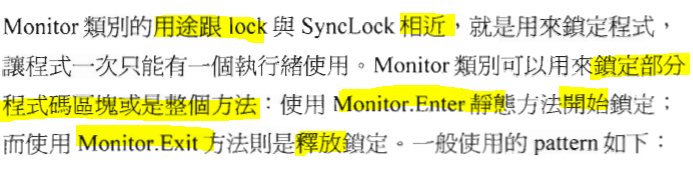


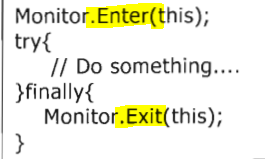




我搜尋了MethodImpl C# 有點複雜，而功能和上面差不多，所以我不實作了。

P 415 Monitor



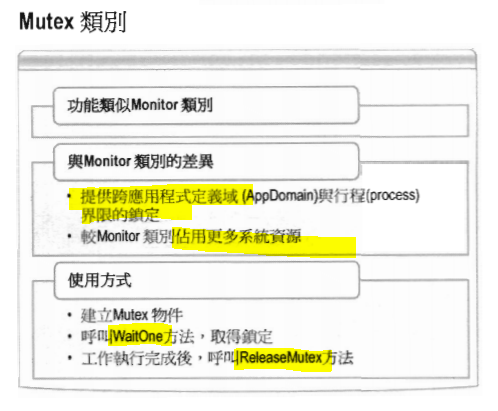


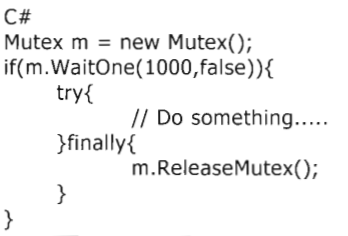
實作︰一直失敗，有保留紀錄。 note3\_ex2

後來成功了，但是發現十分多餘… Monitor\_ex

結論︰用 lock 就夠了

P 416 Mutex





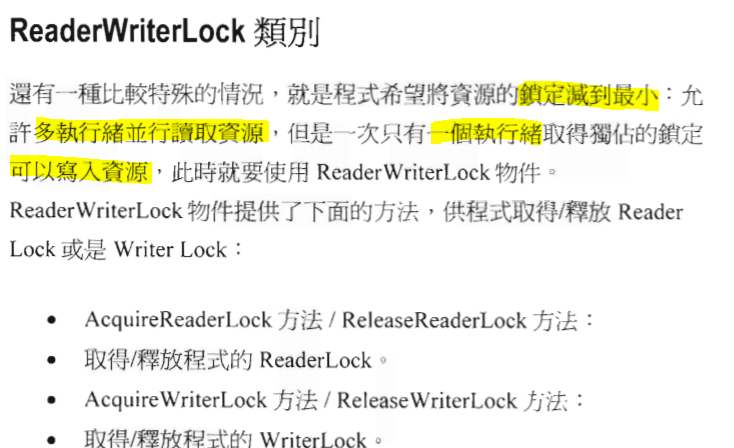
不想實作，沒有用到的地方。沒有需要設計跨應用程式的資源共用方法…

P 417 Semaphore 同時入場人數



PDF 也沒有實作…是知道它有就好？

P 418 ReaderWriterLock 資源的讀取和寫入的權限分開



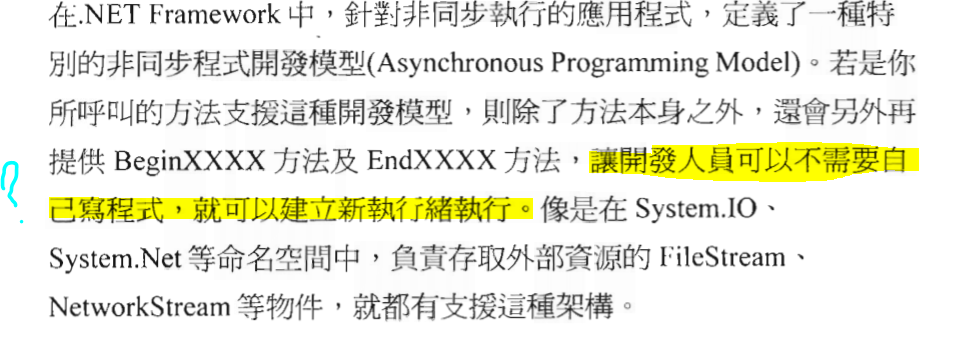
然後又沒有實作了，所以它也是知一下就好？

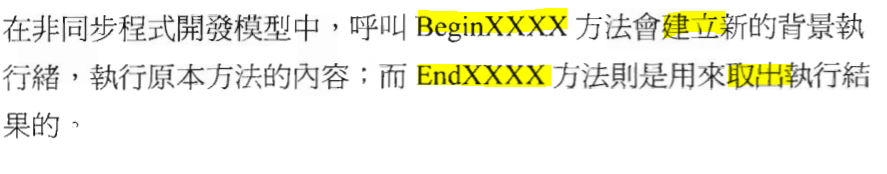
其實有 lock 在，確定了得到資源再進行修改的程式。效果不是一樣的嗎？

….

所以真正有用的地方，是限制「能同時讀取」的人數？…

P 419 Asynchronous Programming Model 非同步程式開模型

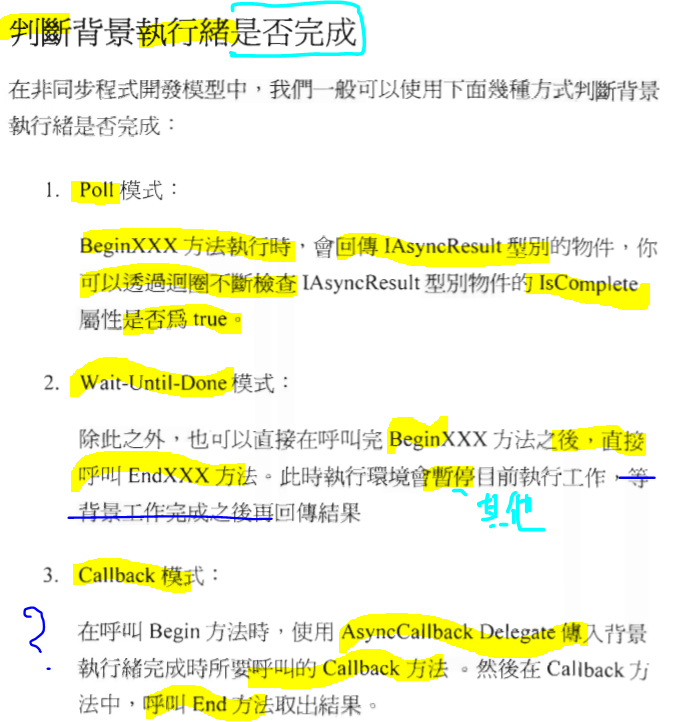




這是一個程式運作的模式。

以呼叫方法 (請參照上述命名方式) 來開啟新的執行緒，給得到該執行的結果

( 所以我推斷 這個執行緖應該至少是個Object，一個(get set)屬性，一個開執行緒去修改它的方法。)

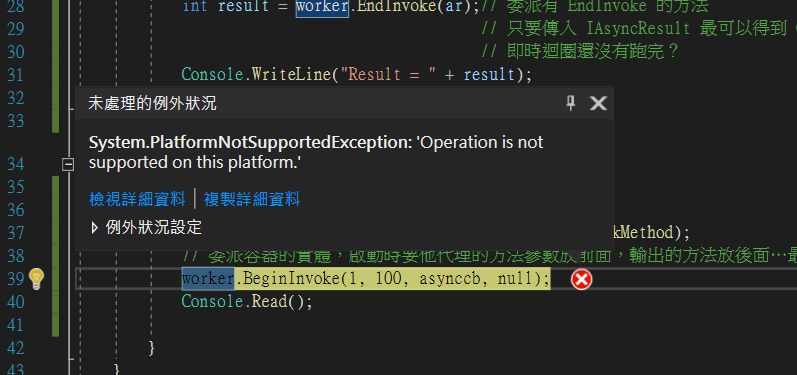


下頁實作︰

P 421 APM 實作於︰

C:\Users\kate\_jou\source\repos\Learning\ASP.NET\note3\_exercise\note3\_ex2\

APM\_Model



在這個平台不支援？

上網找了一下資，詳細於 專案之中。

最後放棄…

這個實作是第三個模式。其實還有頭兩個可以試，可能對平台的要求沒那麼高…

但是我們之前做實作的每一步，要求它們印出來做到哪裡，如 start …end

不多多少少都在做這個東西嗎？

我想測自己發明的方法，由Thread instant 一個物件，

物件有一個 bool，在完成的時候才= true…看這樣可不可以代替上述各種複雜到不行的方法…

有，我的測試成功 C:\Users\kate\_jou\source\repos\Learning\ASP.NET\note3\_exercise\note3\_ex2

\note3\_ex2

行 197

Main:

#region 429 自行測試

MyOwnTest test\_obj = new MyOwnTest(); test\_obj.class\_ID = 100;

MyOwnTest test\_obj2 = new MyOwnTest(); test\_obj2.class\_ID = 200;

test\_obj.MakeAThreadAndRun();

Thread.Sleep(1000);

test\_obj2.MakeAThreadAndRun();

SetTimerForMyTest(test\_obj2);

Thread.Sleep(3000);

SetTimerForMyTest(test\_obj);

// 跑出來︰

// class\_ID: 100-- > Count:10

// 檢查時間︰ 14:30:38.091 物件︰100 未完成

// class\_ID:200-- > Count:10

// 檢查時間︰ 14:30:39.091 物件︰200 未完成

// 檢查時間︰ 14:30:40.091 物件︰100 完成

// 物件 200 沒有顯示 完成

// 原因是數到 10 的那一下，Timer也剛好檢查，但 finish 屬性還沒有來得及 = true;

// 在 Main 因為兩個 obj 都跑完後， Main也會跑完，自然 Timer也關了，所以︰沒有檢查到 200 已完成

Console.ReadLine();// 試試加這一行，不讓Main跑完…

// 檢查時間︰ 14:39:12.157 物件︰100 未完成

//class\_ID:200-- > Count:10

// 檢查時間︰ 14:39:13.155 物件︰200 未完成

// 檢查時間︰ 14:39:14.161 物件︰100 完成

// 檢查時間︰ 14:39:15.156 物件︰200 完成

// 檢查時間︰ 14:39:16.160 物件︰100 完成

// 檢查時間︰ 14:39:17.155 物件︰200 完成

// 檢查時間︰ 14:39:18.159 物件︰100 完成

// 檢查時間︰ 14:39:19.156 物件︰200 完成

// 好了，這樣的話，不按Enter，它就一直跑…

// 這個測試證明我的方法可以，儘管PDF的東東無法實作，但是他的想法啟發了我。

在Program以外的物件藍圖︰

public class MyOwnTest

{

public bool finshed = false;

public int class\_ID;

public void MakeAThreadAndRun()

{

ThreadStart starter = new ThreadStart(Count10);

Thread tl = new Thread(starter);

tl.Start();

}

private void Count10()

{

for (int i = 1; i <= 10; i++)

{

Console.WriteLine("class\_ID:{0}-->Count:{1}",

class\_ID, i);

Thread.Sleep(2000); // 每兩秒數一

}

finshed = true;

}

}

在 Program之下，Main 旁邊的 Timer︰

private static void SetTimerForMyTest(MyOwnTest obj)

{

//bool Done = obj.finshed;

// Create a timer with a two second interval.

aTimer = new System.Timers.Timer(2000);// 間隔 2 秒

// Hook up the Elapsed event for the timer.

// https://stackoverflow.com/questions/9977393/how-do-i-pass-an-object-into-a-timer-event

aTimer.Elapsed += (sender, e) => OnTimedEventForMyTest(sender, e, obj); // 當時間到了要做的事情

aTimer.AutoReset = true; // 自己循還

aTimer.Enabled = true;// 啟動

}

private static void OnTimedEventForMyTest(Object source, ElapsedEventArgs e, MyOwnTest obj)

{

Console.Write(" 檢查時間︰ {0:HH:mm:ss.fff} 物件︰{1}", e.SignalTime, obj.class\_ID);

if (obj.finshed)

Console.WriteLine(" 完成");

else

Console.WriteLine(" 未完成");

}

P 424 Finish