Thread

Experiment

Experiment #1

Test order of thread execution and thread switch

Thread 1 , Thread 2 สลับกันทำงาน

```
1   // simple thread - test order
2   using System;
3   using System.Threading;
4
5   namespace Lab_OS_Concurrency
6   {
7       class Program
8       {
9           static void TestThread1()
10          {
11              int i;
12              for (i = 0; i < 100; i++)
13                  Console.WriteLine("Thread# 1 i={0}", i);
14          }
15          static void TestThread2()
16          {
17              int i;
18              for (i = 0; i < 100; i++)
19                  Console.WriteLine("Thread# 2 i={0}", i);
20          }
21
22          static void Main(string[] args)
23          {
24              Thread th1 = new Thread(TestThread1);
25              Thread th2 = new Thread(TestThread2);
26              th1.Start();
27              th2.Start();
28          }
29      }
30  }
```

Experiment #2

- Resource sharing among threads

แล้วผลลัพธ์ที่ได้จะเป็นอย่างไร

```
1   //test resource sharing
2   using System;
3   using System.Threading;
4
5   namespace Lab_OS_Concurrency01
6   {
7       class Program
8       {
9           static int resource = 10000;
10          static void TestThread1()
11          {
12              Console.WriteLine("Thread# 1 i={0}", resource);
13          }
14          static void TestThread2()
15          {
16              Console.WriteLine("Thread# 2 i={0}", resource);
17          }
18
19          static void Main(string[] args)
20          {
21              Thread th1 = new Thread(TestThread1);
22              Thread th2 = new Thread(TestThread2);
23              th1.Start();
24              th2.Start();
25          }
26      }
27  }
```

Experiment #3

- Pause a thread

แล้วแต่การเข้าใช้งาน → 10,000
→ 55,555

ถ้าใส่ sleep จะเยอะ 55,555

```
//test pause a thread
using System;
using System.Threading;

namespace Lab_OS_Concurrency02
{
    class Program
    {
        static int resource = 10000;
        static void TestThread1()
        {
            resource = 55555;
        }

        static void Main(string[] args)
        {
            Thread th1 = new Thread(TestThread1);
            th1.Start();
            //Thread.Sleep(10);
            Console.WriteLine("resource={0}", resour
        }
    }
}
```

Experiment #3.1

- Pause a thread #2

```
1   //test pause #2
2   using System;
3   using System.Threading;
4
5   namespace Lab_OS_Concurrency01
6   {
7       class Program
8       {
9           static int resource = 10000;
10          static void TestThread1()
11          {
12              int i;
13              for (i = 0; i < 45555; i++)
14              {
15                  resource++;
16                  Console.Write(".");
17              }
18          }
19
20          static void Main(string[] args)
21          {
22              Thread th1 = new Thread(TestThread1);
23              th1.Start();
24              Thread.Sleep(10);
25              Console.WriteLine("Resource = {0}", resource);
26          }
27      }
28  }
```

→ ดูผลลัพธ์ที่ได้

Experiment #3.1 desired result

Experiment #4

- Join thread

ใช้คำสั่ง join เพื่อให้ได้ผลลัพธ์ที่ถูกต้องเสมอ

```
1   //test pause #2
2   using System;
3   using System.Threading;
4
5   namespace Lab_OS_Concurrency01
6   {
7       class Program
8       {
9           static int resource = 10000;
10          static void TestThread1()
11          {
12              int i;
13              for (i = 0; i < 45555; i++)
14              {
15                  resource++;
16                  Console.Write(".");
17              }
18          }
19
20          static void Main(string[] args)
21          {
22              Thread th1 = new Thread(TestThread1);
23              th1.Start();
24              //Thread.Sleep(10);
25              th1.Join();
26              Console.WriteLine("Resource = {0}", resource);
27          }
28      }
29  }
```