

# Thread and UI Demo

```
using System;
using System.Threading;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;

public struct MyData {
    public double pi;
    public int iters;
}

public class Calc {
    private double _pi;
    private int _iters;

    private readonly int TotalIters;
    public Calc(int it) {
        _iters = 1;
        _pi = 0;
        TotalIters = it;
    }

    public MyData PI {
        get {
            MyData pi = new MyData();
            lock (this) {
                pi.pi = _pi;
                pi.iters = _iters;
            }
            return pi;
        }
    }
}

public Thread MakeThread() {
    return new Thread(new ThreadStart(this.ThreadStarter));
}
```

```
}

private void calculate() {
    double series = 0;
    do {
        series++;
        lock (this) {
            _iters += 4;
            _pi = series * 4;
        }
    } while (_iters < TotalIters); } private void ThreadStarter()
{ try { calculate(); } catch (ThreadAbortException e) {
    Console.WriteLine("ThreadAbortException"); } } } public class
Form1 : System.Windows.Forms.Form { private
System.Windows.Forms.Label label1; private
System.Windows.Forms.TextBox PiValue; private
System.Windows.Forms.Label label2; private
System.Windows.Forms.TextBox Iterations; private Calc pi = new
Calc(100000000); private Thread calcThread = null; private
System.Windows.Forms.Timer timer1; private
System.Windows.Forms.Button StopButton; private
System.Windows.Forms.Button Pause; private
System.ComponentModel.IContainer components; public Form1() {
    this.label1 = new System.Windows.Forms.Label(); this.label2 =
    new System.Windows.Forms.Label(); this.Pause = new
    System.Windows.Forms.Button(); this.PiValue = new
    System.Windows.Forms.TextBox(); this.StopButton = new
    System.Windows.Forms.Button(); this.timer1 = new
    System.Windows.Forms.Timer(this.components); this.Iterations =
    new System.Windows.Forms.TextBox(); this.SuspendLayout(); //
    // label1 // this.label1.Location = new
    System.Drawing.Point(8, 24); this.label1.Name = "label1";
    this.label1.TabIndex = 0; this.label1.Text = "Value of PI:";
    // // label2 // this.label2.Location = new
    System.Drawing.Point(8, 72); this.label2.Name = "label2";
    this.label2.TabIndex = 2; this.label2.Text = "Iterations:"; //
    // Pause // this.Pause.Location = new System.Drawing.Point(24,
    112); this.Pause.Name = "Pause"; this.Pause.TabIndex = 5;
```

```
this.Pause.Text = "Pause"; this.Pause.Click += new System.EventHandler(this.Pause_Click); // // PiValue //  
this.PiValue.Location = new System.Drawing.Point(128, 24);  
this.PiValue.Name = "PiValue"; this.PiValue.ReadOnly = true;  
this.PiValue.Size = new System.Drawing.Size(136, 20);  
this.PiValue.TabIndex = 1; this.PiValue.Text = ""; // //  
StopButton // this.StopButton.Location = new  
System.Drawing.Point(200, 112); this.StopButton.Name =  
"StopButton"; this.StopButton.TabIndex = 4;  
this.StopButton.Text = "Stop"; this.StopButton.Click += new  
System.EventHandler(this.StopButton_Click); //  
this.timer1.Enabled = true; this.timer1.Tick += new  
System.EventHandler(this.timer1_Tick); //  
this.Iteratons.Location = new System.Drawing.Point(128, 72);  
this.Iteratons.Name = "Iteratons"; this.Iteratons.ReadOnly =  
true; this.Iteratons.TabIndex = 3; this.Iteratons.Text = "";  
// this.AutoScaleBaseSize = new System.Drawing.Size(5, 13);  
this.ClientSize = new System.Drawing.Size(292, 149);  
this.Controls.AddRange(new System.Windows.Forms.Control[] {  
this.Pause, this.StopButton, this.Iteratons, this.label2,  
this.PiValue, this.label1}); this.Load += new  
System.EventHandler(this.Form1_Load); this.Closed += new  
System.EventHandler(this.Form1_Closed);  
this.ResumeLayout(false); } [STAThread] static void Main() {  
Application.Run(new Form1()); } private void Form1_Load(object  
sender, System.EventArgs e) { calcThread = pi.MakeThread();  
calcThread.Priority = ThreadPriority.Lowest;  
calcThread.Start(); } private void timer1_Tick(object sender,  
System.EventArgs e) { if (this.Pause.Text == "Pause") { MyData  
p = pi.PI; this.PiValue.Text = p.pi.ToString();  
this.Iteratons.Text = p.iters.ToString(); } if  
(calcThread.IsAlive == false) { StopButton.Enabled = false;  
Pause.Enabled = false; timer1.Enabled = false; calcThread =  
null; } } private void StopButton_Click(object sender,  
System.EventArgs e) { StopButton.Enabled = false;  
Pause.Enabled = false; timer1.Enabled = false;  
calcThread.Abort(); calcThread.Join(); calcThread = null; }
```

```
private void Pause_Click(object sender, System.EventArgs e) {  
if (this.Pause.Text == "Pause") { calcThread.Suspend();  
this.Pause.Text = "Resume"; this.StopButton.Enabled = false; }  
else { calcThread.Resume(); this.Pause.Text = "Pause";  
this.StopButton.Enabled = true; } } private void  
Form1_Closed(object sender, System.EventArgs e) { if  
(calcThread != null) { calcThread.Abort(); calcThread.Join();  
} } } [/csharp]
```