

))

إذا مات ابن آدم انقطع عمله إلا من ثلاث :

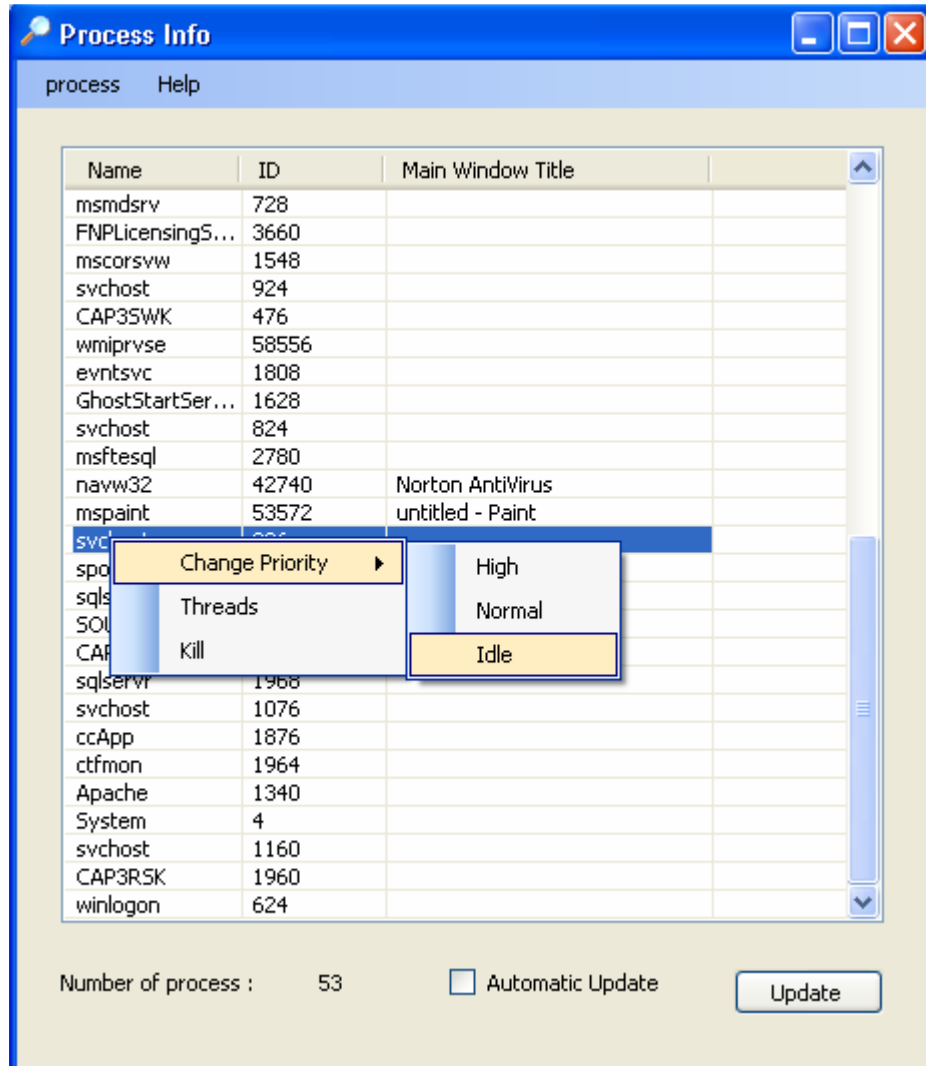
صدقة جارية أو علم ينتفع به أو ولد صالح يدعو له
((

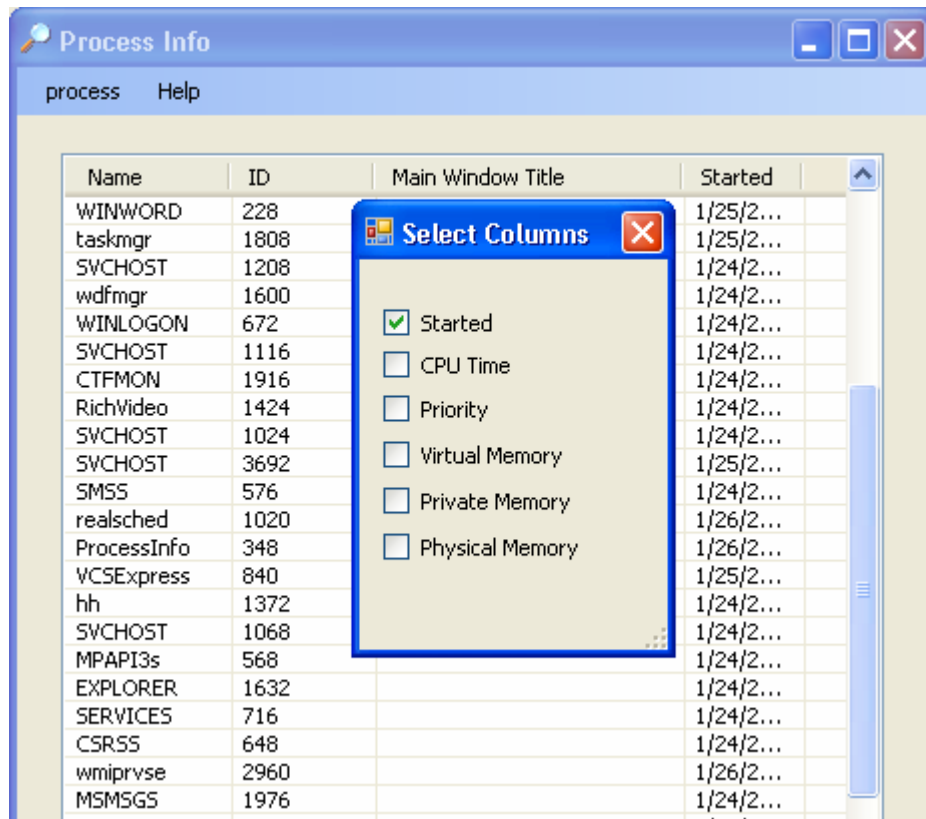
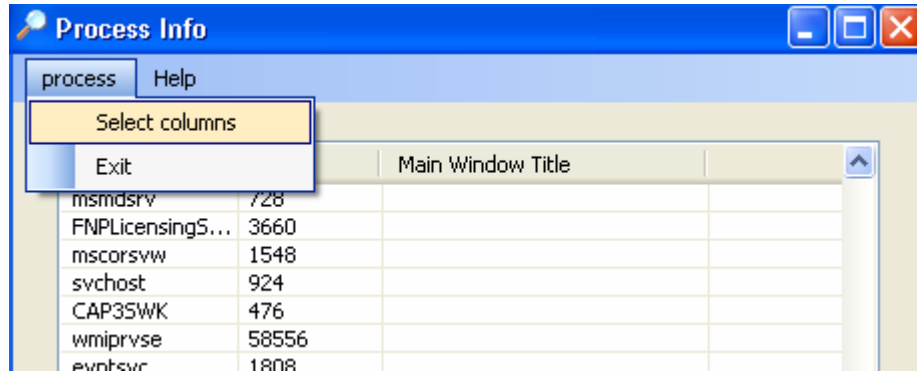
اللهم تقبلها في ميزان حسنات أستاذنا د.عبد العزيز سليمان

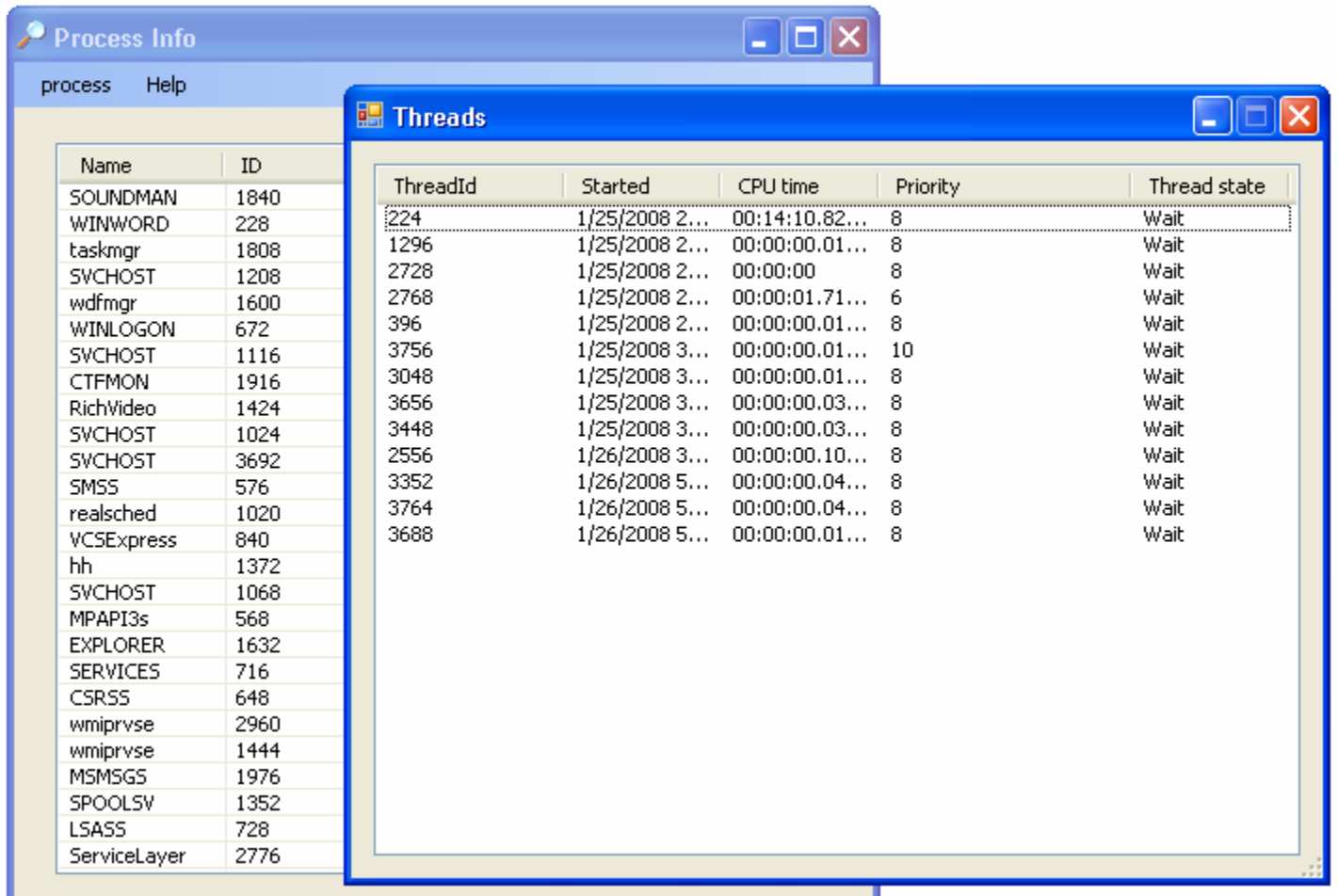
إن الهدف من هذا الدرس هو التعامل مع اكبر عدد ممكن من أَل controls و لذا ربما ستجد البرمجة ركيكة في بعض الأوقات فليس الهدف هو عمل برنامج مشابه لل Task Manager ولكن كما ذكرت الهدف هو معرفة كيفية التعامل مع الكثير من أَل controls و لذا هو درس موجه للمبتدئين جدا.

البرنامج ببساطة سيقوم بعرض أَل processes الشغالة، مع بعض المعلومات عنها.

واجهة البرنامج







بسم الله نبدأ

ملاحظات:

- الأداة الأساسية هي الـ ListView ، سنتعلم كيفية إضافة هذه الأداة وكيفية إضافة أعمدة لها في مرحلة التصميم و كيفية إضافة و حذف أعمدة منها برمجيا .
- سنتعلم كيفية إضافة قوائم و كيفية التعامل معها .
- سنتعلم كيفية إضافة About Dialog .
- سنتعلم كيفية إضافة form جديد و كيفية عرضه .
- سنتعلم كيفية إضافة Timer .

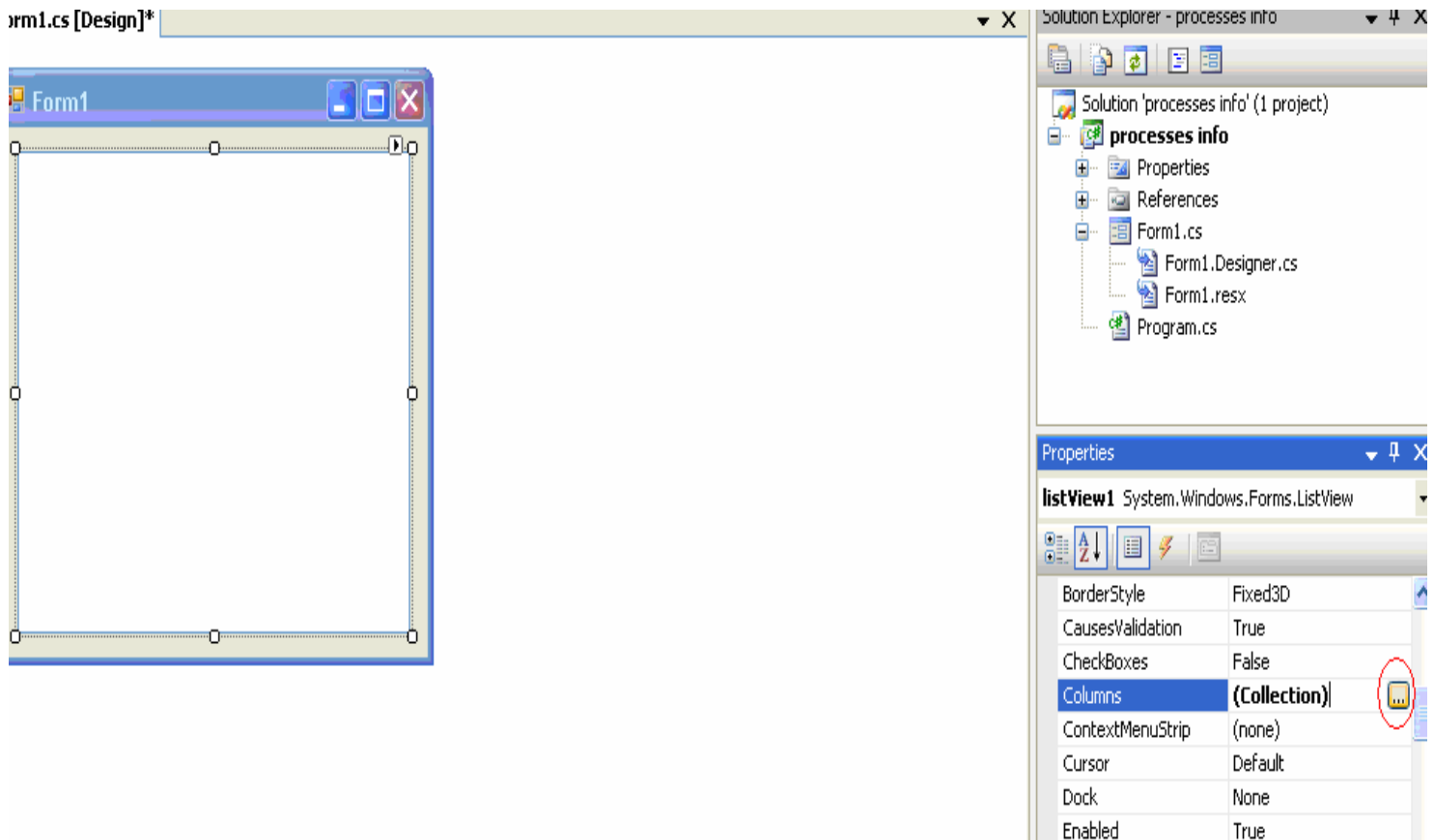
التعامل مع أل ListView

كون مشروع جديد من نوع Windows Application
ثم قم بإضافة الأداة ListView إلى المشروع

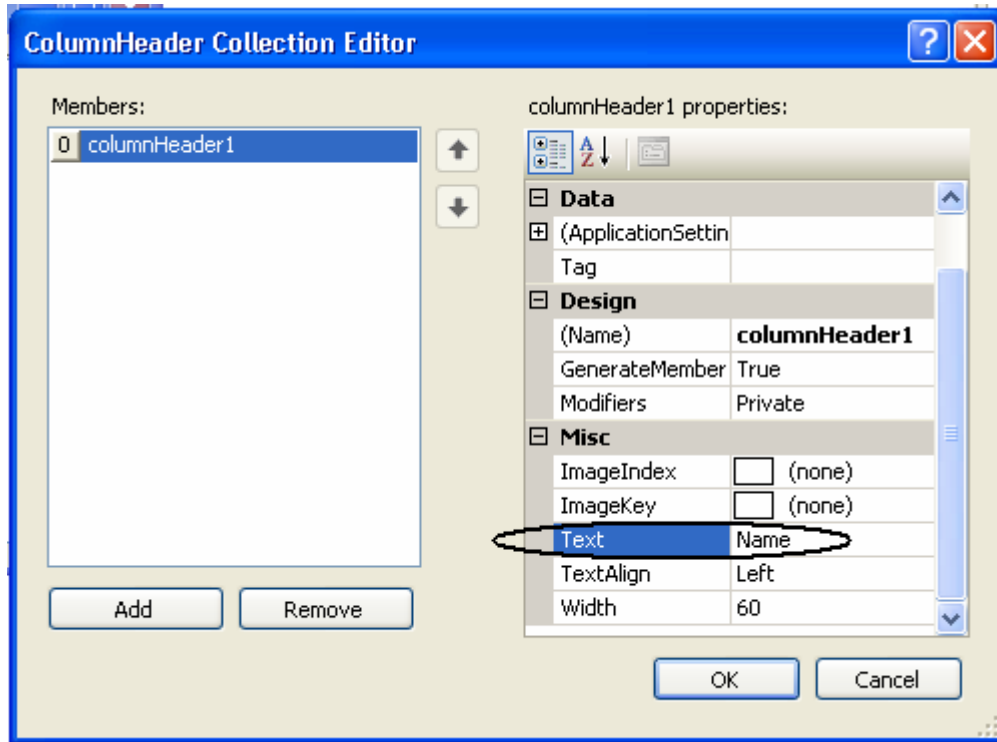
ثم توجه إلى قائمة الخصائص (properties) و غير الخصائص التالية

View = Details
Anchor = Top, Bottom, Left, Right
FullRowSelect = True
GridLines = True
MultiSelect = False

اضغط على الخاصية columns وانقر على الزر البيضي كما في الصورة



ستفتح لك نافذة أخرى يمكنك من إضافة أعمدة إلى أل listview ، كما في الصورة التالية :



اضغط على زر Add و في خانة Text اكتب اسم العمود الذي تريده أن يظهر ،
كرر العملية (أي الضغط على زر Add) بعدد الأعمدة التي تريد أن تضيفها،

مثلا لنضيف عمودين ، احدهما باسم Name و الآخر باسم ID ،
ثم اضغط على زر OK .

ستكون النافذة كالآتي :

The screenshot shows a standard Windows application window titled 'Form1'. Inside the window is a table with two columns: 'Name' and 'ID'. The table has a header row and several empty rows below it, suggesting it is a data grid or list box.

هذا عن كيفية إضافة عمود في مرحلة التصميم ،

أما لو أردنا أن نضيف عمود برمجيا ،

اضغط double click على أل Form الأساسي للمشروع لتنتقل إلى نافذة الكود و تحديدا إلى الحدث Load (أو ممكن اختيار الحدث من قائمة الأحداث)

This screenshot is similar to the first one, showing the 'Form1' window with the 'Name' and 'ID' table. A black arrow points to the top-right corner of the table header area, indicating where to perform a double-click to trigger a code event.

انقر نقرة مزدوجة هنا

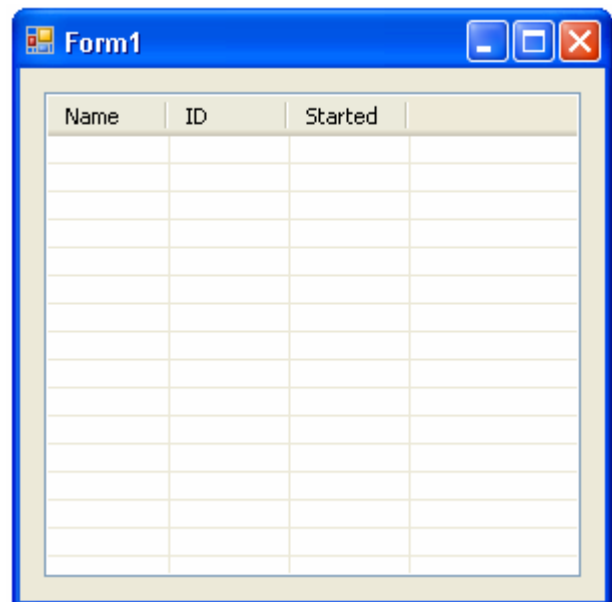
و غير الكود كما موضح في الصورة

```
namespace processes_info
{
    public partial class Form1 : Form
    {
        ColumnHeader addcol;

        public Form1()
        {
            InitializeComponent();
            addcol = new ColumnHeader();
            addcol.Text = "Started";
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            listView1.Columns.Add(addcol);
        }
    }
}
```

الان نفذ البرنامج ولاحظ النتيجة



و لحذف عمود برمجيا ، نكتب

```
listView1.Columns.Remove(colName);
```

ملئ محتويات ال list view

انتقل الى جزء البرمجة ،
و اضع الكود كما في الصورة

```
public partial class Form1 : Form
{
    ColumnHeader addcol;
    ListViewItem file;

    public Form1()
    {
        InitializeComponent();
        addcol = new ColumnHeader();
        addcol.Text = "Started";
    }

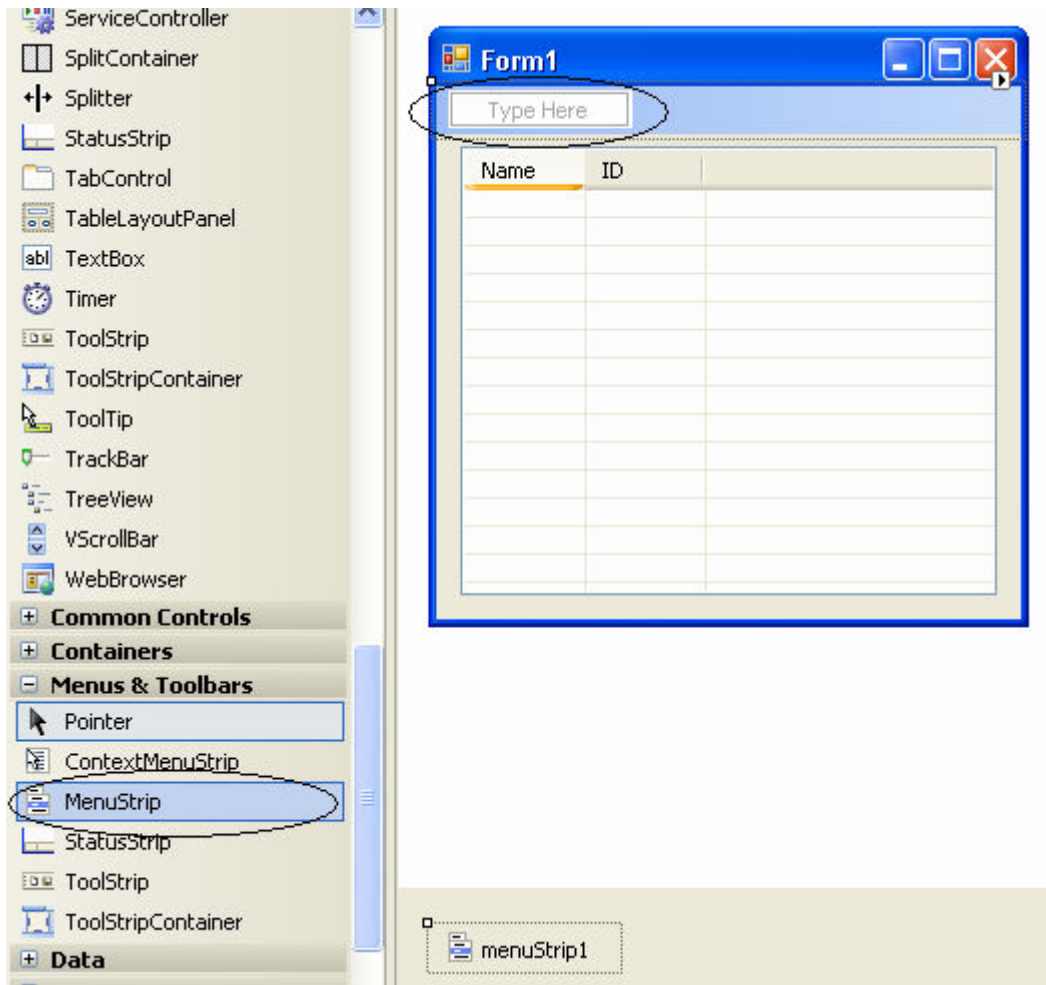
    private void Form1_Load(object sender, EventArgs e)
    {
        listView1.Columns.Add (addcol);

        file = new ListViewItem("process 1");
        file.SubItems.Add("0");
        file.SubItems.Add("1111");

        listView1.Items.Add(file);
    }
}
```

التعامل مع القوائم

ارجع الى نافذة التصميم ، ومن ال Toolbox قم بأضافة MenuStrip
كما في الصورة



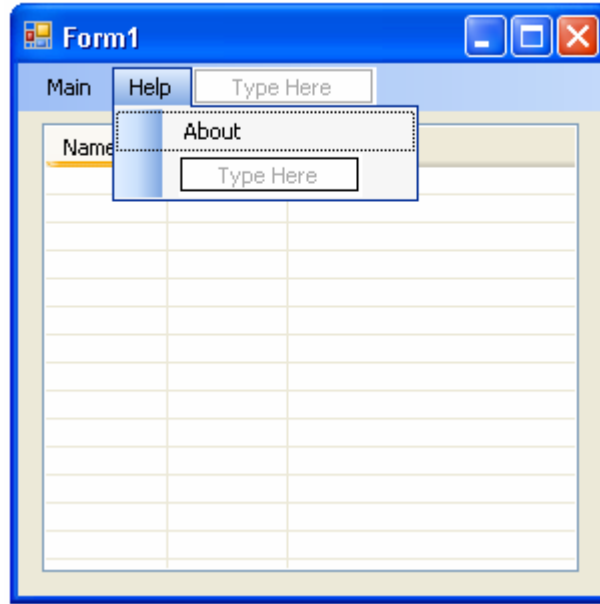
لإضافة عناصر إلى القائمة اضغط على كلمة Type Here واكتب عنوان القائمة الرئيسي، ثم اضغط Enter (أضف عناصر حسب الحاجة)

بالنسبة لنا هنا سنضيف قائمة باسم Main تحوي على عنصرين هما Show و الآخر Exit وقائمة أخرى بعنوان Help تحوي على زر About

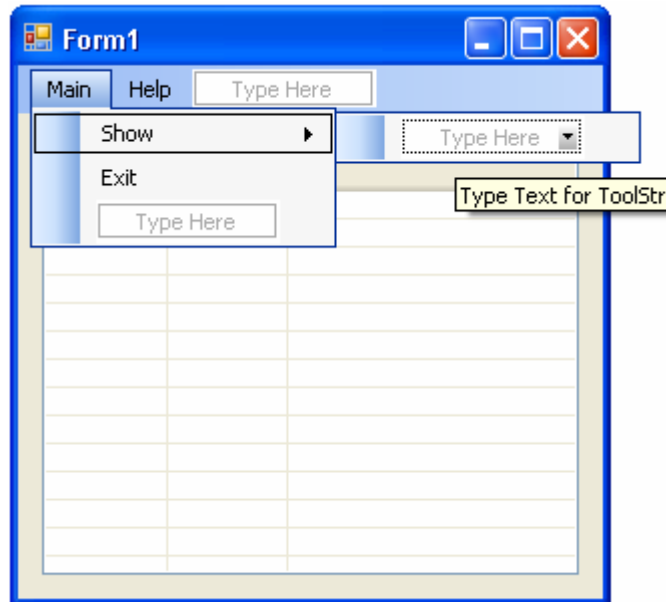
كما توضح الصور التالية

The screenshot shows a Windows application window titled "Form1". At the top, there is a menu bar with two buttons: "Main" and "Type Here". Below the menu bar, there is a sub-menu bar with a button labeled "Type Here". The main area of the form contains a table with 10 rows and 3 columns. The table is currently empty.

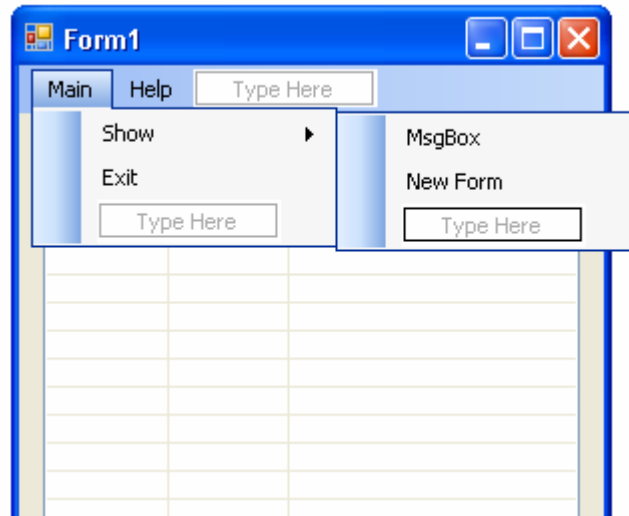
This screenshot shows the same "Form1" window, but with a context menu open over the "Main" button. The context menu has two options: "Show" and "Exit". Below the context menu, there is a button labeled "Type Here". The table below remains empty.



يمكن إضافة قائمة فرعية ،
مثلا اضغط على قائمة Main واضغط على Show سترى ظهور سهم صغير و تفرع يمكنك من إضافة
عناصر أخرى ضمن هذا العنصر

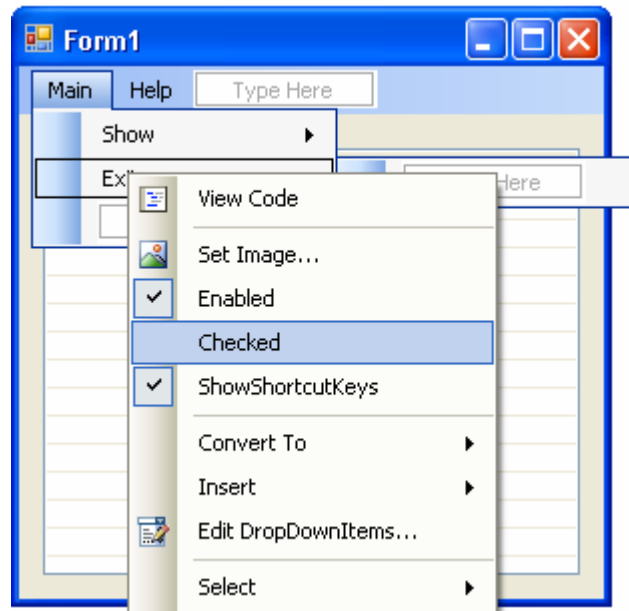


أضف خيارين كالتالي

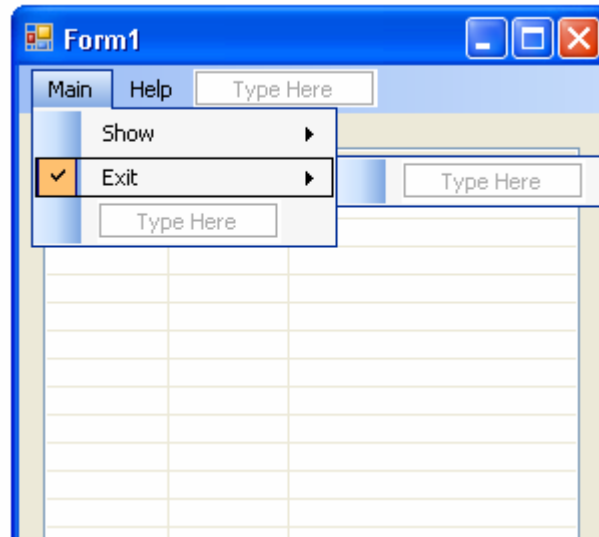


ملاحظة :

- 1- إذا أردنا أن نعمل عنصر في القائمة من نوع Checked ، نضغط بالزر الأيمن للماوس على العنصر و نختار checked كما في الصورة

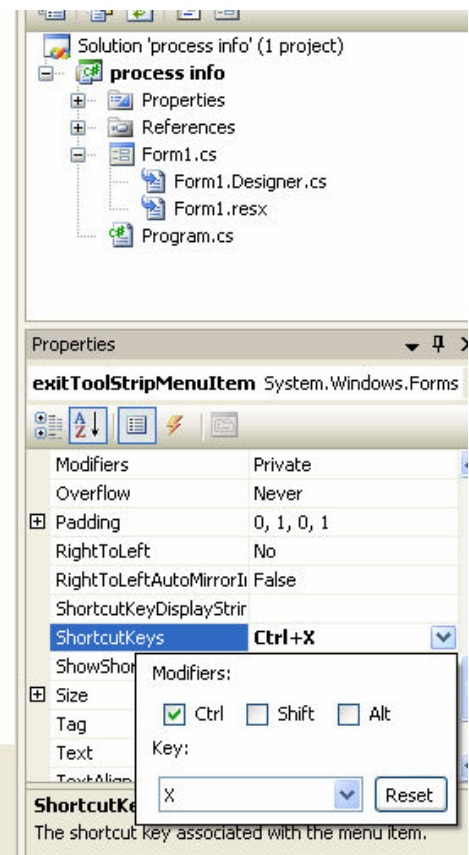
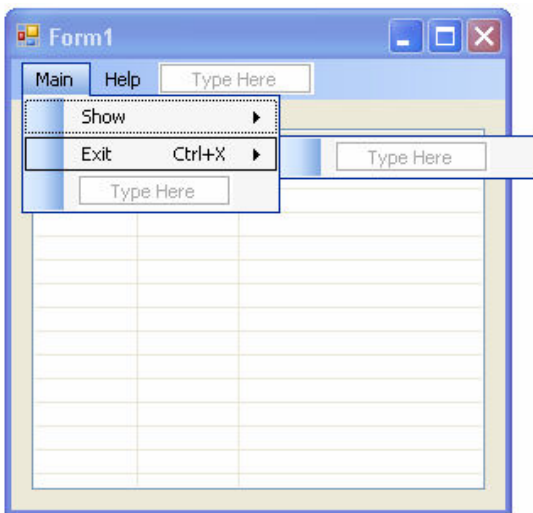


و النتيجة



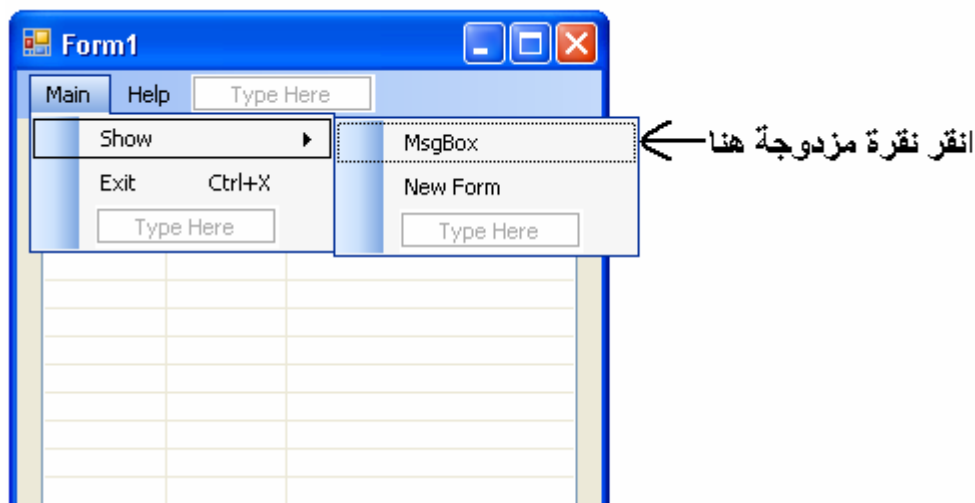
2- لإضافة اختصار لعناصر القائمة ، نضغط على العنصر و من قائمة الخصائص نختار ShortcutKey و نحدد الأحرف التي نريد

مثلا



إضافة الكود لعناصر القائمة

اضغط double click على العنصر المراد إضافة كود له و ستنتقل مباشرة إلى قسم الكود ،
مثلا ،
اتبع الصورة



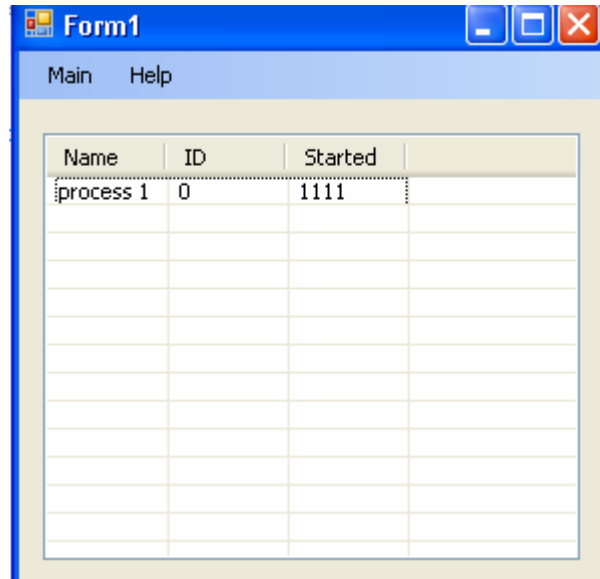
و أضف الكود التالي

```
private void msgBoxToolStripMenuItem_Click(object sender, EventArgs e)
{
    MessageBox.Show(" Hello !"); ← أضف هذا
}
```

بنفس الطريقة ،
انقر نقرة مزدوجة على خيار Exit من القائمة و أضف الكود التالي للخروج من البرنامج

```
private void exitToolStripMenuItem_Click(object sender, EventArgs e)
{
    Application.Exit(); ←
}
```

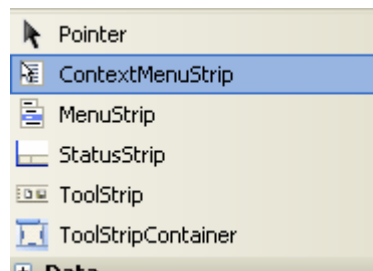
نفذ البرنامج و اختبر القوائم



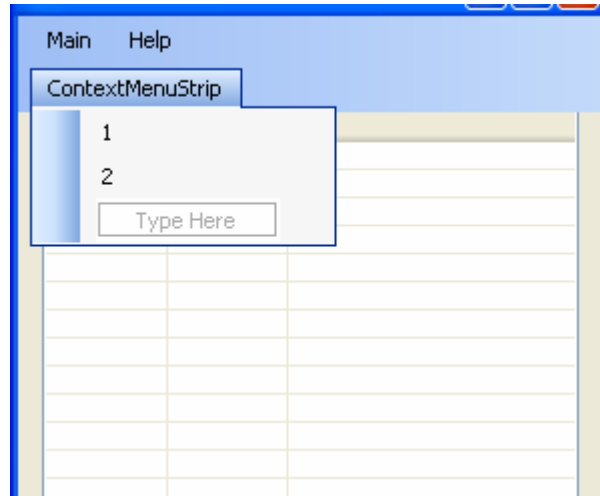
Name	ID	Started
process 1	0	1111

إضافة Context Menu Strip

من أل toolbox انقر نقرة مزدوجة على ContextMenuStrip

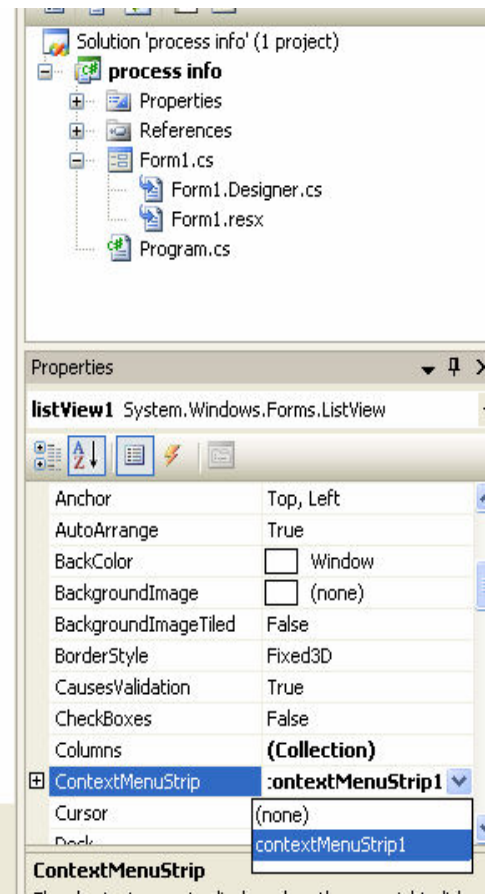
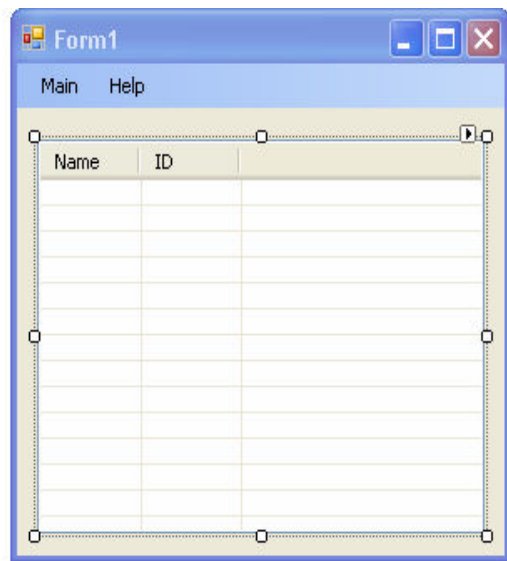


ثم أضف العناصر التي تريد ،



و لربط هذه أل قائمة مع أداة معينة ، مثلا مع أل list view

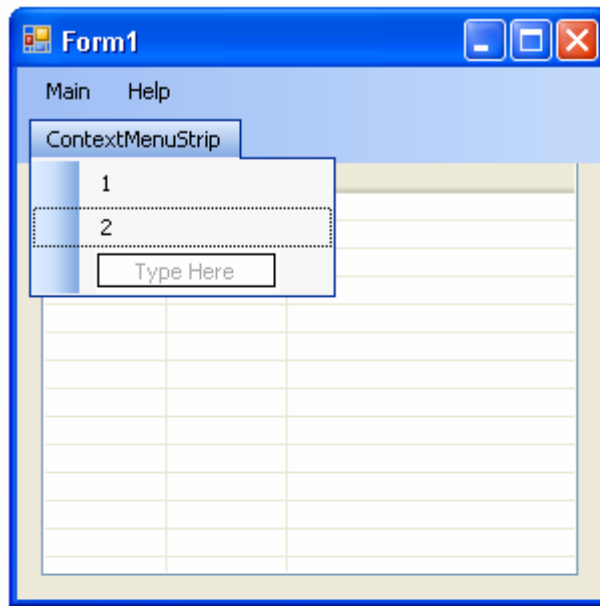
حدد الأداة و من قائمة الخصائص اذهب إلى الخاصية ContextMenuStrip و اختار اسم القائمة التي أضفتها



و الآن نفذ البرنامج ،
وعند التنفيذ اشر بالماوس على أل list view ثم انقر بالزر الأيمن للماوس ولاحظ النتيجة .

و لإضافة كود إلى هذه القائمة ،

نفس الأمر في القائمة العادية ، اضغط double click على العنصر واكتب الكود الذي تريد .



و انقر نقرة مزدوجة ،
مرة على 1
و مرة أخرى على 2

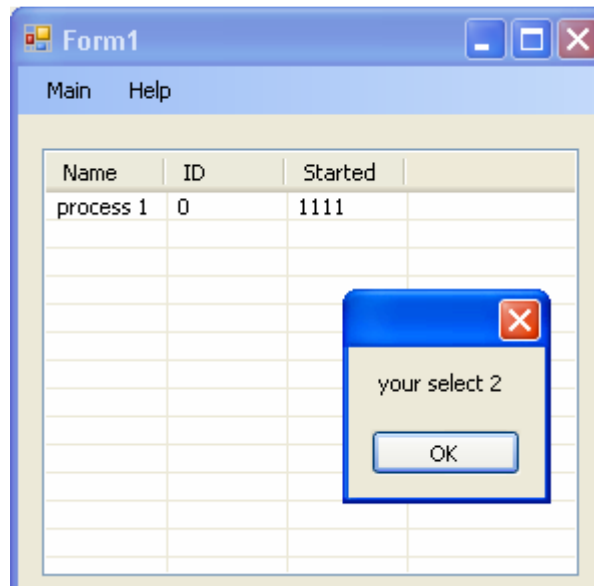


و في جزء الكود مثلاً أضف

```
private void toolStripMenuItem2_Click(object sender, EventArgs e)
{
    MessageBox.Show("your select 1| ");
}

private void toolStripMenuItem3_Click(object sender, EventArgs e)
{
    MessageBox.Show("your select 2 ");
}
```

و نفذ



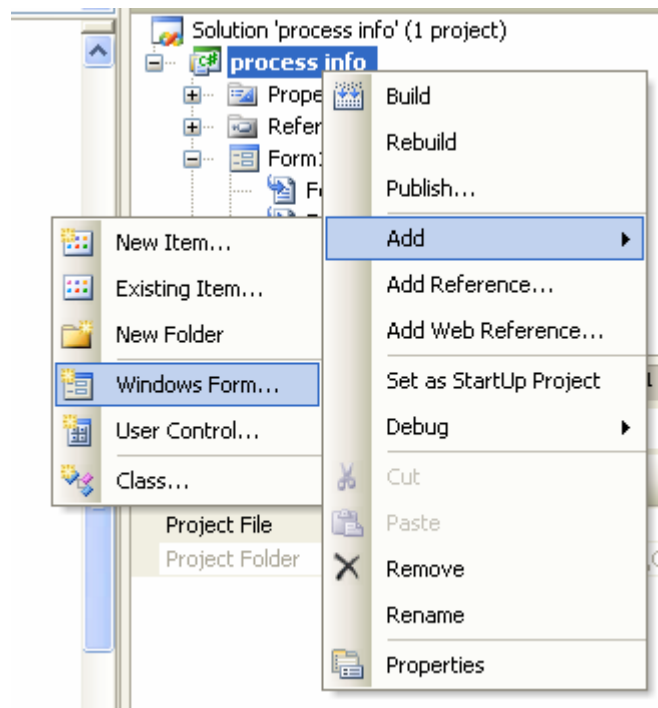
* * * * *

إضافة form جديد

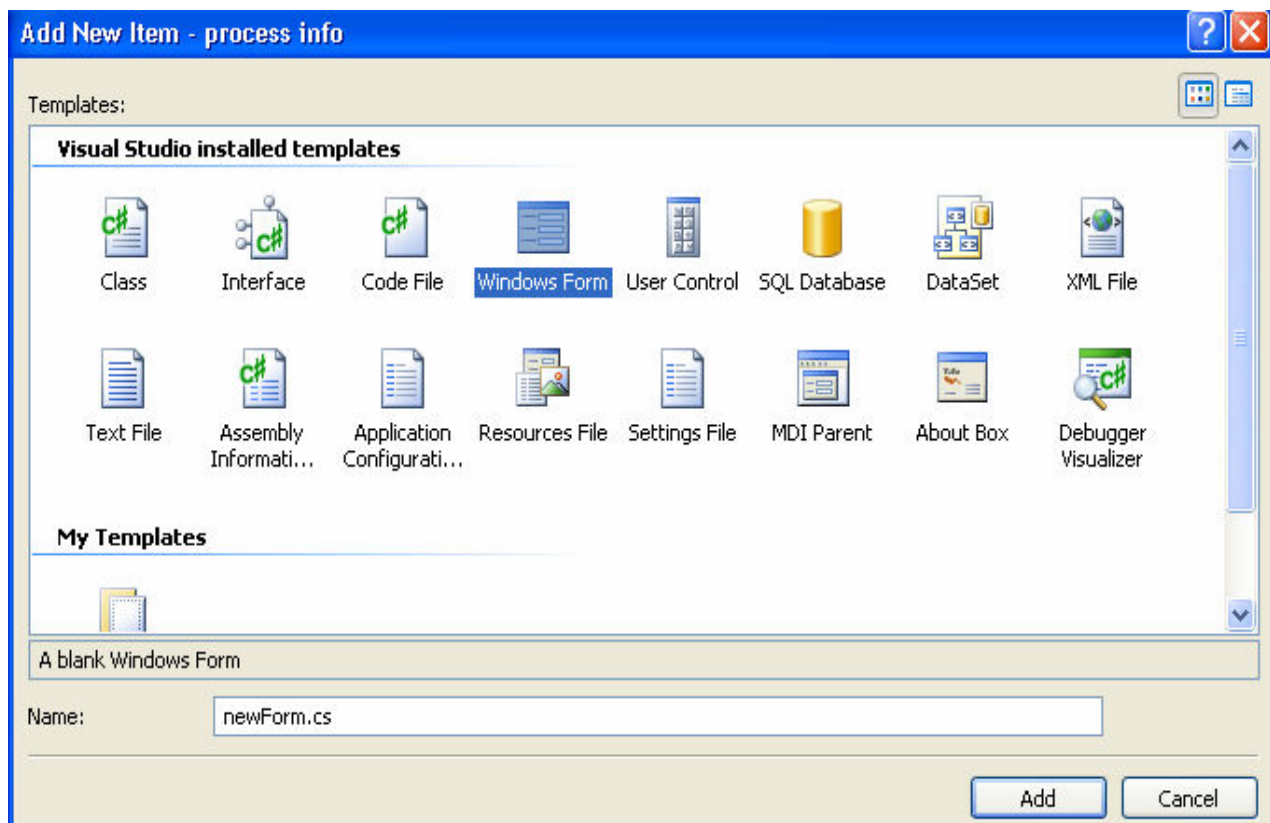
من نافذة أل solution explorer اضغط على اسم المشروع بالزر الأيمن للماوس واختر

Add → Windows Form

كما في الصورة

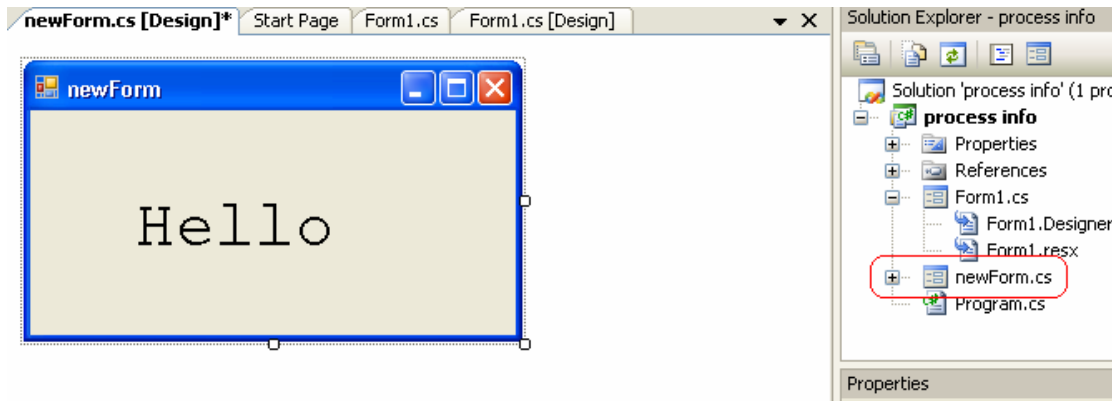


سيظهر لك dialog اكتب فيه اسم ال form الجديد و ليكن newForm

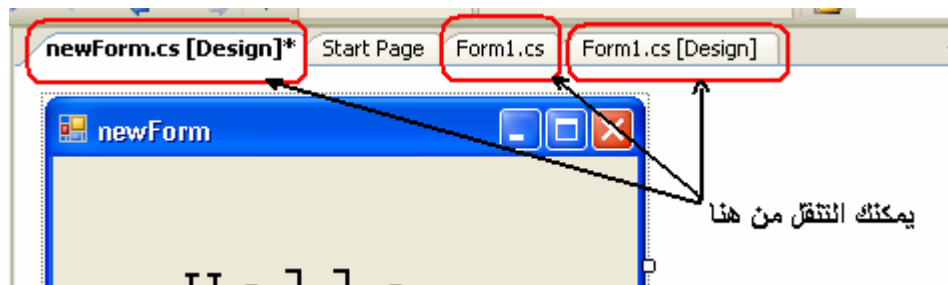


ثم اضغط على زر Add

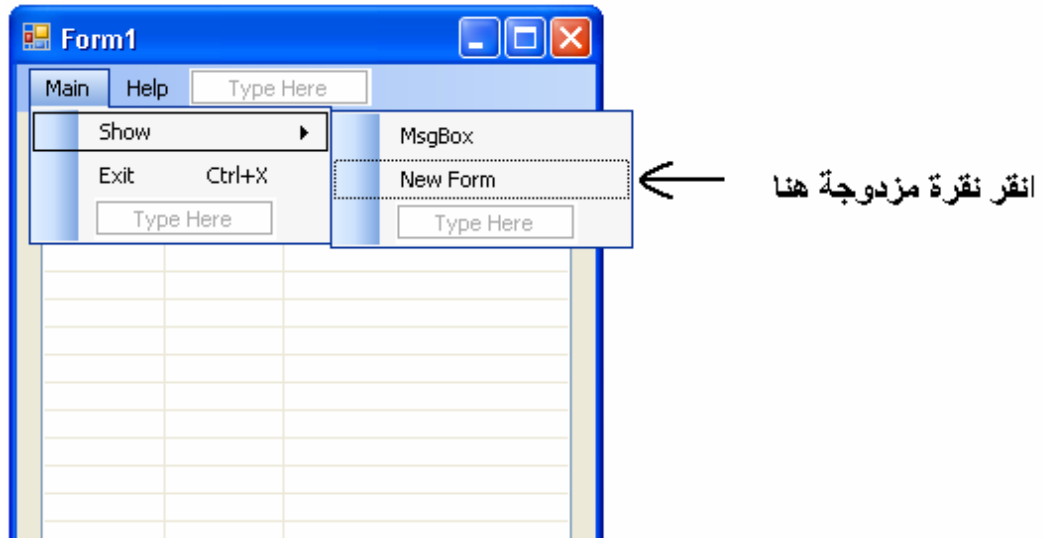
سيظهر لك form جديد ،
أضف له label و غير خاصية أل Text لل label إلى Hello



الان انتقل إلى أل form الأصلي



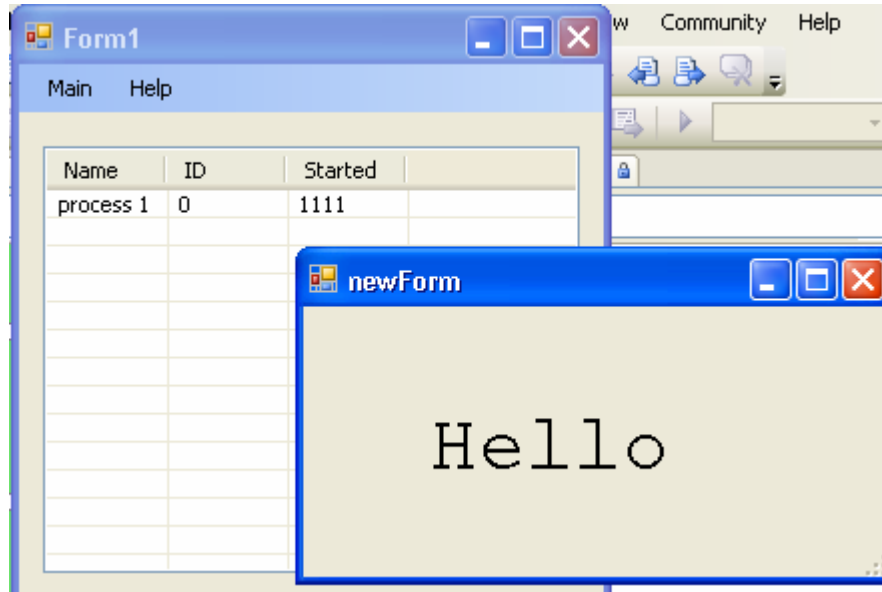
و اضغط كما في الصورة



و لاستدعاء أل form الجديد ، أضف الكود التالي

```
private void newFormToolStripMenuItem_Click(object sender, EventArgs e)
{
    newForm obj = new newForm();
    obj.ShowDialog();
}
```

نفذ الان و لاحظ النتيجة بعد اختيار main → show → New Form .

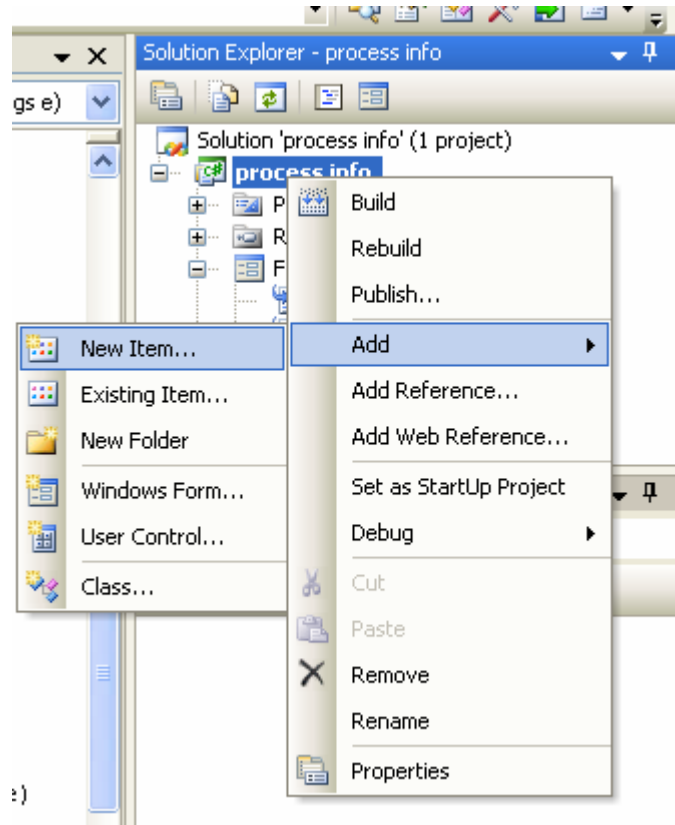


إضافة About box

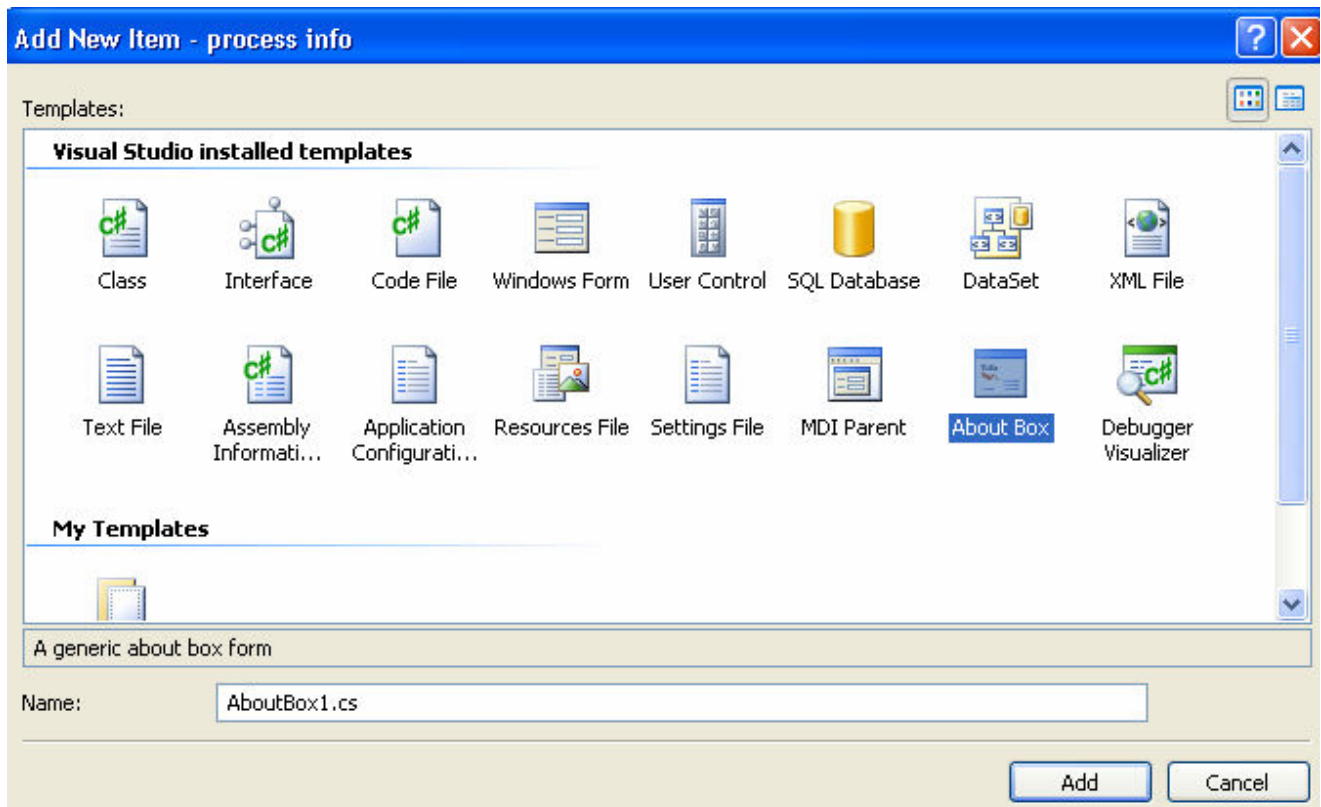
من نافذة أل solution explorer
اضغط على اسم المشروع بالزر الأيمن للماوس واختار

Add → New Item

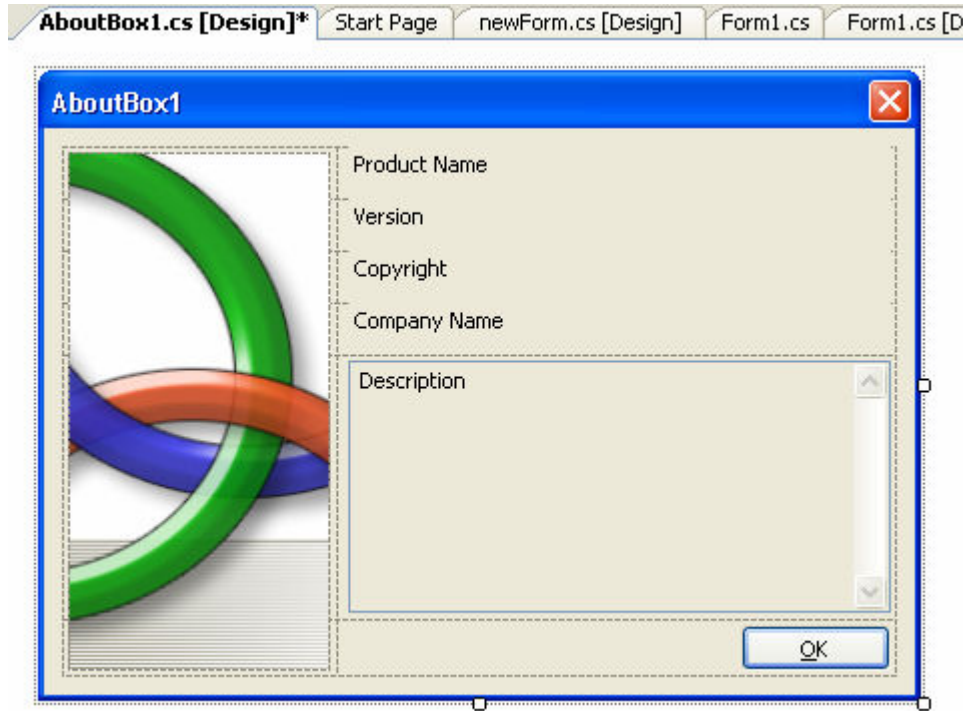
كما في الصورة



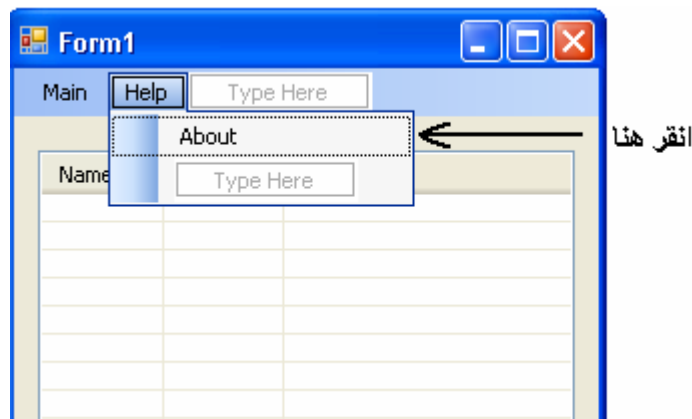
و من أل dialog الذي سيظهر ، اختر About box و اكتب الاسم الذي تريد ثم اضغط على Add



سيظهر لك Form جديد (كما في الصورة التالية) غير خصائصه كما تنشأ من قائمة الخصائص



و لاستدعاء هذا أل dialog ،
انتقل إلى أل form الأصلي ،
وانقر نقرة مزدوجة على زر About الذي أضفناه إلى القائمة لتنتقل إلى جزء الكود الخاص به ،



أضف هذا الكود

```
private void aboutToolStripMenuItem_Click(object sender, EventArgs e)
{
    AboutBox1 abt = new AboutBox1();
    abt.Show();
}
```

و نفذ

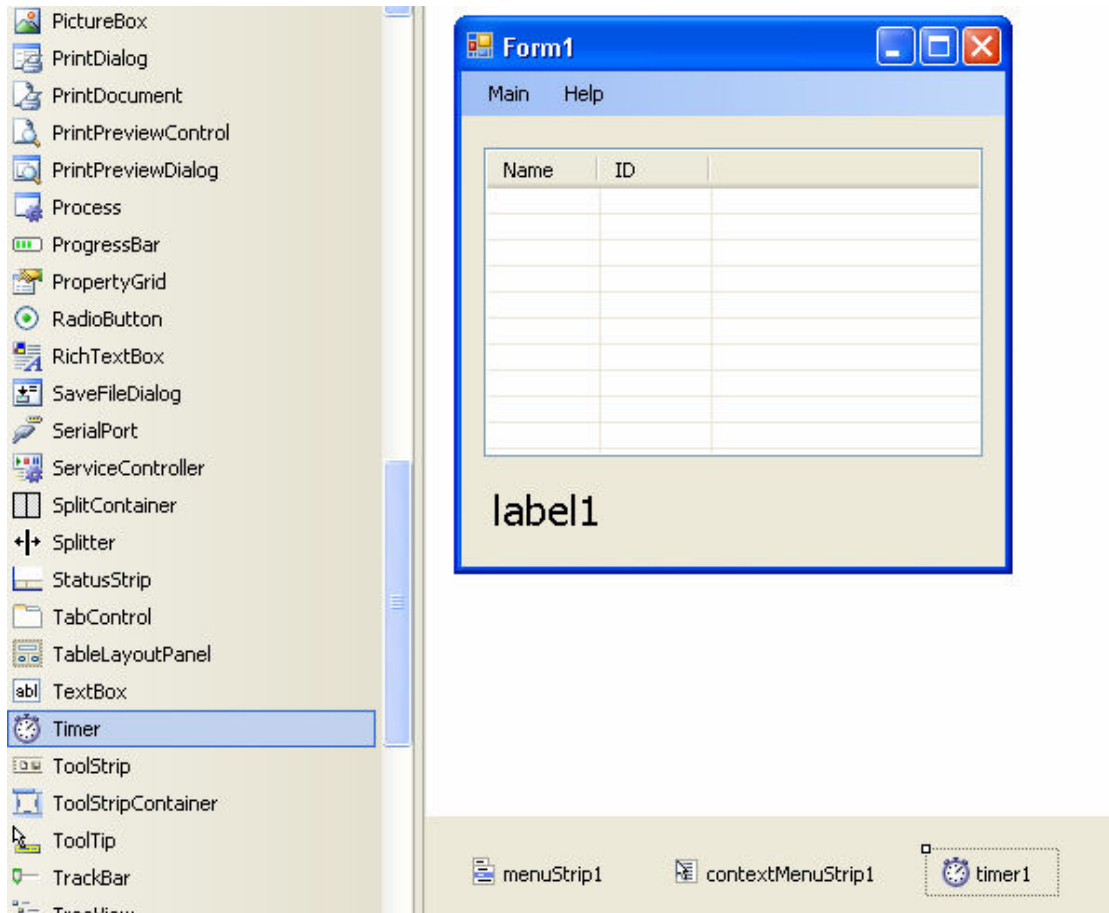
إضافة Timer

أضف إلى واجهة البرنامج الأساسية Label جديد وغير خصائصه كالتالي

AutoSize = False

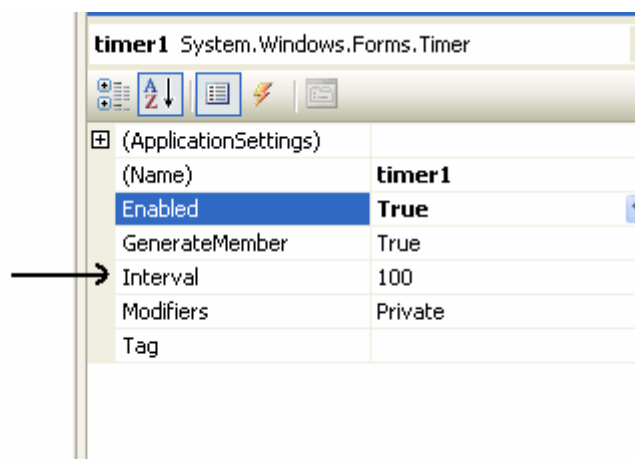
Font → size = 16

ثم أضف أداة أل Timer كما في الصورة

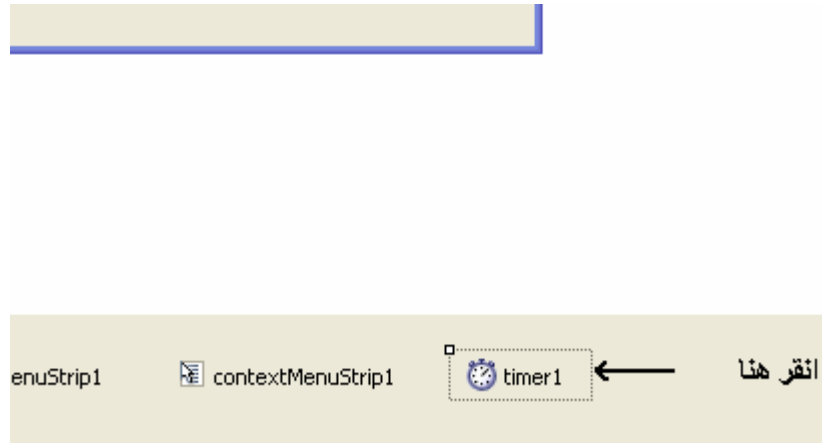


و غير خاصية Enable لل Timer إلى True

المدة التي سيتكرر
عندها المؤقت



الان انقر نقرة على المؤقت لتنتقل إلى جزء الكود الخاص به

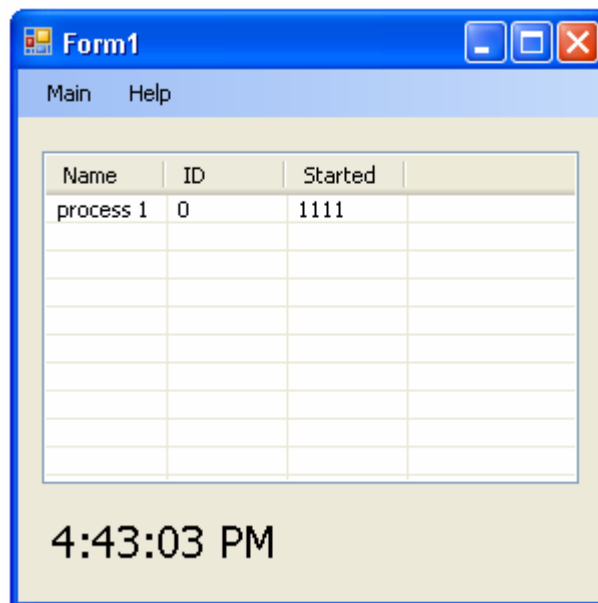


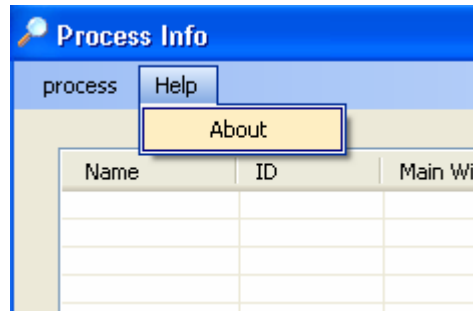
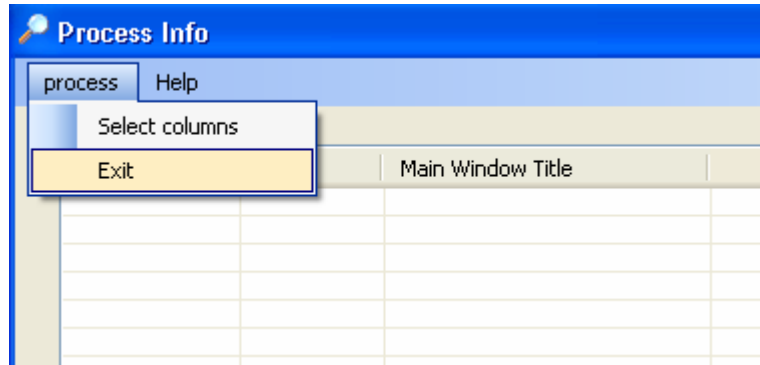
و أضف هذا الكود

```
private void timer1_Tick(object sender, EventArgs e)
{
    label1.Text = DateTime.Now.ToLongTimeString();
}
```

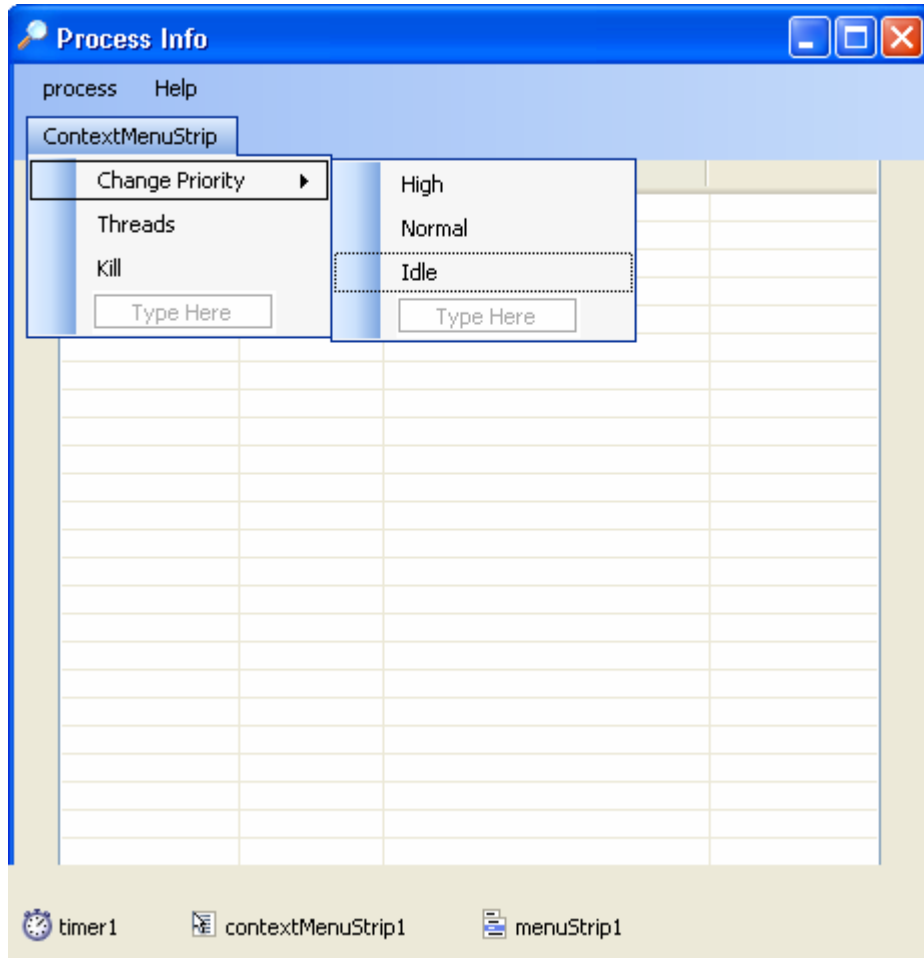
كود لإظهار الوقت

الان نفذ و لاحظ النتيجة



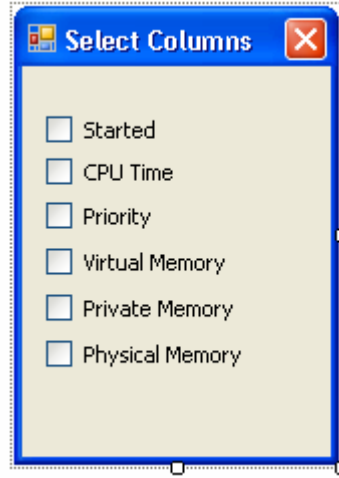


قم بإضافة contextmenustrip و أضف إليها العناصر كما في الصورة التالية



و أربطها مع الأداة ListView كما تعلمت سابقا ،، وإذا نسيت ارجع إلى الصفحة 16

الان قم بإضافة Form جديد وليكن أسمه مثلا SelectColumn ،
حيث عندما يضغط المستخدم خيار select column من القائمة يظهر له هذا أل Form ليحدد منه الأعمدة
التي يريد أن تظهر ،
و غير شكل أل Form الجديد كما في الصورة

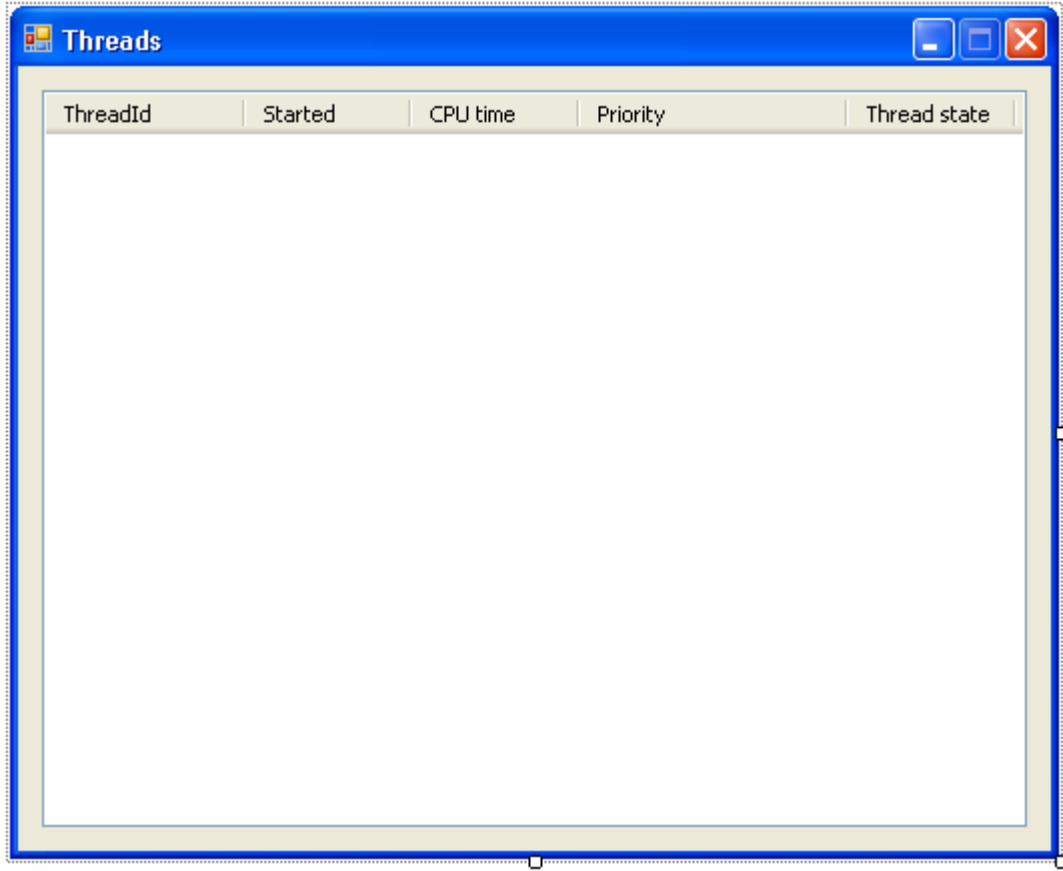


أي أضف له الأداة checkbox و غير خاصية أل text لكل checkbox لتتنطبق مع الصورة التي بالأعلى .

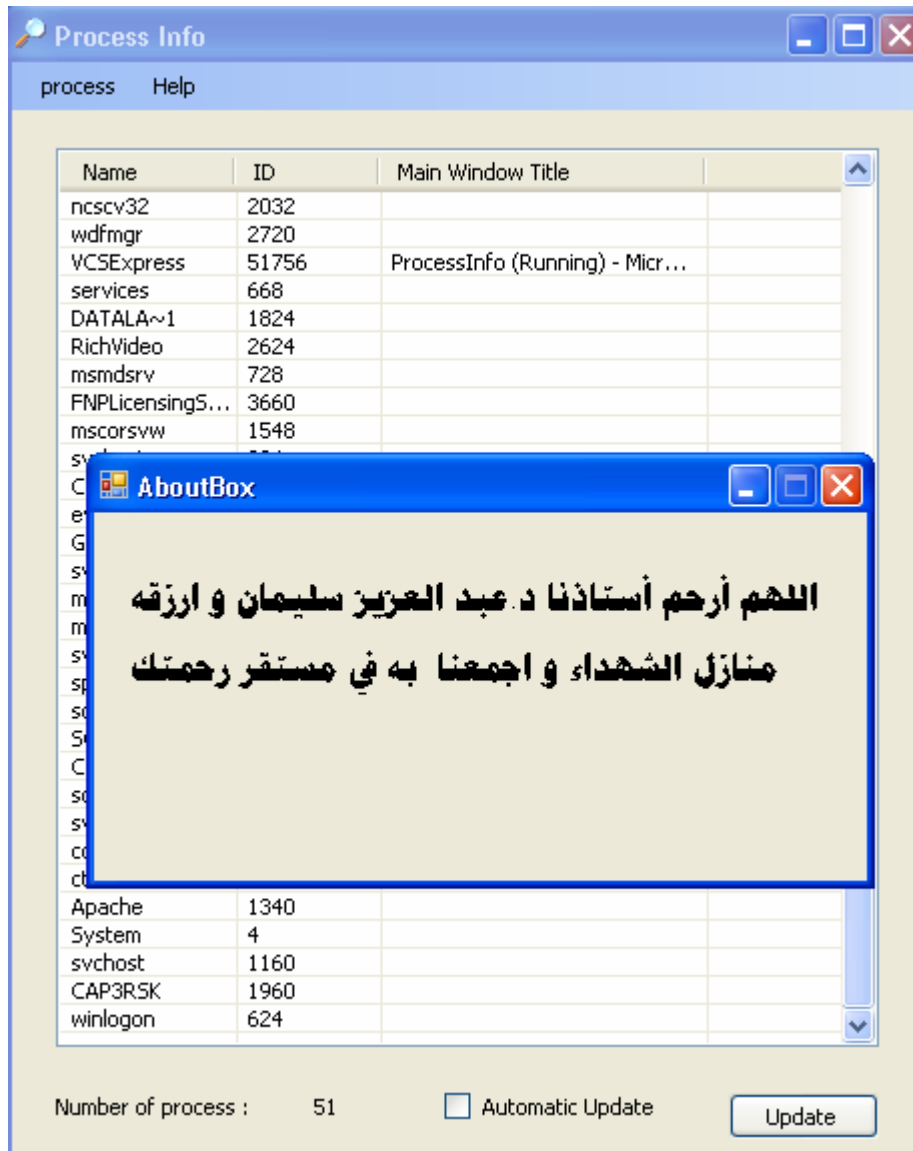
أضف Form آخر و سمه Threads ، هذا أل Form سيظهر عندما يختار المستخدم process معينة و يضغط بالزر الأيمن للماوس و يختار من أل contextmenu خيار Thread حيث سيظهر جميع أل thread الموجودة ضمن هذه أل Process .

أضف إلى أل Form الجديد الأداة ListView و غير خصائصها لتصبح كما في الشكل التالي

اللهم ارحم شهيدنا الأستاذ د. عبد العزيز سليمان وأسكنه فسيح جناتك



أيضا أضف About Box و غيره كما يحلو لك



الآن ارجع إلى أل Form الأصلي و أضف له Button و غير خصائصه كالتالي

Text = Update
Name = updateButton
Anchor = Bottom, Right

و أيضا أضف checkbox و غير خصائصه

Text = Automatic Update
Anchor = Bottom, Right

و أضف Label و غير خصائصه إلى

Text = Number of process :
Anchor = Bottom, Right

بحيث يكون الترتيب النهائي كما في الصورة

msmsgs	148	
lsass	680	
dexplore	52152	
NMBgMonitor	2012	
ProcessInfo.v...	61132	
ncscv32	2032	
wdfmgr	2720	
VCSEExpress	51756	ProcessInfo - Microsoft Visua...
services	668	
DATA~1	1824	

Number of process : 54 ☐ Automatic Update Update

إضافة الكود اللازم (الكود التابع لل Form الأصلي)

سنحتاج إلى إضافة متغيرات نعتبرها ك flag لمعرفة الأعمدة التي سيحددها المستخدم ،
و بما أن الأعمدة موجودة في form آخر إذا علينا أن نجعل هذه المتغيرات مشتركة بين ال two form
(أي بين two class لأن كل form له class خاص به) و لعمل ذلك فقط سنجعل
المتغيرات من نوع static .
هذا هو الكود كاملا لملف ال form الرئيسي
مع إضافة بعض التعليقات عليه

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

namespace ProcessInfo
{
    public partial class Form1 : Form
```

```
{
//
    public string name = "";
//
    هذا المتغير لخرن ال threads في process معينة
    public static ProcessThreadCollection myThread;
//
    المتغيرات التالية للتحكم بعدد الأعمدة التي ستظهر في ال list view
    public static bool m_staeted = false, m_cputime = false,
        m_priority = false, m_memory = false, m_priMemory = false,
        m_physMemory = false;
    public static int m_staetedFlag = 0, m_cputimeFlag = 0,
        m_priorityFlag = 0, m_memoryFlag = 0, m_priMemoryFlag = 0,
        m_physMemoryFlag = 0;
    ColumnHeader header2, header3,
        header4, header5, header6, header7;
    private ListViewItem file;
    public Form1()
    {
        InitializeComponent();
        header2 = new ColumnHeader();
        header3 = new ColumnHeader();
        header4 = new ColumnHeader();
        header5 = new ColumnHeader();
        header6 = new ColumnHeader();
        header7 = new ColumnHeader();
        header2.Text = "Started";
        header3.Text = "CPU Time";
        header4.Text = "Priority";
        header5.Text = "Memory";
        header6.Text = "Private Memory";
        header7.Text = "Physical Memory";
    }
}
```

/* هذه الدالة المسؤولة عن عملية حساب ال processes الشغالة ، سيتم استدعاء هذه الدالة عندما يضغط المستخدم على زر Update و ايضا عند عمل ال Timer */

*/

```
public void Update_Func()
{
    int index = -1;

    listView1.Items.Clear();

    try
    {
        Process[] allProcs = Process.GetProcesses();
        label2.Text = allProcs.Length.ToString();
    }
    catch
    {
        // قبل اضافة أي عمود ، نفحص هل ان العمود مؤشر من ال Form الذي يظهر اسماء الاعمدة (الفورم الثاني)
        // ونتأكد ايضا انه غير مضاف في مرة سابقة ، اذا تحقق الشرطان ، نضيف العمود
    }

    if ((Form1.m_staeted) && (Form1.m_staetedFlag == 0))
    {
        listView1.Columns.Add(header2);
        Form1.m_staetedFlag = 1;
    }

    if ((!Form1.m_staeted) && (Form1.m_staetedFlag == 1))
    {
        listView1.Columns.Remove(header2);
        Form1.m_staetedFlag = 0;
    }

    if ((Form1.m_cputime) && (Form1.m_cputimeFlag == 0))
    {
        listView1.Columns.Add(header3);
        Form1.m_cputimeFlag = 1;
    }

    if ((!Form1.m_cputime) && (Form1.m_cputimeFlag == 1))
    {
        listView1.Columns.Remove(header3);
        Form1.m_cputimeFlag = 0;
    }

    if ((Form1.m_priority) && (Form1.m_priorityFlag == 0))
    {
        listView1.Columns.Add(header4);
        Form1.m_priorityFlag = 1;
    }

    if ((!Form1.m_priority) && (Form1.m_priorityFlag == 1))
    {

```

```
{
    listView1.Columns.Remove(header4);
    Form1.m_priorityFlag = 0;
}

if ((Form1.m_memory) && (Form1.m_memoryFlag == 0))
{
    listView1.Columns.Add(header5);
    Form1.m_memoryFlag = 1;
}

if ((!Form1.m_memory) && (Form1.m_memoryFlag == 1))
{
    listView1.Columns.Remove(header5);
    Form1.m_memoryFlag = 1;
}

if ((Form1.m_priMemory) && (Form1.m_priMemoryFlag == 0))
{
    listView1.Columns.Add(header6);
    Form1.m_priMemoryFlag = 1;
}

if ((!Form1.m_priMemory) && (Form1.m_priMemoryFlag == 1))
{
    listView1.Columns.Remove(header6);
    Form1.m_priMemoryFlag = 0;
}

if ((Form1.m_physMemory) && (Form1.m_physMemoryFlag == 0))
{
    listView1.Columns.Add(header7);
    Form1.m_physMemoryFlag = 1;
}

if ((!Form1.m_physMemory) && (Form1.m_physMemoryFlag == 1))
{
    listView1.Columns.Remove(header7);
    Form1.m_physMemoryFlag = 0;
}

foreach (Process thisProc in allProcs)
{
    index++;
}
```

```
string procName = thisProc.ProcessName;

DateTime started = thisProc.StartTime;
int procID = thisProc.Id;
long memory = thisProc.VirtualMemorySize64;
long priMemory = thisProc.PrivateMemorySize64;
long physMemory = thisProc.WorkingSet64;

ProcessPriorityClass priClass = thisProc.PriorityClass;
TimeSpan cpuTime = thisProc.TotalProcessorTime;

file = new ListViewItem(procName);
file.SubItems.Add(procID.ToString());
file.SubItems.Add(thisProc.MainWindowTitle);
if (Form1.m_staeted)
    file.SubItems.Add(started.ToString());
if (Form1.m_cputime)
    file.SubItems.Add(cpuTime.ToString());
if (Form1.m_priority)
    file.SubItems.Add(priClass.ToString());
if (Form1.m_memory)
    file.SubItems.Add(memory.ToString());
if (Form1.m_priMemory)
    file.SubItems.Add(priMemory.ToString());
if (Form1.m_physMemory)
    file.SubItems.Add(physMemory.ToString());

listView1.Items.Add(file);

if (name == procName)
{
    listView1.Focus();
    listView1.TopItem = listView1.Items[index];
    listView1.Items[index].Selected = true;
}

}
}
catch
{}

}
```

```
private void timer1_Tick(object sender, EventArgs e)
{
    Update_Func();
}

private void checkBox1_CheckedChanged(object sender, EventArgs e)
{
    /*
        Update checkbox سيتم تشغيل ال timer و سيتم تعطيل زر Update
    */
    if (checkBox1.Checked)
    {
        updateButton.Enabled = false; // تعطيل زر التحديث
        timer1.Enabled = true; // تشغيل المؤقت
    }

    else
    {
        updateButton.Enabled = true;
        timer1.Enabled = false;
    }

}

private void exitToolStripMenuItem_Click(object sender, EventArgs e)
{
    Environment.Exit(0);
}

private void selectColumnToolStripMenuItem1_Click(object sender, EventArgs e)
{
    threads thread = new threads();
    thread.ShowDialog();
}

private void updateButton_Click(object sender, EventArgs e)
{
    Update_Func();
}

private void Kill_Click(object sender, EventArgs e)
{
    try
    {

```

```
if (listView1.SelectedItems.Count > 0)
{
    Process.GetProcessesByName(
        listView1.SelectedItems[0].Text)[0].Kill();
    listView1.Items.Remove(listView1.SelectedItems[0]);
}
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message);
}
}

private void listView1_MouseClick(object sender, MouseEventArgs e)
{
    if (e.Button == MouseButtons.Right)
    {
        if (listView1.SelectedIndices.Count > 0)
            listView1.ContextMenuStrip = contextMenuStrip1;
    }
}

private void threads_Click(object sender, EventArgs e)
{
    try
    {
        if (listView1.SelectedItems.Count > 0)
        {
            myThread =
                Process.GetProcessesByName
                    (listView1.SelectedItems[0].Text)[0].Threads;

            Threads0 thread = new Threads0();
            thread.ShowDialog();
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
```

```
}

private void highToolStripMenuItem_Click(object sender, EventArgs e)
{
    //                تغيير الاسبقية ل process معينة الى high
    try
    {
        if (listView1.SelectedItems.Count > 0)
        {
            Process.GetProcessesByName(
                listView1.SelectedItems[0].Text)[0].PriorityClass
                = ProcessPriorityClass.High;
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}

private void normalToolStripMenuItem_Click(object sender, EventArgs e)
{
    //                تغيير الاسبقية ل process معينة الى Normal
    try
    {
        if (listView1.SelectedItems.Count > 0)
        {
            Process.GetProcessesByName(
                listView1.SelectedItems[0].Text)[0].PriorityClass
                = ProcessPriorityClass.Normal;
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
```

```
private void idleToolStripMenuItem_Click(object sender, EventArgs e)
{
    //          تغيير الاسبقية ل process معينة الى Idle

    try
    {
        if (listView1.SelectedItems.Count > 0)
        {
            Process.GetProcessesByName(
                listView1.SelectedItems[0].Text)[0].PriorityClass = ProcessPriorityClass.Idle;
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}

private void aboutToolStripMenuItem_Click(object sender, EventArgs e)
{
    AboutBox abt = new AboutBox();
    abt.Show();
}

private void listView1_Click(object sender, EventArgs e)
{
    name =listView1.SelectedItems[0].Text.ToString() ;
}
}
}
```

الكود الخاص بال form الثاني (select column)

كل ما علينا في هذا الجزء ، انه عندما يؤشر المستخدم على column معين نضع القيمة 1 في أل flag الخاص بذلك العمود و إذا حذف التأشير نضع القيمة 0 (هذا أل flag يتم فحصه عند إضافة الأعمدة في الكود الخاص بال form الأول)

أيضا في حدث أل load نقوم بفحص أل flags الخاصة بكل عمود و ذلك لإعادة تهيئة أل Form (وضع التأشير على أل checkbox بالصورة الصحيحة .

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace ProcessInfo
{
    public partial class threads : Form
    {
        public threads()
        {
            InitializeComponent();
        }

        private void startedcheckBox_CheckedChanged(object sender, EventArgs e)
        {
            if (startedcheckBox.Checked)
            {
                Form1.m_staeted = true;
            }

            else
            {
                Form1.m_staeted = false;
            }
        }

        private void threads_Load(object sender, EventArgs e)
        {
            if( Form1.m_staeted == true )
                startedcheckBox.Checked = true ;
            else
                startedcheckBox.Checked =false ;
        }
    }
}
```

```
if (Form1.m_cputime == true)
    cputimecheckBox.Checked = true;
else
    cputimecheckBox.Checked = false;

if (Form1.m_priority == true)
    prioritycheckBox.Checked = true;
else
    prioritycheckBox.Checked = false;

if (Form1.m_memory == true)
    virMemcheckBox.Checked = true;
else
    virMemcheckBox.Checked = false;

if (Form1.m_priMemory == true)
    privMemcheckBox.Checked = true;
else
    privMemcheckBox.Checked = false;

if (Form1.m_physMemory == true)
    physMemcheckBox.Checked = true;
else
    physMemcheckBox.Checked = false;

}

private void cputimecheckBox_CheckedChanged(object sender, EventArgs e)
{
    if (cputimecheckBox.Checked)
    {
        Form1.m_cputime = true;
    }
    else
    {
        Form1.m_cputime = false;
    }
}

private void prioritycheckBox_CheckedChanged(object sender, EventArgs e)
{
    if (prioritycheckBox.Checked)
    {
        Form1.m_priority = true;
    }
}
```

```
else
{
    Form1.m_priority = false;
}
}

private void virMemcheckBox_CheckedChanged(object sender, EventArgs e)
{
    if (virMemcheckBox.Checked)
    {
        Form1.m_memory = true;
    }
    else
    {
        Form1.m_memory = false;
    }
}

private void privMemcheckBox_CheckedChanged(object sender, EventArgs e)
{
    if (privMemcheckBox.Checked)
    {
        Form1.m_priMemory = true;
    }
    else
    {
        Form1.m_priMemory = false;
    }
}

private void physMemcheckBox_CheckedChanged(object sender, EventArgs e)
{
    if (physMemcheckBox.Checked)
    {
        Form1.m_physMemory = true;
    }

    else
    {
        Form1.m_physMemory = false;
    }
}
```

```
}  
}  
}
```

الكود الخاص بال Form الثالث (Threads)

هذا الكود فقط يقوم بعرض جميع أَل Threads (المخزونة في متغير myThread و الذي تم تهيئته في الكود الخاص بال form الأول) في listBox عن طريق جملة foreach

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Text;  
using System.Windows.Forms;  
using System.Diagnostics;  
  
namespace ProcessInfo  
{  
    public partial class Threads0 : Form  
    {  
  
        public Threads0()  
        {  
            InitializeComponent();  
        }  
  
        private void Threads0_Load(object sender, EventArgs e)  
        {  
            try  
            {  
  
                foreach (ProcessThread pt in Form1.myThread )  
                {  
                    DateTime startTime = pt.StartTime;  
                    TimeSpan cpuTime = pt.TotalProcessorTime;  
                    int priority = pt.BasePriority;  
                    ThreadState ts = pt.ThreadState;
```

```
ListViewItem file = new ListViewItem(pt.Id.ToString());
file.SubItems.Add(startTime.ToString());
file.SubItems.Add(cpuTime.ToString());
file.SubItems.Add(priority.ToString());
file.SubItems.Add(ts.ToString());

listView1.Items.Add(file);

    }
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message);
}

}
}
}
```

إذا أردت عمل setup للمشروع يمكنك ذلك ببساطة ،
من قائمة Build اختر publish ثم تتبع أل wizard .

