



الجمهورية اليمنية
جامعة الحديدة
كلية التربية
معلم حاسوب

010010

البرمجة بلغة السي شارب

C#



0100

إعداد

أ/ أحمد القاسمي

للعام الجامعي 2011-2012

010010

الإهداء

إلى من أوصلونا إلى هذه الدروب

إلى من مدوا إلينا يد العطاء بالحب والحنان

إلى أمهاتنا وآبائنا وإخواننا

إلى كل يد شريفة بنت وعمرت أرض اليمن السعيدة

إلى صرح العلم الشامخ **جامعة الحديدة**

إلى منارة العلم **كلية التربية**

إلى لؤلؤة العلوم **قسم معلم الحاسوب**

إلى كل من صحح الخطأ وقوم السلوك

إلى من بذل كل الجهود

إلى كل من أرشدنا إلى الدروب السليمة .

خصائص النموذج (Form)

قم بإنشاء نموذج واجعل النموذج يظهر كالتالي

أولاً: بالعنوان "البرنامج الأول"، وسط الشاشة، وخلفية زرقاء وبدون أزرار تحكم.

ثانياً: أظهر النموذج ملء الشاشة بدون إطارات.

يدوياً: أنشئ نموذج (form).

قم بضبط خصائص النموذج التالية:

الوصف	القيمة	الخاصية
الاسم البرمجي الذي سنستخدمه أثناء البرمجة	Frm	Name
النص الذي سيظهر كعنوان للنموذج	البرنامج الأول	text
ونحدد منها لون خلفية النموذج		BackColor
تخفي أزرار التحكم (تصغير، تكبير، إغلاق)	false	ControlBox

أضغظ F5 وشاهد التغييرات على النموذج كما في الصورة التالية.



ثم قم بضبط خصائص النموذج التالية:

الوصف	القيمة	الخاصية
تخفي إطارات النموذج	None	FormBorderStyle
تغير خلفية النموذج بالصورة التي تحددها		BackgroundImage
تجعل النموذج يأخذ حجم الصورة	Stretch	BackgroundImageLayout
تجعل النموذج يظهر بحجم الشاشة	Maximized	WindowState

يمكنك إضهار النموذج وسط الشاشة
باستخدام الطريقة: `CenterToScreen();`
أو وسط حدود النموذج الأب بالطريقة
`CenterToParent();`



ثم أضغط shift+f5 لإلغاء التنفيذ.

برمجيا: نقوم بعمل الشفرة داخل حدث تحميل النموذج (form_load) وذلك بالنقر نقرة مزدوجة في مكان من النموذج لتظهر شاشة كتابة الشفرة كما يلي:

```
private void Form1_Load(object sender, EventArgs e)
{
    Text="الأول البرنامج";
    BackColor = Color.Blue;
    ControlBox = false;
    CenterToScreen();
}
```

شاهد التغييرات أولا ثم انه التنفيذ وأضف الأوامر الأخرى.

```
private void Form1_Load(object sender, EventArgs e)
{
    Text="الأول البرنامج";
    BackColor = Color.Blue;
    ControlBox = false;
    FormBorderStyle = FormBorderStyle.None;
    WindowState = FormWindowState.Maximized;
}
```

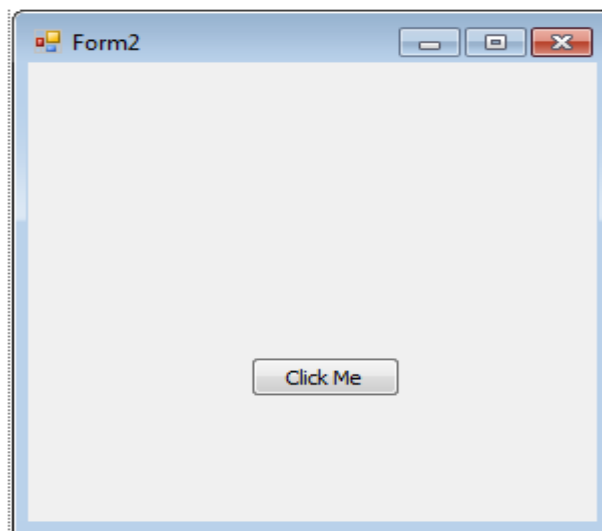
التحكم في النموذج و عناصر التحكم

قم بإنشاء النموذج وأضف عنصر التحكم button1 وأجعل الخاصية Text تأخذ القيمة Click Me كما توضح الصورة التالية:

من خلال الخاصية Size بإمكانك تغيير حجم النموذج أو عناصر التحكم

والخاصية location تمكنك من تحريكها وتغيير موقعها .

بإمكانك عمل ذلك بطريقة يدوية من خلال تغيير قيم الخاصيتين أثناء التصميم أو بطريقة برمجية .



داخل حدث Click للزر button1 أكتب الشفرة التالية ولاحظ النتيجة بعد التنفيذ

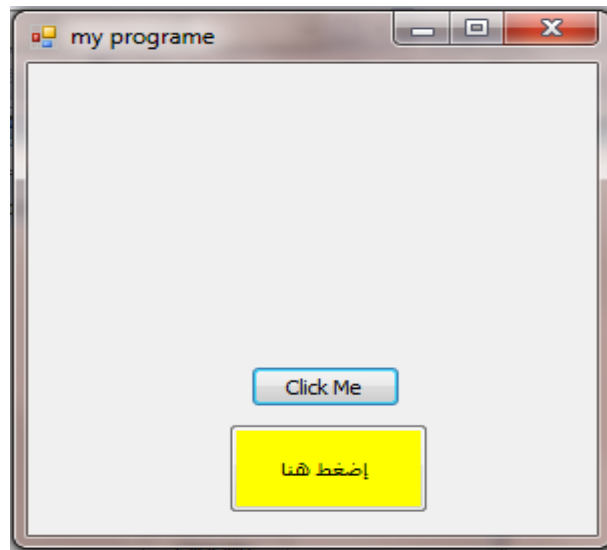
```
private void button1_Click(object sender, EventArgs e)
{
    Size = new Size(500, 500);
    Location = new Point(400, 100);
    button1.Size = new Size(button1.Size.Width + 75,
        button1.Size.Height + 50);
    button1.Location = new Point(100,50);
}
```

إضافة وإزالة عناصر التحكم برمجياً

قم بإنشاء النموذج وأضف عنصر التحكم button1 وأجعل الخاصية Text تأخذ القيمة Click Me .

داخل حدث Click للزر button1 أكتب الشفرة التالية ولاحظ النتيجة بعد التنفيذ

```
private void button1_Click(object sender, EventArgs e)
{
    Button bu1 = new Button();
    bu1.Size = new Size(100, 50);
    bu1.Location = new Point(100, 200);
    bu1.Text = "هنا اضغط";
    bu1.BackColor = Color.Yellow;
    Controls.Add(bu1);
}
```



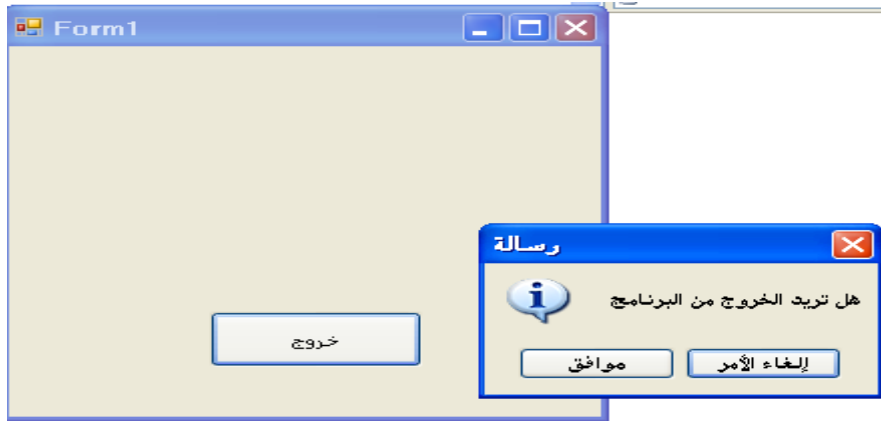
عرض مربعات الرسائل باستخدام الطريقة (MessageBox.Show)

قم بإنشاء نموذج يظهر للمستخدم رسالة "هل تريد إغلاق البرنامج" إذا كانت الإجابة موافق يغلق النموذج وإذا إلغاء الأمر يغلق الرسالة .

- 1 - أضف زر (Button) من قائمة الأدوات واضبط الخاصية (خروج=Text) والخاصية Exite.name.
- 2 ثم أنقر نقرة مزدوجة على الزر واكتب الشفرة كما يلي:

```
private void Exite_Click(object sender, EventArgs e)
{
```

```
DialogResult re;
re = MessageBox.Show("البرنامج من الخروج تريد هل", "رسالة",
    MessageBoxButtons.OKCancel, MessageBoxIcon.Information,
    MessageBoxDefaultButton.Button2);
if (re == DialogResult.OK)
    System.Environment.Exit(0);
```



التدريب 1: < (اسم المستخدم وكلمة المرور)

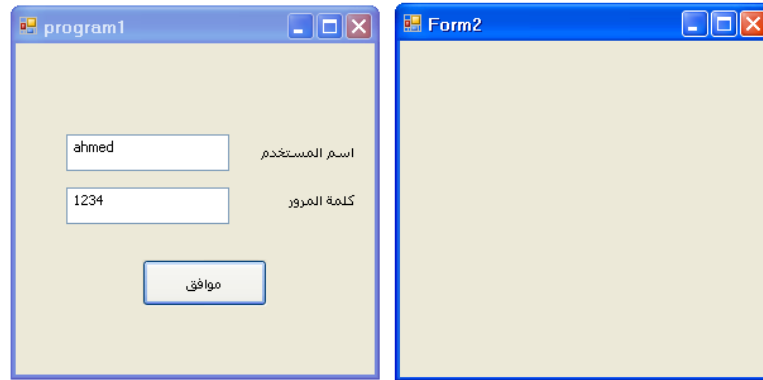
قم بإنشاء نموذج وأضف التالي:

- 2 مربع نص (TextBox) واجعل الخاصية Text فارغة.
- (label) واجعل الخاصية "اسم المستخدم" Text="اسم المستخدم" و (label) آخر واجعل الخاصية "كلمة المرور" Text="كلمة المرور".
- زر (Button) واجعل الخاصية "موافق" Text="موافق".

عند الضغط على الزر موافق وكانت اسم المستخدم وكلمة المرور صحيحة يتم الانتقال لنموذج آخر وإذا كانت غير صحيحة يعطيك ثلاث محاولات فقط ثم يغلق البرنامج.
إذا كانت اسم المستخدم وكلمة المرور غير صحيحة.



إذا كان اسم المستخدم وكلمة المرور صحيحة

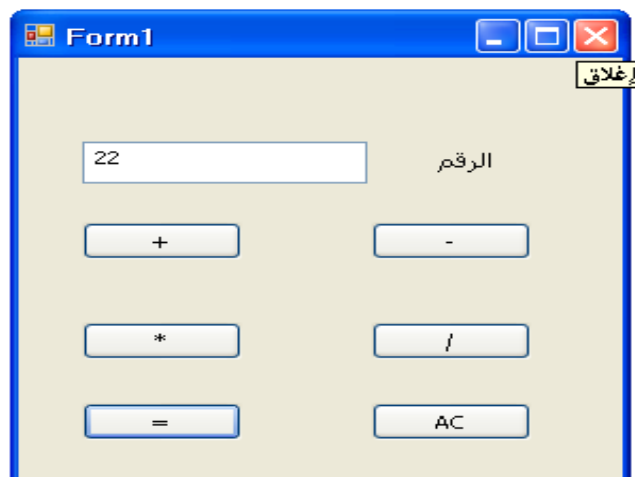


```
public partial class Form1 : Form
{
    string x = "ahmed", y = "1234";
    int i=1;
    Form2 OB = new Form2();

    private void button1_Click(object sender, EventArgs e)
    {
        if (i <= 3)
        if (textBox1.Text == x && textBox2.Text == y)
        OB.Show();
        else
        {
            i++;
            MessageBox.Show("أخرى مرة حاول", "رسالة");
        }
        else
        Close();
    }
}
```

التدريب 2: <آلة حاسبة>

أنشئ نموذج وأضف إليه الأدوات كما توضح الصورة وغير الخصائص مثلما تعلمت سابقا.



```

public partial class Form1 : Form
{ // تعريف المتغيرات
double x, y, z; int i;
//+ شفرة الزر
private void SubBtn_Click(object sender, EventArgs e)
{
x = double.Parse(textBox1.Text);
textBox1.Text = "";
i = 1;
}
// شفرة الزر يساوي
private void EqBtn_Click(object sender, EventArgs e)
{
y = double.Parse(textBox1.Text);
if (i == 1)
z = x + y;
else if (i == 2)
z = x - y;
else if (i == 3)
z = x * y;
else
z=x/y;

textBox1.Text =z.ToString() ;
}
// شفرة الزر -
private void MinBtn_Click(object sender, EventArgs e)
{
x = double.Parse(textBox1.Text);
textBox1.Text = "";
i = 2;
}
// شفرة الزر *
private void ProBtn_Click(object sender, EventArgs e)
{
x = double.Parse(textBox1.Text);
textBox1.Text = "";
i = 3;
}
// شفرة الزر /
private void DifBtn_Click(object sender, EventArgs e)
{
x = double.Parse(textBox1.Text);
textBox1.Text = "";
i = 4;
}
// شفرة الزر AC
private void button6_Click(object sender, EventArgs e)
{
textBox1.Text = "";
}}

```


تمكين النموذج وعناصر التحكم وإلغاء تمكينها

الخاصية `Enable` هي الخاصية المسؤولة عن تمكين العناصر وعدم تمكينها وتحمل القيمة `true or false` عند إلغاء تمكين النموذج فإن كل العناصر على النموذج يلغى تمكينها، بالعكس بالنسبة لعناصر التحكم فإن العنصر الذي نلغى تمكينه يلغى تمكينه هو فقط.

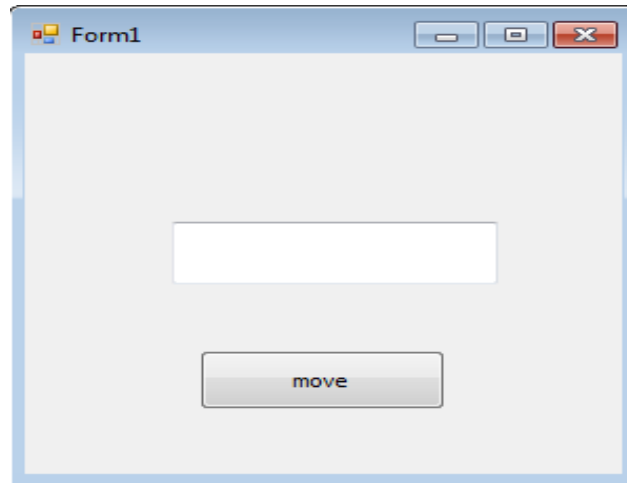
الخاصية `visible` وتحمل القيمة `true or false` هي الخاصية المسؤولة عن إظهار وإخفاء عناصر التحكم من على النموذج.

كما يمكنك إظهار إخفائها باستخدام الطريقة `Hide();Show();`.

الطريقة `Focus()` تمكنك من نقل التركيز للعنصر أو الأداة التي تحددها.

معالجة أحداث الفأرة ولوحة المفاتيح

أنشئ نموذج وأضف إليه الأدوات كما توضح الصورة وغير الخصائص مثلما تعلمت سابقا.



الحدث `MouseMove` يفعل هذا الحدث عند تحريك الماوس على الأداة.

```
private void button1_MouseMove(object sender, MouseEventArgs e)
{
    textBox1.Text = "welcome c#";
}
```

الحدث `MouseUp` يفعل هذا الحدث عند النقر بالماوس بينما يتم تحرير المؤشر من فوق الأداة.

```
private void Form1_MouseUp(object sender, MouseEventArgs e)
{
    FormBorderStyle = FormBorderStyle.None;
    textBox1.ForeColor = Color.Red;
    textBox1.ReadOnly = true;
    textBox1.RightToLeft = RightToLeft.Yes;}
```

الحدث MouseDown يفعل هذا الحدث عند النقر بالماوس بينما يكون المؤشر مازال فوق الأداة.

```
private void Form1_MouseDown(object sender,
MouseEventArgs e)
{
    button1.BackColor = Color.Red;
}
```

الحدث MouseDown يفعل هذا الحدث عندما يغادر مؤشر الماوس من فوق الأداة.

```
private void button1_MouseLeave(object sender,
EventArgs e)
{
    textBox1.Text = "";}
```

الحدث KeyPress يفعل هذا الحدث عند الضغط على المفتاح من لوحة المفاتيح .
وهذا مثال يوضح عملية إدخال الأرقام فقط إلى مربع النص

```
private void textBox1_KeyPress(object sender,
KeyPressEventArgs e)
{
    if (e.KeyChar >= '0' && e.KeyChar <= '9')
        e.Handled = false;
    else
        e.Handled = true;
}
```

الحدث KeyUp يفعل هذا الحدث عند الضغط على المفتاح من لوحة المفاتيح وتحريره.
وهذا مثال يوضح عملية الإغلاق عند الضغط على المفتاح Esc من لوحة المفاتيح.

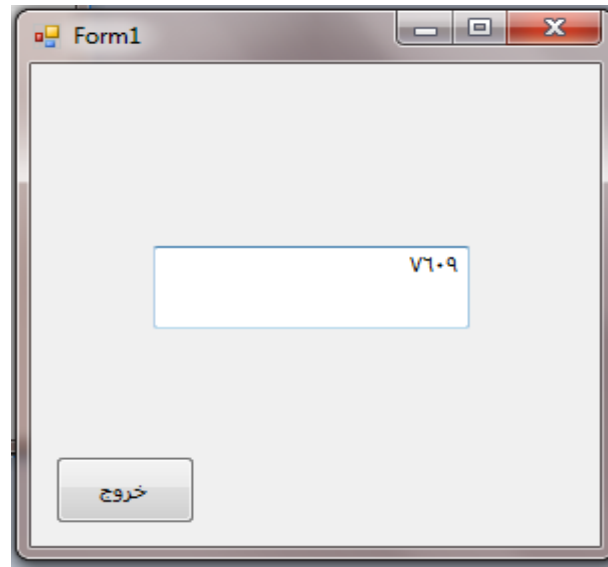
```
Private void button1_KeyUp(object sender, KeyEventArgs e)
{
    if (e.KeyCode.ToString() == "Escape")
        System.Environment.Exit(0);
}
```

الحدث KeyDown يفعل هذا الحدث عند الضغط على المفتاح من لوحة المفاتيح والاستمرار في الضغط.
وهذا مثال يوضح عملية الإغلاق عند الضغط على المفتاح Esc من لوحة المفاتيح.

```
private void button1_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode.GetHashCode() == 27)
        System.Environment.Exit(0);
}
```

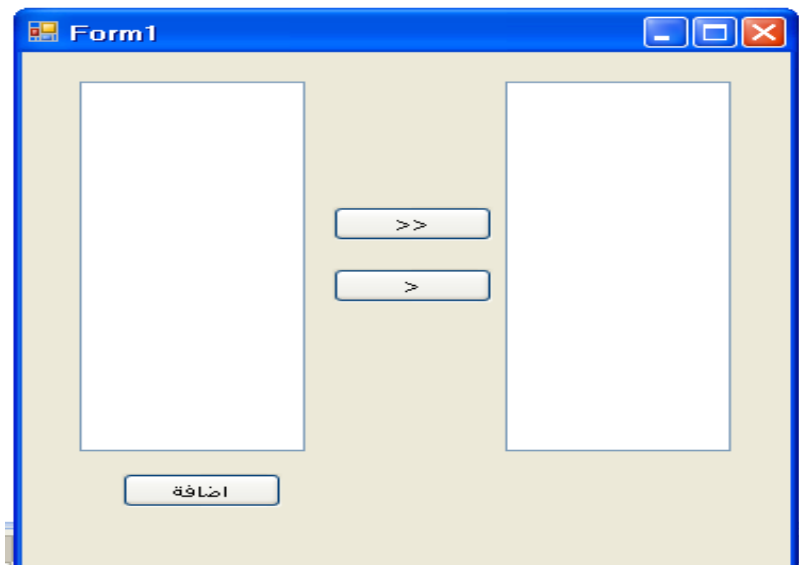
الدالة أو الطريقة GetHashCode تعطي شفرة المفتاح المضغوط من لوحة المفاتيح

الخاصية Handle تأخذ القيمة true or false وهي تمكن صندوق النص من إمكانية إستقبال قيمة في حالة
إذا كانت true والعكس إذا كانت false .



التعامل مع القوائم (ComboBox & ListBox)

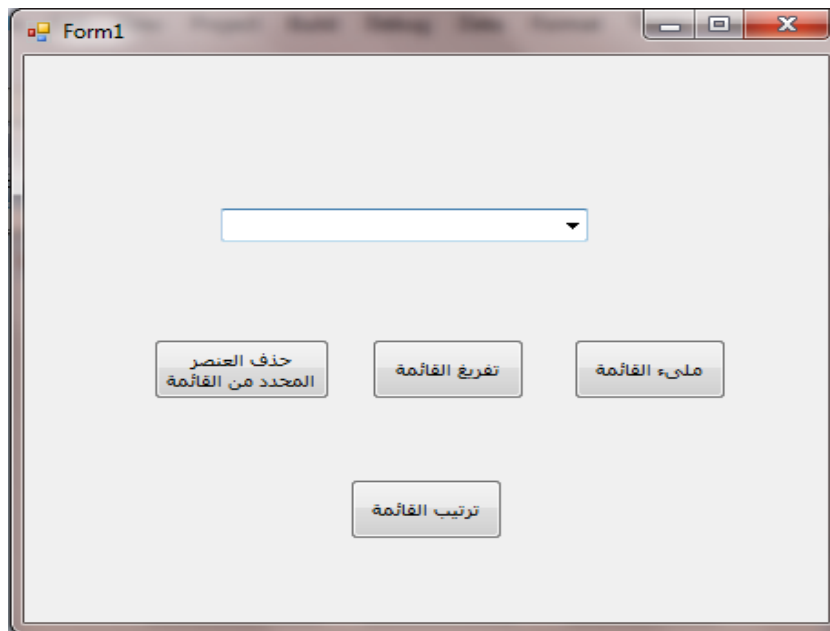
أنشئ نموذج وأضف إليه الأدوات كما توضح الصورة وغير الخصائص مثلما تعلمت سابقا.



```
// شفرة الزر >>
private void button2_Click(object sender, EventArgs e)
{
    listBox2.Items.Clear();
    if (listBox1.Items.Count == 0)
        MessageBox.Show("listbox1 is empty");

    for (int i = 0; i < listBox1.Items.Count; i++)
    {
        listBox2.Items.Add(listBox1.Items[i]);
    }
}
```

```
// شفرة الزر إضافة
private void button1_Click(object sender, EventArgs e)
{
    listBox1.Items.Clear();
    string[] Array = new string[12];
    for (int i = 0; i < 12; i++)
        Array[i] = ("number "+i.ToString());
    listBox1.Items.AddRange(Array);
}
// شفرة الزر >
private void button3_Click(object sender, EventArgs e)
{
    if (listBox1.Items.Count == 0)
        MessageBox.Show("listbox1 is empty");
    else if (listBox1.SelectedIndex != -1)
    {
        listBox2.Items.Add(listBox1.SelectedItem);
        listBox1.Items.Remove(listBox1.SelectedItem);
    }
    else
    {
        MessageBox.Show("القائمة من عنصر إختار");
    }
}
```



```
private void button1_Click(object sender, EventArgs e)
{
    comboBox1.Items.Clear();
    primary(15);
}
```

```
private void primary(int x)
{
    for (int j = 1; j <= x; j++)
        comboBox1.Items.Add(j);
}

private void button2_Click(object sender, EventArgs e)
{
    comboBox1.Items.Clear();
    primary(15);
}

private void button3_Click(object sender, EventArgs e)
{
    comboBox1.Items.Remove(comboBox1.SelectedItem);
}

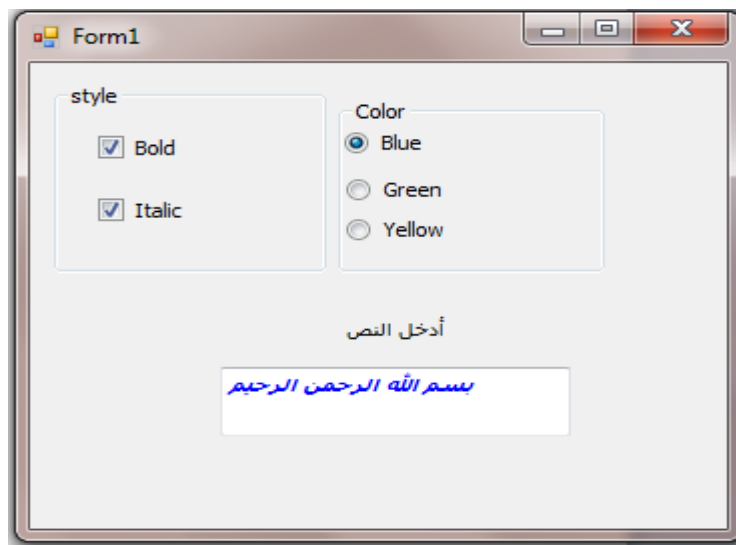
private void button4_Click(object sender, EventArgs e)
{
    comboBox1.Sorted = true;
}
```

التعامل مع الأدوات (CheckBox, RadioButton, GroupBox)

تمكنك الأداة GroupBox من إنشاء مجموعات ضمن النموذج .

الأداة RadioButton والأداة CheckBox متشابهة من حيث إمكانية تفعيلها وعدم تفعيلها باستخدام الخاصية Checked وتحمل هذه الخاصية القيمة true or false والفرق بينهما ان الأداة CheckBox صممت لتعمل ضمن مجموعات بحيث يمكنك تحديد أكثر من زر اختيار معا حتى ولو كانت ضمن مجموعة بينما الأداة RadioButton لايمكنك تحديد أكثر من زر اختيار واحد إذا كانت ضمن مجموعة.

أنشئ نموذج وأضف الأدوات كما توضح الصورة التالية



```
// شفرة الأداة Bold
private void CheckBold_CheckedChanged(object sender,
EventArgs e)
{
    if (CheckBold.Checked == true & CheckItalic.Checked ==
false )
        textBox1.Font = new Font(textBox1.Font,
FontStyle.Bold);
    else if (CheckBold.Checked == false & CheckItalic.Checked
== true )
        textBox1.Font = new Font(textBox1.Font,
FontStyle.Italic);
    else if (CheckBold.Checked == true & CheckItalic.Checked
== true )
        textBox1.Font = new Font(textBox1.Font,
FontStyle.Italic | FontStyle.Bold);

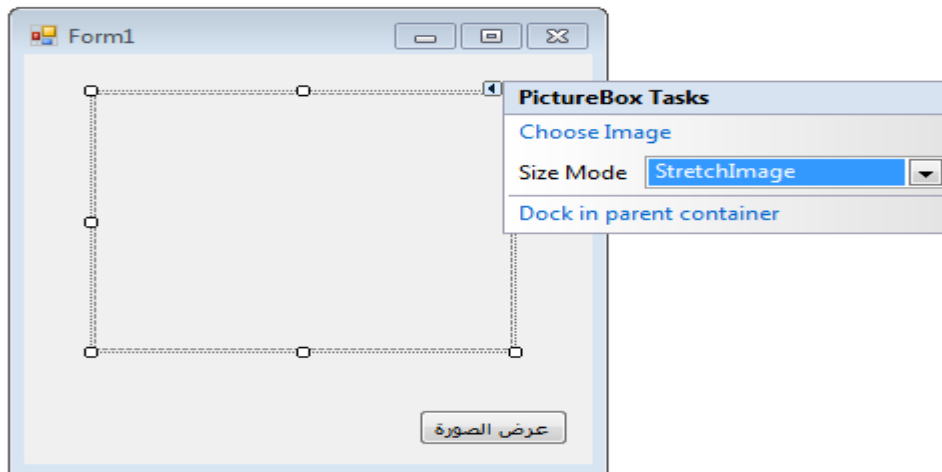
    else
        textBox1.Font = new Font(textBox1.Font,
FontStyle.Regular);
}
// شفرة الأداة Italic
private void CheckItalic_CheckedChanged(object sender,
EventArgs e)
{
    if (CheckBold.Checked == true & CheckItalic.Checked ==
false )
        textBox1.Font = new Font(textBox1.Font,
FontStyle.Bold);
    else if (CheckBold.Checked == false & CheckItalic.Checked
== true)
        textBox1.Font = new Font(textBox1.Font,
FontStyle.Italic);
    else if (CheckBold.Checked == true & CheckItalic.Checked
== true )
        textBox1.Font = new Font(textBox1.Font,
FontStyle.Italic | FontStyle.Bold);

    else
        textBox1.Font = new Font(textBox1.Font,
FontStyle.Regular);
}
// شفرة الزر Blue
private void radioButton1_CheckedChanged(object sender,
EventArgs e)
{
    textBox1.ForeColor = Color.Blue;
}
```

```
// شفرة الزر Green
private void radioButton2_CheckedChanged(object sender,
EventArgs e)
{
    textBox1.ForeColor = Color.Green;
}
// شفرة الزر Yellow
private void radioButton3_CheckedChanged(object sender,
EventArgs e)
{
    textBox1.ForeColor = Color.Yellow;
}
```

(التعامل مع صندوق الصور PictureBox)

أضف الأداة PictureBox إلى النموذج وأضبط الخاصية sizemode بالقيمة StretchImage كما توضح الصورة التالية.



اذهب إلى مجلد المشروع وأضف 4 صور من نوع jpg داخل المجلد Depuge واجعل إسمها 1,2,3,4 ثم أكتب الشفرة التالية.

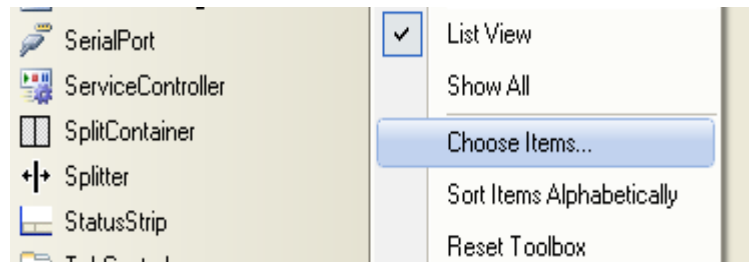
```
public partial class Form1 : Form
{
    int i = 1;
    private void button1_Click(object sender, EventArgs e)
    {
        if (i>4)
            i=1;
        else
            pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;
            pictureBox1.Image = Image.FromFile(Application.StartupPath
+ "/" + i + ".jpg");
            i++;
    }
}
```

بنفس الطريقة يمكن تغيير خلفية الفورم بصورة عن طريق الخاصية `BackgroundImage`
`BackgroundImage = Image.FromFile(Application.StartupPath +`
`"\\a.jpg");`
`BackgroundImageLayout = ImageLayout.Stretch;`

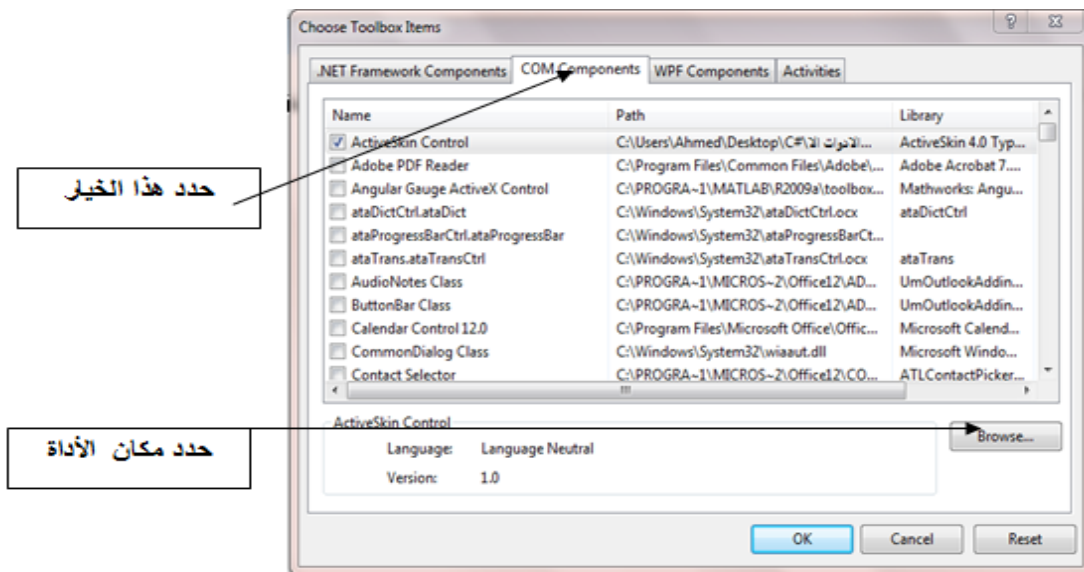
(تشغيل ملفات الصوت والفيديو)

الأدوات المستخدمة (MCIWindx Control , TrackBar , Timer)

- قم بتحميل الأداة **MCIWindx** إلى صندوق الأدوات وذلك بالنقر بالزر الأيمن للماوس في مكان فارغ من صندوق الأدوات واختيار الأمر **Chose Items** كما توضح الصورة.



ثم انتظر حتى تظهر الصورة التالية:



ثم اضغط **ok** ليتم إضافتها إلى صندوق الأدوات

- أنشى نموذج وأضف الأدوات الآتية:
- 1 - `playbar=fase` واجعل الخاصية `MCIWindx control`.
- 2 - `TrackBar`.
- 3 - `groubBox` وضع فيها `2.Button,Label`.
- 4 - ثلاث أزرار (`Button`) للتحكم بالفيديو.
- 5 - مؤقت `timer`.

كما توضح الصورة التالية.



```
// شفرة تحميل الملف
private void Form1_Load(object sender, EventArgs e)
{
    axMCIWnd1.FileName = Application.StartupPath + "\\A.AVI";
    axMCIWnd1.Command = "open";
    trackBar1.Maximum = axMCIWnd1.Length;
}

// شفرة تشغيل
private void playBtn_Click(object sender, EventArgs e)
{
    axMCIWnd1.Command = "play";
    button1.Enabled = false;
    button2.Enabled = true;
    button3.Enabled = true;
    timer1.Enabled = true;
}

// شفرة إيقاف مؤقت
private void pauseBtn_Click(object sender, EventArgs e)
{
    axMCIWnd1.Command = "pause";
    button2.Enabled = false;
    button3.Enabled = false;
    button1.Enabled = true;
    timer1.Enabled = true;
}

// شفرة إيقاف
private void stopBtn_Click(object sender, EventArgs e)
{
    axMCIWnd1.Command = "stop";
    button3.Enabled = false;
}
```

```

axMCIWnd1.Position = 0;
button2.Enabled = false;
button1.Enabled = true;
timer1.Enabled = true;
}
// شفرة نعم
private void yesBtn_Click(object sender, EventArgs e)
{
    axMCIWnd1.Command = "play";
    groupBox1.Visible = false;
    timer1.Enabled = true;
}
// شفرة لا
private void NoBtn_Click(object sender, EventArgs e)
{
    axMCIWnd1.Command = "stop";
    axMCIWnd1.Position = 0;
    groupBox1.Visible = false; ;
    axMCIWnd1.Command = "play";
    timer1.Enabled = true;
}
// شفرة المؤقت
private void timer1_Tick(object sender, EventArgs e)
{
    trackBar1.Value = axMCIWnd1.Position;
    if (axMCIWnd1.Position >= 100 && axMCIWnd1.Position < 101)
    {

        axMCIWnd1.Command = "pause";
        groupBox1.Visible = true;
        timer1.Enabled = false;
    }
}
}

```

الأداة (shockwaveflash object, Realplyer G2 control)

(التعامل مع ملفات الصوت والفلش)

قم بإنشاء نموذج جديد وأضف الأدوات كما يلي :



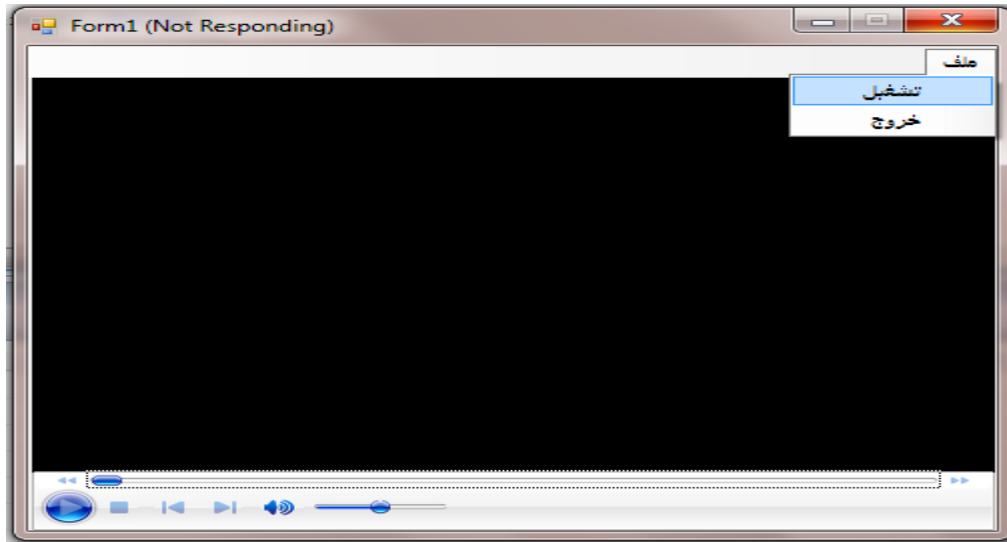
```
// شفرة تحميل ملف الصوت والفلش
private void Form1_Load(object sender, EventArgs e)
{
    axShockwaveFlash1.LoadMovie(0, System.Environment.CurrentDirectory + "\\v.swf");
    axShockwaveFlash1.Stop();
    axRealAudio1.Source = Application.StartupPath + "\\A.mp3";
}

// شفرة الزر تشغيل
axShockwaveFlash1.Play();
// شفرة الزر ايقاف
axShockwaveFlash1.Stop();
```

• الأداة (Windowsmediaplayer , menustrip, openFileDialog)

أنشئ نموذج وأضف إليه الأدوات التالية

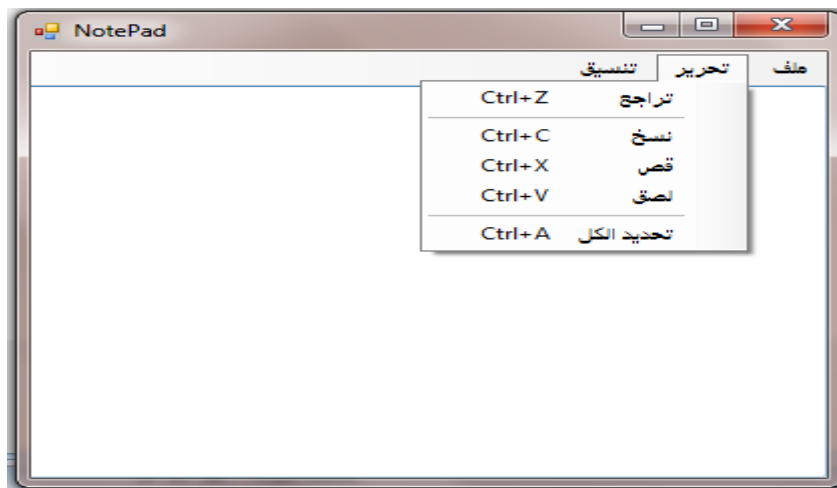
- Windows media player
- openFileDialog , menustrip وهما تظهرا في شريط أسفل النموذج
- غير الخاصية text لـ MenuStrip كما توضح الصورة والخاصية name كما توضح الشفرة.



```
// شفرة تشغيل
private void RunFile_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "vedio|*.avi|sound|*.mp3";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
        axWindowsMediaPlayer1.URL = openFileDialog1.FileName;
}

// شفرة خروج
private void Exit_Click(object sender, EventArgs e)
{
    Close();
}
```

برنامج المفكرة:



إضبط الخاصية Righttoleft للأداة menustrep على القيمة yes لتجعلها تعرض من اليمين لليساار. والخاصية shortcutkey تمكنك من إضافة مفاتيح إختصار لكل واحد من محتويات القائمة.

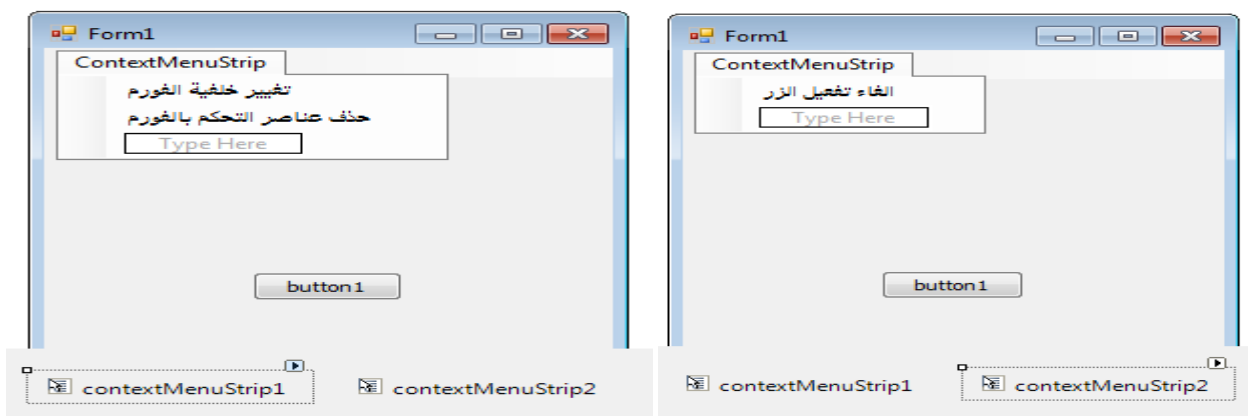
```
// شفرة تراجع
private void Undo_Click(object sender, EventArgs e)
{
    Edite.Undo();
}
// شفرة نسخ
private void Copy_Click(object sender, EventArgs e)
{
    Clipboard.SetDataObject(Edite.SelectedText);
}
// شفرة قص
private void Cutt_Click(object sender, EventArgs e)
{
    Clipboard.SetDataObject(Edite.SelectedText);
    Edite.SelectedText="";
}
// شفرة لصق
private void Past_Click(object sender, EventArgs e)
{
    if
    (Clipboard.GetDataObject().GetDataPresent(DataFormats.Text)
    == true)
    {
        Edite.SelectedText =
        Clipboard.GetDataObject().GetData(DataFormats.Text).ToString
        ();
    }
}
// شفرة تحديد الكل
private void SeselectAll_Click(object sender, EventArgs e)
{
    Edite.SelectAll();
}
```

الأداة (contestMenuStrip)

أنشئ نموذج وأضف الأدوات التالية :

كما توضح الصورة :

Button -2 contestMenuStrip 2 - 1



ومن خصائص الأداة التي نريد إظهار محتوى الـ contextMenuStrip عند النقر بالزر الأيمن للفأرة عليها نختار الخاصية contextMenuStrip ونحدد الـ contextMenuStrip الذي نريد إظهار محتواه.

فمثلاً هنا سنجعل الخاصية contextMenuStrip للفورم تأخذ الـ contextMenuStrip1 و الخاصية contextMenuStrip الأداة Button تأخذ الـ contextMenuStrip2

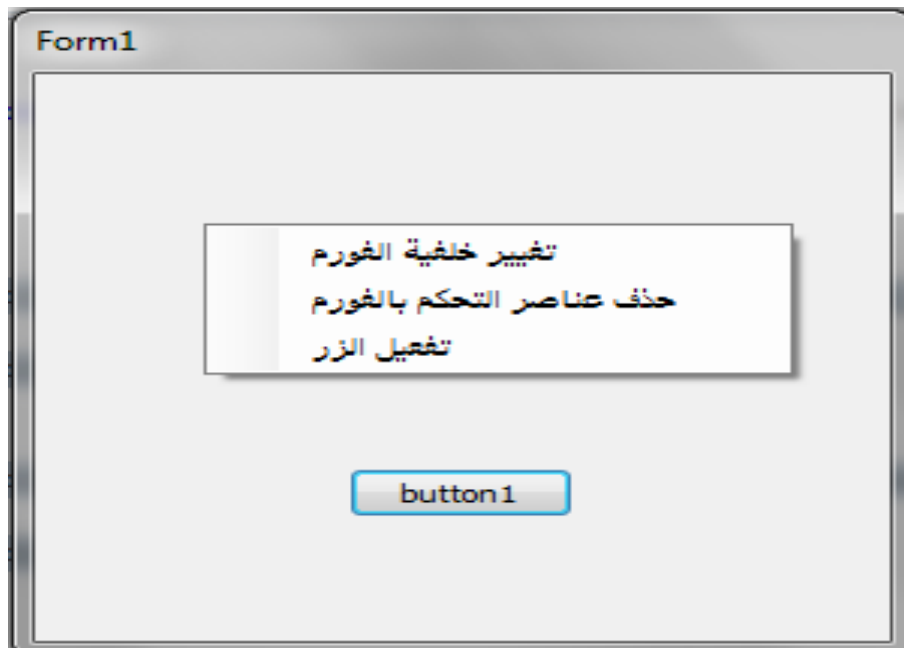
ثم نقوم بكتابة الشفرة بالنقر على كل واحد من محتويات الـ contextMenuStrip .

```
private void ToolStripMenuItem_Click(object sender, EventArgs e)
{
    button1.Enabled = false;
}

private void ToolStripMenuItem_Click(object sender, EventArgs e)
{
    BackColor = Color.Brown;
}

private void ToolStripMenuItem_Click(object sender, EventArgs e)
{
    ControlBox = false;
}
```

وعند التنفيذ يمكننا الآن إظهار عمل الاداة بالضغط بزر الفأرة الأيمن على الفورم أو الزر كما توضح الصورة



أداة (DateTimePicker و monthCalendar و Tooltip)

(التعامل مع منتقى الوقت والتاريخ وأداة إظهار التلميحات)

أفتح مشروع جديد وأضف نموذج وأضف الأداة (DateTimePicker) والأداة Tooltip وعنصر تقويم الشهر .monthCalendar

أضف button واجعل الخاصية text="أنقر" .

أضف .TextBox

الأداة Tooltip يمكنك من إظهار التلميحات على الأدوات الأخرى داخل النموذج.

تظهر هذه الأداة عند اختيارها في شريط أسفل من الفورم.

بإمكاننا الآن أن نظيف تلميح للزر " أنقر " وليكن في الحدث Form_Load باستخدام الطريقة SetToolTip



```
private void Form1_Load(object sender, EventArgs e)
{
    tooltip1.SetToolTip(button1, " فظلك من
    أنقر هنا ");
    dateTimePicker1.MinDate = new DateTime(1985, 6,
    20);
    dateTimePicker1.MaxDate = DateTime.Today;

    dateTimePicker1.CustomFormat = "dddd dd - MMMMM
    -YYYY";
    dateTimePicker1.Format =
    DateTimePickerFormat.Custom;
}
```

الخاصية **format** تمكنك من تحديد الصيغة التي تريد أن تظهر بها الأداة.

في حالة جعلت هذه الخاصية تأخذ القيمة **custom** فيمكنك تحديد صيغة التاريخ أو الوقت الذي ستظهر به الأداة من الخاصية **custom format** كما يوضح الشفرة السابقة .

في حالة لو كان التقويم لديك بالهجري فإنه بإمكانك إظهار القيمة بالتقويم الميلادي كما في الكود التالي.

أنقر على **button** وأضف الكود التالي:

```
private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show(dateTimePicker1.Value.ToString("dddd dd - 
    MMMMM -yyyy",
    System.Globalization.CultureInfo.InvariantCulture));
}
```

العنصر **monthCalendar1** :

يمتلك العنصر **monthCalendar1** الحدثان المهمان **DateChange** وهو الحدث الافتراضي للعنصر ويقع عندما يتغير التاريخ في التقويم و **DateSelected** ويقع هذا الحدث عندما نحدد على أي تاريخ في التقويم.

وكذلك الطريقة **SelectionStart** و **SelectionRange.End** و **SelectionRange.Stare** و **SelectionEnd** والتي تمكننا من الحصول على غرض من نوع **DateTime** .

فمثلا يمكننا أن نحدد أي أيام الأسبوع ينتمي اليوم الذي سنحدد عليه كما يلي :

```
private void monthCalendar1_DateSelected(object sender,
DateRangeEventArgs e)
{
    textBox1.Text=
monthCalendar1.SelectionStart.DayOfWeek.ToString();
}

private void monthCalendar1_DateChanged(object sender,
DateRangeEventArgs e)
{
    textBox1.Text =
monthCalendar1.SelectionRange.Start.Day.ToString();
}

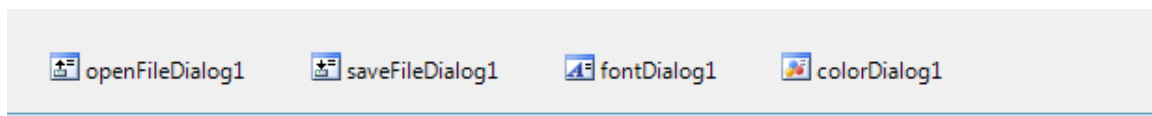
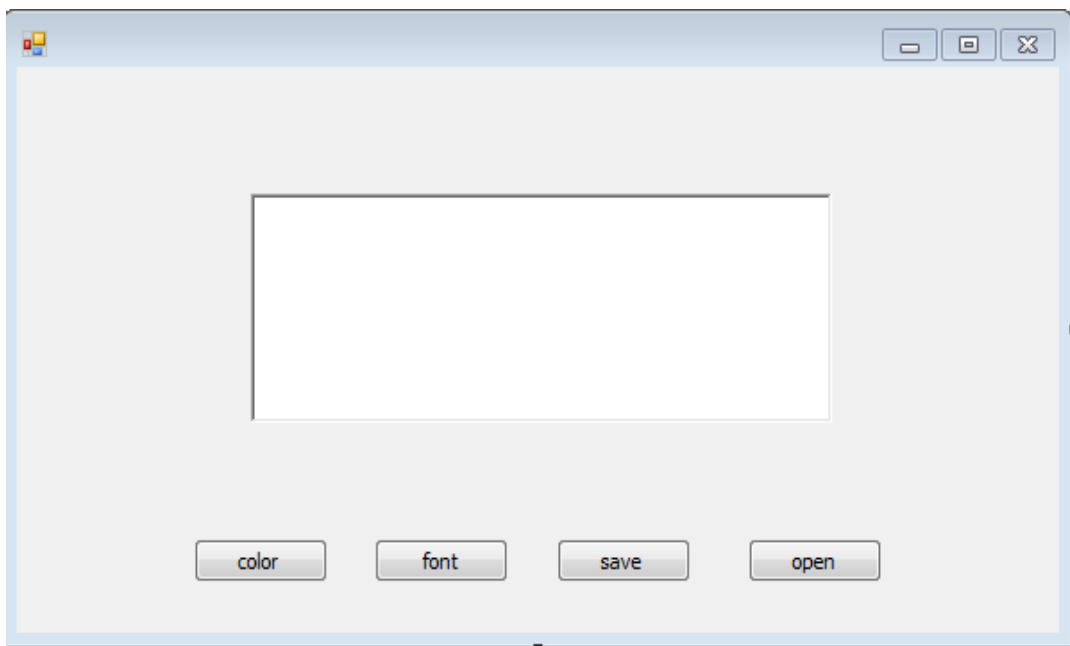
}
```


صناديق الحوار (Dialog Box)

openFileDialog saveFileDialog
fontDialog colorDialog

(التعامل مع صناديق الحوار (Dialog))

- 1 - أضف الأداة RichTextBox من صندوق الأدوات.
- 2 - أضف صناديق الحوار (openFileDialog, saveFileDialog, colorDialog, fontDialog).
- 3 - أضف 4 أزرار تحكم (Button) وغير الخاصية text لها كما توضح الصورة، واجعل الخاصية name لها كما تظهر في شفرة الحدث Click.



```
// شفرة الزر open
private void OpenBtn_Click(object sender, EventArgs e)
{
    openFileDialog1.Title = "open RTF file";
    openFileDialog1.Filter = "RTF files|*.rtf";
    openFileDialog1.InitialDirectory = @"C:\";
    openFileDialog1.ShowDialog();
    openFileDialog1.ShowHelp = true
}
```

الخاصية ShowHelp تظهر زر المساعدة (Help) على صناديق الحوار وتأخذ القيمة true or false

```
// شفرة الزر save
private void SaveBtn_Click(object sender, EventArgs
e)
{
    saveFileDialog1.Title = "open RTF file";
    saveFileDialog1.Filter = "RTF files|*.rtf";
    saveFileDialog1.InitialDirectory = @"C:\";
    saveFileDialog1.ShowDialog();
    saveFileDialog1.ShowHelp = true;
}
```

ينفذ محتوى شفرة HelpRequest الحدث عندما ينقر المستخدم على زر Help من صندوق الحوار.

```
// شفرة الحدث HelpRequest لصندوق الحوار openFileDialog
private void openFileDialog1_HelpRequest(object
sender, EventArgs e)
{
    MessageBox.Show("Help Button open");
}
```

```
// شفرة الحدث HelpRequest لصندوق الحوار saveFileDialog
private void saveFileDialog1_HelpRequest(object
sender, EventArgs e)
{
    MessageBox.Show("Help Button save");
}
```

ينفذ محتوى شفرة FileOk الحدث عندما ينقر المستخدم على زر OK من صندوق الحوار.

```
// شفرة الحدث FileOk لصندوق الحوار saveFileDialog
private void openFileDialog1_FileOk(object sender,
CancelEventArgs e)
{
    richTextBox1.LoadFile(openFileDialog1.FileName);
}
```

```
// شفرة الحدث FileOk لصندوق الحوار saveFileDialog
private void saveFileDialog1_FileOk(object sender,
CancelEventArgs e)
{
    richTextBox1.SaveFile(saveFileDialog1.FileName);
}
```

ينفذ محتوى شفرة الحدث APPLY عندما ينقر المستخدم على زر APPLY من صندوق الحوار (fontDialog).

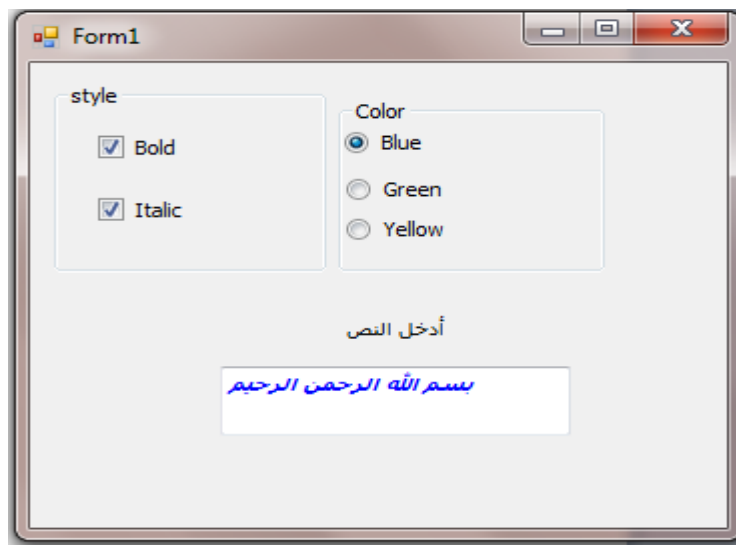
```
private void fontDialog1_Apply(object sender, EventArgs e)
{
    richTextBox1.SelectionFont = fontDialog1.Font;
    richTextBox1.SelectionColor = fontDialog1.Color;
}
```

```
//شفرة الزر
private void FontBtn_Click(object sender, EventArgs e)
{
    if (fontDialog1.ShowDialog() == DialogResult.OK)
    {
        richTextBox1.SelectionFont = fontDialog1.Font;
        richTextBox1.SelectionColor = fontDialog1.Color;
    }
}
//شفرة الزر color
private void ColorBtn_Click(object sender, EventArgs e)
{
    if (colorDialog1.ShowDialog() == DialogResult.OK)
    richTextBox1.SelectionColor = colorDialog1.Color;
}
}
```

الأداة (CheckBox, RadioButton, GroupBox)

(التعامل مع مربعات الاختيار وأزرار الراديو)

أنشئ نموذج وأضف الأدوات كما توضح الصورة التالية



```
//شفرة الأداة Bold
private void CheckBold_CheckedChanged(object sender,
EventArgs e)
{
    if (CheckBold.Checked == true & CheckItalic.Checked ==
false )
        textBox1.Font = new Font(textBox1.Font,
FontStyle.Bold);
    else if (CheckBold.Checked == false & CheckItalic.Checked
== true )
```

```

        textBox1.Font = new Font(textBox1.Font,
        FontStyle.Italic);
        else if (CheckBold.Checked == true & CheckItalic.Checked
        == true )
            textBox1.Font = new Font(textBox1.Font,
        FontStyle.Italic | FontStyle.Bold);
        else
            textBox1.Font = new Font(textBox1.Font,
        FontStyle.Regular);
    }
    // شفرة الأداة Italic
    private void CheckItalic_CheckedChanged(object sender,
    EventArgs e)
    {
        if (CheckBold.Checked == true & CheckItalic.Checked ==
        false )
            textBox1.Font = new Font(textBox1.Font,
        FontStyle.Bold);
        else if (CheckBold.Checked == false & CheckItalic.Checked
        == true)
            textBox1.Font = new Font(textBox1.Font,
        FontStyle.Italic);
        else if (CheckBold.Checked == true & CheckItalic.Checked
        == true )
            textBox1.Font = new Font(textBox1.Font,
        FontStyle.Italic | FontStyle.Bold);

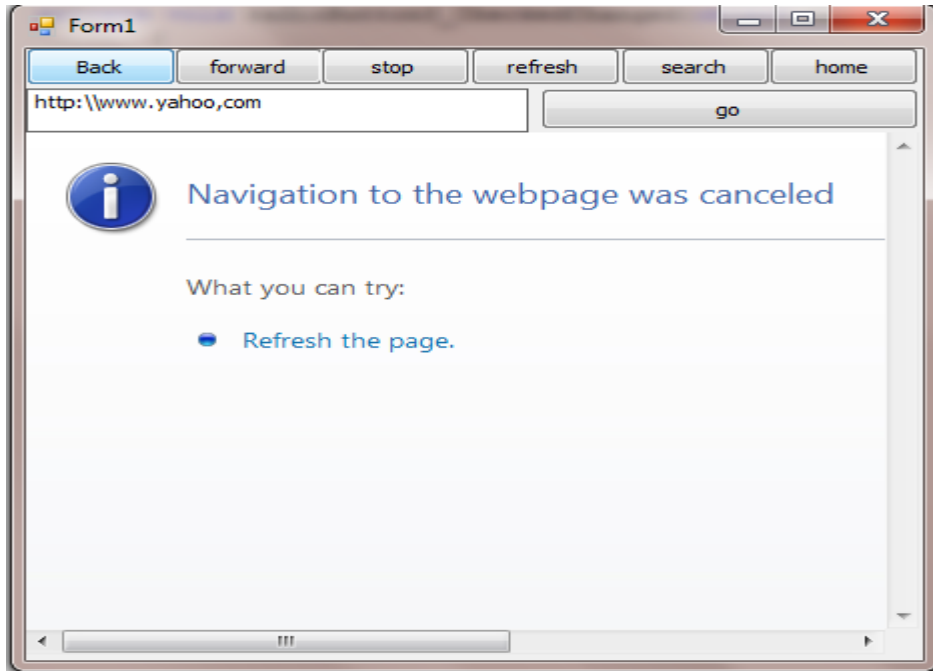
        else
            textBox1.Font = new Font(textBox1.Font,
        FontStyle.Regular);
    }
    // شفرة الزر Blue
    private void radioButton1_CheckedChanged(object sender,
    EventArgs e)
    {
        textBox1.ForeColor = Color.Blue;
    }
    // شفرة الزر Green
    private void radioButton2_CheckedChanged(object sender,
    EventArgs e)
    {
        textBox1.ForeColor = Color.Green;
    }
    // شفرة الزر Yellow
    private void radioButton3_CheckedChanged(object sender,
    EventArgs e)
    {
        textBox1.ForeColor = Color.Yellow;
    }

```

الأداة (WebBrowser)

(تصميم متصفح انترنت بسيط)

أنشئ نموذج وأضف الأدوات كما توضح الصورة.



```
// شفرة الزر Back
private void button1_Click(object sender, EventArgs e)
{
    webBrowser1.GoBack();
}

//شفرة الزر forward
private void button2_Click(object sender, EventArgs e)
{
    webBrowser1.GoForward();
}

// شفرة الزر stop
private void button3_Click(object sender, EventArgs e)
{
    webBrowser1.Stop();
}

//شفرة الزر refresh
private void button4_Click(object sender, EventArgs e)
{
    webBrowser1.Refresh();
}

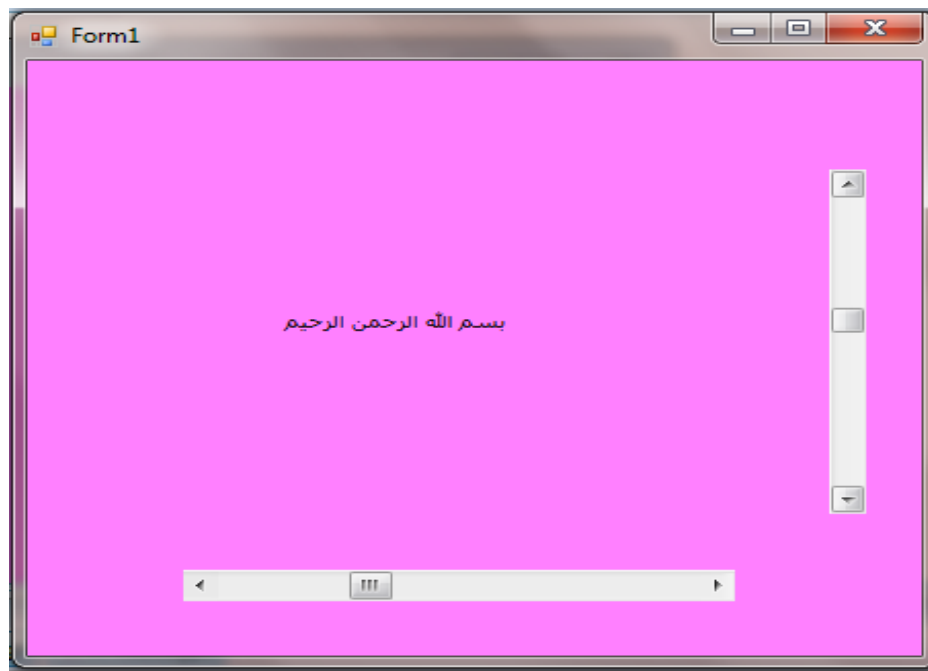
// شفرة الزر search
private void button5_Click(object sender, EventArgs e)
{
    webBrowser1.GoSearch();
}
```

```
// شفرة الزر Home
private void button6_Click(object sender, EventArgs e)
{
    webBrowser1.GoHome();
}
//شفرة الزر Go
private void button7_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate(textBox1.Text);
}
```

أداة (HScrollBar,VScrollBar)

(التعامل مع أشرطة التمرير)

أضف أشرطة التمرير إلى النموذج وأضف Label كما توضح الصورة.



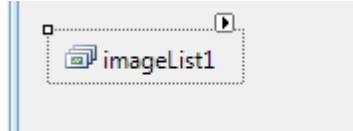
```
private void vScrollBar1_Scroll(object sender,
ScrollEventArgs e)
{
    label1.Location = new Point(label1.Location.X,
e.NewValue * this.Size.Height / 100);
}

private void hScrollBar1_Scroll(object sender,
ScrollEventArgs e)
{
    label1.Location = new
Point(e.NewValue*this.Size.Width/100,label1.Location.Y);
}
```

أداة (ImageList)

(التعامل مع قائمة الصور)

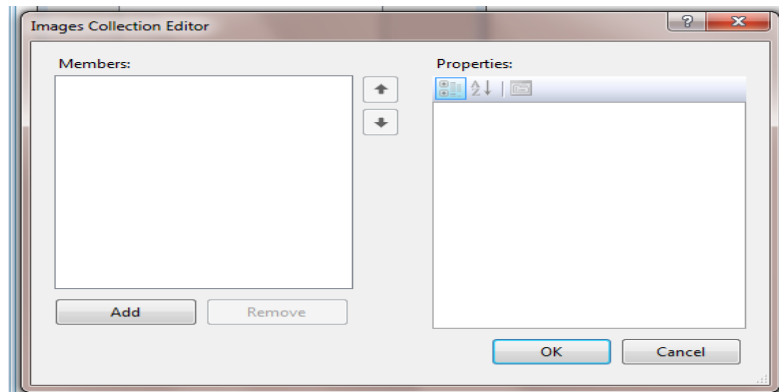
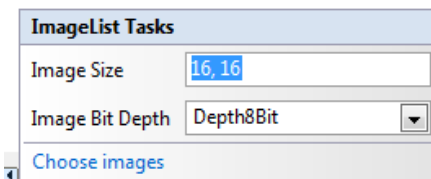
أذهب إلى ال Toolbox ومن المجموعة Components اختر المكون ImageList كما تبين الصورة.



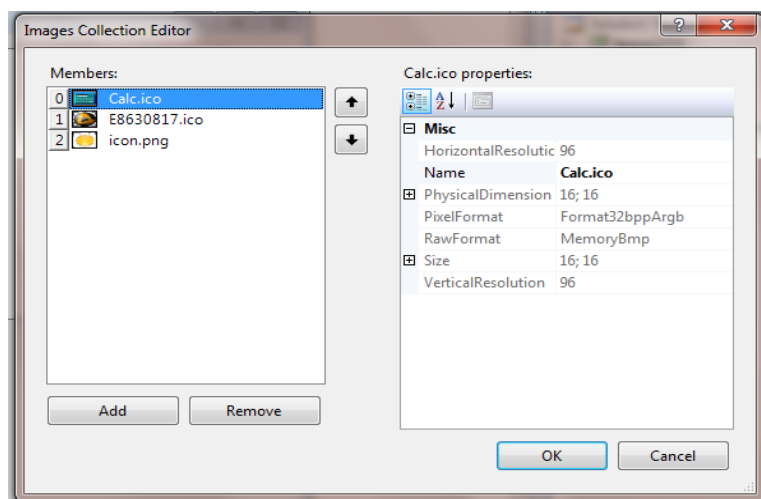
انقر على السهم الموجود في الزاوية العليا من المكون ImageList1

انقر على البند Choose Images من النافذة (ImageList Tasks)

ومن ثم انقر الزر Add من النافذة Images Collections Editor لإضافة الصور



سيظهر امامك مربع الحوار Open أختار الأيقونات وبعدها اضغط على الزر Open كما في الصورة.



ونستطيع إضافة الصور من الخاصية Images للأداة ImageList

بإمكانك الآن إضافة الصور برمجيا لعمل ذلك قم بالآتي

. أضف openFileDialog وزر Button الخاصة name=AddImage

. في الحدث Click اكتب الشفرة التالية.

```
private void AddImage_Click(object sender, EventArgs e)
{
    openFileDialog1.ShowDialog();
    openFileDialog1.Filter = "Icon|*.ico";
    Image Img =
Image.FromFile(openFileDialog1.FileName);
    imageList1.Images.Add(Img);
}
```

من صندوق الحوار الذي سيظهر إذهب إلى مجلد الأيقونات لديك وحدد الأيقونات التي تريد إضافتها لـ ImageList1

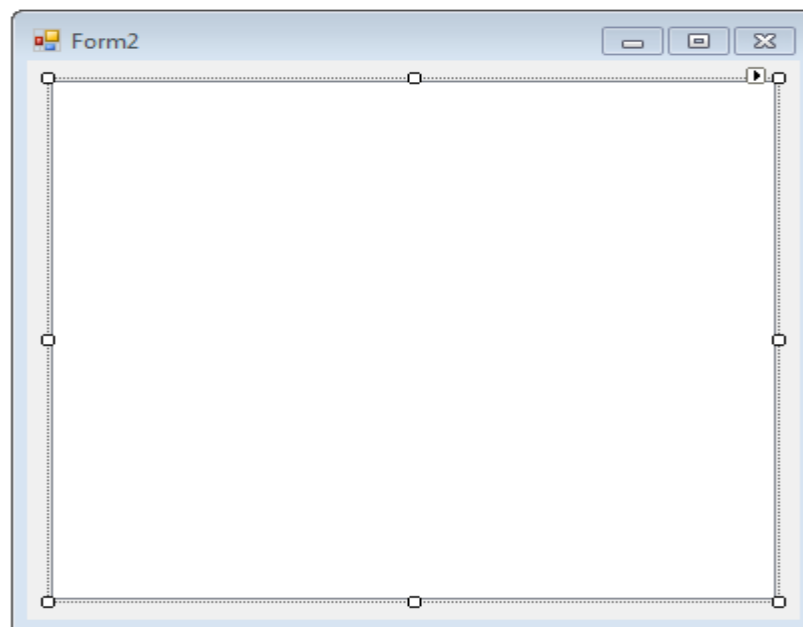
الأداة (ListView)

(التعامل مع ListView)

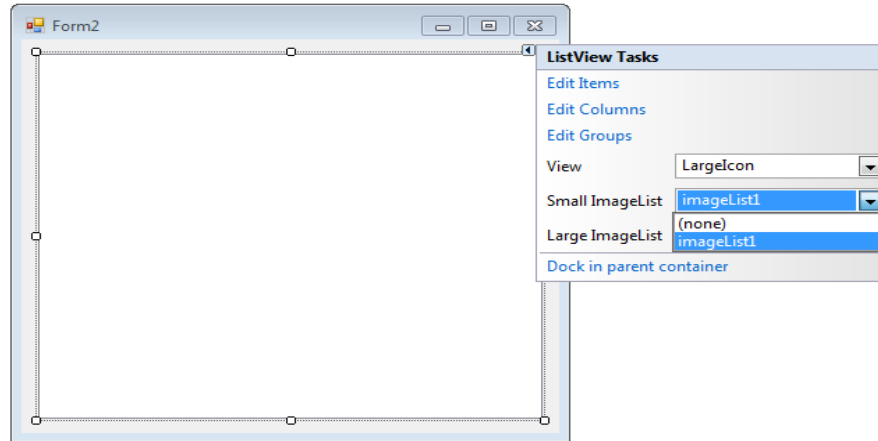
في هذا التدريب سنتعلم كيفية التعامل مع المكون أو الأداة ListView بطريقة Wizarded .

أضف المكون ImageList كما تعلمت سابقا وأملنها بالأيقونات التي تريدها.

أضف المكون ListView وبقية العناصر الأخرى كما توضح الصورة.



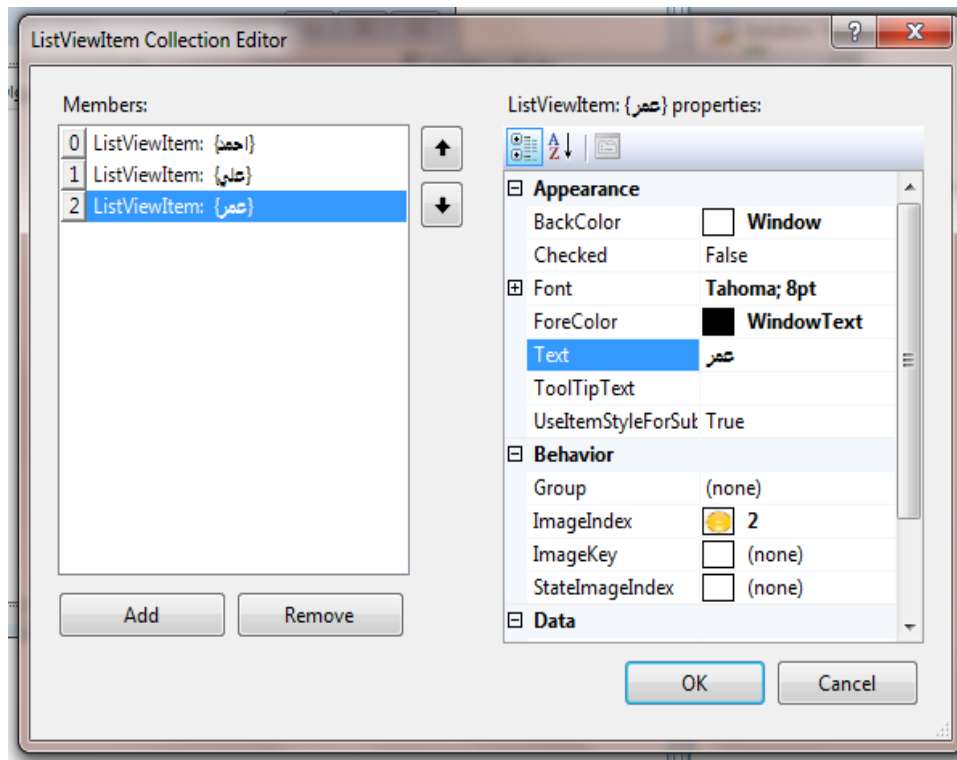
1. اضغط السهم وأربط الخاصية SmallImageList بالمكون ImageList الموجود لديك كما توضح الصورة.



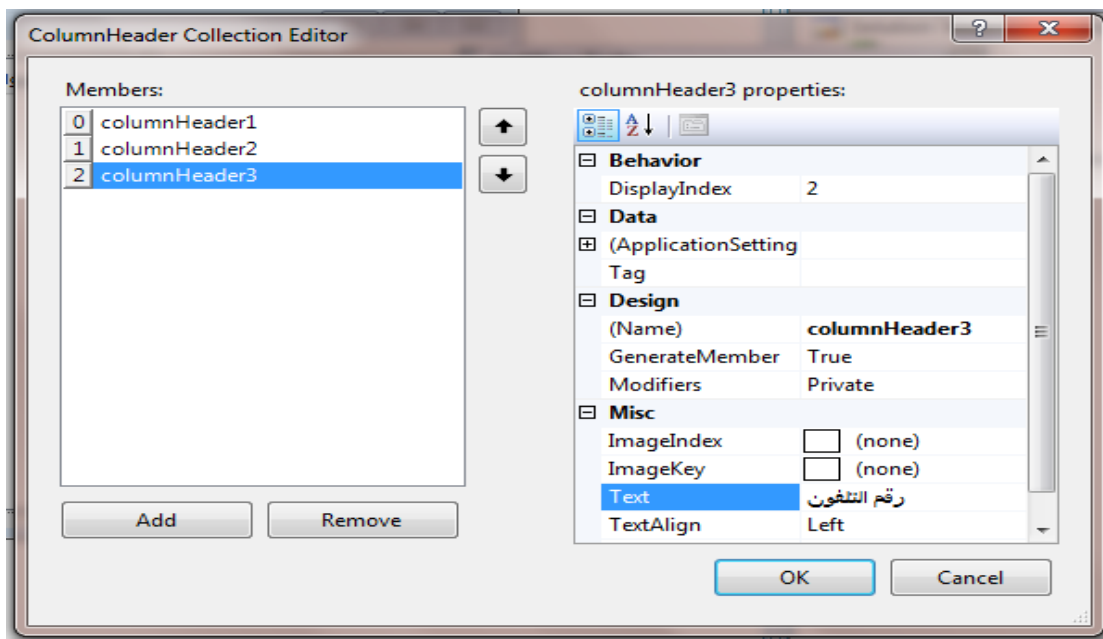
أو من خصائص

المكون **ListView** إجعل الخاصية **SmallImageList** بالمكون **ImageList** الموجود لديك .

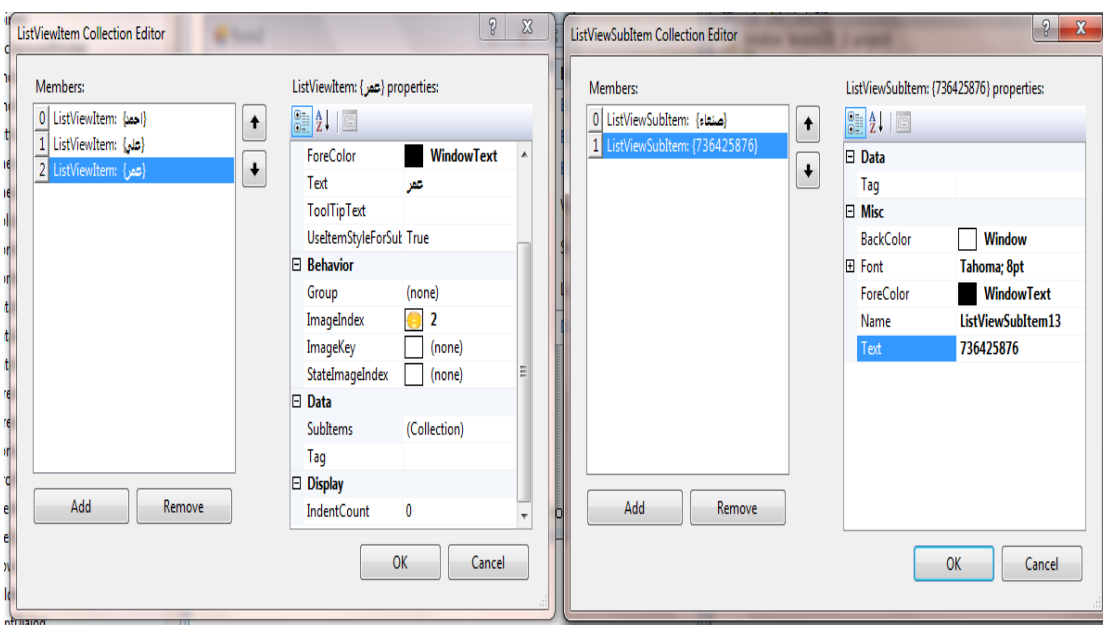
2. أضغط الارتباط **edit Items** وأضف الأسماء التي تريدها . وذلك بالضغط على الزر **Add** ومن خصائص كل عنصر غير الخاصية **text** بالاسم الذي تريده والخاصية **ImageIndex** أجعلها تأخذ رقم الصورة التي تم إدخالها من قبل في المكون **ImageList**.



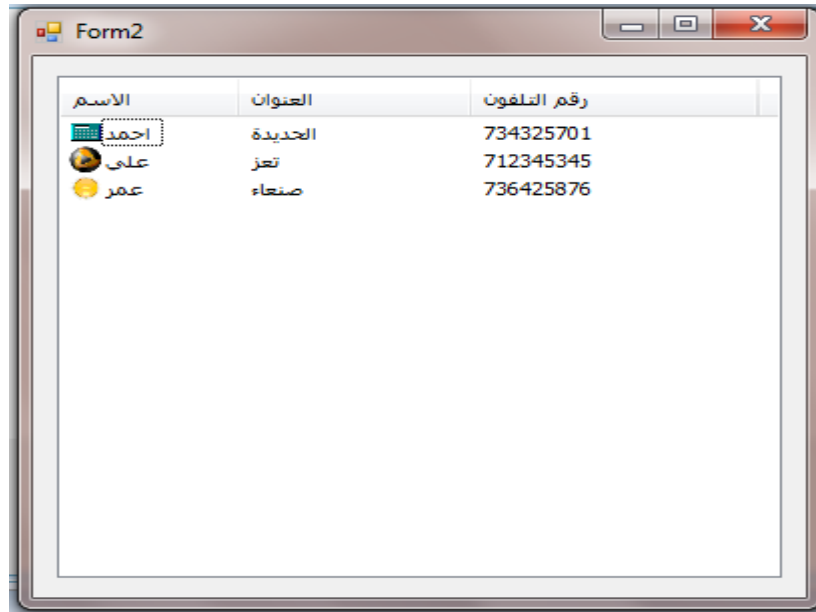
3. أضغط الارتباط **edit columns** وأضف العناوين التي تريدها.



4. عود إلى الارتباط edit Items وعند كل إسم إذهب الى الخاصية SubItemsView أضف بقية المعلومات لكل شخص.



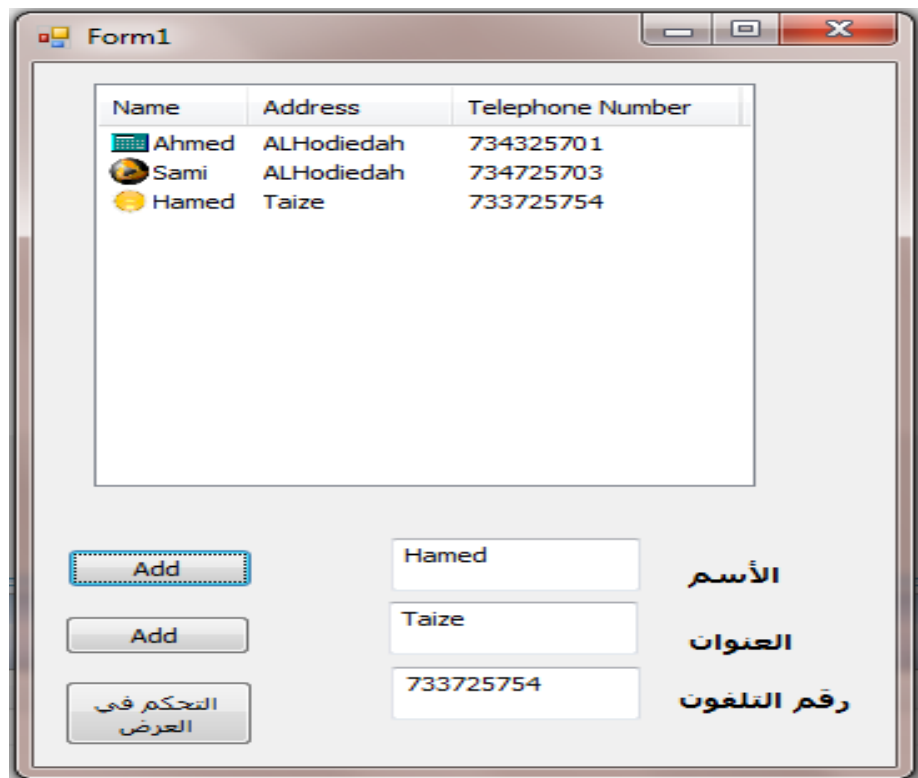
5. إضغط F5 ولاحظ التنفيذ.



الآن سنتعلم كيفية التعامل مع المكون أو الأداة ListView برمجيا .

أضف المكون ImageList كما تعلمت سابقا وأملئها بالأيقونات التي تريدها.

أضف المكون ListView وبقية العناصر الأخرى كما توضح الصورة.



```
public partial class Form1 : Form
{
    int index = 0;
```

```
// تهيئة الأدوات وتكوين الأعمدة الرأسية
private void Form1_Load(object sender, EventArgs e)
{
    listView1.View = View.Details;
    listView1.SmallImageList = imageList1;
    //=====
    ColumnHeader columnHeader1 = new ColumnHeader();
    ColumnHeader columnHeader2 = new ColumnHeader();
    ColumnHeader columnHeader3 = new ColumnHeader();
    //=====
    columnHeader1.Text = "Name";
    columnHeader1.ImageIndex = 0;
    columnHeader2.Text = "Address";
    columnHeader2.ImageIndex = 1;
    columnHeader3.Text = "Telephone Number";
    columnHeader3.ImageIndex = 2;
    listView1.Columns.Add(columnHeader1);
    listView1.Columns.Add(columnHeader2);
    listView1.Columns.Add(columnHeader3);
    //=====
}
```

تحليل الشفرة

```
//=====
listView1.View = View.Details;
listView1.SmallImageList = imageList1;
//=====
```

في السطر الأول من هذا المقطع جعلنا طريقة العرض للمكون listView هي Deaile وفي السطر الثاني ربطنا هذا المكون بالمكون Imagelist عن طريق الخاصية SmallImageList

```
//=====
ColumnHeader columnHeader1 = new ColumnHeader();
ColumnHeader columnHeader2 = new ColumnHeader();
ColumnHeader columnHeader3 = new ColumnHeader();
//=====
```

هنا تم تعريف ثلاث ColumnHeader وهي التي سنظهر في رأس القائمة listView1

```
//=====
columnHeader1.Text = "Name";
columnHeader1.ImageIndex = 0;
columnHeader2.Text = "Address";
columnHeader2.ImageIndex = 1;
columnHeader3.Text = "Telephone Number";
columnHeader3.ImageIndex = 2;
listView1.Columns.Add(columnHeader1);
listView1.Columns.Add(columnHeader2);
listView1.Columns.Add(columnHeader3);
```

//=====

في هذا المقطع حددنا عناوين تلك الأعمدة عن طريق الخاصية **Text** وحددنا الدليل لكل منها بالخاصية **ImageIndex** ثم قمنا بإضافتها إلى القائمة **ListView**.

زر الأضافة الاول//

```
private void button1_Click(object sender, EventArgs e)
{
    string[] array = { AddressText.Text,
MobileText.Text };

    listView1.Items.Add(NameText.Text).SubItems.AddRange(array);
    listView1.Items[index].ImageIndex = index;
    index++;
}
```

في هذه الشفرة عرفنا مصفوفة من نوع **string** وجعلنا مصدرها هي القيم التي سترسل لكل عضو في القائمة والتي تمثل المجموعة الجزئية له. وفي السطر الثاني أضفنا العضو والذي يتمثل بالـ **Text** الأول والمجموعة الجزئية **array** أما بالنسبة للسطر الثالث فإن دليل الصور في القائمة يأخذ القيمة **Index** التي تحدد الصورة التي سيتم إضافتها من القائمة **Imagelist**.

زر الأضافة الثاني //

```
private void button2_Click(object sender, EventArgs e)
{
    ListViewItem listViewItem1 = new ListViewItem();
//=====
System.Windows.Forms.ListViewItem.ListViewSubItem
AddressSubItem = new
System.Windows.Forms.ListViewItem.ListViewSubItem();
//=====
System.Windows.Forms.ListViewItem.ListViewSubItem
MobileSubItem = new
System.Windows.Forms.ListViewItem.ListViewSubItem();
//=====
    listViewItem1.Text = NameText.Text;
    listViewItem1.ImageIndex = index;
    index++;
    AddressSubItem.Text = AddressText.Text;
    MobileSubItem.Text = MobileText.Text;
//=====
    listViewItem1.SubItems.Add(AddressSubItem);
    listViewItem1.SubItems.Add(MobileSubItem);
    listView1.Items.Add(listViewItem1);
//=====
}
```

تحليل الشفرة

```
ListViewItem listViewItem1 = new ListViewItem();

//=====
```

في هذا المقطع عرفنا عضو ListViewItem من القائمة ListViewItem

```
System.Windows.Forms.ListViewItem.ListViewSubItem
AddressSubItem = new
System.Windows.Forms.ListViewItem.ListViewSubItem();

//=====
```

```
System.Windows.Forms.ListViewItem.ListViewSubItem
MobileSubItem = new
System.Windows.Forms.ListViewItem.ListViewSubItem();

//=====
```

وفي هذا المقطع عرفنا العضوين AddressSubItem و MobileSubItem من القائمة الجزئية ListViewItem .

```
//=====
listViewItem1.Text = NameText.Text;
listViewItem1.ImageIndex = index;
index++;
AddressSubItem.Text = AddressText.Text;
MobileSubItem.Text = MobileText.Text;
//=====
listViewItem1.SubItems.Add(AddressSubItem);
listViewItem1.SubItems.Add(MobileSubItem);
listView1.Items.Add(listViewItem1);
//=====
```

وفي هذا المقطع جعلنا تلك الأعضاء تأخذ القيم من الـ Text الموجودة في النموذج وأضفناها إلى القائمة والقائمة الفرعية.

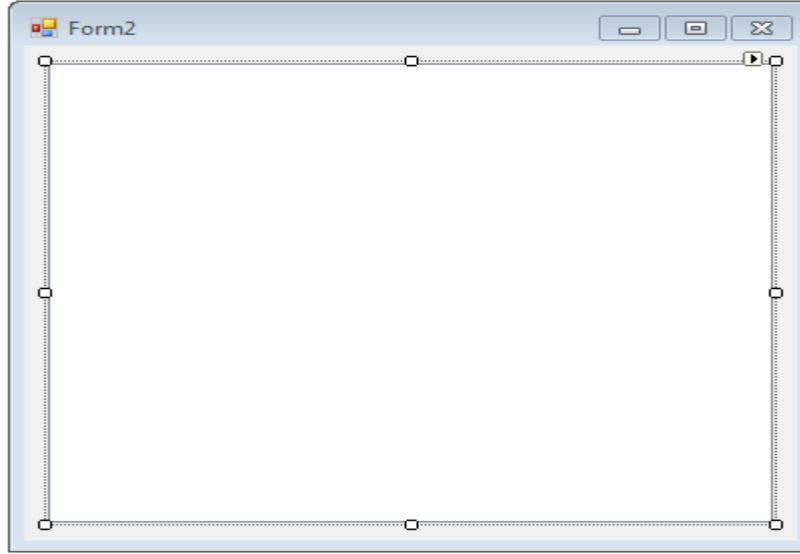
```
// شفرة الزر التحكم في العرض
private void button3_Click(object sender, EventArgs e)
{
    Random r = new Random();
    int I = r.Next(5);
    listView1.View = (View)I;
}
}
```

هذه الشفرة تولد أرقام عشوائية نستخدمها في تغيير طريق العرض حسب قيمة المتغير I والذي يحدد لنا رقم الطريق التي تستخدم لعرض محتوى ListView.

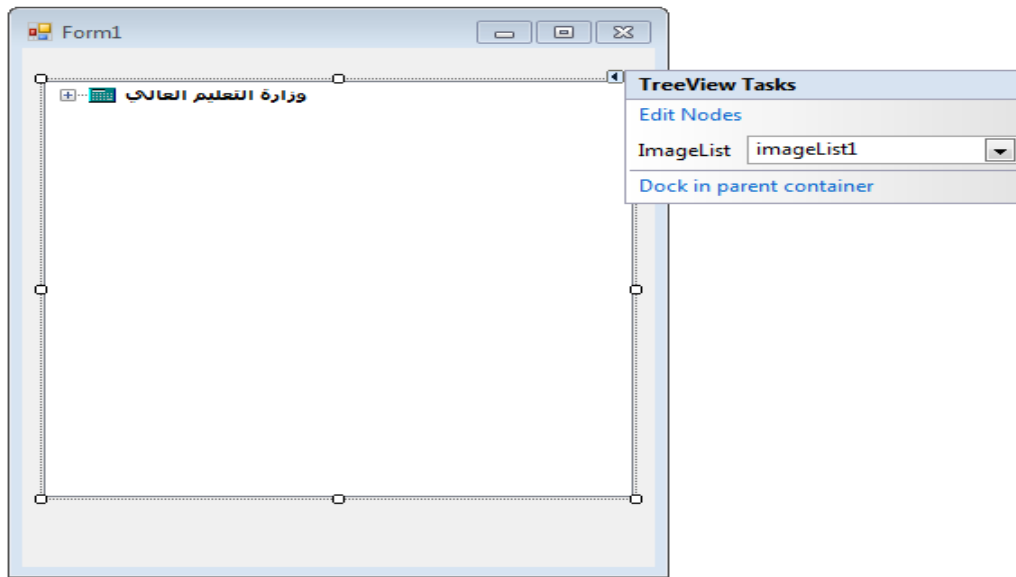
أداة (TreeView)

التدريب السادس عشر: < (التعامل مع TreeView):

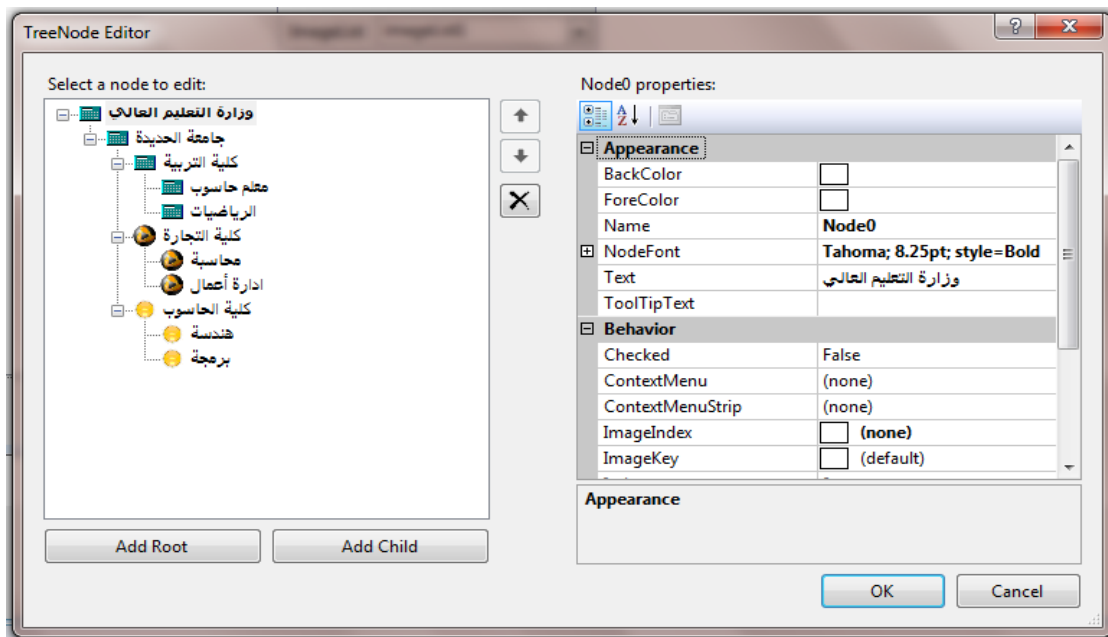
في هذا التدريب سنتعلم كيفية التعامل مع المكون أو الأداة TreeView بطريقة Wizarded أضف المكون ImageList كما تعلمت سابقا وأملئها بالأيقونات التي تريدها. أضف المكون TreeView وبقية العناصر الأخرى كما توضح الصورة.



1. أضغط السهم وأربط الخاصية ImageList بالمكون ImageList1 الموجود لديك كما توضح الصورة.
2. أضغط الارتباط edit Nodes وأضف الأسماء التي تريدها. وذلك بالضغط على الزر Add Root ومن خصائص كل عنصر غير الخاصية text بالاسم الذي تريده والخاصية ImageIndex أجعلها تأخذ رقم الصورة التي تم إدخالها من قبل في المكون ImageList.



3. عند إضافة ابناء لكل عنصر حدد عليّة وأضغظ **Add Child**. وغير الخصائص كما في السابق.



و الآن نستطيع إن نفعل برمجيا ما فعلناه سابقا:

- أضف نموذج جديد إلى المشروع
- أضف مكون من النوع **TreeView** إلى النموذج (الفورم)
- أضف مكون من النوع **Button** واجعل الخاصية **text** لهذا المكون تساوي **Add Childs**
- انقر على هذا الزر ثم اكتب الكود التالي:

```
private void button1_Click(object sender, EventArgs e)
{
    treeView1.BeginUpdate();
    treeView1.Nodes.Clear();
    treeView1.Nodes.Add("العلي التعليم وزارة");
}
```



```

treeView1.ImageIndex=0;

treeView1.Nodes[0].Nodes.Add("الحديقة جامعة");
treeView1.Nodes[0].ImageIndex = 0;

treeView1.Nodes[0].Nodes.Add("عدن جامعة");
treeView1.Nodes[0].Nodes[1].ImageIndex = 1;

treeView1.Nodes[0].Nodes[0].Nodes.Add("التربية كلية");
treeView1.Nodes[0].Nodes[0].Nodes[0].ImageIndex = 0;

treeView1.Nodes[0].Nodes[0].Nodes.Add("التجارة كلية");
treeView1.Nodes[0].Nodes[0].Nodes[1].ImageIndex = 1;

treeView1.Nodes[0].Nodes[0].Nodes.Add("الحاسوب كلية");
treeView1.Nodes[0].Nodes[0].Nodes[2].ImageIndex = 2;

treeView1.Nodes[0].Nodes[0].Nodes[0].Nodes.Add("معلم
حاسوب");
treeView1.Nodes[0].Nodes[0].Nodes[0].Nodes[0].ImageIndex =
0;

treeView1.Nodes[0].Nodes[0].Nodes[0].Nodes.Add("رياضيات");
treeView1.Nodes[0].Nodes[0].Nodes[0].Nodes[1].ImageIndex =
0;

treeView1.Nodes[0].Nodes[0].Nodes[1].Nodes.Add("محاسبة");
treeView1.Nodes[0].Nodes[0].Nodes[1].Nodes[0].ImageIndex =
1;

treeView1.Nodes[0].Nodes[0].Nodes[1].Nodes.Add("ادارة
اعمال");
treeView1.Nodes[0].Nodes[0].Nodes[1].Nodes[1].ImageIndex =
1;

treeView1.Nodes[0].Nodes[0].Nodes[2].Nodes.Add("برمجة");
treeView1.Nodes[0].Nodes[0].Nodes[2].Nodes[0].ImageIndex =
2;

treeView1.Nodes[0].Nodes[0].Nodes[2].Nodes.Add("هندسة");
treeView1.Nodes[0].Nodes[0].Nodes[2].Nodes[1].ImageIndex =
2;
treeView1.EndUpdate();
}

```

بعد التنفيذ شاهد الصورة التالية



صناديق الطباعة: (printDialog , printDocument , printPrivewDialog)

(التعامل مع مربعات الطباعة):

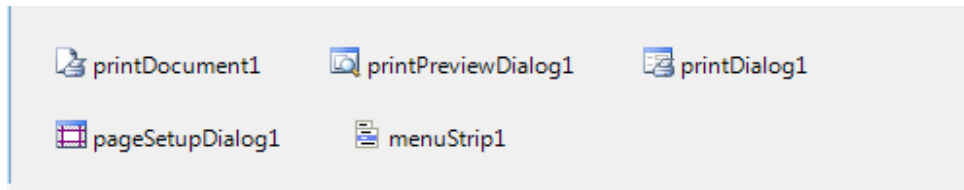
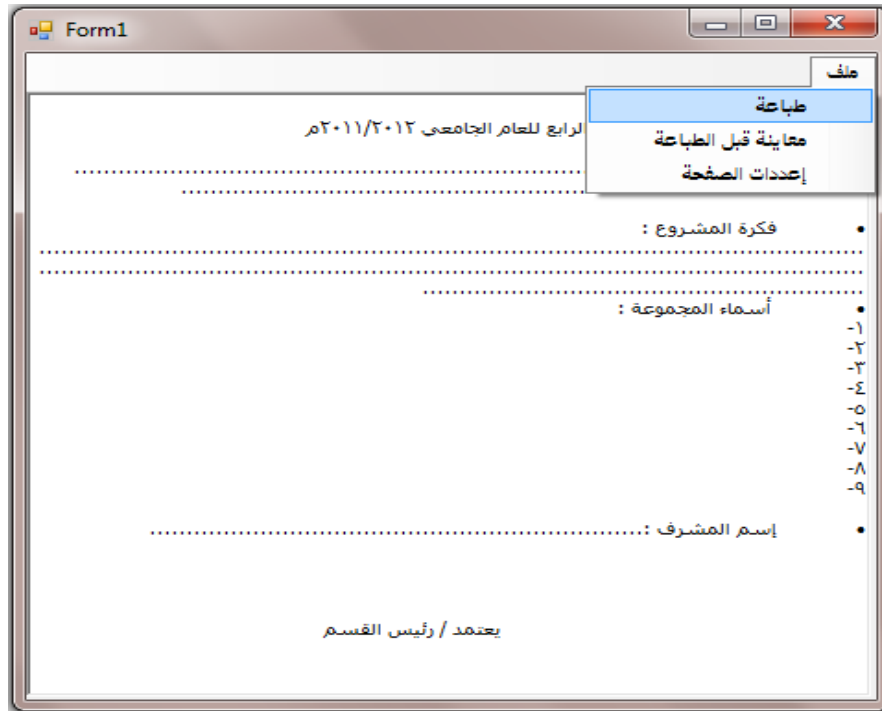
عند إسناد المكون `printDialog` إلى النموذج يجب إسناد الغرض أو المكون `printDocument` أيضا وإسناده إلى الخاصية `Document` من المكون `printDialog` وكذلك إسناد الغرض `printerSettings` (وهي تحوي إعدادات الطباعة) للمكون `printDocument` إلى الخاصية `printerSettings` من المكون `printDialog` ثم نستخدم الطريقة `print()` من المكون `printDocument` لطباعة الملف .

كذلك بالنسبة للمكون `printPrivewDialog` والمكون `pageSetupeDialog` عند إسناده للنموذج يجب إسناد الغرض أو المكون `printDocument` إلى الخاصية `Document` .

وسيتم توضيح ذلك من خلال الشفرة.

والآن :

- أفتح مشروع جديد وأضف نموذج إلى كود كما تعلمت سابقا.
- أضف المكون `printDialog` والمكون `printDocument` والمكون `PrintPrivewDialog`
- غير الخصائص للمكون `printDialog` وأجعلها كلها تأخذ القيمة `true` .
- قم بإضافة المكون `MenuStrip` والمكون `richTextBox1`.
- أجعل تصميم النموذج يظهر بالشكل التالي بعد إضافة النص للمكون `richTextBox1`.



بعدما قمت بتصميم النموذج غير الخاصية name لكل عنصر من عناصر ال-MenuStrip كما ستظهر في الشفرة التالية.

```
// شفرة العنصر طباعة
private void Print_Click(object sender, EventArgs e)
{
    printDialog1.Document = printDocument1;
    printDialog1.PrinterSettings = printDocument1.PrinterSettings;
    if (printDialog1.ShowDialog() == DialogResult.OK)
    {
        printDocument1.PrinterSettings =
        printDialog1.PrinterSettings;
        printDocument1.Print();
    }
}

// شفرة العنصر معاينة قبل الطباعة
private void PeforPrint_Click(object sender, EventArgs e)
{
    printPreviewDialog1.Document = printDocument1;
    printDocument1.PrintPage +=
    new System.Drawing.Printing.PrintPageEventHandler(
    printDocument1_PrintPage);
    if (printPreviewDialog1.ShowDialog() == DialogResult.OK)
```

```

printDocument1.Print();
}

// شفرة إعدادات الصفحة
private void Pagesetup_Click(object sender, EventArgs e)
{
    pageSetupDialog1.Document = printDocument1;
    pageSetupDialog1.PrinterSettings =
    printDocument1.PrinterSettings;

    if (pageSetupDialog1.ShowDialog() == DialogResult.OK)
    {
        printDocument1.PrinterSettings =
        pageSetupDialog1.PrinterSettings;
        printDocument1.Print();
    }
}

// شفرة الحدث printpage
private void printDocument1_PrintPage(object sender,
System.Drawing.Printing.PrintPageEventArgs e)
{
    System.Drawing.Font printFont =
    new Font("Arial", 25, FontStyle.Regular);

    e.Graphics.DrawString(richTextBox1.Text, printFont,
    Brushes.Black, 10, 10);
}

```

هذه الشفرة تنفذ عندما يتم الضغط على زر الطباعة .

الوقت والتاريخ	
الوقت	10:28:12 م
التاريخ	08/06/2012
الساعة	10
الدقيقة	28
الثانية	12
السنة	2012
الشهر	6
اليوم	8
اليوم من الأسبوع	Friday

قم بتغيير الاسم البرمجي (الخاصية name) لكل label كما توضح الشفرة.

```
private void Form1_Load(object sender, EventArgs e)
{
    TimeandDate.Text = DateTime.Now.ToString();
    Time.Text = DateTime.Now.ToLongTimeString();
    Date.Text = DateTime.Now.ToLongDateString();
    Hour.Text = (DateTime.Now.Hour-12).ToString();
    Minute.Text = DateTime.Now.Minute.ToString();
    Second.Text = DateTime.Now.Second.ToString();
    Yere.Text = DateTime.Now.Year.ToString();
    Month.Text = DateTime.Now.Month.ToString();
    Day.Text = DateTime.Now.Day.ToString();
    DayOfWeek.Text =
    DateTime.Now.DayOfWeek.ToString();
}
```

بعد التنفيذ

قبل التنفيذ



timer1

الشفرة

```
private void timer1_Tick(object sender, EventArgs e)
{
    if (button1.Enabled == false && textBox1.Text != "")
        if (DateTime.Now.ToShortTimeString()
==textBox1.Text)
        {
            axRealAudio1.Source =
Application.StartupPath + "\\A.mp3";
            axRealAudio1.DoPlay();
        }
}

private void button1_Click(object sender, EventArgs
e)
{
    button1.Enabled = false;
    textBox1.ReadOnly = true;
    button2.Enabled = true;
}

private void button2_Click(object sender, EventArgs
e)
{
    axRealAudio1.DoStop();
    button1.Enabled = true;
    button2.Enabled = false;
    textBox1.ReadOnly = false;
}
```