

دورة كن خبيراً

في لغة

C#



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ
(ربنا لا تؤاخذنا ان نسرينا او اخطأنا
ربنا ولا تحمل علينا اصرا كما حملته
على الذين من قبلنا ربنا ولا تحملنا
ما لا طاقة لنا به واعف عنا و اغفر
لنا و ارحمنا انت مولانا فانصرنا على
القوم الكافرين)
صدق الله العظيم

(آية: 286)

الإهداء

للمجباله

تقريباً.....ورخواناً.

إلى:

من أفنى ذاته في ذات الحق . فناء الحبة في الأرض لتنتبت سبع سنابل ، في كل سنبله مائة حبة . منقذ البشرية (محمد صل الله عليه وسلم) .

إلى:

منسوعوا وشقوا النعم بالراحه الهناء الذين لم يخلوا ابشيمنا جلد فعا فطر يقال نجاحا الذين علمونا أن نرتقي سلم الحياة بحكمة صبر مناكو نيهما ثلثيا لأعلو فخر يوا اعتزاز يقود تيفيحياتي..... (أبي) .

إلى:

الينا بيعة التيا تملال لعطاء المنح كنسعادتنا بخيو طمنسوجة منقلوبهنا إلى نكنا تفيعينها دمعهو في قلبها خفقوهو عللسانهاد عاء منبعالتضحيقو بحر الحنان وحضن . الأمان..... (والديا الحبيبة) .

إلى:

منحبه ميجر يفيعر وقنا ويله جذكر اهمفؤ ادنا إلى أر كانا لعائلة منينتنظر ونمينا أنكونا خاهما الأكبر وأملأنيك برو اليكونوا سنـد الشـدة وعونالزمن..... (أخوانيو أخواتي) .

إلى:

منعلمونا حروف فامنذهبوكلمات مندررو عبارات منأسمو أجلعبار اتقيا لعلمنا المنصاغو لنا علمهم حروف فامنف كرهمنارة تنير لنا سيرة العلم والنجاحو الذين ما بخلوا علينا وساعدونا ولو بكلمة إلى .. (أساتذتنا الأوفياء) .

(فإلى هؤلاء نهدي ثمرة جهدنا المتواضع هذا)

إعداد :

م / محمد كمال أحمد

Email: kamal2902796@yahoo.com

م / شريف محمد رضا

Email: sherifmansiti@gmail.com

الفهرس

(1) بيئة التصميم Design environment

(2) المتغيرات Variables

(3) الجمل الشرطية Conditional Logic

(4) جمل التكرار والدورات الشرطية Loop

(5) الدوال Methods

(6) المصفوفات Arrays

(7) الأحداث Events

(8) التعامل مع الملفات Manipulating Files

(9) تصحيح أخطاء التطبيق Debugging

(10) التاريخ والوقت في لغة C#

(11) برمجة الكائنات الموجهة Object-Oriented (OOP) Programming

(12) برمجة تطبيقات الويندوز

(13) مشاريع للتعامل مع قواعد البيانات

(الإصدار الثاني من الكتاب)

بإذن الله تعالى

مقدمة

بيئة التصميم

Design environment بيئة التصميم (1)

كل البرامج تحتوى على بيئة التشغيل الخاصة بها وهى بمثابة الحاوية التى تضم كل مكونات البرنامج التى قد يحتاجها المبرمج فى عمله .

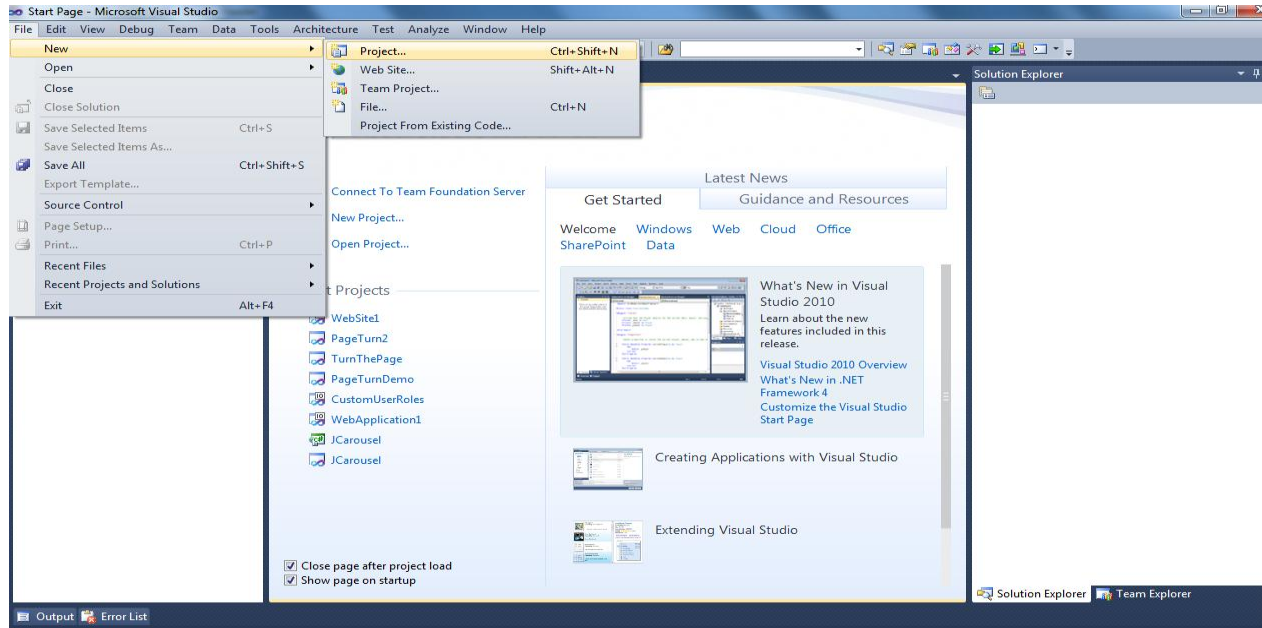
للدخول إلى نافذة التصميم الخاص بمحرر التطوير (Visual Studio)

أولا قم بإنشاء مشروع جديد وذلك عن طريق فتح برنامج

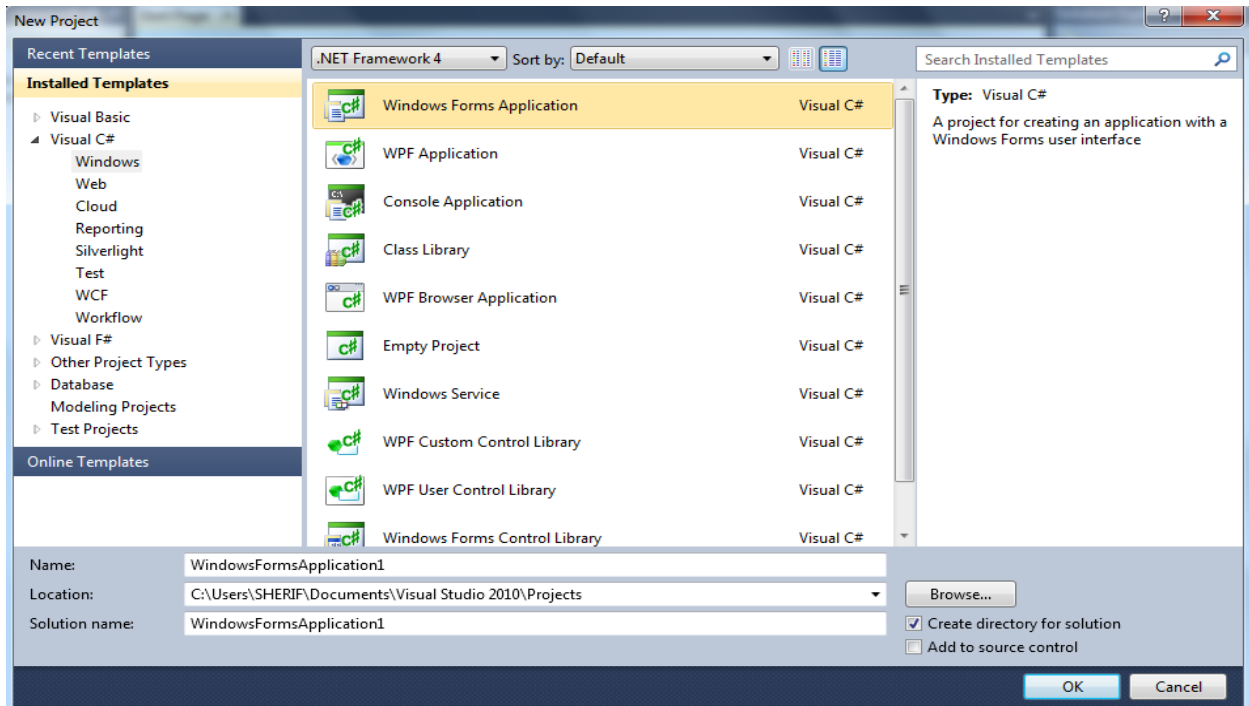
(Visual Studio 2010) أو (Visual Studio 2012) من قائمة Start <-- ثم

الذهاب إلى قائمة File واختيار New Project

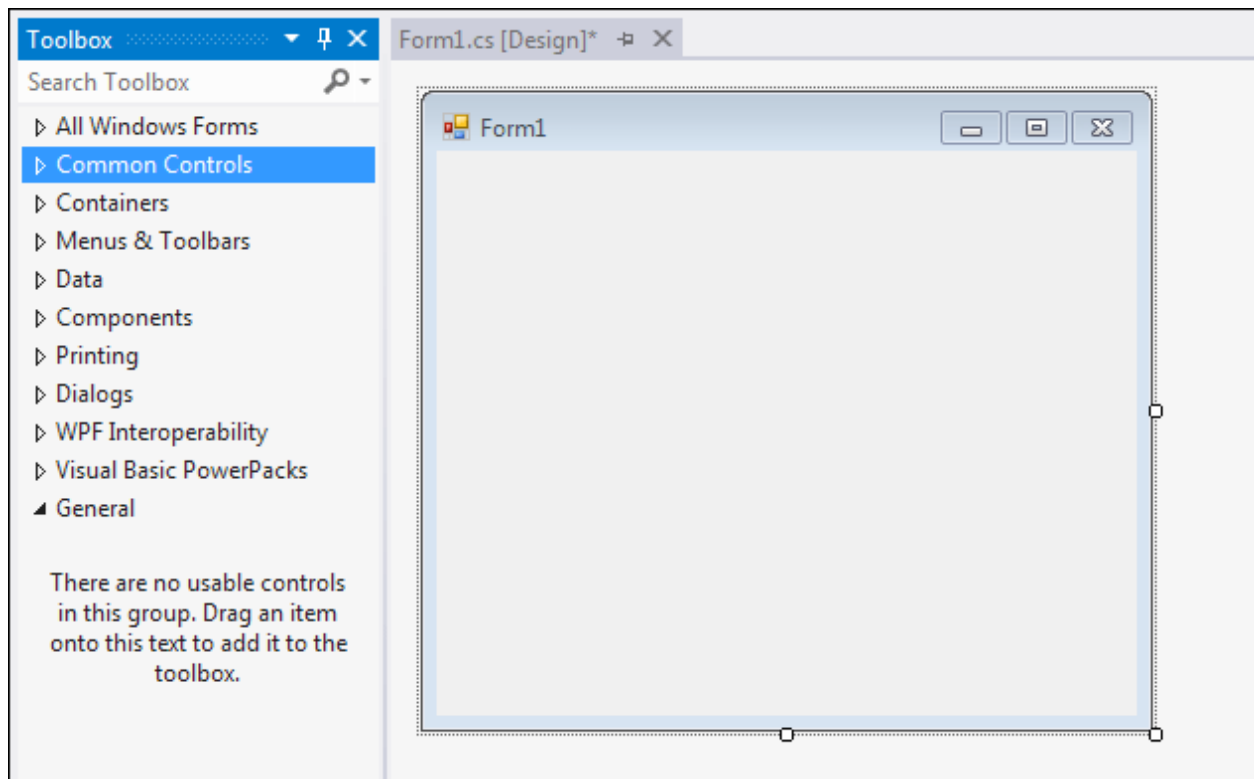
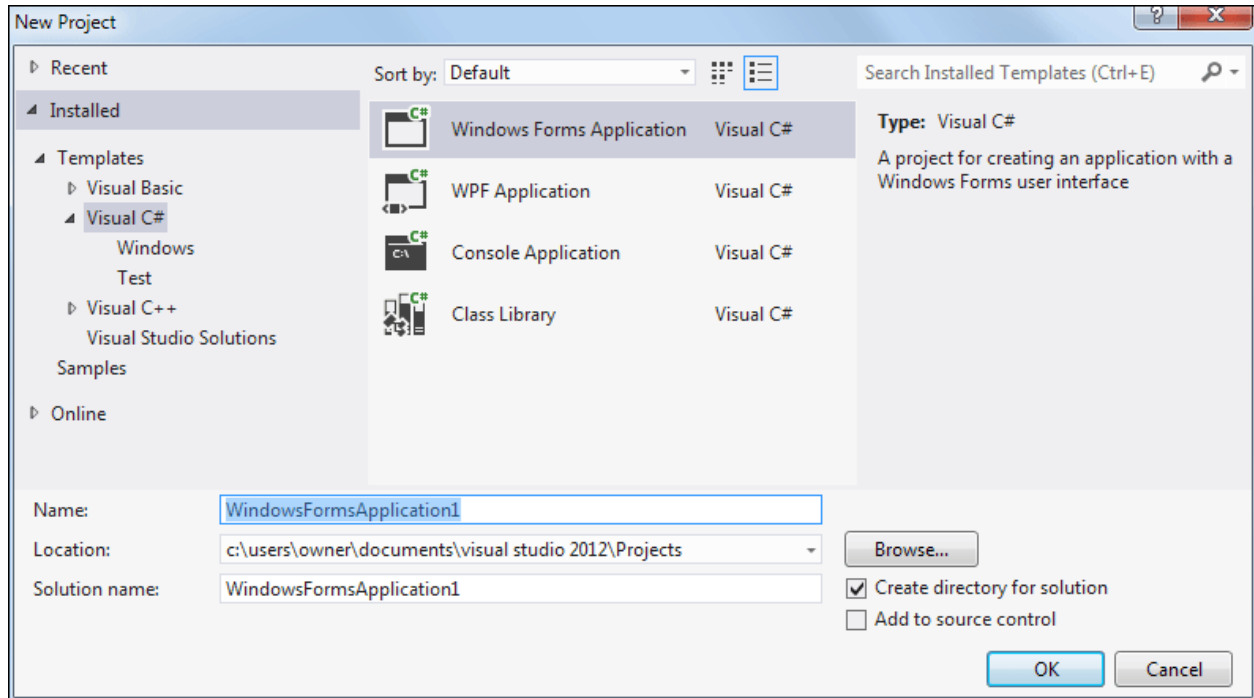
(Visual Studio 2010)



ستظهر لك هذه النافذة كما يمكنك تغيير اسم البرنامج ومسار حفظه ثم نضغط OK.



(Visual Studio 2012)

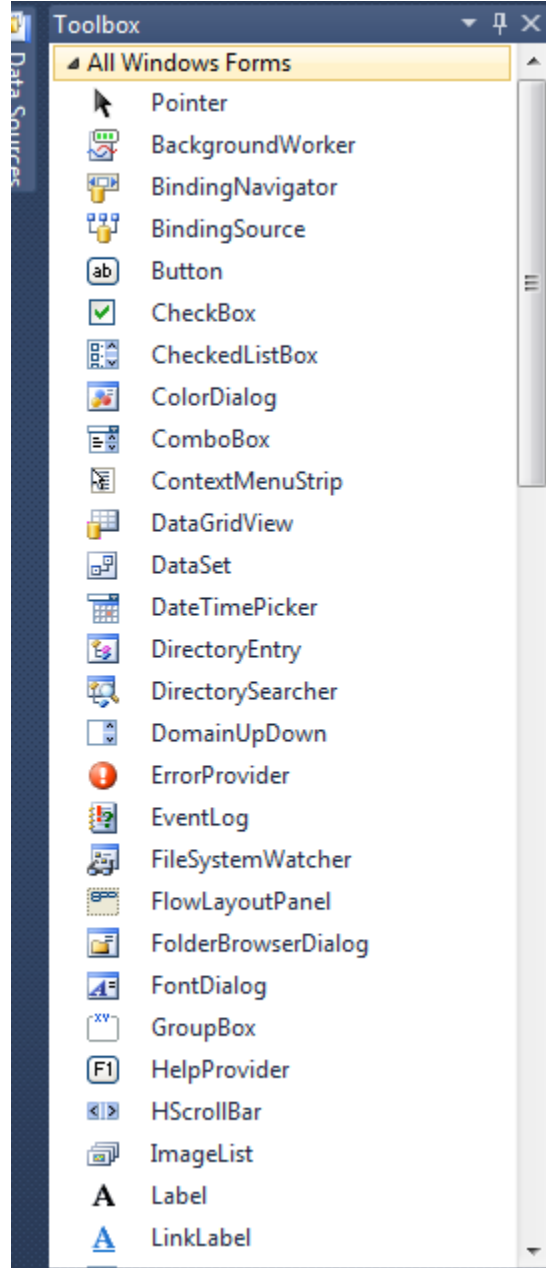


النافذة رقم (1) :

تسمى Toolbox وهى تضم كل الأدوات التى قد يحتاجها برنامجك

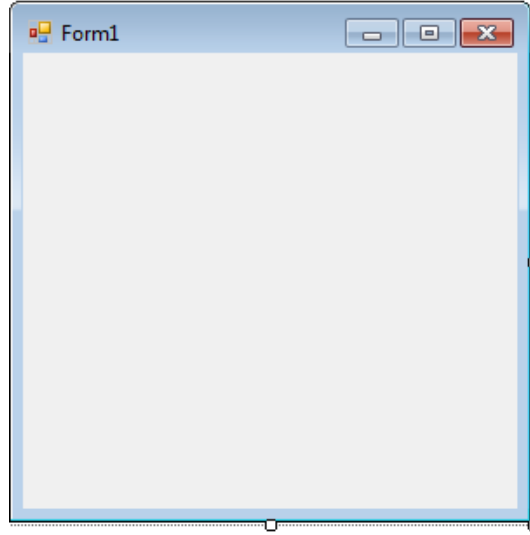
(Buttons, Menus,) إن لم تكن ظاهرة لديك فإذهب إلى قائمة View

ثم اخترها لكي تظهر أو اكتب بالضغط على الاختصار **Ctrl + Alt + X** كما يلي:



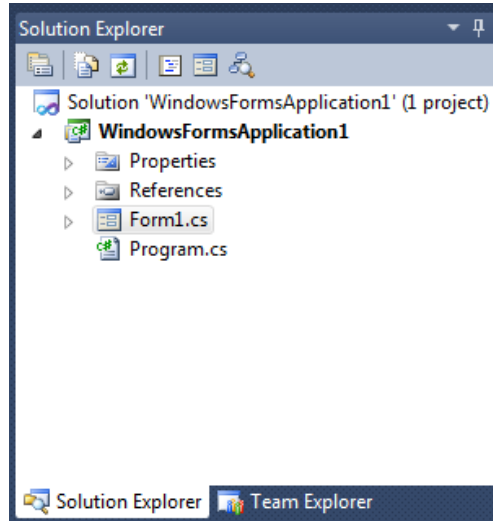
النافذة رقم (2) :

تسمى Form الذى سنضع عليه الأدوات اللازمة لبناء المشروع كما يمكنك إضافة أكثر من واجهة لمشروعك بعد ذلك كما يلي :



النافذة رقم (3) :

تسمى Solution Explorer متصفح المشروع وسمى كذلك لأنه يعرض كل الملفات التى يضمها المشروع ويمكن إظهاره من قائمة View واختيار Solution Explorer أو الإكتفاء بالضغط على الإختصار **Ctrl + Alt + L** كما يلي :



Variables المتغيرات (2)

2.1 المتغيرات النصية Strings

المتغيرات النصية تتكون من مجموعة الحروف والأرقام الموجودة في لوحة المفاتيح بجهاز الكمبيوتر .

لتعريف متغير نصي كالتالي :

(Type) Variable Name = Value

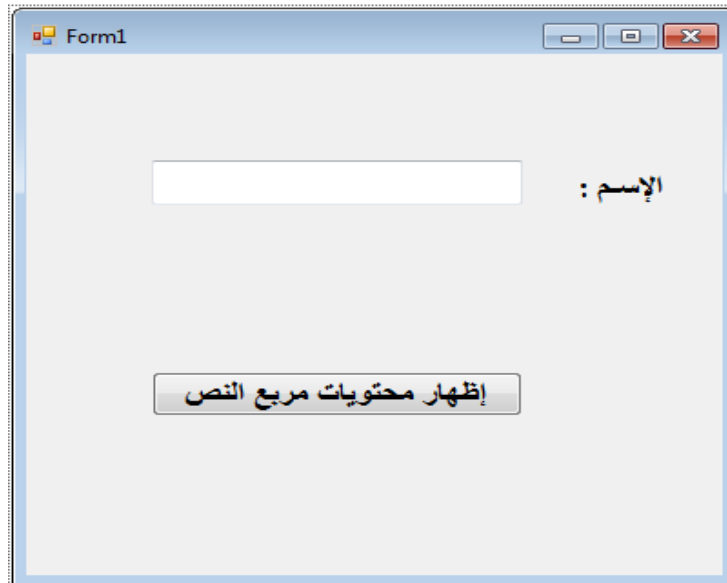
String strVariable = "Internet";

برنامج (1)

أنشئ مشروع جديد من قائمة **File -> New -> Project** من فيجوال ستوديو

واختر نوع المشروع **Windows Forms Application** باسم **Strings** .

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:



1- Label:

Name: label1

Text: الإسم

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Button:

Name: button1

Text: إظهار محتويات مربع بالنص

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- TextBox1

Name: textBox1

Text: إظهار محتويات مربع بالنص

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-to-left: Yes

بالضغط مرتين بالماوس على Button الخاص بإظهار النص ينشئ لنا

الحدث Button1_Click()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    /*
        This is a Comment
    */

    //This is a String Example

    string firstName;
    firstName = textBox1.Text;

    MessageBox.Show(" محتو مبربعالنص هو " + firstName);
}
```

لاحظ عملية دمج النصوص في MessageBox بالعلامة + وهي تعمل

على دمج محتويات النصوص مع المتغيرات النصية كما هو موضح بالكود.

كما تلاحظ كتابة التعليقات على الكود أعلى باللون الأخضر ولعمل التعليقات
Comments بلغة C# يتم كتابة // في أول السطر الذي تكتبه هذا لكتابة

سطر واحد من التعليق أما إذا رغبت في كتابة عدة أسطر من التعليقات لفهم
ما يتم عمله بالكود وإسترجاع ما يتم تنفيذه يتم كتابة

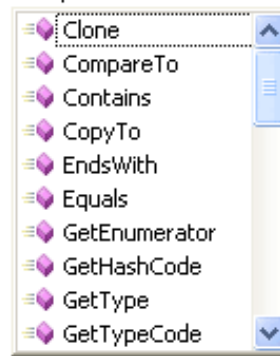
/* يتم كتابة أسطر التعليقات هنا */ /

عمليات على المتغيرات النصية (String Manipulation) :

1- إجراء عمليات على المتغير النصي بحيث يظهر بالحروف الكبيرة (Capital) تستخدم الدالة `ToUpper()` لذلك .

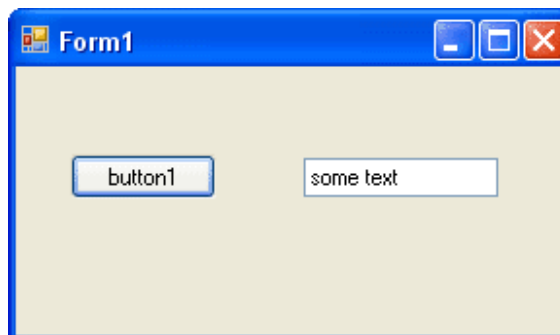
```
private void button1_Click(object sender, EventArgs e)
{
    string stringVar = textBox1.Text;

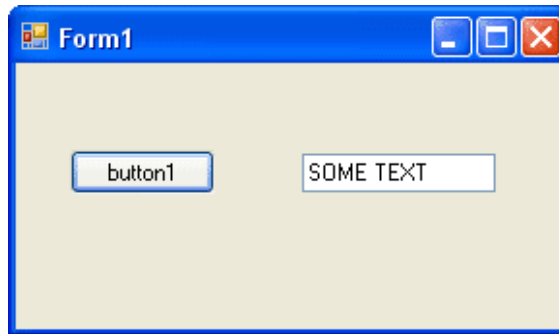
    textBox1.Text = stringVar.
}
```



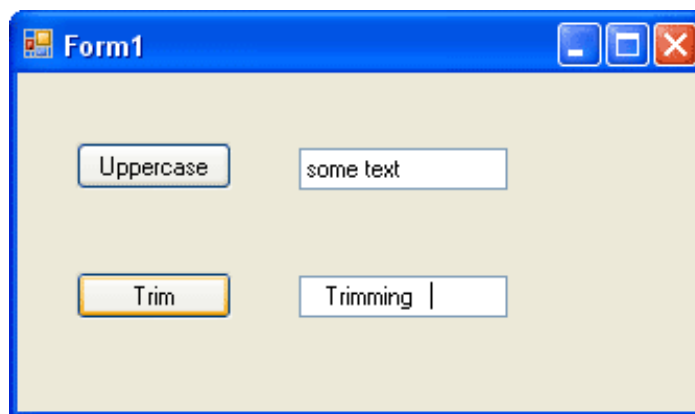
```
private void button1_Click(object sender, EventArgs e)
{
    string stringVar = textBox1.Text;

    textBox1.Text = stringVar.ToUpper();
}
```





2 لحذف المسافات الفارغة في بداية النص وآخره يتم استخدام الدالة Trim()



```
private void button2_Click(object sender, EventArgs e)
{
    string stringTrim = textBox2.Text;
    stringTrim = stringTrim.Trim();
    int stringLength = stringTrim.Length;
    MessageBox.Show( stringLength.ToString() );
}
```

3 لحذف المسافات الفارغة أو الحروف في بداية النص يتم استخدام الدالة TrimStart()

```
stringTrim = stringTrim.TrimStart( trimChars );
```

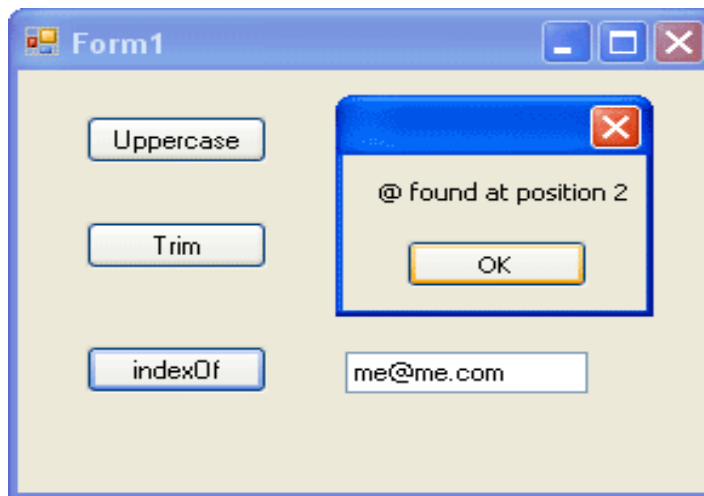
4 للبحث عن حرف أو مجموعة حروف في النص يتم استخدام الدالة Contains()

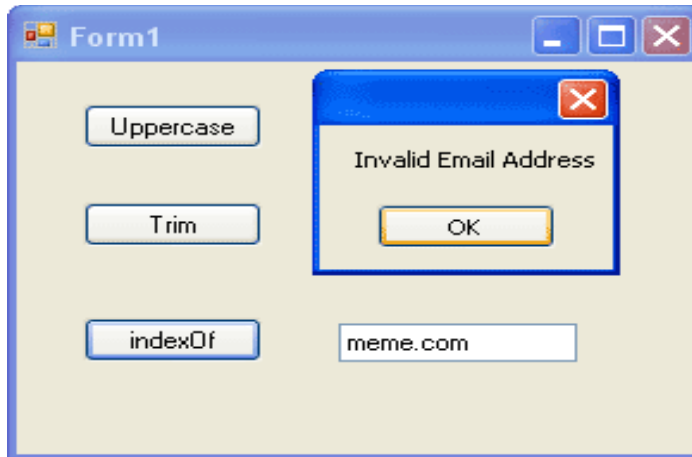
```
private void button4_Click(object sender, EventArgs e)
{
    string stringVar = "Some Text---";

    if (stringVar.Contains("-"))
    {
        MessageBox.Show("TRUE");

        //CALL HYPHEN REMOVE METHOD HERE
    }
}
```

5 لمعرفة موقع حرف معين من بين مجموعة حروف يتم استخدام الدالة IndexOf()



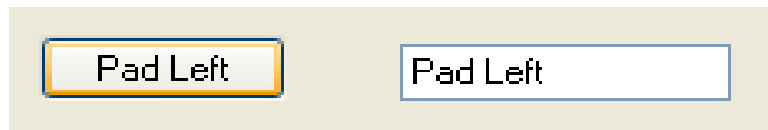


6 لإضافة حرف أو مجموعة من حروف لنص معين يتم استخدام الدالة `Insert()`

```
string someText = "Some Text";
```

```
someText = someText.Insert( 5, "More " );
```

7 لإضافة حرف أو مجموعة من حروف لنص معين يتم استخدام الدالة `PadLeft()`



```
string paddingLeft = textBox5.Text;
```

```
paddingLeft = paddingLeft.PadLeft( 20 );
```



```
paddingLeft = paddingLeft.PadLeft(20 , '*');
```



8 لحذف حرف أو مجموعة من حروف لنص معين يتم استخدام الدالة Remove()

```
string oldString = "some text text text";  
MessageBox.Show(oldString);  
string newString = oldString.Remove(10, 9);  
MessageBox.Show(newString);
```

9 لإحلال حرف أو مجموعة من الحروف محل نص معين تستخدم الدالة Replace(old_Word , new_Word)

```
string spellingError = "mistak";  
spellingError = spellingError.Replace(spellingError,  
"mistake");
```

10 - للحصول على مقطع من نص معين يبدأ من حرف معين نستخدم الدالة

Substring(start_position, no_of_chars_to_grab)

```
private void button11_Click(object sender, EventArgs e)
{
    string emailAddress = "me@me.com";
    string result = "";

    result = emailAddress.Substring(5, 4);

    if (result == ".com")
    {
        MessageBox.Show("Email Address OK");
    }
    else
    {
        MessageBox.Show("Bad Email Address");
    }
}
```

11 - دالة تستخدم لفصل مجموعة من الحروف من نص معين بناءً على حرف معين ويمكنك وضع النص الناتج في مصفوفة تسمى الدالة Split()

```
private void button12_Click(object sender, EventArgs e)
{
    string lineOfText = "item1, item2, item3";

    string[] wordArray = lineOfText.Split(',');

    MessageBox.Show( wordArray[0] );
    MessageBox.Show( wordArray[1] );
    MessageBox.Show( wordArray[2] );
}
```

12 - دالة تستخدم لتجميع مجموعة من الحروف من نص معين بناءً على حرف معين تسمى الدالة Join()

```
private void button13_Click(object sender, EventArgs e)
{
    string lineOfText = "item1, item2, item3";
    string[] wordArray = lineOfText.Split(',');

    string joinedText;

    joinedText = String.Join("-", wordArray);

    MessageBox.Show(joinedText);
}
```

هي متغيرات رقمية ولا تقبل نصوص ولا فواصل عشرية
ويتم تعريفها كالتالي :

(Type) Variable Name = Value

```
int intValue = 10;
```

2.3 المتغيرات الرقمية Float

هي متغيرات رقمية ولا تقبل نصوص وتقبل فواصل عشرية وتتسع لسبعين
خانات من الأرقام ولا بد من كتابة الحرف F نهاية القيمة المعطاة للمتغير
وحجمها من ذاكرة الكمبيوتر هو :

Float: 1.5×10^{-45} to 3.4×10^{38}

ويتم تعريفها كالتالي :

(Type) Variable Name = Value

```
float floatValue = 10.56F;
```

2.4 المتغيرات الرقمية Double

هي متغيرات رقمية ولا تقبل نصوص وتقبل فواصل عشرية وتتسع لـ 16 خانة من الأرقام

وحجمها من ذاكرة الكمبيوتر هو :

Double: 5.0×10^{-324} to 1.7×10^{308}

ويتم تعريفها كالتالي :

(Type) Variable Name = Value

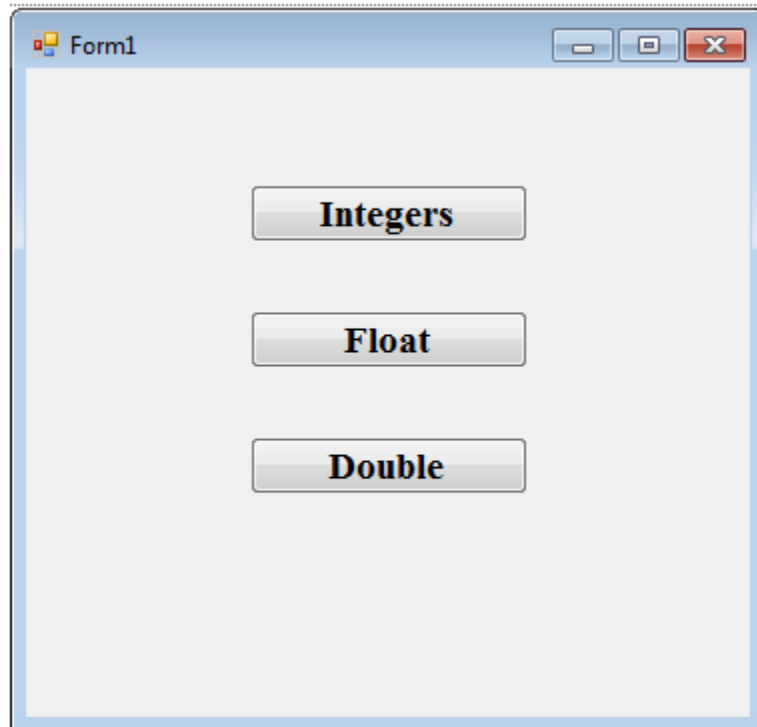
doubledoubleVariable = 10.56;

برنامج (2)

أنشئ مشروع جديد من قائمة **File -> New -> Project** من فيجوال ستوديو

واختر نوع المشروع **Windows Forms Application** باسم **Numbers** .

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:



1- Button:

Name: button1

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Button:

Name: button2

Text: Float

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- Button:

Name: button3

Text: Double

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

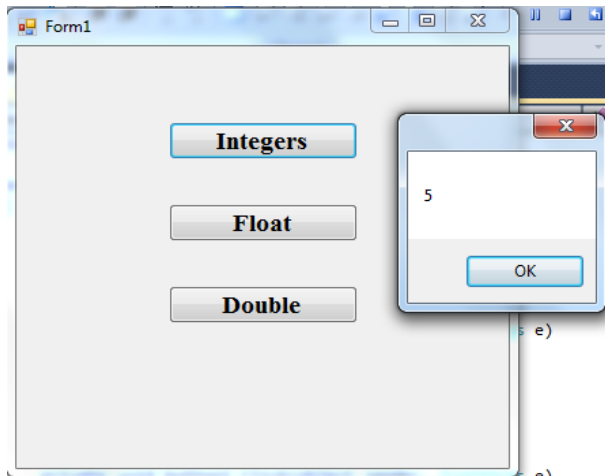
Bold: True

بالضغط مرتين بالماوس على Button Integers ينشئ لنا

الحدث Button1_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    int myInteger = 5;
    MessageBox.Show(myInteger.ToString ());
}
```

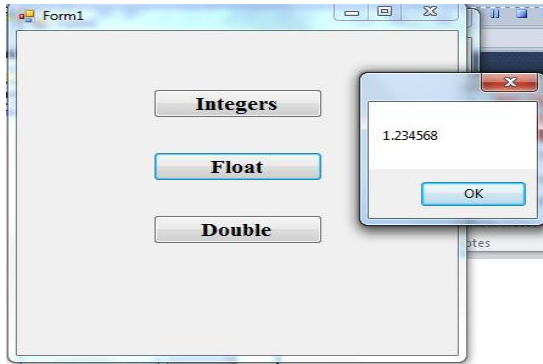


بالضغط مرتين بالماوس على Button Float ينشئ لنا

الحدث Button2_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button2_Click(object sender, EventArgs e)
{
    float myFloat = 1.2345678F;
    MessageBox.Show(myFloat.ToString());
}
```

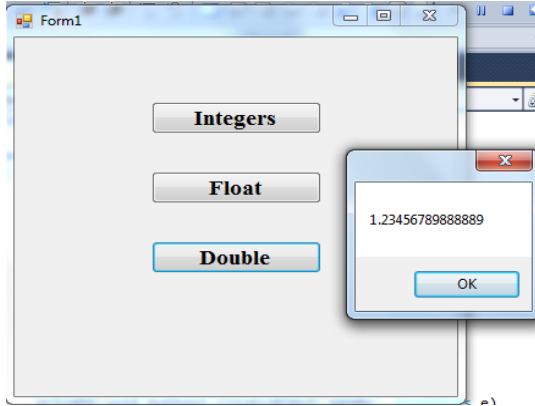


بالضغط مرتين بالماوس على Button Double ينشء لنا

الحدث Button3_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button3_Click(object sender, EventArgs e)
{
    double myDouble = 1.23456789888888888888;
    MessageBox.Show(myDouble.ToString());
}
```



Conditional Logic **الجملة الشرطية** (3)

1 جملة IF :

تقوم باختبار الشرط إذا تحقق الشرط و كان صحيحا تنفذ الجمل التي بعدها بداخل الأقواس {} وإذا كان هناك شرط ثانى لجملة IF يسمى Else if إذا تحقق الشرط و كان صحيحا تنفذ الجمل التي بعدها بداخل الأقواس {} أما إذا لم يتحقق نفذ الجملة بعد الأقواس مباشرة لو كان هناك else تنفذ إذا لم يتم تنفيذ أى من الشروط.

IF (Condition)

```
{  
    Statements;  
}
```

IF (Condition)

```
{  
    Statements;  
}
```

Else if (Condition)

```
{  
    Statements;  
}
```

Else

{

Statements;

}

برنامج (2)

أنشئ مشروع جديد من قائمة **File -> New -> Project** من فيجوال ستوديو
واختر نوع المشروع **Windows Forms Application** باسم **Calculator**.

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: button1

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Button:

Name: button2

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- Button:

Name: button3

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Button:

Name: button4

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

5- Button:

Name: button5

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

6- Button:

Name: button6

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

7- Button:

Name: button7

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

8- Button:

Name: button8

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

9- Button:

Name: button9

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

10- Button:

Name: btnZero

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

11- Button:

Name: btnFraction

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

12- Button:

Name: btnAdd

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

13- Button:

Name: btnSubtract

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

14- Button:

Name: btnMultiply

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

15- Button:

Name: btnDivide

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

16- Button:

Name: btnEqual

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

17- Button:

Name: btnClear

Text: Integers

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

18- TextBox1

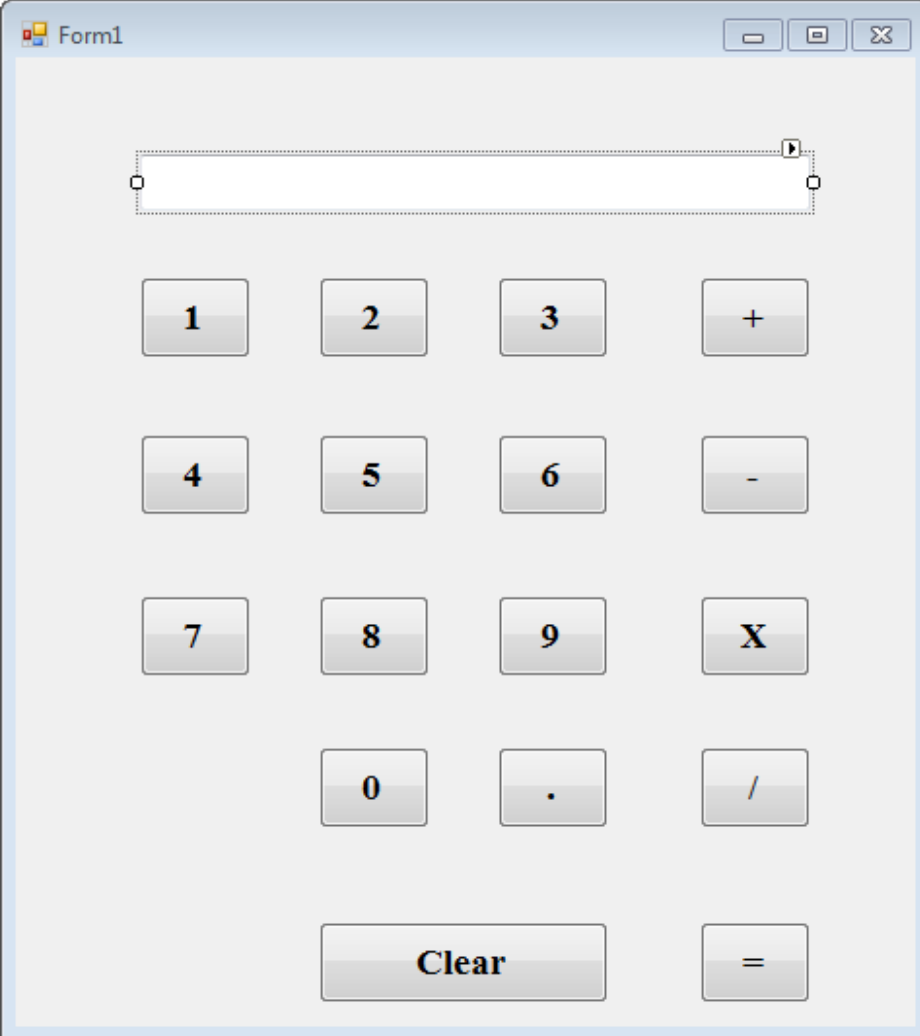
Name: txtDisplay

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-to-left: Yes



The image shows a screenshot of a Windows application window titled "Form1". The window contains a calculator interface. At the top, there is a text box with a dotted border and a small arrow on the right side. Below the text box is a grid of buttons. The first row contains buttons for "1", "2", "3", and "+". The second row contains buttons for "4", "5", "6", and "-". The third row contains buttons for "7", "8", "9", and "X". The fourth row contains buttons for "0", ".", and "/". At the bottom, there is a "Clear" button and an "=" button. The buttons are arranged in a grid that is right-to-left, as indicated by the text above.

وبعد الضغط مرتين على كل حدث خاص بالأزرار على الفورم أكتب سطور الكود التالية :

```
privatevoid button1_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button1.Text;
}

privatevoid button2_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button2.Text;
}

privatevoid button3_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button3.Text;
}

privatevoid button4_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button4.Text;
}

privatevoid button5_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button5.Text;
}

privatevoid button6_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button6.Text;
}

privatevoid button7_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button7.Text;
}

privatevoid button8_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button8.Text;
}

privatevoid button9_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button9.Text;
}

privatevoid btnFraction_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + btnFraction.Text;
}

privatevoid btnZero_Click(object sender, EventArgs e)
{
```

```

        txtDisplay.Text = txtDisplay.Text + btnZero.Text;
    }

privatevoid btnClear_Click(object sender, EventArgs e)
    {
        txtDisplay.Clear();
    }

//Calculator Variables

double total1 = 0;

double total2 = 0;

bool plusButton = false;
bool subtractButton = false;
bool multiplyButton = false;
bool divideButton = false;

privatevoid btnAdd_Click(object sender, EventArgs e)
    {

        total1 = double.Parse(txtDisplay.Text) + total1 ;
        txtDisplay.Clear();

//TRUE
        plusButton = true ;
        subtractButton = false;
        multiplyButton = false;
        divideButton = false;

    }

privatevoid btnEqual_Click(object sender, EventArgs e)
    {

if (plusButton == true )
    {
        total2 = total1 + double.Parse(txtDisplay.Text) ;
    }
elseif (subtractButton ==true)
    {
        total2 = total1- double.Parse(txtDisplay.Text) ;
    }
elseif (multiplyButton == true)
    {
        total2 = total1 *double.Parse(txtDisplay.Text);
    }
    }

```

```

elseif (divideButton == true)
    {
        total2 = total1 / double.Parse(txtDisplay.Text);
    }

    txtDisplay.Text = total2.ToString();
    total1 = 0;

}

privatevoid btnSubtract_Click(object sender, EventArgs e)
{
    total1 = double.Parse(txtDisplay.Text)- total1 ;
    txtDisplay.Clear();

    plusButton = false;

//TRUE
    subtractButton = true;
    multiplyButton = false;
    divideButton = false;
}

privatevoid btnMultiply_Click(object sender, EventArgs e)
{
    total1 = double.Parse(txtDisplay.Text);
    txtDisplay.Clear();

    plusButton = false;
    subtractButton = false;

//TRUE
    multiplyButton = true;
    divideButton = false;
}

privatevoid btnDivide_Click(object sender, EventArgs e)
{
    total1 = double.Parse(txtDisplay.Text) ;
    txtDisplay.Clear();

    plusButton = false;
    subtractButton = false;
    multiplyButton = false ;

//TRUE
    divideButton = true;
}

```

Switch جملة 2

تقوم باختبار الشرط بناءا على مجموعة من الحالات تسمى Cases إذا تحققت حالة من الحالات بداخل جملة Switch يتم تنفيذ ما بداخل Case ويتم الخروج من جملة Switch بكلمة **break;**

```
Switch ()
{
    Case "Your_Test_Here_1":
        // Your Code Here!
        Break;
    Case "Your_Test_Here_2":
        // Your Code Here!
        Break;
    Case "Your_Test_Here_2":
        // Your Code Here!
        Break;
    Default:
        // Your Code Here!
        Break;
}
```

برنامج (4)

أنشئ مشروع جديد من قائمة **File -> New -> Project** من فيجوال ستوديو

واختار نوع المشروع **Windows Forms Application** باسم

. Calculator Using Switch Statement

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها نفس الخصائص السابقة ثم أضف الكود التالي :

```
privatevoid button1_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button1.Text;
}

privatevoid button2_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button2.Text;
}

privatevoid button3_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button3.Text;
}

privatevoid button4_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button4.Text;
}

privatevoid button5_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button5.Text;
}

privatevoid button6_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button6.Text;
}

privatevoid button7_Click(object sender, EventArgs e)
{
    txtDisplay.Text = txtDisplay.Text + button7.Text;
}

privatevoid button8_Click(object sender, EventArgs e)
{
```



```

        txtDisplay.Text = txtDisplay.Text + button8.Text;
    }

privatevoid button9_Click(object sender, EventArgs e)
    {
        txtDisplay.Text = txtDisplay.Text + button9.Text;
    }

privatevoid btnFraction_Click(object sender, EventArgs e)
    {
        txtDisplay.Text = txtDisplay.Text + btnFraction.Text;
    }

privatevoid btnZero_Click(object sender, EventArgs e)
    {
        txtDisplay.Text = txtDisplay.Text + btnZero.Text;
    }

privatevoid btnClear_Click(object sender, EventArgs e)
    {
        txtDisplay.Clear();
    }

//Calculator Variables

double total1 = 0;

double total2 = 0;

bool plusButton = false;
bool subtractButton = false;
bool multiplyButton = false;
bool divideButton = false;

string theOperator;

privatevoid btnAdd_Click(object sender, EventArgs e)
    {

        total1 = double.Parse(txtDisplay.Text) + total1 ;
        txtDisplay.Clear();

        theOperator = "+";

//TRUE

        plusButton = true ;
        subtractButton = false;
        multiplyButton = false;
        divideButton = false;

    }

```

```

privatevoid btnEqual_Click(object sender, EventArgs e)
    {

switch (theOperator)
    {

case "+":
        total2 = total1 + double.Parse(txtDisplay.Text);
break;

case "-":
        total2 = total1 - double.Parse(txtDisplay.Text);
break;

case "*":
        total2 = total1 * double.Parse(txtDisplay.Text);
break;

case "/":
        total2 = total1 / double.Parse(txtDisplay.Text);
break;

default:
break;

    }

    txtDisplay.Text = total2.ToString();
    total1 = 0;

    }

privatevoid btnSubtract_Click(object sender, EventArgs e)
    {

    total1 = double.Parse(txtDisplay.Text)- total1 ;
    txtDisplay.Clear();

    theOperator = "-";

    plusButton = false;

//TRUE
    subtractButton = true;
    multiplyButton = false;
    divideButton = false;
    }

privatevoid btnMultiply_Click(object sender, EventArgs e)
    {

    total1 = double.Parse(txtDisplay.Text);
    txtDisplay.Clear();

    theOperator = "*";

```

```
        plusButton = false;
        subtractButton = false;
//TRUE
        multiplyButton = true;
        divideButton = false;
    }
privatevoid btnDivide_Click(object sender, EventArgs e)
    {
        total1 = double.Parse(txtDisplay.Text) ;
        txtDisplay.Clear();

        theOperator = "/";

        plusButton = false;
        subtractButton = false;
        multiplyButton = false ;
//TRUE
        divideButton = true;
    }
```

(4) جمل التكرار و الدورات الشرطية Loops

1- جملة For() :

تقوم هذه الجملة بتكرار عدد معين من الجمل بناءا على عدد معين يتم تحديده داخل جملة For

```
for (StartValue ; End_Value ; Update_Expression)
{
    Statements;
}
```

```
for (int i = 0; i < length; i++)
{
    Statements;
}
```

برنامج (5)

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو
واختر نوع المشروع Windows Forms Application باسم

.For Loop

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: button1

Text: For Loop

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Textbox:

Name: textBox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-to-left: Yes

3- Textbox:

Name: textBox2

Right-to-left: Yes

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Label:

Name: label1

Text : بداية التكرار

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Label:

Name: label2

Text : نهاية التكرار

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

5- Listbox:

Name: listBox1

Right-to-left: Yes

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



The screenshot shows a Windows application window titled "Form1". Inside the window, there are two text input fields. The first field is labeled "بداية التكرار" (Start of loop) and the second is labeled "نهاية التكرار" (End of loop). To the right of the first field is a button labeled "For Loop". Below these fields is a large list box labeled "listBox1". The list box is currently empty.

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button1_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    listBox1.Items.Clear();

    int answer = 0;

    int loopStart = int.Parse(textBox1.Text);
    int loopEnd = int.Parse(textBox2.Text);

    for (int i = loopStart; i <= loopEnd; i++)
    {
        answer = answer + i;
        listBox1.Items.Add(answer.ToString());
    }
}
```

برنامج (6) تدريب عملي

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختار نوع المشروع Windows Forms Application باسم

For Loop Total Number لإظهار مجموع الأرقام من 1 إلى 100 وإظهار

الناتج في MessageBox و Label

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: button1

Text: مجموع الأعداد من 1-100

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

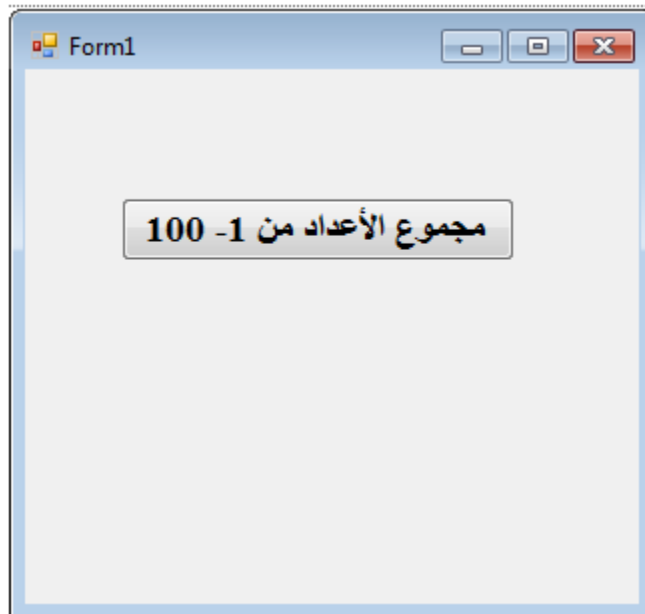
2- Label:

Name: label2

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button1_Click ()

قم بكتابة الكود التالي بداخله كالتالي :


```
private void button1_Click(object sender, EventArgs e)
{
    int answer = 0;
    for (int i = 0; i < 101; i++)
    {
        answer = answer + 1;
        label1.Text = "الناتج =" + answer.ToString ();
    }
    MessageBox.Show("الناتج =" + answer.ToString());
}
```

2- جملة () While :

تقوم هذه الجملة بتكرار عدد معين من الجمل بناءً على عدد معين يتم تحديده داخل شرط الجملة مع وضع عداد للجمل بين أقواس الجملة {} While

برنامج (7)

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو واختر نوع المشروع Windows Forms Application باسم

While Loop

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: button1

Text: While Loop

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Textbox:

Name: textBox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-to-left: Yes

3- Textbox:

Name: textBox2

Right-to-left: Yes

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Label:

Name: label1

Text : بداية التكرار

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Label:

Name: label2

Text : نهاية التكرار

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

5- List box:

Name: listBox1

Right-to-left: Yes

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

The screenshot shows a Windows application window titled "Form1". Inside the window, there are two text input fields. The first field is labeled "بداية التكرار" (Start of loop) and the second is labeled "نهاية التكرار" (End of loop). To the right of the first input field is a button labeled "While Loop". Below these elements is a large list box labeled "listBox1". The text in the form is right-aligned, consistent with the "Right-to-left: Yes" property mentioned in the text.

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button1_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    int loopStart = int.Parse(textBox1.Text);
    int loopEnd = int.Parse(textBox2.Text);

    int i = 0;
    int answer = 0;
    while (i < loopEnd )
    {
        answer = answer + i;
        listBox1.Items.Add(answer.ToString());

        i++;
    }
}
```

3- جملة { } While();

تقوم هذه الجملة بتكرار عدد معين من الجمل بناءً على عدد معين يتم تحديده داخل شرط الجملة مع وضع عداد للجمل بين أقواس الجملة do While(); والفرق بينها وبين جملة While() أنه في جملة do{}While() يتم تنفيذ جمل الكود بداخل do على الأقل مرة واحدة قبل تنفيذ الشرط .

برنامج (8) تدريب عملي

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

DoWhileLoop

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: button1

Text:do While Loop

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Textbox:

Name: textBox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-to-left: Yes

3- Textbox:

Name: textBox2

Right-to-left: Yes

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Label:

Name: label1

Text : بداية التكرار

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Label:

Name: label2

Text : نهاية التكرار

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

5- List box:

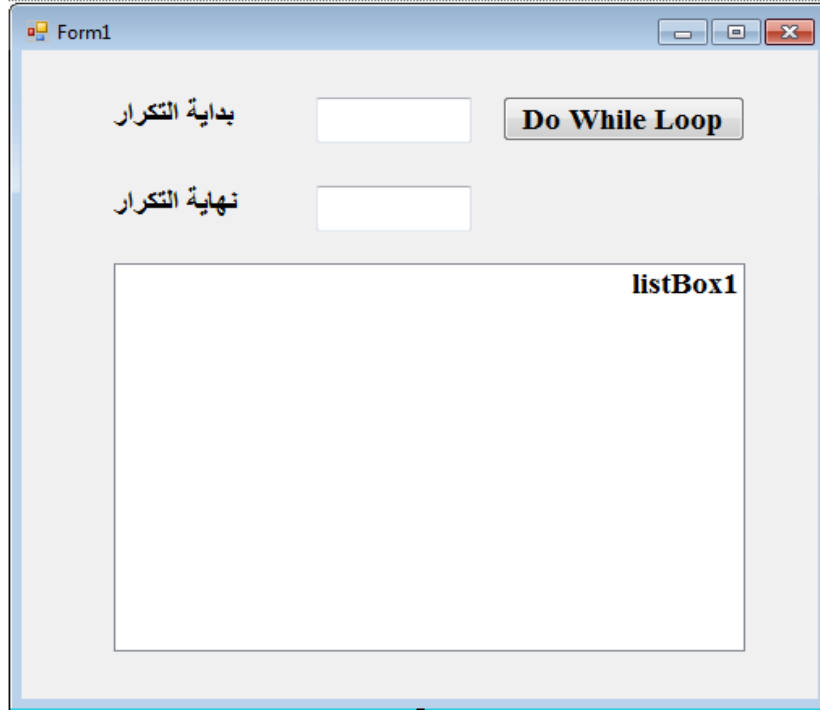
Name: listBox1

Right-to-left: Yes

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button1_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    int loopStart = int.Parse(textBox1.Text);
    int loopEnd = int.Parse(textBox2.Text);

    int i = 0;
    int answer = 0;

    do
    {
        answer = answer + i;
        listBox1.Items.Add(answer.ToString());

        i++;
    } while (i < loopEnd);
}
```

Methods الدوال (5)

الدوال تقوم بتنفيذ مجموعة من سطور الكود داخل الأقواس {} ويمكن أن ترجع قيمة كانت نصية أو رقمية ويمكن أن نكتب عدد من المدخلات تسمى Parameters أو نتركها بدون Parameters ويمكن للقيمة أن نضعها في متغير وصيغة الدوال هي كما يلي :

(Type)MethodName(Param1 , Param2 , ,ParamN)

```
{  
  
    Statements;  
  
    Return ;  
  
}
```

برنامج (9) تدريب عملي

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم Methods

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Label:

Name: label1

Text : الرقم الأول

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Label:

Name: label2

Text : الرقمائى

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- Label:

Name: lblResult

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Textbox:

Name: textBox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-to-left: Yes

5- Textbox:

Name: textBox2

Right-to-left: Yes

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

6- Button:

Name: button1

Text: جمع

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

7- Button:

Name: button1

Text: طرح

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

8- Button:

Name: button1

Text: ضرب

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

9- Button:

Name: button1

Text:قسمة

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

كتابة الدوال التالية لإستدعاءها فى كل Button :

```
// Add Method
private int Add(int Num1, int Num2)
{
    int answer = 0;

    answer = Num1 + Num2;

    return answer;
}

// Subtract Method
private int Subtract(int Num1, int Num2)
{
    int answer = 0;

    answer = Num1 - Num2;

    return answer;
}

// Multiply Method
```

```

private int Multiply(int Num1, int Num2)
{
    int answer = 0;

    answer = Num1 * Num2;

    return answer;
}

// Divide Method
private int Divide(int Num1, int Num2)
{
    int answer = 0;

    answer = Num1 / Num2;

    return answer;
}

// Clear Method
private void Clear()
{
    textBox1.Text = String.Empty;
    textBox2.Text = "";
    lblResult.Text = "";
}

```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button1_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```

private void button1_Click(object sender, EventArgs e)
{
    int Number1 = int.Parse(textBox1.Text);
    int Number2 = int.Parse(textBox2.Text);

    int result = 0;

    result = Add(Number1 ,Number2);

    lblResult.Text = " = الناتج " + result.ToString();

}

```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button2_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid button2_Click(object sender, EventArgs e)
{
    int Number1 = int.Parse(textBox1.Text);
    int Number2 = int.Parse(textBox2.Text);

    int result = 0;

    result = Subtract(Number1, Number2);

    lblResult.Text = "الناتج =" + result.ToString();
}
```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button3_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid button3_Click(object sender, EventArgs e)
{
    int Number1 = int.Parse(textBox1.Text);
    int Number2 = int.Parse(textBox2.Text);

    int result = 0;

    result = Multiply(Number1, Number2);

    lblResult.Text = "الناتج =" + result.ToString();
}
```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button4_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button4_Click(object sender, EventArgs e)
{

    int Number1 = int.Parse(textBox1.Text);
    int Number2 = int.Parse(textBox2.Text);

    int result = 0;

    result = Divide(Number1, Number2);

    lblResult.Text = "النتيجة =" + result.ToString();
}
```

Arrays (6) المصفوفات

هي مجموعة من العناصر لها نفس النوع مرتبة بما يسمى Index ويبدأ بالرقم 0 ويتم تعريفها كالتالي :

`Type array_name[position_in_array] = array_value;`

وتحديد عدد One-Dimensional Arrays لتعريف المصفوفة ذات البعد واحد العناصر التي يمكن أن تتسع لها المصفوفة

`int[] lottery_numbers = new int[49];`

لإعطاء قيم للمصفوفة بطريقتين كالتالي :

(أ) الطريقة الأولى :

`Numbers [0] = 1;`

`Numbers [1] = 2;`

`Numbers [2] = 3;`

`Numbers [3] = 4;`

(ب) الطريقة الثانية :

`int[] Numbers = new int[4] { 1, 2, 3, 4 };`

برنامج (10)

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

OneDimentionArray مصفوفة ذات البعد الواحد

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: button1

Text:OneDimensionArray

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- List box:

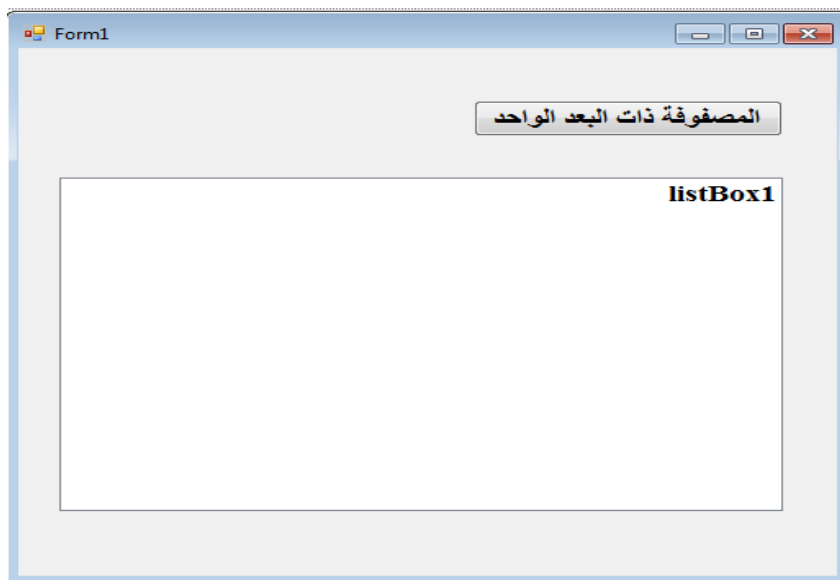
Name: listBox1

Right-to-left: Yes

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button1_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    int[] Numbers = new int[4];

    Numbers[0] = 1;
    Numbers[1] = 2;
    Numbers[2] = 3;
    Numbers[3] = 4;

    // بطريقة for
    for (int i = 0; i < Numbers.Length; i++)
    {
        listBox1.Items.Add(Numbers[i]);
    }

    // بطريقة foreach
    foreach (int item in Numbers)
    {
        listBox1.Items.Add(item);
    }
}
```

جملة Foreach() :

وهي مثل جملة for ولكنها ليس لها عدد محدد من العناصر بل تقوم هذه الجملة بعمل تكرار على مجموعة من العناصر داخل مصفوفة

```
foreach (type item in Collection)
{
    Statements;
}
```

تعريف مصفوفة ذات البعدين : Two-Dimensional Arrays :

(أ) الطريقة الأولى :

`String [,] Names ;`

(أ) الطريقة الثانية :

`String [][] Names ;`

	Column 0	Column 1	Column 2	Column 3
Row 0	<code>a[0][0]</code>	<code>a[0][1]</code>	<code>a[0][2]</code>	<code>a[0][3]</code>
Row 1	<code>a[1][0]</code>	<code>a[1][1]</code>	<code>a[1][2]</code>	<code>a[1][3]</code>
Row 2	<code>a[2][0]</code>	<code>a[2][1]</code>	<code>a[2][2]</code>	<code>a[2][3]</code>

```
int[,] a =int[3,4]={
{0,1,2,3},/* initializers for row indexed by 0 */
{4,5,6,7},/* initializers for row indexed by 1 */
{8,9,10,11}/* initializers for row indexed by 2 */
};
```

وللوصول لعنصر في المصفوفة يتم بطريقة Subscript وهو معرفة

Row index & Column index of the Array كالتالى :

```
int val = a[2,3];
```

تعريف مصفوفة ذات ثلاثة أبعاد : Three-Dimensional Arrays :

(ب) الطريقة الأولى :

`String [, ,] Names ;`

(ب) الطريقة الثانية :

`String [][][] Names ;`

وهكذا تتم تعريف المصفوفات كلما أرت زيادة البعد لها

برنامج (11)

أنشئ مشروع جديد من قائمة **File -> New -> Project** من فيجوال ستوديو

واختار نوع المشروع **Windows Forms Application** باسم

TwoDimentionArray مصفوفة ذات البعدين

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: button1

Text:TwoDimensionArray

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Rich Textbox:

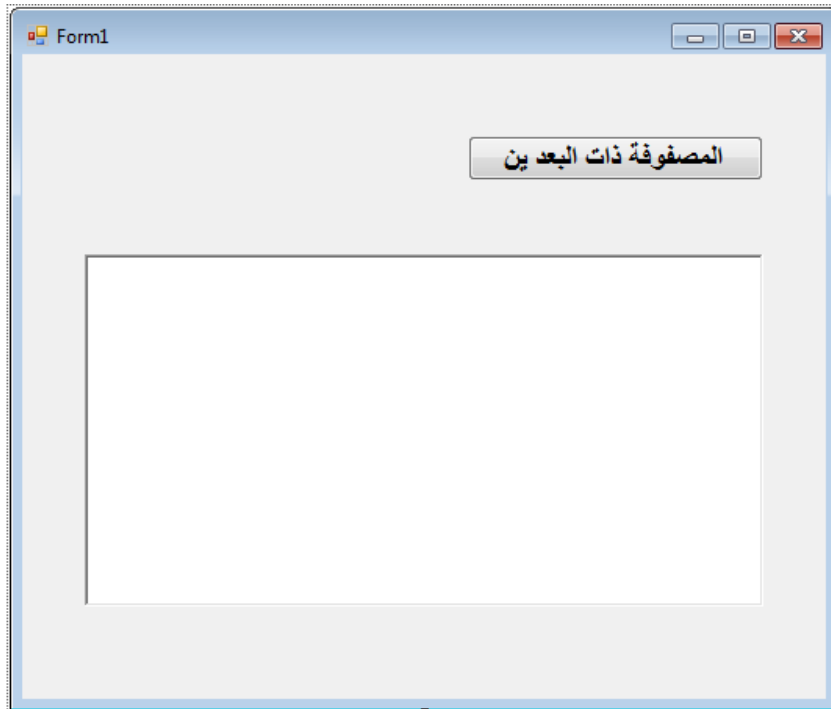
Name: richTextBox1

Right-to-left: Yes

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث Button1_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    /* an array with 5 rows and 2 columns*/
    int[,] a = new int[5, 2] { { 0, 0 }, { 1, 2 }, { 2, 4 }, { 3, 6 }, { 4, 8 } };

    int i, j;

    /* output each array element's value */
    for (i = 0; i < 5; i++)
    {
        for (j = 0; j < 2; j++)
        {

        }
    }

    //Subscripting

    // 0 0
    // 1 2
    // 2 4
    // 3 6
    // 4 8
}
```

```
// |  
// v  
  
//a[0,0] = 0  
//a[0,1] = 0  
//a[1,0] = 1  
//a[1,1] = 2  
//a[2,0] = 2  
//a[2,1] = 4  
//a[3,0] = 3  
//a[3,1] = 6  
//a[4,0] = 4  
//a[4,1] = 8
```

```
richTextBox1.Text += "a["+i+", "+j+"] = " + a[i, j] + "\n";
```

```
    }  
}
```

Events الأحداث (7)

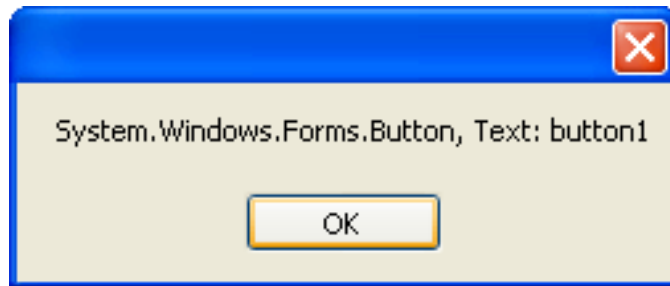
1 - حدث Click

انقر فوق الحدث للزر:

```
private void button1_Click(object sender, EventArgs e)
{
}
}
```

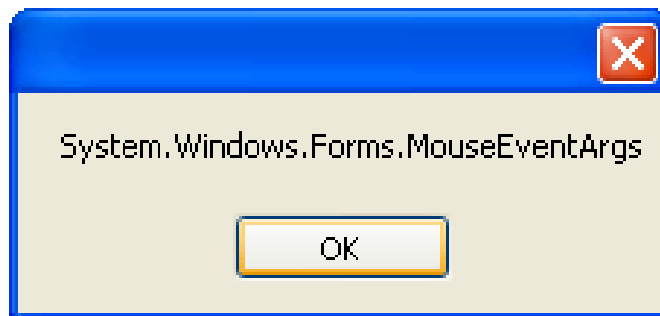
تشغيل البرنامج وانقر فوق الزر. يجب أن تشاهد هذا:

```
MessageBox.Show( sender.ToString() );
```



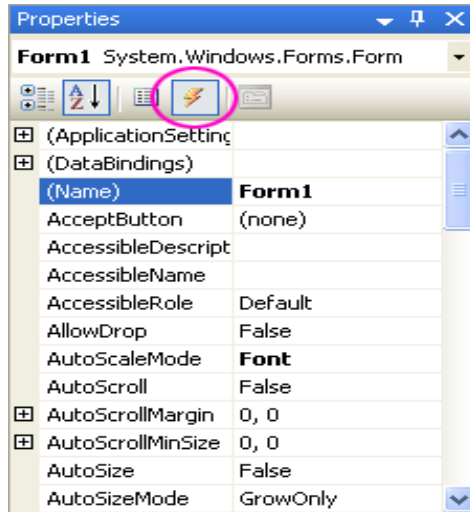
```
MessageBox.Show( e.ToString());
```

مربع النص سوف يظهر ما يلي :

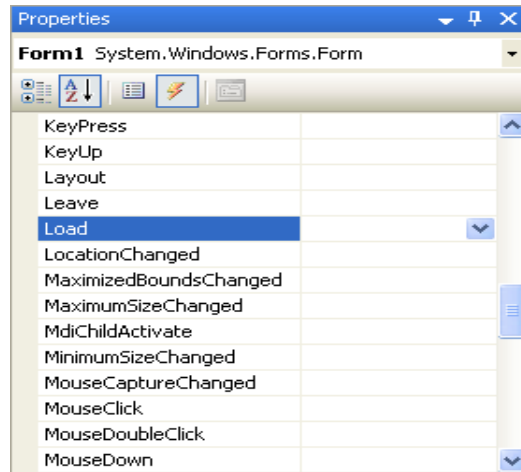


The MouseDown Event in C# .NET - 2

الخصائص: لترى ما هي الأحداث المتاحة للنموذج نفسه، انقر فوق الجزء العلوي من قائمة الخصائص، كما في الصورة أدناه،



عند الضغط على الجزء العلوي الأصفر في قائمة الخصائص ستظهر الأحداث كما يلي :



كما يمكن استخدام جملة if لاختبار أي من ازرار الماوس تقوم باختياره كما يلي :

```

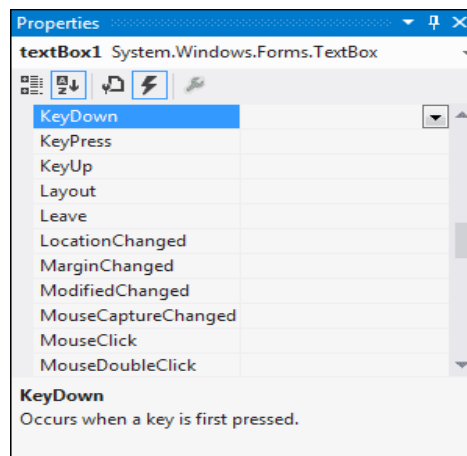
private void Form1_MouseDown(object sender, MouseEventArgs e)
{
    if (e.Button == MouseButtons.Left)
    {
        MessageBox.Show("Left button clicked");
    }
    else if (e.Button == MouseButtons.Right)
    {
        MessageBox.Show("Right button clicked");
    }
}

```

عند تشغيل البرنامج والنقر فوق أي من أزرار الماوس على النموذج يجب أن تشاهد عرض مربع رسالة.

The KeyDown Event in C# .NET - 3

من قائمة الخصائص حدد حدث KeyDown لمربع النص الخاص :



```

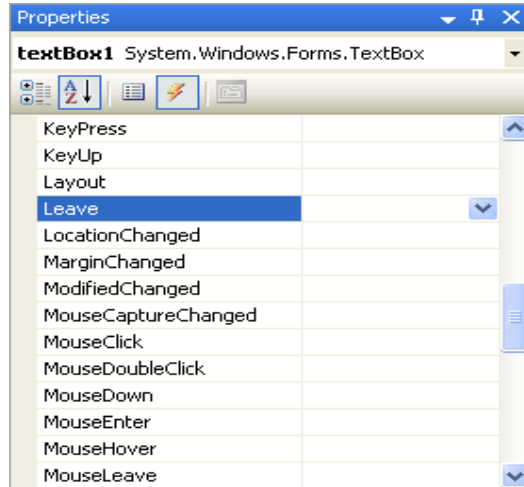
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    |
}

if (e.KeyData == Keys.A)
{
    MessageBox.Show("Letter A Pressed");
}

```

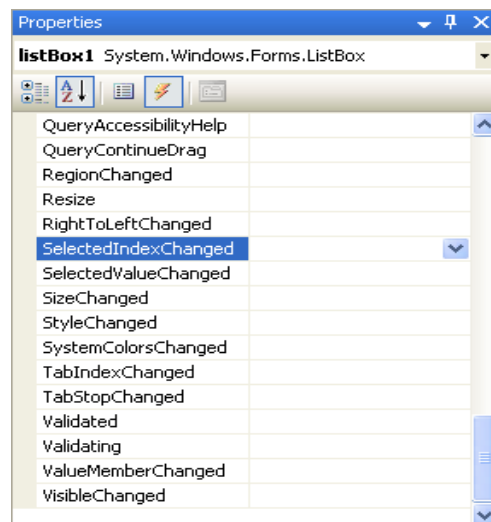
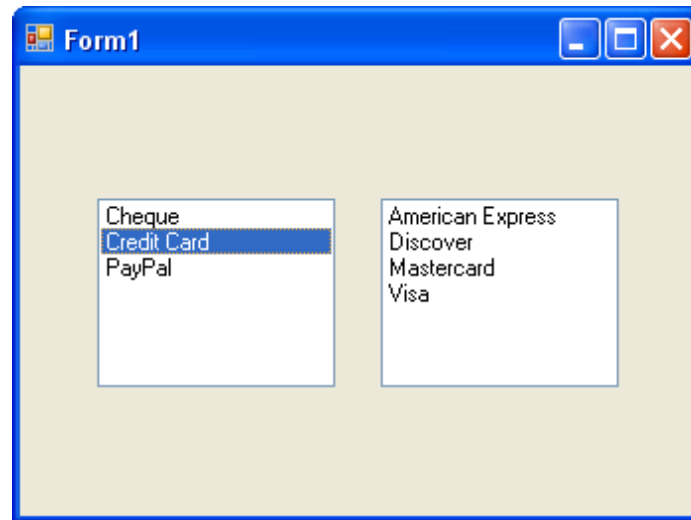

The Leave Event in C# .NET - 4

هذا الحدث مفيد جدا يمكنك استخدامها لمربعات النص هو حدث Leave. فإنه يسمح لك للتحقق من صحة مربع النص عندما يحاول المستخدم إلى تركها.



```
private void textBox1_Leave(object sender, EventArgs e)
{
    if (textBox1.Text == "")
    {
        MessageBox.Show("You can't leave this box blank");
        textBox1.Focus();
    }
    else
    {
        textBox1.Text = textBox1.Text.ToUpper();
    }
}
```

5- أحداث ListBox



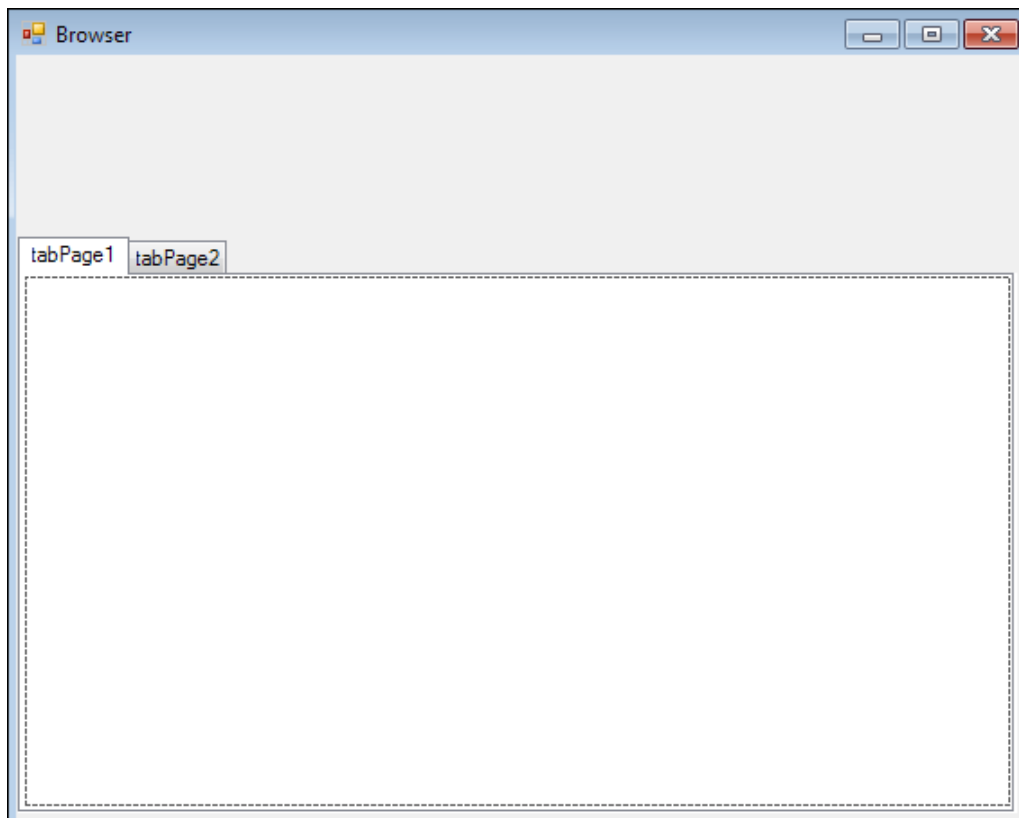
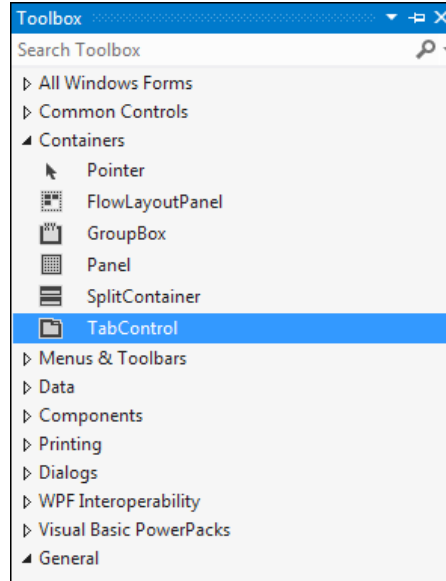
```
private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    if (listBox1.SelectedIndex == 1)
    {
        loadListBox();
    }
}

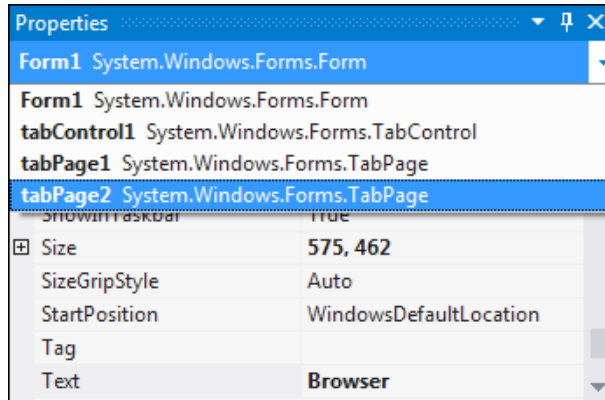
private void loadListBox()
{
    listBox2.Items.Clear();

    listBox2.Items.Add("American Express");
    listBox2.Items.Add("Discover");
    listBox2.Items.Add("Mastercard");
    listBox2.Items.Add("Visa");
}
```

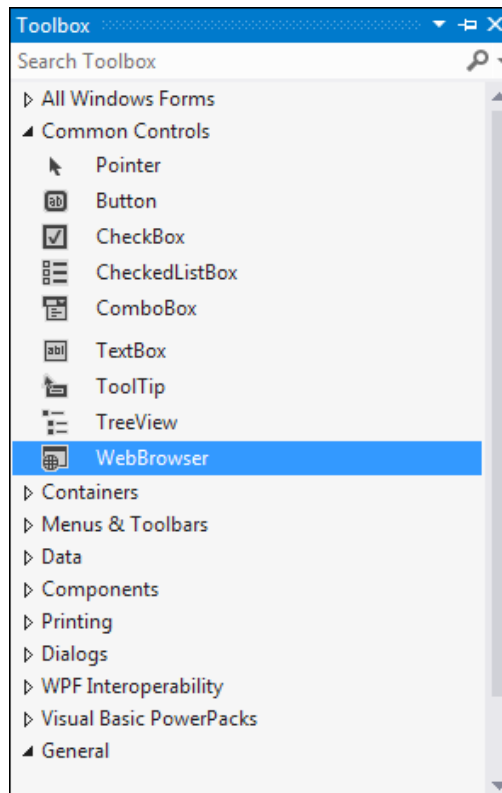
7 - عمل متصفح الخاص بلغة C#

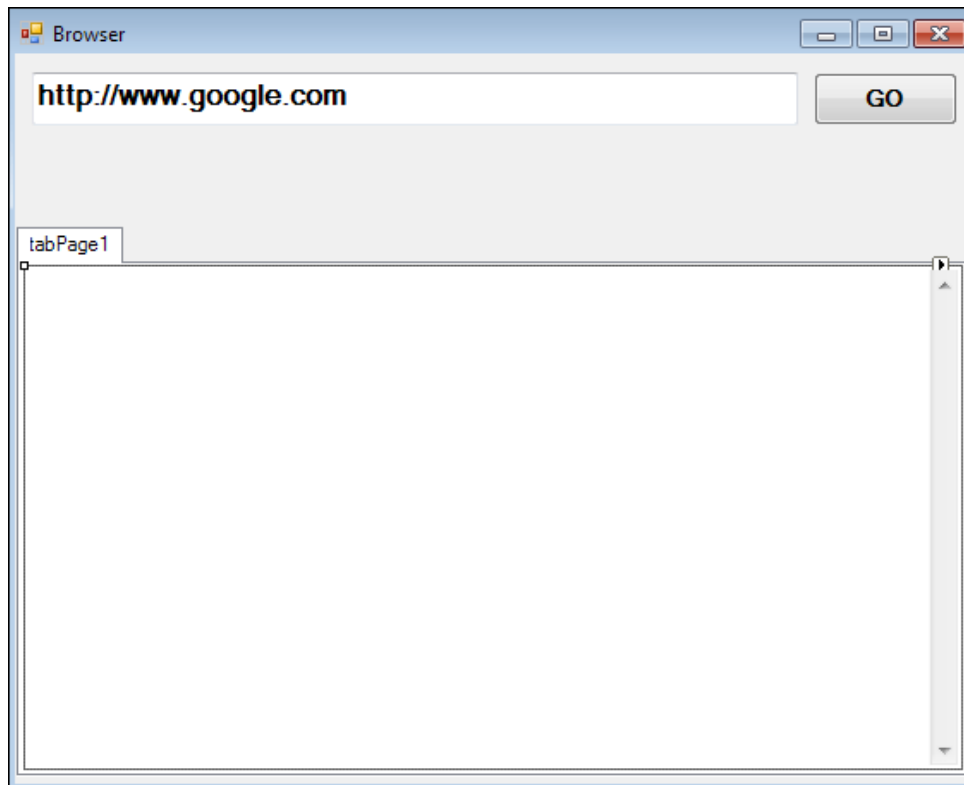
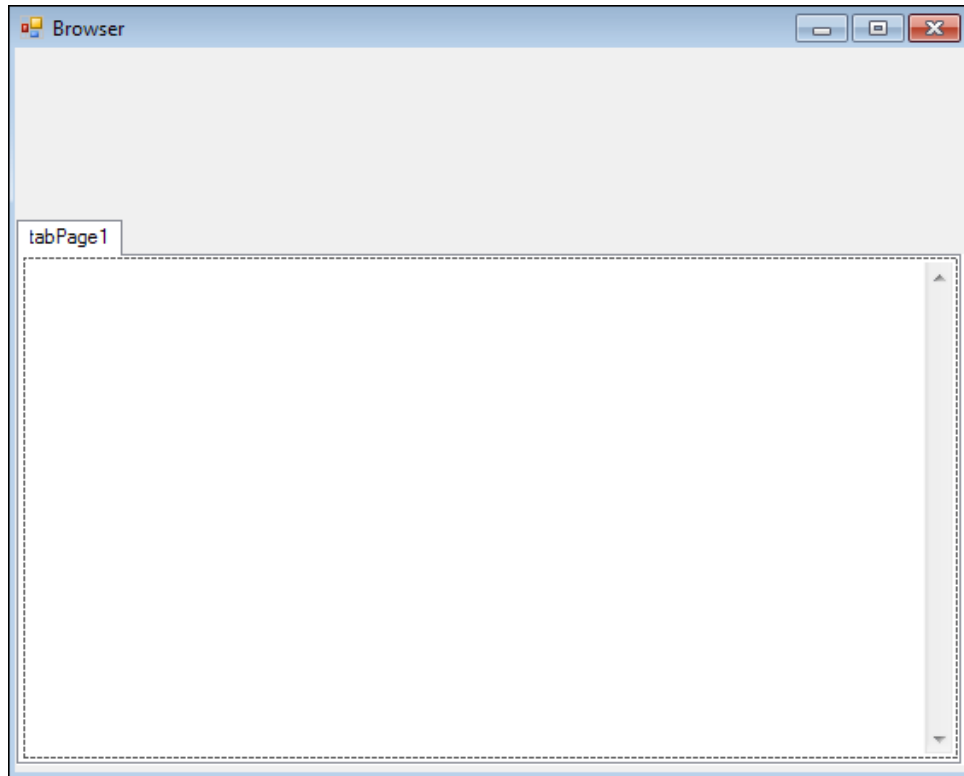
Build your own C# Custom Web Browser





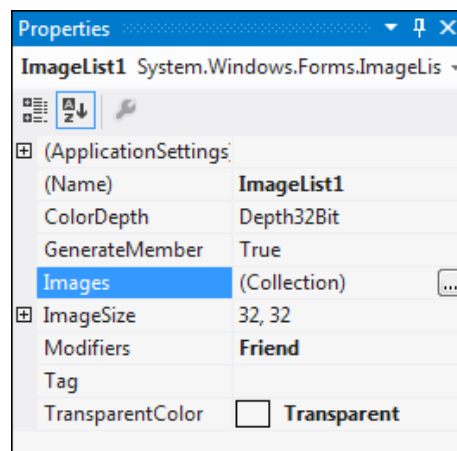
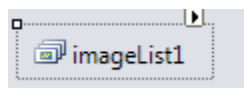
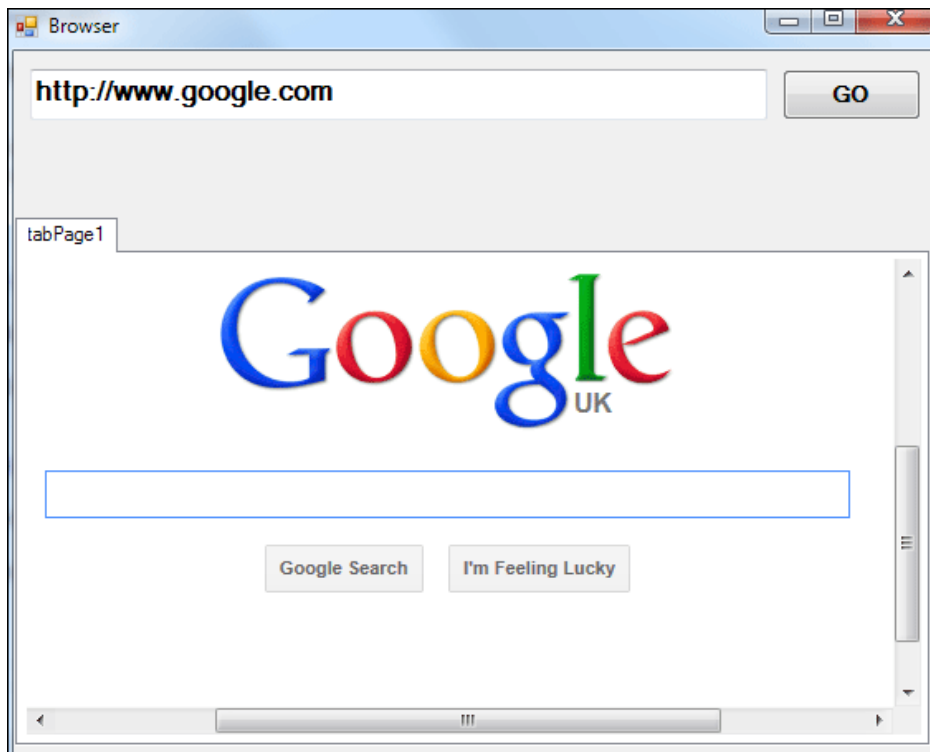
: أداة Web Browser Control

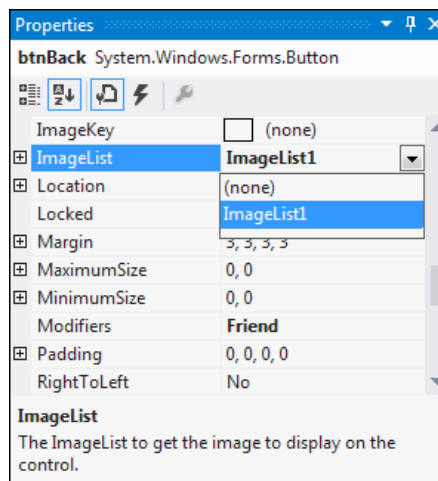
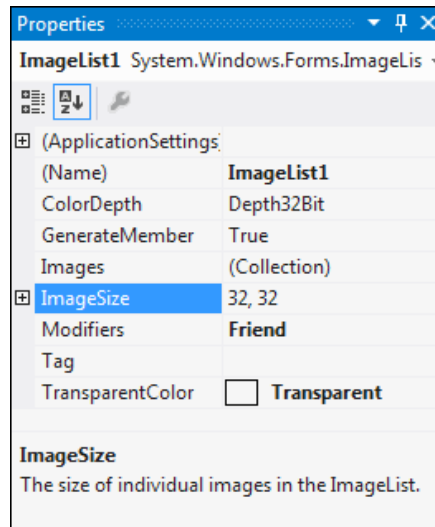
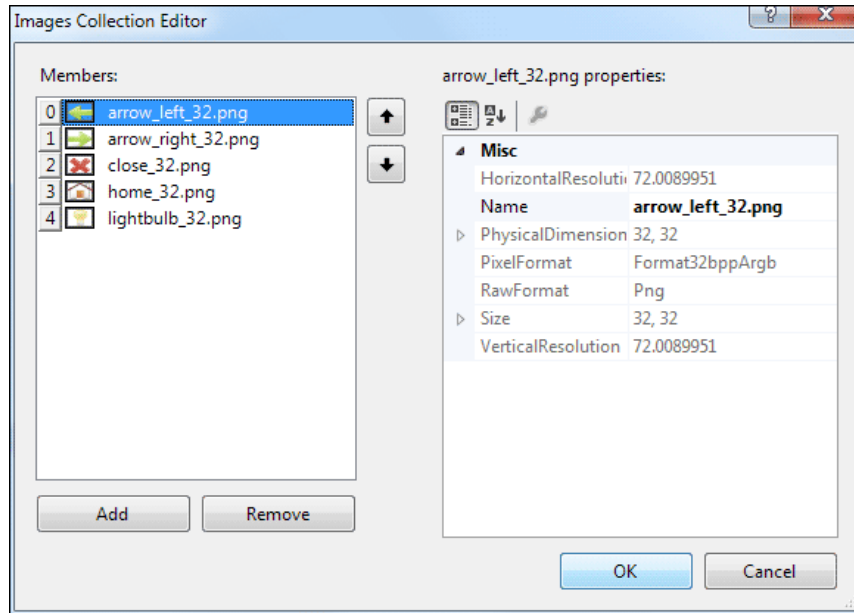


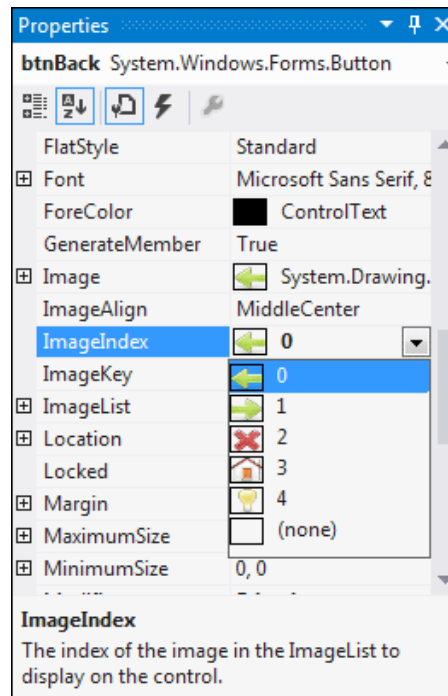


```
string WebPage = txtAddress.Text.Trim();
```

```
webBrowser1.Navigate(WebPage);
```







: Button Back كود

```
if (webBrowser1.CanGoBack)
{
    webBrowser1.GoBack();
}
```

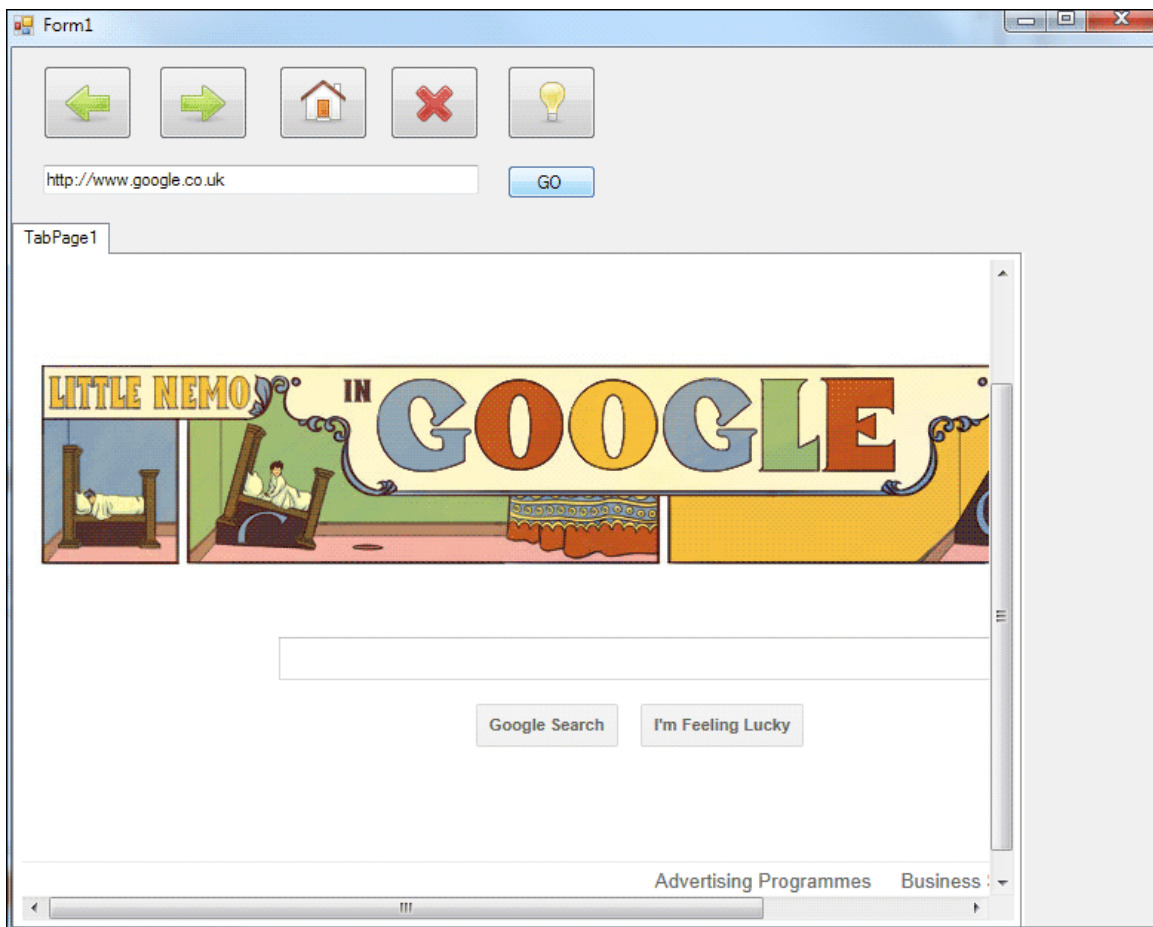
Code for Button Forward:

```
if (webBrowser1.CanGoForward)
{
    webBrowser1.GoForward();
}
```

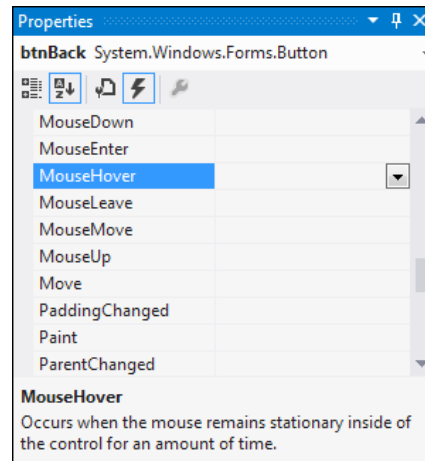
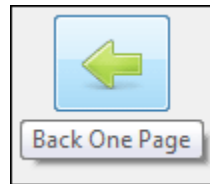

Code for Button Stop:webBrowser1.Stop();

Code for Button Home:webBrowser1.GoHome();

Code for Button Refresh:webBrowser1.Refresh();



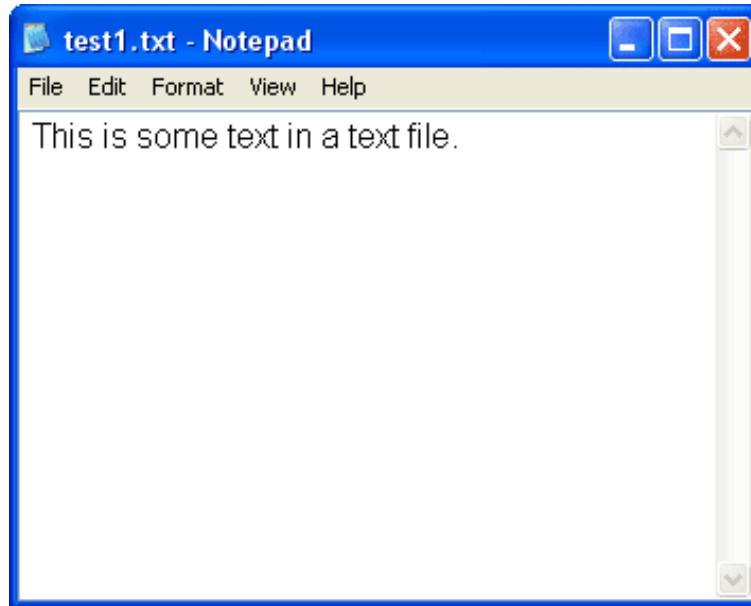
8-إضافة ToolTips in C# .NET



```
toolTip1.SetToolTip( btnBack, "Back One Page" );
```

Manipulating Files التعامل مع الملفات (8)

1- فتح Text File in C# .NET



```
private void btnOpen_Click(object sender, EventArgs e)
{
    string file_name = "C:\\test1111.txt";

    if (System.IO.File.Exists(file_name) == true)
    {
        System.IO.StreamReader objReader;
        objReader = new System.IO.StreamReader(file_name);

        textBox1.Text = objReader.ReadToEnd();

        objReader.Close();
    }
    else
    {
        MessageBox.Show("No such file " + file_name);
    }
}
```

2- قراءة ملف سطر سطر

```
private void btnOpen_Click(object sender, EventArgs e)
{
    string file_name = "C:\\test1.txt";
    string textLine = "";

    System.IO.StreamReader objReader;
    objReader = new System.IO.StreamReader(file_name);

    do
    {
        textLine = textLine + objReader.ReadLine() + "\r\n";
    } while (objReader.Peek() != -1);

    textBox1.Text = textLine;

    objReader.Close();
}
```

الدالة Peek تختبر ما إذا كان هناك حرف معين في الإختبار أم لا إذا لم يكن هناك حرف فإنها تسترجع القيمة -1 .

3 للكتابة لملف Text File in C#

```
private void btnWrite_Click(object sender, EventArgs e)
{
    string file_name = "C:\\test1.txt";

    System.IO.StreamWriter objWriter;
    objWriter = new System.IO.StreamWriter(file_name);

    objWriter.Write(textBox1.Text);
    objWriter.Close();

    MessageBox.Show("Wrote File");
}
```

4 لنسخ ملف باستخدام C#

لنسخ ملف يمكن أن نستخدم الدالة **Copy** التابعة للفصيلة **File Class**

System.IO.File.Copy(fileToCopy, newLocation);

```
private void btnCopy_Click(object sender, EventArgs e)
{
    string fileToCopy = "C:\\test1.txt";
    string newLocation = "C:\\copiedFiles\\test1.txt";
    string folderLocation = "C:\\copiedFiles";

    if (System.IO.Directory.Exists(folderLocation))
    {
        if (System.IO.File.Exists(fileToCopy))
        {
            System.IO.File.Copy(fileToCopy, newLocation, true);
            MessageBox.Show("File Copied");
        }
        else
        {
            MessageBox.Show("No such File");
        }
    }
    else
    {
        MessageBox.Show("No such Directory");
    }
}
```

5 لنقل ملف باستخدام C#

لنقل ملف يمكن أن نستخدم الدالة **Move** التابعة للفصيلة **File Class** لمكان جديد

System.IO.File.Move(fileToMove, fileLocation);

6 لحذف ملف باستخدام C#

لحذف ملف يمكن أن نستخدم الدالة **Delete** التابعة للفصيلة **File Class** لمكان جديد

System.IO.File.Delete(fileToMove, fileLocation);

Debugging (9) تصحيح أخطاء التطبيق

Errors at Design-Time 1

Blue Wavy Lines

```
textBox2.Text = x;
```

The name 'textBox2' does not exist in the current context

Red Wavy Lines

```
textBox1.Text = x
```

; expected

Green Wavy Lines

```
private void button1_Click(object sender, EventArgs e)  
{
```

```
    string CompilerWarning = "";
```

The variable 'CompilerWarning' is assigned but its value is never used

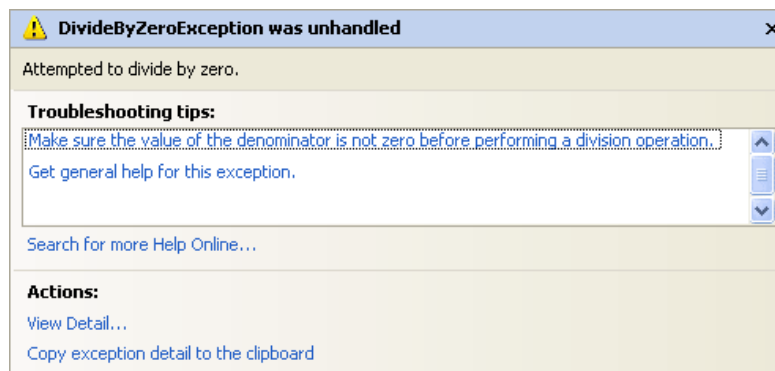
```
    textBox1.Text = "";
```

```
}
```

Run-Time Error -2

```
private void button1_Click(object sender, EventArgs e)
{
    int Num1 = 10;
    int Num2 = 0;
    int answer;

    answer = Num1 / Num2;
}
```



Logic Errors -3

Logic errors are ones where you don't get the result you were expecting.

```
private void button1_Click(object sender, EventArgs e)
{
    int startLoop = 11;
    int endLoop = 1;
    int answer = 0;

    for (int i = startLoop; i < endLoop; i++)
    {
        answer = answer + i;
    }

    MessageBox.Show("answer =" + answer.ToString());
}
```

Breakpoints in c# -4

```
18 private void button1_Click(object sender, EventArgs e)
19 {
20     int LetterCount = 0;
21     string strText = "Debugging";
22     string letter;
23
24     for (int i = 0; i < strText.Length; i++)
25     {
26         letter = strText.Substring(i, i + 1);
27
28         if (letter == "g")
29         {
30             LetterCount++;
31         }
32     }
33
34     textBox1.Text = "g appears " + LetterCount + " times";
35 }
```

```
18 private void button1_Click(object sender, EventArgs e)
19 {
20     int LetterCount = 0;
21     string strText = "Debugging";
22     string letter;
23
24     for (int i = 0; i < strText.Length; i++)
25     {
26         letter = strText.Substring(i, i + 1);
27
28         if (letter == "g")
29         {
30             LetterCount++;
31         }
32     }
33
34     textBox1.Text = "g appears " + LetterCount + " times";
35 }
```



```
18 private void button1_Click(object sender, EventArgs e)
19 {
20     int LetterCount = 0;
21     string strText = "Debugging";
22     string letter;
23
24     for (int i = 0; i < strText.Length; i++)
25     {
26         letter = strText.Substring(i, i + 1);
27
28         if (letter == "g")
29         {
30             LetterCount++;
31         }
32     }
33
34     textBox1.Text = "X appears " + LetterCount + " times";
```

```
for (int i = 0; i < strText.Length; i++)
{
    letter = strText.Substring(i, i + 1);
    if (letter == "g")
    {
        LetterCount++;
    }
}
```

```
for (int i = 0; i < strText.Length; i++)
{
    letter = strText.Substring(i, i + 1);
    if (letter == "g")
    {
        LetterCount++;
    }
}
```

Local window in c# -5

اختار من القائمة Windows > Locals

```
for (int i = 0; i < strText.Length; i++)
{
    letter = strText.Substring(0, 1);

    if (letter == "g")
    {
        LetterCount++;
    }
}
```

Locals	
Name	Value
this	{debugging.Form1, Text: Form1}
sender	{Text = "button1"}
e	{X = 33 Y = 13 Button = Left}
i	0
LetterCount	0
strText	"Debugging"
letter	"D"



تلاحظ أن قيمة المتغير تتغير F10 اضغط على مفتاح

```
for (int i = 0; i < strText.Length; i++)
{
    letter = strText.Substring(0, 1);

    if (letter == "g")
    {
        LetterCount++;
    }
}
```

Locals	
Name	Value
this	{debugging.Form1, Text: Form1}
sender	{Text = "button1"}
e	{X = 33 Y = 13 Button = Left}
i	3
LetterCount	0
strText	"Debugging"
letter	"D"

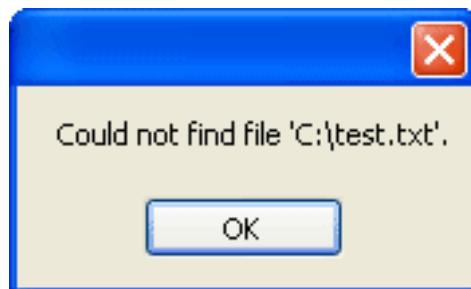
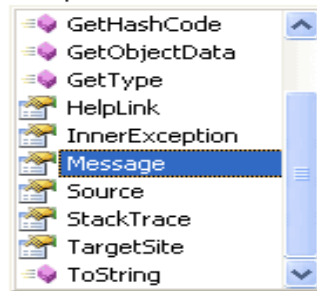


6 – استخدام try Catch in C#

يتم استخدامها للتعامل مع الأخطاء المحتملة بوضع الكود في try ويمكن تصيد الأخطاء باستخدام Catch بالكود التالي :

```
try  
{  
  
}  
catch  
{  
  
}
```

```
try  
{  
    rtb.LoadFile("C:/test.txt");  
}  
catch (System.Exception excep)  
{  
    MessageBox.Show(excep.);  
}
```



إذا كنت تعرف الخطأ الناتج يمكن استخدام الكود التالي :

```
catch (System.IO.FileNotFoundException)
{
    MessageBox.Show("File not found");
}
```

لمعرفة نوع الخطأ الناتج يمكن استخدام الكود التالي :

```
catch (System.Exception excep)
{
    MessageBox.Show( excep.GetType().ToString() );
}
```

C# التاريخ والوقت فى لغة C# (10)

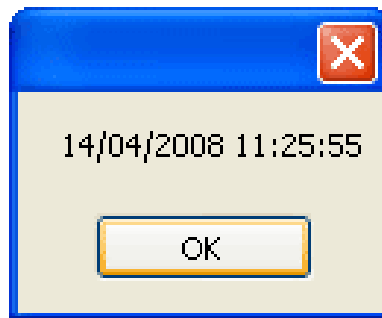
Dates and Times in C#

Add this to you button code:

```
DateTime theDate;
```

```
theDate = DateTime.Now;
```

```
MessageBox.Show( theDate.ToString() );
```

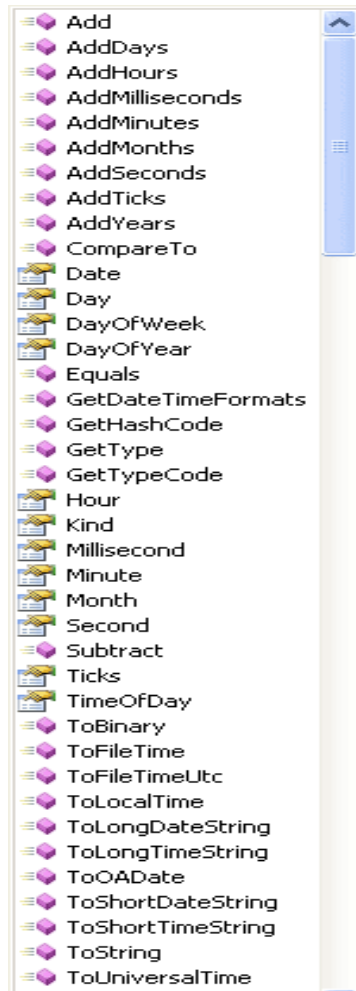


You can also have this, instead of Now:

```
theDate = DateTime.Today;
```

And even this

```
theDate = DateTime.UtcNow;
```



To use this, then, you can set up a new integer variable and hand it the day:

```
theDate = DateTime.UtcNow;  
int theDay = theDate.Day;
```

The Month and Year are also Integers, so the code is similar:

```
int theMonth = theDate.Month;
```

Or

```
int theYear = theDate.Year;  
DateTime firstDate = new DateTime(2008, 01, 14);
```



// Convert To Hijri Date Method

```
public string HijriDate(int Year, int Month, int Day)  
  
{  
  
    CultureInfo arSA = CultureInfo.CreateSpecificCulture("ar-SA");  
  
    DateTime dt = new DateTime(Year, Month, Day);  
  
    //String stringDate = "yyyy-MM-dd";  
  
    String stringDate = "dd-MM-yyyy";  
  
    string s = dt.ToString(stringDate, arSA);  
  
    return s.ToString();  
  
}
```

برمجة الكائنات الموجهة (11)

Object-Oriented Programming (OOP)

برمجة الكائنات الموجهة Object – Oriented Programming

لغة البرمجة تكون Full OOP

لابد وأن تتوافر بها الأربعة شروط التالية :

Encapsulation -1

Abstraction -2

Inheritance -3

Polymorphism -4

الشرط الأول: Encapsulation

ومعناها هو تجميع الأشياء ووضعها آمنة بمكانها .

ومناسمها تستطيع تمييز ذلك معناها الكبسولة

ولتعريفها جزأين :

الجزء الأول : (Class) ونضع فيها المتغيرات والدوال التي لها علاقة ببعضها

هيتعريف مكان من ثم نسند إليها اسماً

الجزء الثاني : هذا المكان لا يستطيع أحد الدخول إليها إلا بصلاحيات محددة من قبل المبرمج فيوجد

وهو القسم العام قسمان رئيسيان لها Public والPrivate وهو القسم الخاص

أما Protected يتم استخدامه إذا تم التعامل مع فئات مورثة

Inherited Classes

الشرط الثاني: Abstraction:

هي إضافة دالة عامة Method داخل Class لأقوم بتحديد التفاصيل لها
إلا في قسم البرنامج الرئيسى .

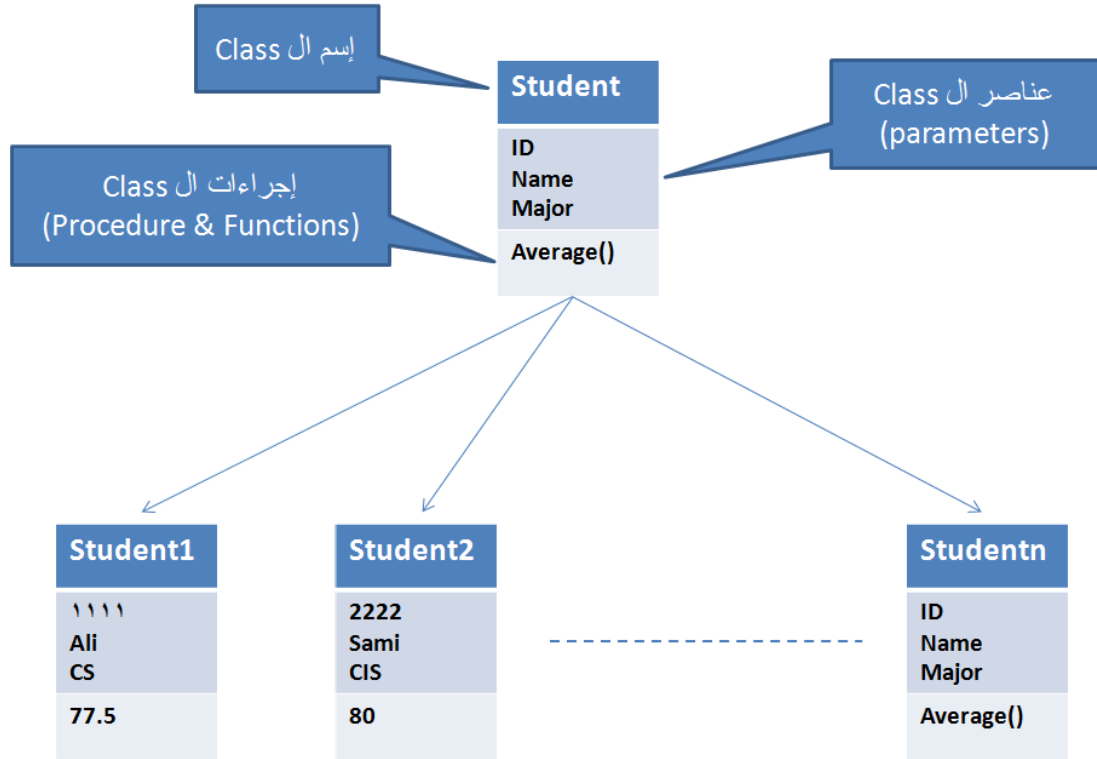
فمثلاً

(Class) أو Modules في جامعة معينة فينتج من ذلك كائن للدكتور وكائن
نريد صنع عدة كائنات للطالب وآخر للمدير وآخر للعاملو ... إلخ وذلك لتوفير الوقت والجهد
عند كتابتك للكود .

الفصيلة (Class)

هي مجموعة من النصوص البرمجية تقوم بوظيفة معينة وتقوم فكرتها على أنه
يمكن إستخدام الكود أكثر من مرة بنفس صيغة الكود داخل مشروعك أو مشاريع
أخرى لتحقيق مبدأ reusability أى أن نفس الكود يمكن إستخدامه أكثر من مرة
ولا حاجة لتكرار كتابته مرة أخرى وتحتوى على خصائص ودوال ويمكن إعتبارها
كقالب Template أو مصنع Factory يتم من خلاله إنتاج ما يسمى بالهدف
Object والذي نستخدمه للوصول والتعامل مع خصائص ودوال الفصيلة .

مثال لذلك :



نطاق أو مجال الكود Namespace :

يعرف ببداية أى برنامج بلغة C#Namespaces المتطلبه بمجرد إنشاء البرنامج عن طريق كلمة using ويستخدم لتحديد أسماء الفصائل التي يتم تعريفها بداخله حتى لا يحدث خطأ أثناء إستدعاء الفصيلاء كالتالى :

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace SunSchoolNameSpace
{
    class Student
    {

    }
  
```

```
}  
  
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
namespace MicrosoftSchoolNameSpace  
{  
    class Student  
    {  
  
    }  
}
```

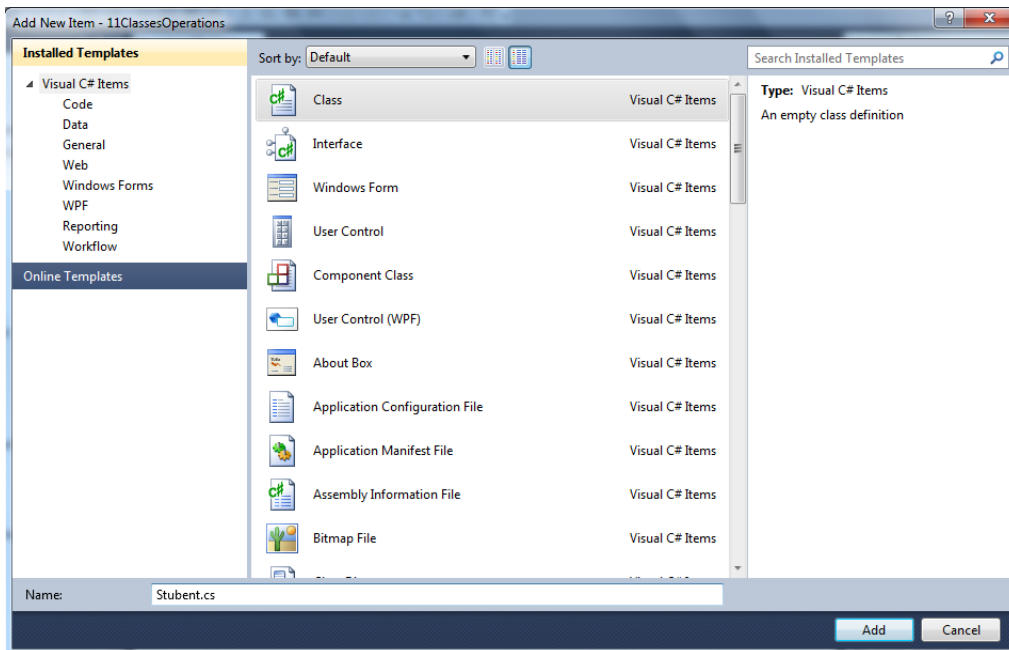
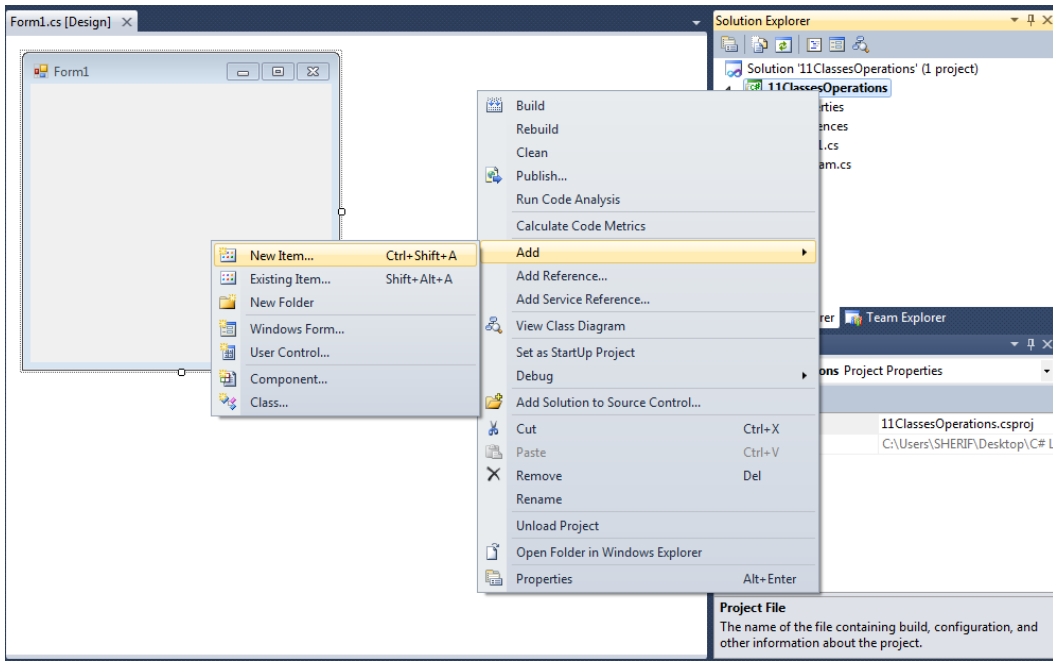
لإستدعاء الفصيلة الخاصة بكل Namespace كالتالى :

- 1- SunSchoolNameSpace. Student
- 2- MicrosoftSchoolNameSpace. Student

: Class

لتعريف الفصيلة أضف ملف جديد لمشروعك عن طريق الضغط بالماوس

Right-Click على اسم المشروع الخاص بك وإضافة **New Item** وحدد اسم الفصيلة وليكن **Student** كما بالشكل التالى :



```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace _11ClassesOperations
{
    class Stubent
    {

```

```
}  
}
```

: Object

يتم تعريفه بالصيغة التالية :

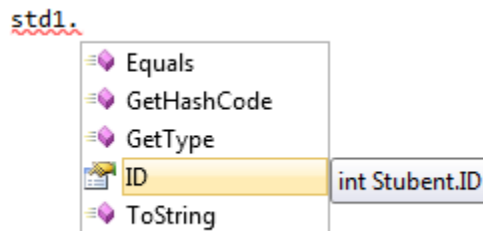
```
Student stu1 = new Student();
```

ويمكنه الوصول والتعامل مع خصائص الفصيله بعد أن قمنا بتعريفه .

:Properties

تتم تعريفها بالصيغة التالية :

```
private int sid;  
  
public int ID  
{  
    set  
    {  
        sid = value;  
    }  
    get  
    {  
        return sid;  
    }  
}
```



وذلك للوصول للخصائص عن طريق اسم الهدف .

:Constructor

يتم من خلاله إعطاء قيم ابتدائية للمتغيرات المستخدمة في الفصيلة ويأخذ نفس إسم الفصيلة ويتم تعريفه كالتالى :

```
public Student()  
{  
    sid = 0;  
    sname = "Name";  
    smajor = "Major";  
}
```

كما يمكن إعطاء قيم للمتغيرات أثناء التعريف كالتالى :

```
public Student(int id, string name, string  
major)  
{  
    sid = id;  
    sname = name;  
    smajor = major;  
}
```

الشرط الثالث : Inheritance

هو أن ترث الفصيلة الابن Child Class خصائص ودوال الفصيلة الأب

Parent Class دون أن ترث الفصيلة الأب شىء من الفصيلة الابن أى أنها علاقة بين فصيليتين أحدهما يرث خصائص وصفات الفصيلة الأخرى دون أن يرث الأول منها شىء وصيغتها كالتالى :

```
public class ChildClass : ParentClass  
{  
    // محتوى class  
}
```

الشرط الرابع : Polymorphism(Overloading - Overriding) :

: Overloading

هو تعريف الدوال بنفس الإسم والنوع ولكن مع إختلاف عدد المتغيرات التي تأخذها الدالة .

مثال :

```
public double Average(int mark1, int mark2)
{
    return (mark1 + mark2) / 2;
}
```

```
public double Average(int mark1, int mark2, int mark3)
{
    return (mark1 + mark2 + mark3) / 3;
}
```

: Overriding

وهو عبارة عن اعادة تعريف Method موجود داخل ال parent class بال child class بحيث يحمل نفس الإسم والنوع وعدد ونوع المتغيرات لكن بكود داخلي خاص بال child class. وهذا يعني أنه يتم استدعاء ال Method المعروف داخل ال child class فقط عند تعريف object من نوعه، وأيضا يتم استدعاء ال Method الذي يحمل نفس الإسم في ال parent class فقط عند تعريف object من نوعه، وكل منهم ينفذ الكود الخاص به. بالإضافة إلى أنه عند استدعاء ال Method سواء من student object أو parson object لن يظهر لنا إلا خيار واحد فقط ، وهو ال Method الموجود داخل هذه ال class .

مثال :

يتم تعريف الدالة في Parent Class مسبوقة بكلمة Virtual أى أن هذه الدالة حدث لها توارث داخل Child Class ولكن بكلمة override وهى نفس الدالة مع اختلاف محتوى الدالة داخل Child Class .

وصيغتها في Parent Class كالتالى :

```
public virtual string search(int id)
{
    return "person";
}
```

وصيغتها في Child Class كالتالى :

```
public override string search(int id)
{
    return "student";
}
```


برنامج (12)

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختار نوع المشروع Windows Forms Application باسم

ClassesOperations العمليات على الفصائل

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: btnProperties

Text:Properties

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Button:

Name: btnConstructor

Text:Constructor

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- Button:

Name: btnInheritance

Text:Inheritance

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Button:

Name: btnOverloading

Text:Overloading

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

5- Button:

Name: btnOverride

Text:Override

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

6- Label:

Name: label1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

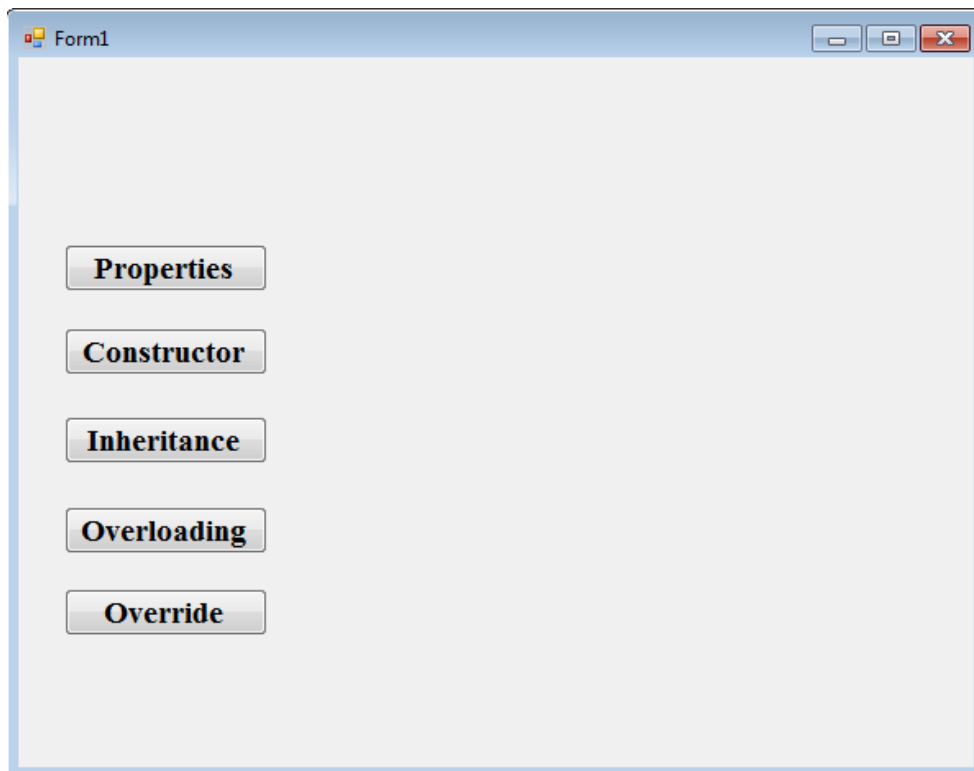
7- Label:

Name: label2

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



أولا أضف إلى مشروعك الفصيلتين التاليتين :

الفصيلة الأب Parent Class

```
public class Person
{
    // Properties For Inheritance
    int pssn;
    string address;

    public int SSN
    {
        get
        {
            return pssn;
        }
        set
        {
            pssn = value;
        }
    }
}
```

```
publicstring Address
    {
    get
        {
        return paddress;
        }
    set
        {
            paddress = value;
        }
    }
}
```

// Overriding Method

```
publicvirtualstring search(int id)
    {
    return"person";
    }
}
```

Child Class الفصيلة الابن

```
publicclassStudent:Person
    {

    privateint sid;
    privatestring sname;
    privatestring smajor;

    // Properties

    publicint ID
        {
        set
            {
                sid = value;
            }
        get
            {
                return sid;
            }
        }

    publicstring Name
        {
        set
            {
                sname = value;
            }
        get
            {
            }
        }
    }
}
```

```

        {
return sname;
        }
    }

public string Major
    {
set
        {
            smajor = value;
        }
get
        {
return smajor;
        }
    }

// Constructors

public Student()
    {
        sid = 20;
        sname = "Ali";
        smajor = "Pharmacy";
    }

public Student(int id, string name, string major)
    {
        sid = id;
        sname = name;
        smajor = major;
    }

// Overloading

public double Average(int mark1, int mark2)
    {
return (mark1 + mark2) / 2;
    }

public double Average(int mark1, int mark2, int mark3)
    {
return (mark1 + mark2 + mark3) / 3;
    }

```

```
    }

    // Overriding Method
    public override string search(int id)
    {
        return "Student";
    }
}
```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث btnProperties_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void btnProperties_Click(object sender, EventArgs e)
{
    label2.Text = String.Empty;

    Student std1 = new Student();

    // للتعامل مع الخصائص Properties
    std1.ID = 111;
    std1.Name = "Mohamed";
    std1.Major = "Computer Science";

    label1.Text = "ID=" + std1.ID.ToString() + "\n" + "Name=" +
    std1.Name.ToString() + "\n" + "Major=" + std1.Major.ToString();
}
```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث btnConstructor_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void btnConstructor_Click(object sender, EventArgs e)
{
    // Constructor للتعامل مع
    Student std2 = new Student();
    Student std3 = new Student(10, "Ahmed", "Engineering");

    label1.Text = "ID=" + std2.ID.ToString() + "\n" + "Name=" +
std2.Name.ToString() + "\n" + "Major=" + std2.Major.ToString();

    label2.Text = "ID=" + std3.ID.ToString() + "\n" + "Name=" +
std3.Name.ToString() + "\n" + "Major=" + std3.Major.ToString();
}
```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث btnInheritance_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void btnInheritance_Click(object sender, EventArgs e)
{
    Person per1 = new Person();
    per1.SSN = 100;
    per1.Address = "Cairo";

    label1.Text = "SSN=" + per1.SSN.ToString() + "\n" + "Address=" +
per1.Address.ToString();

    Student std1 = new Student();

    std1.SSN = per1.SSN;
    std1.Address = per1.Address;

    label2.Text = "ID=" + std1.ID.ToString() + "\n" + "Name=" +
std1.Name.ToString() + "\n" + "Major=" + std1.Major.ToString() + "\n" + "SSN=" +
std1.SSN.ToString() + "\n" + "Address=" + std1.Address.ToString() + "\n";
}
```

```
}
```

بالضغط مرتين بالماوس على **Button Double** ينشئ لنا

الحدث **btnOverloading_Click ()**

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid btnOverloading_Click(object sender, EventArgs e)
{
    // Method Overloading للتعامل مع

    Student std4 = newStudent();
    double Avg1 = std4.Average(20, 20);
    double Avg2 = std4.Average(30, 30 ,30);

    label1.Text = "Average1=" + Avg1.ToString();
    label2.Text = "Average2=" + Avg2.ToString();
}
```

بالضغط مرتين بالماوس على **Button Double** ينشئ لنا

الحدث **btnOverride_Click ()**

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid btnOverride_Click(object sender, EventArgs e)
{
    Student std1 = newStudent();
    string std1String = std1.search(10);

    label1.Text = "Child Search =" + std1String;

    Person per1 = newPerson();
    string per1String = per1.search(10);

    label2.Text = "Parent Search =" + per1String;
}
```


برمجة تطبيقات الويندوز (12)

لبرمجة تطبيقات الويندوز بلغة C# لابد من التعامل مع بيئة التطوير Visual Studio باستخدام الأدوات Controls اللازمة لصنع برنامجك أو مشروعك الذى تقوم ببرمجته لذا فيتم هنا استخدام الأدوات الهامة لكل مبرمج يتطلع إلى الاحترافية فى برمجة الواجهات من خلال برامج تفصيلية لكل أداة أو عدة أدوات كما تتطلبه البرامج .

الأدوات Controls :

1 أداة الزر Button

وهى أداة يقوم بتنفيذ حدث معين أثناء الضغط عليها بالماوس أو تمرير الماوس عليها وهى ترتبط بأحداث الماوس مثل Click – DoubleClick وغيرها من الأحداث التى يتم تنشيطها عند الضغط على الزر أو استخدام أى حدث له ولها خواص مثل خاصية Text و Visible وغيرها من الخواص والتى يمكنك التعرف عليها من قائمة Properties فى Visual Studio .

برنامج (13) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

PrintMessage

ضع على الفورم التى تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

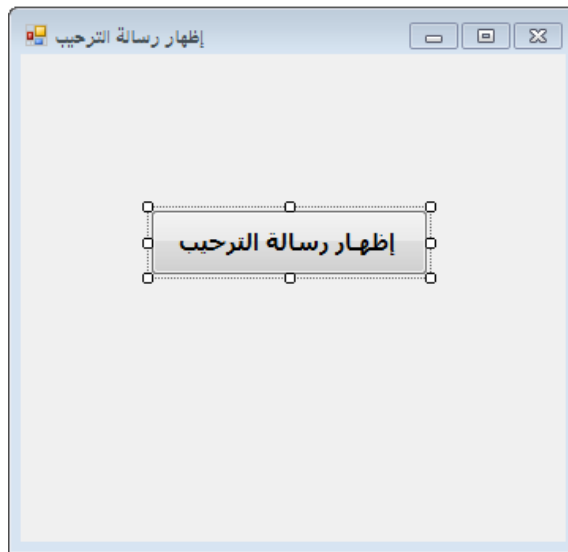
Name: btnMessage

Text:Properties

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث btnMessage_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void btnMessage_Click(object sender, EventArgs e)
{
    lblMessage.Text = " مرحباكمفتعلمتطبيقاتالويندوزبلغة C# ";
}
```

برنامج (14) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

ButtonProperties

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: btnChangeColor

Text: تغيير لون خلفية الشاشة

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Button:

Name: btnFont

Text: Change Form Font

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- Button:

Name: btnMessage

Text: إظهار رسالة

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Button:

Name: btnExit

Text:خروج

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث (`btnChangeColor_Click ()`)

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid btnChangeColor_Click(object sender, EventArgs e)
{
    BackColor = Color.Green;
}
```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث btnFont_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid btnFont_Click(object sender, EventArgs e)
{
    this.Font = newFont("CASTELLAR", 16);
}
```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث btnMessage_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid btnMessage_Click(object sender, EventArgs e)
{
    MessageBox.Show("مرحبابكم");
}
```

بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث btnExit_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid btnExit_Click(object sender, EventArgs e)
{
    Close();
}
```

2 صندوق النص TextBox

وهي عبارة عن أداة لإدخال النصوص والتي يتم من خلالها قراءة النص الذي يدخله المستخدم من الخاصية Text و TextAlign وغيرها من الخواص والتي يمكنك التعرف عليها من قائمة Properties في Visual Studio .

برنامج (15) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم TextBox

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: btnChangeColor

Text: طباعة محتويات صندوق النص

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- TextBox:

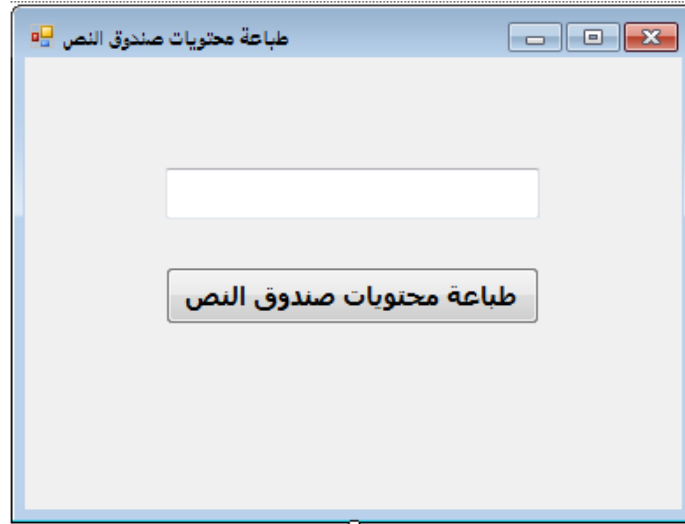
Name: txtMessage

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

RightToLeft: Yes



بالضغط مرتين بالماوس على Button Double ينشئ لنا

الحدث `btnMessage_Click ()`

قم بكتابة الكود التالي بداخله كالتالي :

```
private void btnPrintMessage_Click(object sender, EventArgs e)
{
    lblMessage.Text = txtMessage.Text;
}
```

3 صندوق النص المتعدد RichTextBox

وهي عبارة عن أداة لإدخال النصوص والتي يتم من خلالها قراءة النص الذي يدخله المستخدم وهي مثل أداة TextBox إلا أنها تحتوى على خصائص غير موجودة في أداة TextBox كونها تحتوى على نص متعدد الألوان والخطوط والأحجام ولها بعض الخواص مثل Text و TextAlign غيرها من الخواص والتي يمكنك التعرف عليها من قائمة Properties في Visual Studio .

برنامج (16) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو واختر نوع المشروع Windows Forms Application باسم RichTextBox ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: btnRed

Text: لون الخط أحمر

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Button:

Name: btnGreen

Text: لون الخط أخضر

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- Button:

Name: btnBlue

Text: لونا الخط أزرق

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Label:

Name: txtMessage

Text : أكتب النص الذي تريده هنا

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

5- TextBox:

Name: textbox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

RightToLeft: Yes

6- RichTextBox:

Name: richTextBox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

RightToLeft: Yes



بالضغط مرتين بالماوس على Button btnRed ينشئ لنا

الحدث **btnRed_Click ()**

قم بكتابة الكود التالي بداخله كالتالي :

```
private void btnRed_Click(object sender, EventArgs e)
{
    richTextBox1.SelectionFont = new Font("Timew New Roman", 40, FontStyle.Bold);
    richTextBox1.SelectionColor = Color.Red;
}
```

بالضغط مرتين بالماوس على Button btnGreen ينشئ لنا

الحدث **btnGreen_Click ()**

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid btnGreen_Click(object sender, EventArgs e)
{
    richTextBox1.SelectionFont = newFont("Timew New Roman", 40, FontStyle.Bold);
    richTextBox1.SelectionColor = Color.Green;
}
```

بالضغط مرتين بالماوس على Button btnBlue ينشئ لنا

الحدث btnBlue_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid btnBlue_Click(object sender, EventArgs e)
{
    richTextBox1.SelectionFont = newFont("Timew New Roman", 40, FontStyle.Bold);
    richTextBox1.SelectionColor = Color.Blue;
}
```

بالضغط مرتين بالماوس على TextBox textbox1 ينشئ لنا

الحدث txtMessage_TextChanged()

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid txtMessage_TextChanged(object sender, EventArgs e)
{
    this.richTextBox1.Text = this.textbox1.Text;
}
```

4- أدوات إظهار النص Label , LinkLabel

وهي عبارة عن أداة تلعب دور مثل الملصق على الأجهزة الكهربائية أو الأدوات المنزلية في إظهار النص وهي أداة Label أما الأداة الأخرى LinkLabel فهي نفس الأداة الأولى ولكن تزيد عليها كونها عبارة عن رابط انترنت ولهما بعض الخواص مثل Text و TextAlign غيرها من الخواص والتي يمكنك التعرف عليها من قائمة Properties في Visual Studio .

برنامج (17) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو واختر نوع المشروع Windows Forms Application باسم

LabelAndLinkLabel

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Label:

Name: txtMessage

Text : This is a Label

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- LinkLabel:

Name: txtMessage

Text : This is a Link Label

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بالضغط مرتين بالماوس على `LinkLabel linkLabel1` ينشئ لنا

الحدث `linkLabel1_LinkClicked ()`

قم بكتابة الكود التالي بداخله كالتالي :

```
private void linkLabel1_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    System.Diagnostics.Process.Start("www.google.com");
}
```

5 صندوق الإختيار CheckBox

وهى عبارة عن أداة تسمح للمستخدم بالإختيار من متعدد وتتعامل مع الخصائص التالية : Checked – Enabled و غيرها من الخواص والتي يمكنك التعرف عليها من قائمة Properties فى Visual Studio .

برنامج (18) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

CheckBox

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: btnBlue

Text: لقد قمت باختيار

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Label:

Name: txtMessage

Text : ما هيلغة البرمجة المفضلة لديك ؟

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- CheckBox:

Name: cb_cs

Text:CSharp.Net

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- CheckBox:

Name: cb_vb

Text:Visual Basic.Net

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



The screenshot shows a Windows application window titled "Form1". The window contains the following elements:

- A question in Arabic: "ما هي لغة البرمجة المفضلة لديك ؟" (Which programming language do you prefer?)
- Two checkboxes with labels: Visual Basic.Net and CSharp.Net.
- A button at the bottom with the Arabic text "لقد قمت باختيار" (I have chosen).

بالضغط مرتين بالماوس على Button button1 ينشئ لنا

الحدث button1_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    if (cb_vb.Checked == true && cb_cs.Checked == false)
    {
        MessageBox.Show("Your Favourite language is : Visual Basic.net");
    }
    elseif (cb_vb.Checked == false && cb_cs.Checked == true)
    {
        MessageBox.Show("Your Favourite language is : CSharp.net");
    }
    elseif (cb_vb.Checked == false && cb_cs.Checked == false)
    {
        MessageBox.Show("Please Choose Your language !");
    }
    else
    {
        MessageBox.Show("Your Favourite language is : Visual Basic.net and CSharp.Net");
    }
}
```

6 زر الإختيار RadioButton

وهي عبارة عن أداة تشترك مع أداة CheckBox تقريبا في الخواص إلا أنها لا تتيح للمستخدم إمكانية تعدد الإختيارات وتتعامل مع الخصائص التالية :
Checked – Enabled وغيرها من الخواص والتي يمكنك التعرف عليها من قائمة Properties في Visual Studio .

برنامج (19) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

RadioButton

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Label:

Name: label1

Text:اختيار النوع:

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Button:

Name: button1

Text:اختيار كهو:

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- RadioButton:

Name: radioButton1

Text:ذكر:

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- RadioButton:

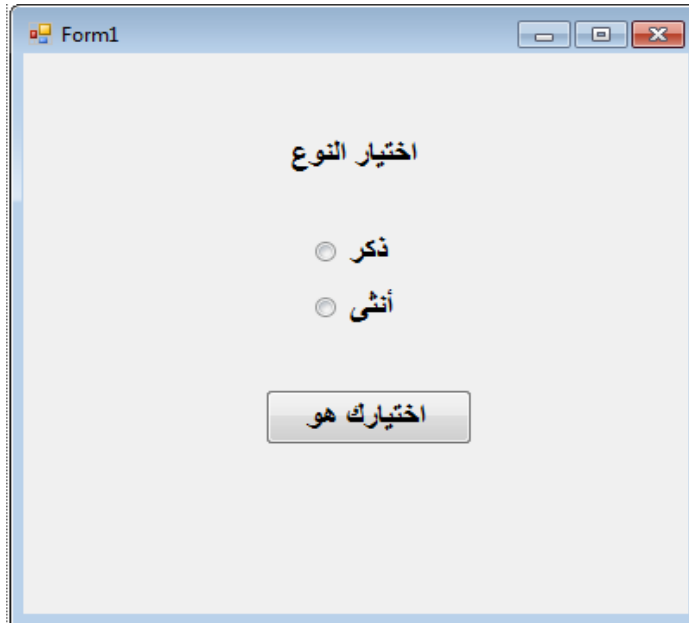
Name: radioButton2

Text: أنثى

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بالضغط مرتين بالماوس على Button button1 ينشئ لنا

الحدث **button1_Click ()**

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    if (radioButton1.Checked == true)
    {
        MessageBox.Show(" ذكر ");
    }
    else
    {
        MessageBox.Show(" أنثى ");
    }
}
```

7- علبه القائمه ListBox

تقوم هذه الأداة بعرض البيانات على شكل قائمة يمكن للمستخدم من خلالها أن يختار مايشاء ويمكن تعبئتها يدويا بأى عناصر كما تشاء من الخاصة Items وغيرها من الخواص والتي يمكنك التعرف عليها من قائمة Properties فى Visual Studio .

برنامج (20) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

ListBox1

ضع على الفورم التى تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Label:

Name: label1

Text: قائمة هواياتك المفضلة:

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- ListBox:

Name: listBox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-To-Left: Yes



بالضغط مرتين بالماوس على الفورم Form1 ينشئ لنا

الحدث Form1_Load ()

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid Form1_Load(object sender, EventArgs e)
{
    this.listBox1.Items.Add("قراءةالقصصالدينية");
    this.listBox1.Items.Add("السباحة");
    this.listBox1.Items.Add("ركوبالخيول");
}
```

برنامج (21) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

ListBox2

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Button:

Name: button1

Text: أضف إلى القائمة

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- Button:

Name: button2

Text: احذف العنصر الذي تم اختياره

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- Button:

Name: button3

Text: حذف جميع العناصر

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- Label:

Name: label1

Text: أكتباسمكهنأ :

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-To-Left: Yes

5- TextBox:

Name: textBox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-To-Left: Yes

6- ListBox:

Name: listBox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-To-Left: Yes



بالضغط مرتين بالماوس على الفورم Button button1 ينشئ لنا

الحدث (`button1_Click ()`)

قم بكتابة الكود التالي بداخله كالتالي :

```
private void button1_Click(object sender, EventArgs e)
{
    this.listBox1.Items.Add(textBox1.Text);
    this.textBox1.Text = String.Empty;
}
```

بالضغط مرتين بالماوس على الفورم Button button2 ينشئ لنا

الحدث (`Button2_Click ()`)

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid button2_Click(object sender, EventArgs e)
{
this.listBox1.Items.Remove(listBox1.SelectedItem);
//this.listBox1.Items.RemoveAt(listBox1.SelectedIndex);
}
```

بالضغط مرتين بالماوس على الفورم Button button3 ينشئ لنا

الحدث Button3_Click ()

قم بكتابة الكود التالي بداخله كالتالي :

```
privatevoid button3_Click(object sender, EventArgs e)
{
this.listBox1.Items.Clear();
}
```

8 صندوق الكومبوجو ComboBox

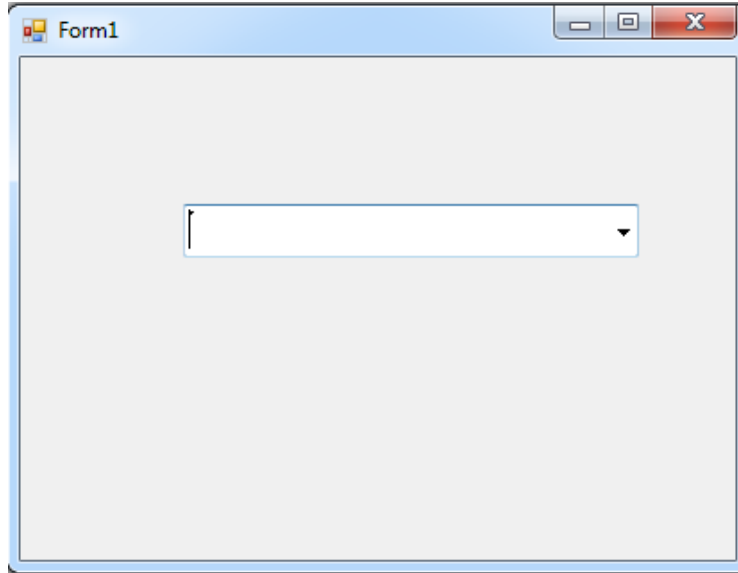
هذه الأداة تستعمل لتمكين المستخدم من اختيار بعض البيانات مثال عند التسجيل بموقع مث الياهو يطلب منك اختيار الدولة التي تنتمي إليها وخصائصها وأحداثها شبيهة بصندوق القوائم ListBox ويمكن تعبئتها يدويا بأى عناصر كما تشاء من الخاصة Items وغيرها من الخواص والتي يمكنك التعرف عليها من قائمة Properties فى Visual Studio .

برنامج (22) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

ComboBox



ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- ComboBox:

Name: comboBox1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

بالضغط مرتين بالماوس على الفورم Form1 ينشئ لنا

الحدث Form1_Load ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void Form1_Load(object sender, EventArgs e)
{
    string[] Countries = { "مصر", "السعودية", "الجزائر", "العراق", "تونس", "ليبيا" };
    this.comboBox1.Items.AddRange(Countries);
}
```

بالضغط مرتين بالماوس على ComboBox comboBox1 ينشئ لنا

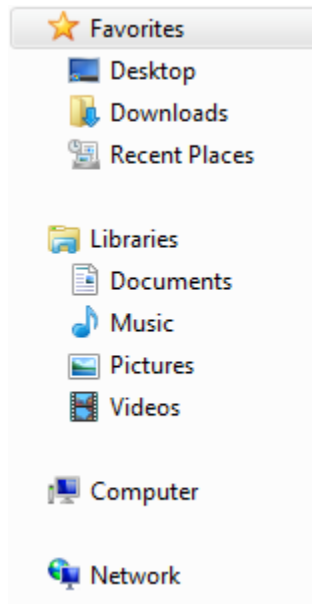
الحدث () comboBox1_SelectedIndexChanged

قم بكتابة الكود التالي بداخله كالتالي :

```
private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    MessageBox.Show(" : لقاخرتدولة " + comboBox1.Text);
}
```

9 القائمة الشجرية TreeView

سبق وأن شاهدت هذه الأداة مع متصفح الويندوز



وهي أداة تقوم بإظهار البيانات تنازليا على شكل بيانات شجرية متشعبة من الأصل إلى الفروع وتستعمل غالبا في تصفح وحدات الحاسوب Drives والمجلدات Folders والملفات Files ولها العديد من الخواص مثل Nodes و Right-To-Left وغيرها من الخواص والتي يمكنك التعرف عليها من قائمة Properties في Visual Studio .

برنامج (23) :

أنشئ مشروع جديد من قائمة **File -> New -> Project** من فيجوال ستوديو
واختر نوع المشروع **Windows Forms Application** باسم
TreeView1

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- TreeView:

Name: treeView1

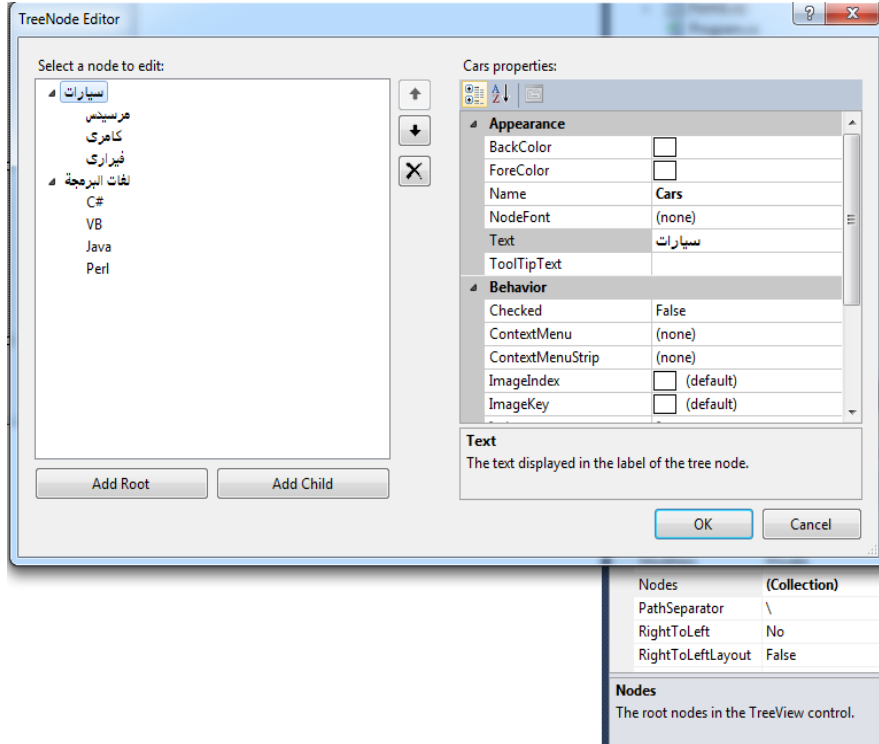
Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



أضف عناصر للقائمة الشجرية من الخاصية Nodes كالتالى :



برنامج (24) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

TreeView2

ضع على الفورم التى تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Label:

Name: label1

Text: انقر بالماوس مرتين لإختيار العنصر من الشجرة

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- ImageList:

Name: imageList1

Images: (Inside Image Collection Editor put Image called Folder.jpg)

3- TreeView:

Name: treeView1

ImageList: imageList1

ImageIndex: 0

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

Right-To-Left: Yes

Right-To-Left-Layout: True

SelectedImageIndex: 0



بالضغط مرتين بالماوس على الفورم Form1 ينشئ لنا

الحدث Form1_Load ()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void Form1_Load(object sender, EventArgs e)
{

    TreeNode tNode;
    tNode = treeView1.Nodes.Add("Websites");

    treeView1.Nodes[0].Nodes.Add("Net-informations.com");
    treeView1.Nodes[0].Nodes[0].Nodes.Add("CLR");

    treeView1.Nodes[0].Nodes.Add("Vb.net-informations.com");
    treeView1.Nodes[0].Nodes[1].Nodes.Add("String Tutorial");
    treeView1.Nodes[0].Nodes[1].Nodes.Add("Excel Tutorial");

    treeView1.Nodes[0].Nodes.Add("Csharp.net-informations.com");
    treeView1.Nodes[0].Nodes[2].Nodes.Add("ADO.NET");
    treeView1.Nodes[0].Nodes[2].Nodes[0].Nodes.Add("Dataset");

}
```

بالضغط مرتين بالماوس على `treeView1` `TreeView` ينشئ لنا

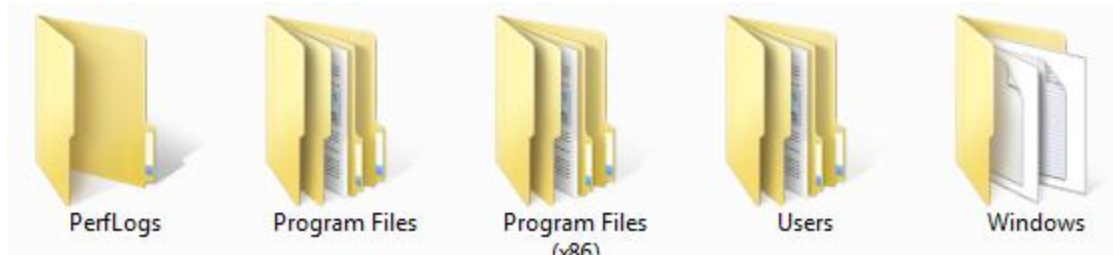
الحدث `treeView1_MouseDoubleClick()`

قم بكتابة الكود التالي بداخله كالتالي :

```
private void treeView1_MouseDoubleClick(object sender, MouseEventArgs e)
{
    TreeNode Node = treeView1.SelectedNode;
    MessageBox.Show(string.Format("لقد اخترت: {0}", Node.Text));
}
```

10 - قائمة العرض `ListView`

هذه الأداة شبيهة بصندوق القوائم `ListBox` إلا أنها أفضل منها حيث أنها تستطيع عرض البيانات بطريقة منسقة وجذابة كما تستطيع عرض الصور كما في متصفح الويندوز كما في متصفح الويندوز مصمم بهذه الأداة .



برنامج (25) :

أنشئ مشروع جديد من قائمة `File -> New -> Project` من فيجوال ستوديو

واختر نوع المشروع `Windows Forms Application` باسم

`ListView1`

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- Label:

Name: label1

Text: استعراض المجلدات

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

2- ComboBox:

Name: comboBox1

DropDownStyle: DropDownList

Items:

(Collection)

(Largelcon

Details

Smalllcon

List

Title)

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- ImageList:

Name: imageList1

Images: (Inside Image Collection Editor put Image called Folder.jpg)

4- ListView:

Name: listView1

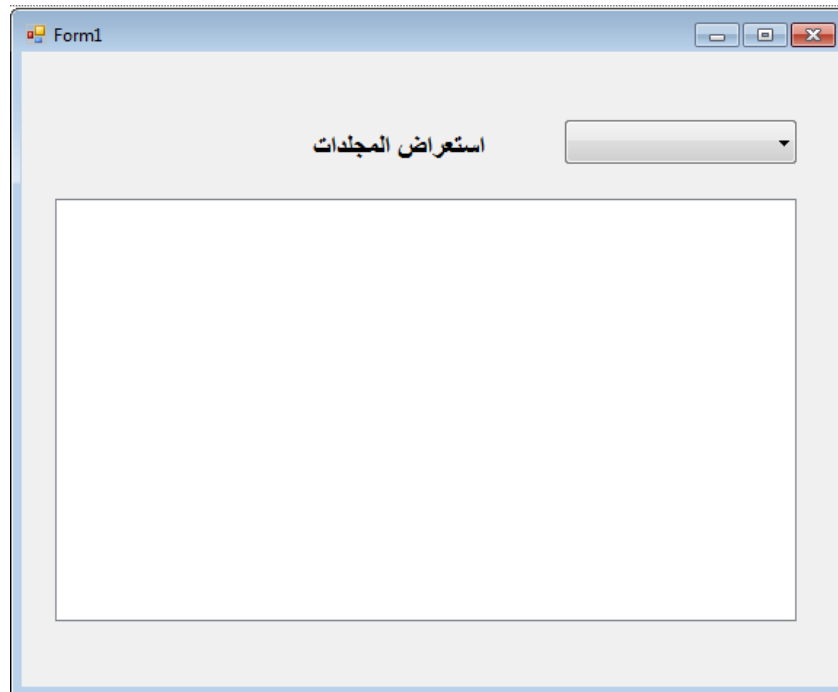
LargeImageList: imageList1

SmallImageList: imageList1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بكتابة هذا الكود داخل Constructor Form1() لإظهار أسماء وتفاصيل المجلدات داخل ListView

```
public Form1()
{
    InitializeComponent();

    listView1.Columns.Add("الإسم");
    listView1.Columns.Add("التفاصيل");

    listView1.Items.Add("المجلد 1", 0);
    listView1.Items.Add("المجلد 2", 0);
    listView1.Items.Add("المجلد 3", 0);
    listView1.Items.Add("المجلد 4", 0);
    listView1.Items.Add("المجلد 5", 0);
}
```

بالضغط مرتين بالماوس على **comboBox1** ينشئ لنا

الحدث **comboBox1_SelectedIndexChanged()**

قم بكتابة الكود التالي بداخله كالتالي :

```
private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    if (comboBox1.Text == "LargeIcon")
    {
        listView1.View = View.LargeIcon;
    }
    elseif (comboBox1.Text == "Details")
    {
        listView1.View = View.Details;
    }
    elseif (comboBox1.Text == "SmallIcon")
    {
        listView1.View = View.SmallIcon;
    }
    elseif (comboBox1.Text == "List")
    {
        listView1.View = View.List;
    }
    else
    {
        listView1.View = View.Tile;
    }
}
```

برنامج (26) :

أنشئ مشروع جديد من قائمة **File -> New -> Project** من فيجوال ستوديو
واختر نوع المشروع **Windows Forms Application** باسم

ListView2

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- ImageList:

Name: imageList1

Images: (Inside Image Collection Editor put Images called →

Drive.png, Folder.png, File.png)

2- ListView:

Name: lvDrives

LargeImageList: imageList1

SmallImageList: imageList1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

3- ListView:

Name: lvFiles

LargeImageList: imageList1

SmallImageList: imageList1

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True

4- TreeView:

Name: tvFolders

ImageIndex: 0

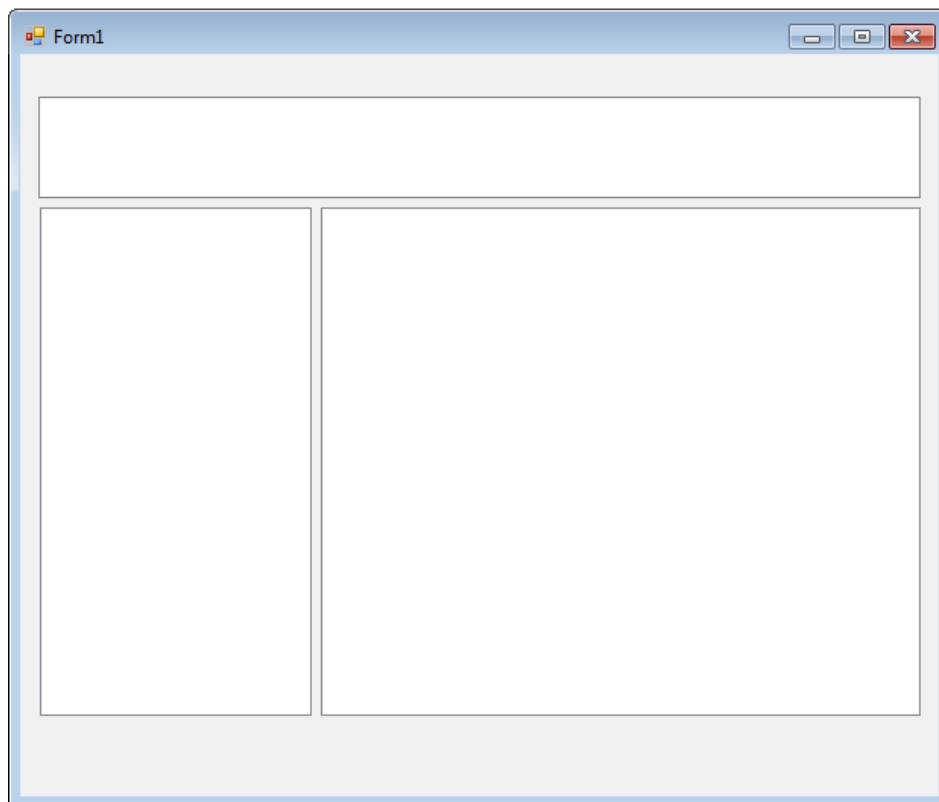
ImageList: imageList1

SelectedImageIndex: 0

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



قم بكتابة أكواد الدوال التالية للتعامل مع وحدات الحاسوب Drives و المجلدات Folders والملفات Files كالتالى :

```
string strDrive;

// First Method
void getDrives()
{
    try
    {
        foreach (string drive inDirectory.GetLogicalDrives())
        {
            this.lvDrives.Items.Add(drive, 0);
        }
    }

    catch (Exception ex)
    {
        ex.Message.ToString();
        lblError.Text = "لم يتم الوصول لوحدات الحاسب";
    }
}

// Second Method
void getFolders()
{
    try
    {
        strDrive = lvDrives.FocusedItem.Text;
        DirectoryInfo Dir = newDirectoryInfo(strDrive);
        tvFolders.Nodes.Clear();
        foreach (DirectoryInfo Folder in Dir.GetDirectories())
        {
            tvFolders.Nodes.Add("", Folder.Name, 1, 1);
        }
    }

    catch (Exception ex)
    {
        ex.Message.ToString();
        lblError.Text = "لم يتم الوصول للمجلدات الحاسب";
    }
}

// Third Method
void getFiles(string strPath)
{
    try
    {
```

```

ListViewItem lvi;
DirectoryInfo Dir = newDirectoryInfo(strDrive + strPath);
lvFiles.Items.Clear();
foreach (FileInfo file in Dir.GetFiles())
{
    lvi = lvFiles.Items.Add(file.Name, 2);
    lvi.SubItems.Add(file.Length.ToString());
    lvi.SubItems.Add(file.LastAccessTime.ToString());
    lvi.SubItems.Add(file.Extension);
}

}

catch (Exception ex)
{
    ex.Message.ToString();
    lblError.Text = "لم يتم الوصول إلى الملف";
}

}

```

قم بإستدعاء الدالة **getDrives()** داخل **Form1() Constructor** كالتالى :

```

public Form1()
{
    InitializeComponent();

    getDrives();
}

```

باختيار الحدث **AfterSelect** الخاص بالأداة **TreeView tvFolders**

قم بكتابة الكود التالى بداخله كالتالى :

```

privatevoid tvFolders_AfterSelect(object sender, TreeViewEventArgs e)
{
    try
    {
        getFiles(e.Node.FullPath);
        TreeNode node;
        DirectoryInfo Dir = newDirectoryInfo(strDrive +
            e.Node.FullPath);
        foreach (DirectoryInfo folder in Dir.GetDirectories())
        {
            node = newTreeNode(folder.Name, 1, 1);
            e.Node.Nodes.Add(node);
        }
    }
    catch (Exception ex)
    {
    }
}

```

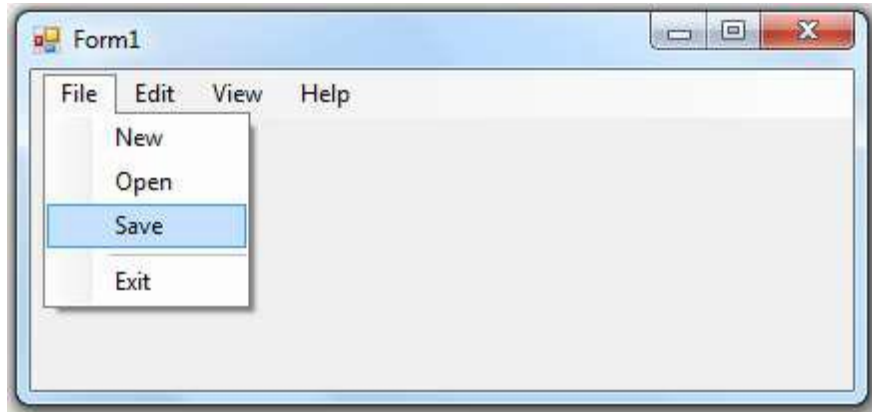
```

ex.Message.ToString();
lblError.Text = "لم يتم الوصول للمجلدات الحاسب";
}
}

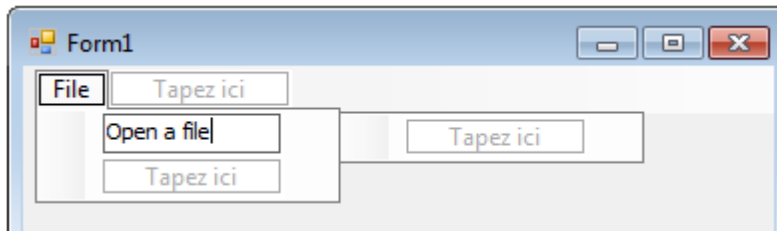
```

11 - أداة القائمة الرئيسية MenuStrip

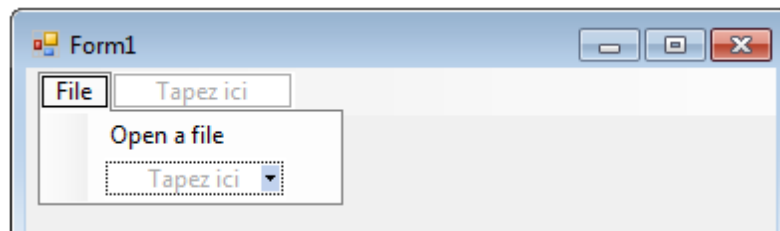
تعد هذه الأداة من من أهم الأدوات التي يحفل بها معظم البرامج وهذه صورة لها :



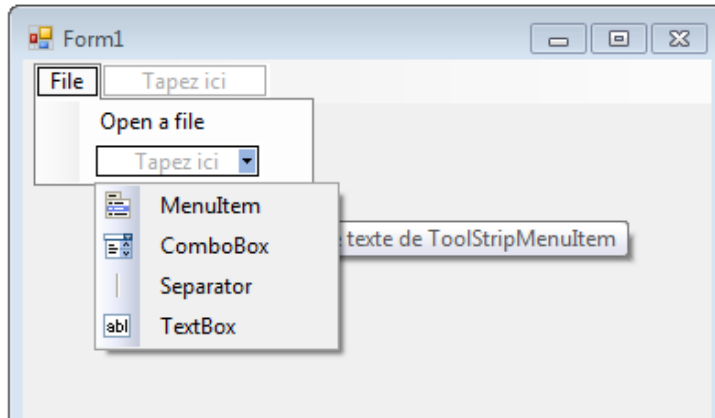
من علبة الأدوات Toolbox قم بجذب أداة القائمة MenuStrip إلى الفورم جرب أن تضيف بعض القوائم الرئيسية وبعض القوائم الفرعية لها



إذا مررت بال مؤشر فوق مكان كتابة اسم القائمة سيظهر لك سهم صغير موجه نحو الأسفل كما تظهر الصورة التالية :



إذا قمت بالضغط عليه ستظهر لك الإختيارات التالية :



برنامج (27) :

أنشئ مشروع جديد من قائمة **File -> New -> Project** من فيجوال ستوديو

واختر نوع المشروع **Windows Forms Application** باسم

MenuStrip

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- MenuStrip:

Name: menuStrip1

Text : menuStrip1

2- fileToolStripMenuItem:

Name: fileToolStripMenuItem

Text: &File

3-openToolStripMenuItem

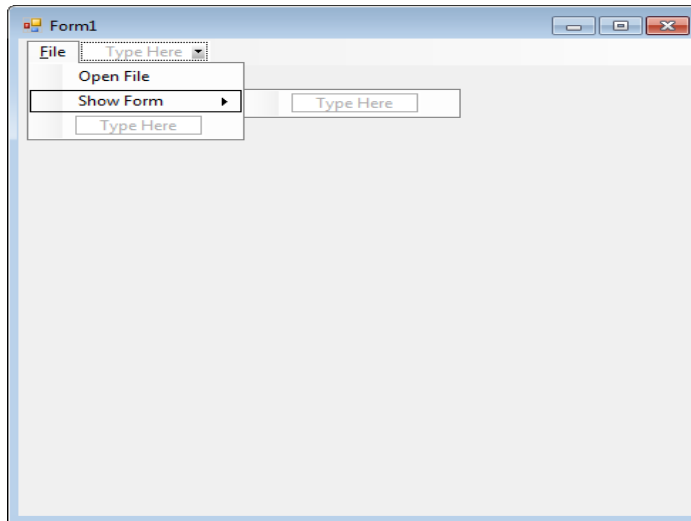
Name: openToolStripMenuItem

Text: Open File

4-showFormToolStripMenuItem

Name: showFormToolStripMenuItem

Text: Show Form



بالماسع لنزر بالضغط مرتين

Button showFormToolStripMenuItem

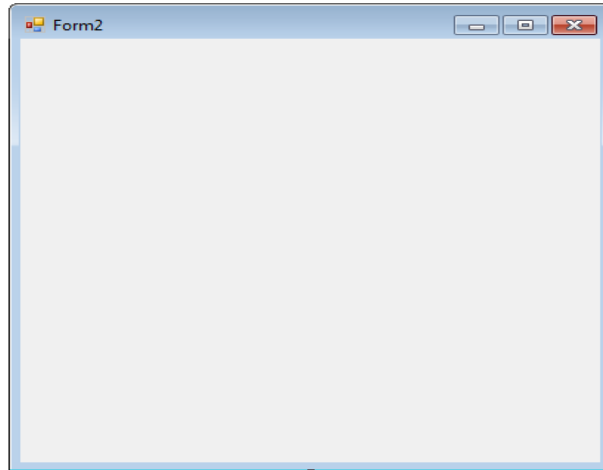
ينشئ لنا الحدث

showFormToolStripMenuItem_Click()

قم بكتابة الكود التالي بداخله كالتالي :

```
private void showFormToolStripMenuItem_Click(object sender, EventArgs e)
{
    Form2 frm = new Form2();
    frm.Show();
}
```

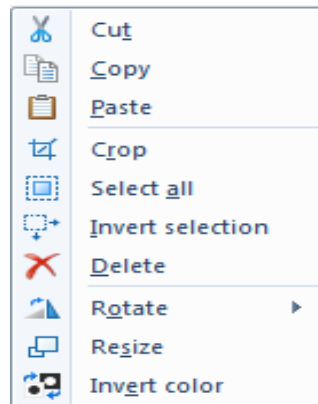
إنشاء فورم أخرى Form2 يتم مناداتها من خلال الزر الموجود بالفورمة السابقة



12 - أداة القائمة المنسدلة ContextMenuStrip

طبعا سبق لك وأن رأيت القائمة المنسدلة بالضغط على يمين الماوس

كالتالى :



برنامج (28) :

أنشئ مشروع جديد من قائمة File -> New -> Project من فيجوال ستوديو

واختر نوع المشروع Windows Forms Application باسم

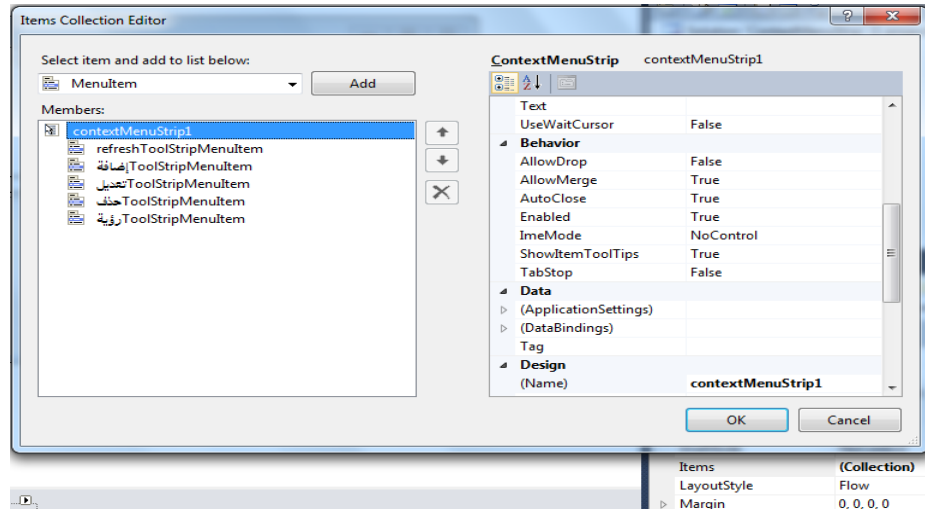
ContextMenuStrip

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

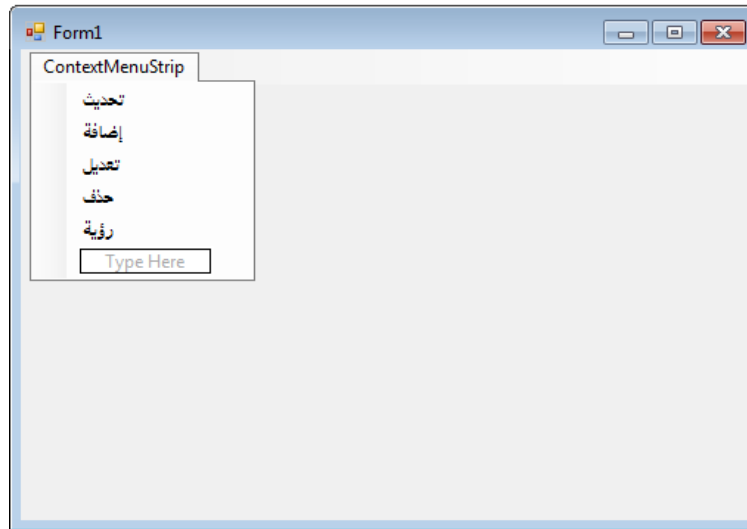
1- ContextMenuStrip:

Name: contextMenuStrip1

Items: Collection



عند التصميم تظهر الفورم فى التصميم بالشكل التالى :



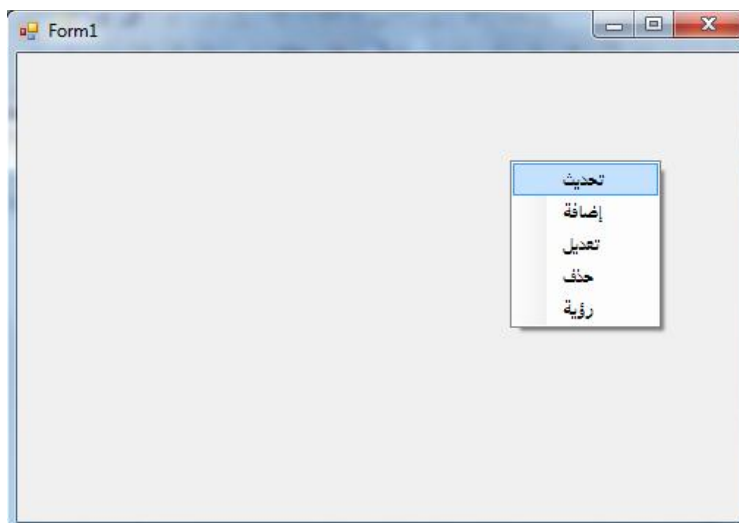
2-Form1

ContextMenuStrip: contextMenuStrip1

Name: Form1

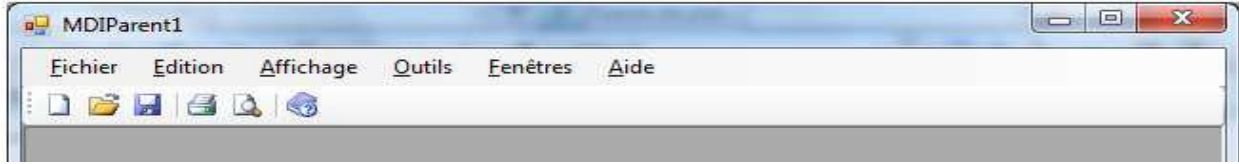
Text: Form1

عند تشغيل البرنامج تظهر الفورم بالشكل التالى :



13 - أداة شريط الأدوات ToolStrip

غالبا ما يكون الغرض من هذه الأداة هو عرض محتوى القوائم الفرعية بالصورة وبشكل واضح يجعل المستخدم أكثر ارتياحا وهذه صورة الأداة .



برنامج (29) :

أنشئ مشروع جديد من قائمة **File -> New -> Project** من فيجوال ستوديو واختر نوع المشروع **Windows Forms Application** باسم

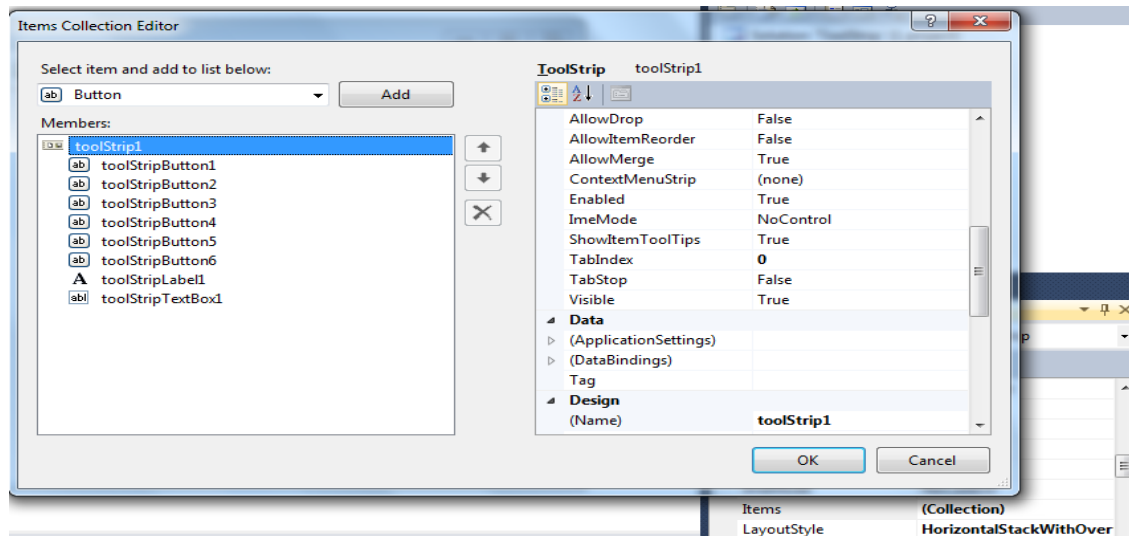
ToolStrip

ضع على الفورم التي تظهر لك الأدوات التالية وأعطى لها الخصائص الآتية:

1- ToolStrip

Name: toolStrip1

Items: Collection



LayoutStyle: HorizontalStackWithOverflow

Text: toolStrip1

1- Label:

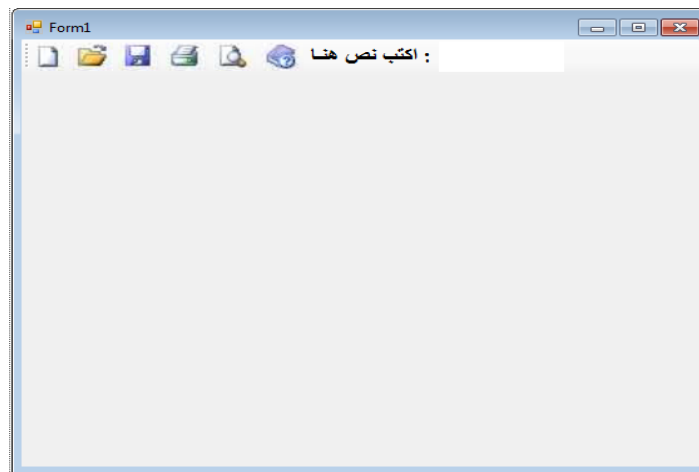
Name: label1

Text:

Font Name: Times New Roman, 14pt, style=Bold

Size: 14

Bold: True



بالضغط مرتين بالماوس على toolStrip1textBox1 ينشئ لنا

الحدث () toolStrip1textBox1_TextChanged

قم بكتابة الكود التالي بداخله كالتالي :

```
private void toolStrip1textBox1_TextChanged(object sender, EventArgs e)
{
    this.label1.Text = toolStrip1textBox1.Text;
}
```

(13) مشاريع للتعامل مع قواعد البيانات

(الإصدار الثاني من الكتاب)

وآخر دعوانا

(أن الحمد لله رب العالمين)