



# Microsoft Visual Basic

# دليل المبتدئين... جدا جدا Express 2010

مقدم من مدونة البوهريزم



<http://algharism.blogspot.com>

```

57 'For i = 0 To 10
58 '    MessageBox.Show("The value of i is: " & i)
59 'Next
60
61 ' Example 8 - 'do while/loop' recursion statement
62 'Dim i As Integer = 0
63 'Do While i < Integer.Parse(textBox1.Text)
64 '    i += 1
65 '    i = i + 1
66 'Loop
67 'MessageBox.Show("The final value was: " & i)
68
69
70 ' Example 9 - combining array and branching statements
71 'Dim myArray() As String = {"أحمد", "الجمهرى", "أحمد"}
72
73 'Dim i As Integer
74 'For i = 0 To myArray.Length - 1
75 '    If myArray(i) = "أحمد" Then
76 '        MessageBox.Show("تم إيجاد أحمد")
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96

```

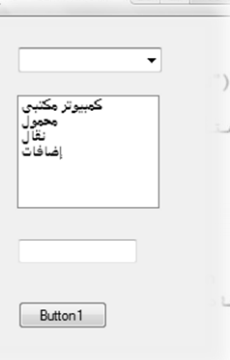
```

' ComboBox1.Text = "ممدوح"
' End If
' Else
'     MessageBox.Show("لمست ممدوح")
' End If
' Else
'     MessageBox.Show("تم إيجاد أحمد")
' End If
' Next

' Example 10 - recursion and branching combined
Dim i As Integer
For i = 0 To ListBox1.SelectedItems.Count - 1
    Select Case ListBox1.SelectedItems(i).ToString()
        Case "كمبيوتر مكتبي"
            MessageBox.Show("كمبيوتر")
        Case "نقال"
            MessageBox.Show("نقال")
        Case "محمول"
            MessageBox.Show("محمول")
        Case "إضافات"
            MessageBox.Show("إضافات")
    End Select
Next

```

جميع الحقوق محفوظة ©



## بسم الله الرحمن الرحيم

### تقديم

يسعد مدونة الجوهريزم أن تقدم لكم كتاب " **Microsoft Visual Basic 2010 Express دليل للمبتدئين جدا** "، فما هو إلا ملخص و ترجمة لبعض المواد التعليمية الخاصة بتعليم لغة الـ Visual Basic حيث اقتصر دورى على ترجمة و توضيح بعض المواد الأجنبية و العربية و تنقيتها لتناسب المستخدم العربى المبتدئ فى البرمجة، و قد روعى عند اختيار شرح هذه اللغة أنها سهلة و تقترب فى أوامرها من اللغة الانجليزية ، و روعى أيضا ترجمة المفاهيم الانجليزية الصعبة بلغة عربية مع أمثلة لتوضيح المعنى.

هذا الدليل مقدم للمبتدئ من الصفر فى لغة البرمجة ، فالدليل لا يبدأ كما تبدأ معظم الكتب و إنما يبدأ بإنشاء برنامج دون معرفة أى قواعد عن اللغة ثم نتطرق من خلال هذا البرنامج فى شرح بعض المقومات و المفاهيم الأساسية فى لغة الـ visual basic.

استخدمنا فى الشرح برنامج Microsoft Visual Studio Express 2010 و هو مجانى و يمكنك تنزيله من :

<http://www.microsoft.com/express/Downloads>

و أخيرا أرجو أن يكون عملا مفيدا و نيتى الإفادة ، و كل ما أتمناه دعوة صالحة لظهر الغيب ، و يمكنكم التواصل عن طريق :

المدونة: <http://algharism.blogspot.com>

صفحة الفيس بوك: <http://www.facebook.com/algharism>

البريد الإلكتروني: [algharism@gmail.com](mailto:algharism@gmail.com)

الجوهريزم

الحياة بعيون جوهريية

مارس 2011

## الفهرس

### الدرس الأول

1. التشغيل: ..... 6
2. مكونات الصفحة الرئيسية ..... 8
3. أول برنامج Hello world ..... 11
4. تدريب ..... 15

### الدرس الثاني

1. الخصائص Properties ..... 17
2. أهم العناصر فى صندوق الأدوات(الأدوات الشائعة) ..... 22
3. أهم العناصر فى صندوق الأدوات(أدوات القائمة و صندوق الأدوات) ..... 25
4. نصائح فى تصميم واجهة المشاريع ..... 32
5. تدريب ..... 39

### الدرس الثالث

1. مقدمة عن Events ..... 41
2. كيف أكتب الكود؟ ..... 43
3. أدوات تساعدك فى كتابة الكود ..... 48
4. تدريب ..... 51

### الدرس الرابع

1. مقدمة عن Variables المتغيرات ..... 53
2. كيفية تعريف المتغيرات ..... 56
3. كيفية التحويل بين أنواع المتغيرات ..... 59
4. بعض التعريفات فى المتغيرات ..... 64
5. تدريب ..... 65

### الدرس الخامس

1. مقدمة ..... 67
- 1.2. الجمل التشعبية: If...Then ..... 68
- 2.2. الجمل التشعبية: Select Case ..... 72
3. متغيرات متعددة المحتوى Arrays ..... 73
- 1.4. الجملة الحلقية التكرارية: for/each ..... 77
- 2.4. الجملة التكرارية: for/next ..... 78
5. تركيبات بين الجمل التكرارية و التشعبية ..... 80

.....83.....
.....85.....
.....86.....
.....90.....
.....91.....
.....92.....
.....94.....
.....97.....

6.تدريب

### الدرس السادس

1.إنشأ مشروع جديد

2.أبدأ فى التصميم

3.غير أسماء العناصر

4.أضف مربعات الحوار

5.ابدأ فى كتابة الكود

6.لإخراج البرنامج مستقل

الملحقات

---

# Microsoft VB Express للمبتدئين 2010

---

الدرس الأول

---

[algharism.blogspot.com](http://algharism.blogspot.com)

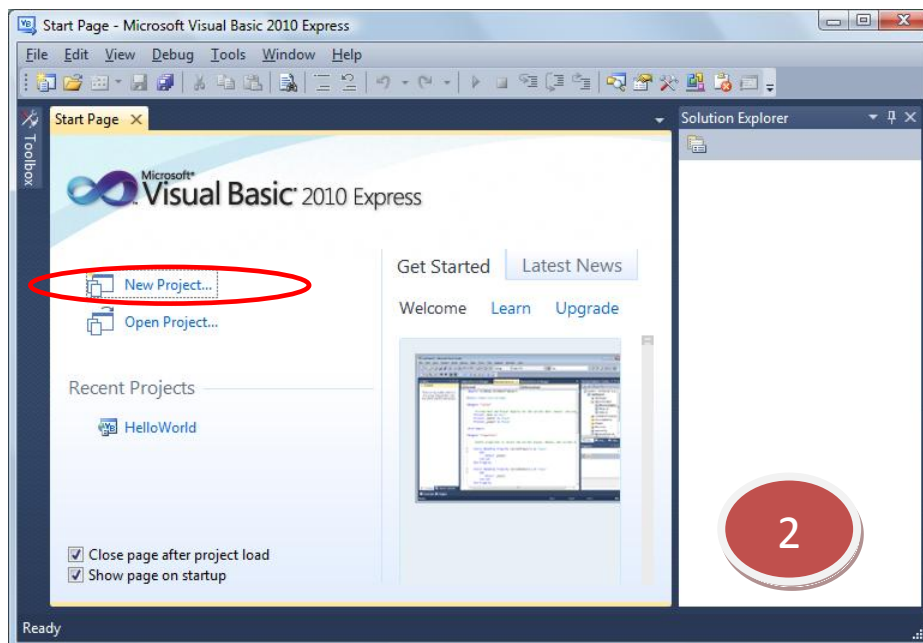
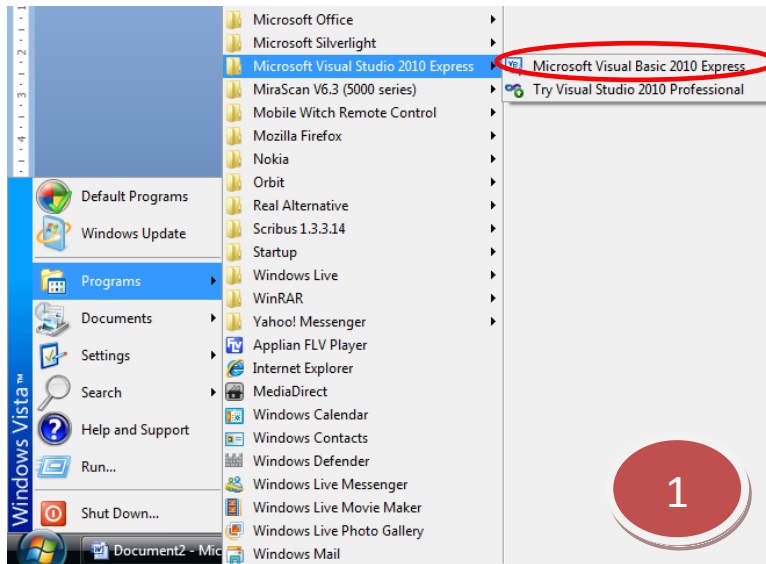
---

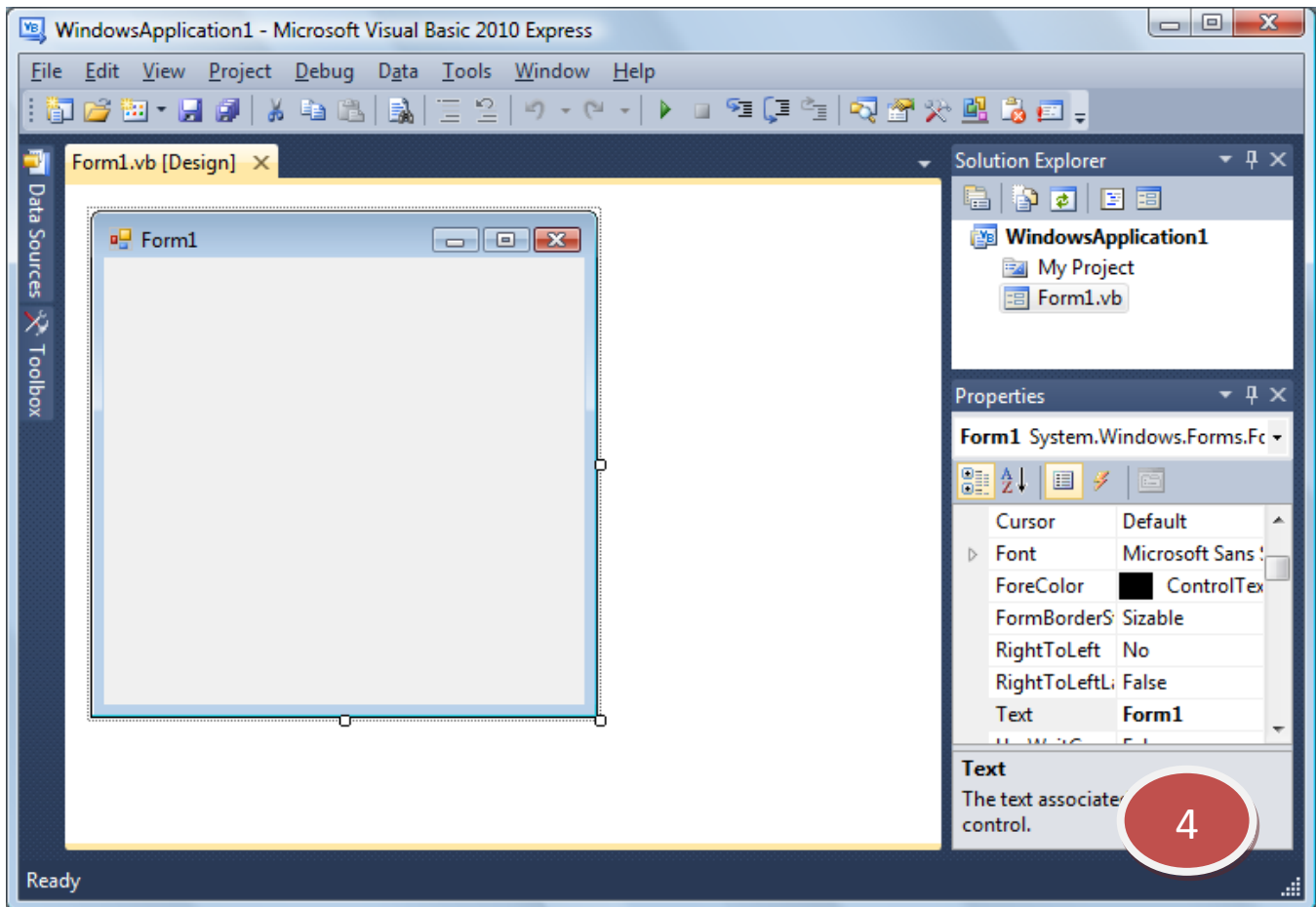
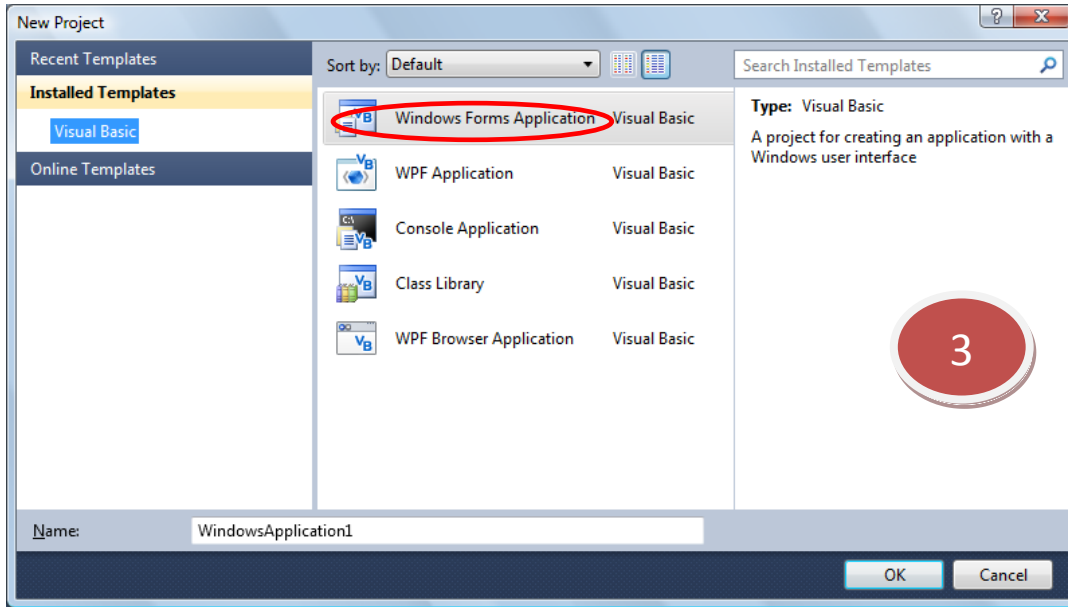
الدرس الأول يتضمن مقدمة سريعة لواجهة التشغيل و  
انشاء و حفظ أول مشروع Hello World.

# 1. التشغيل:



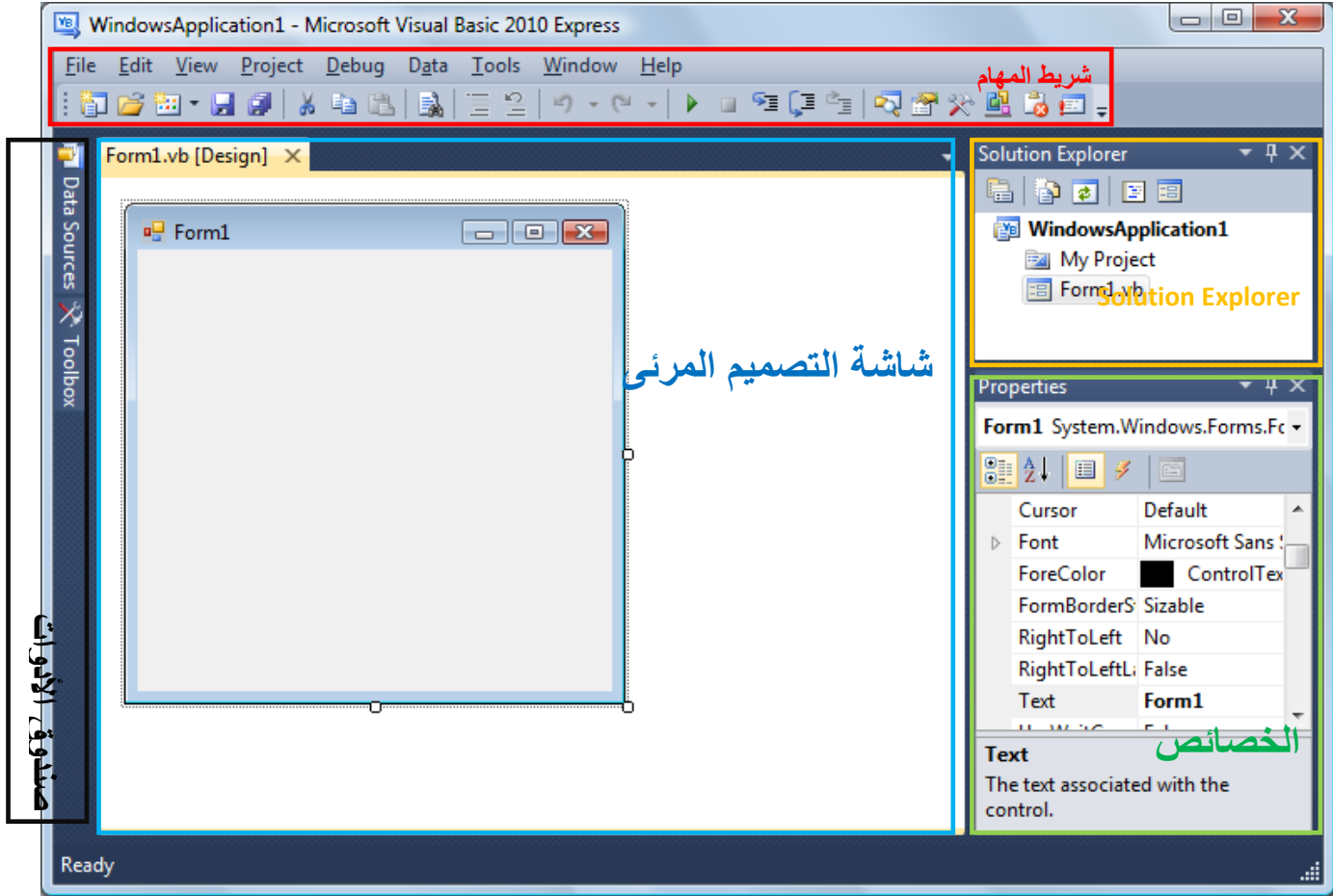
1. Start menu >> Programs >> Microsoft Visual Studio 2010 Express >> Microsoft Visual Basic 2010 Express
2. New Project... مشروع جديد
3. Windows Forms Application انشاء برنامج يستخدم النوافذ
4. النافذة الرئيسية للبرنامج



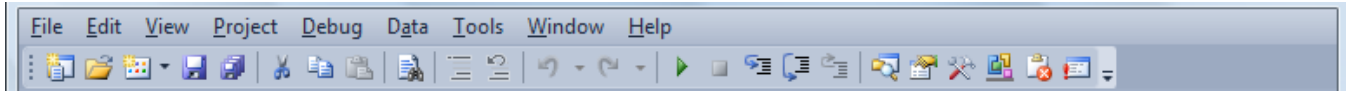


## 2. مكونات الصفحة الرئيسية

سنعرض المكونات الأساسية للصفحة الرئيسية للبرنامج حتى تكون مألوفة عند الشرح فيما بعد. سيتم توضيح بعض الخصائص و التعمق أكثر فيها عند التقدم في البرنامج.



1 - شريط المهام

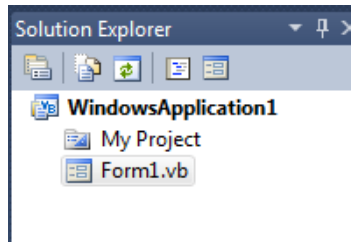


يكون في أعلى الصفحة و يمكنك من القيام بجميع خصائص البرنامج

2 - Solution Explorer

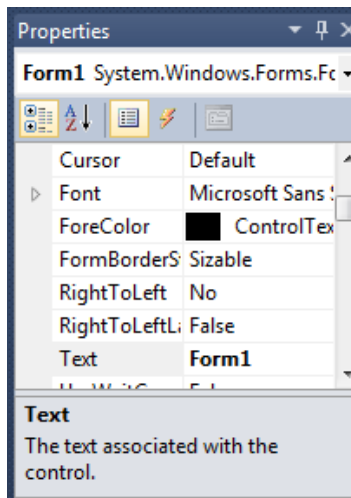
يكون في الجانب الأيمن من الصفحة و يوضح لك العناصر و الكائنات المستخدمة في المشروع الخاص بك





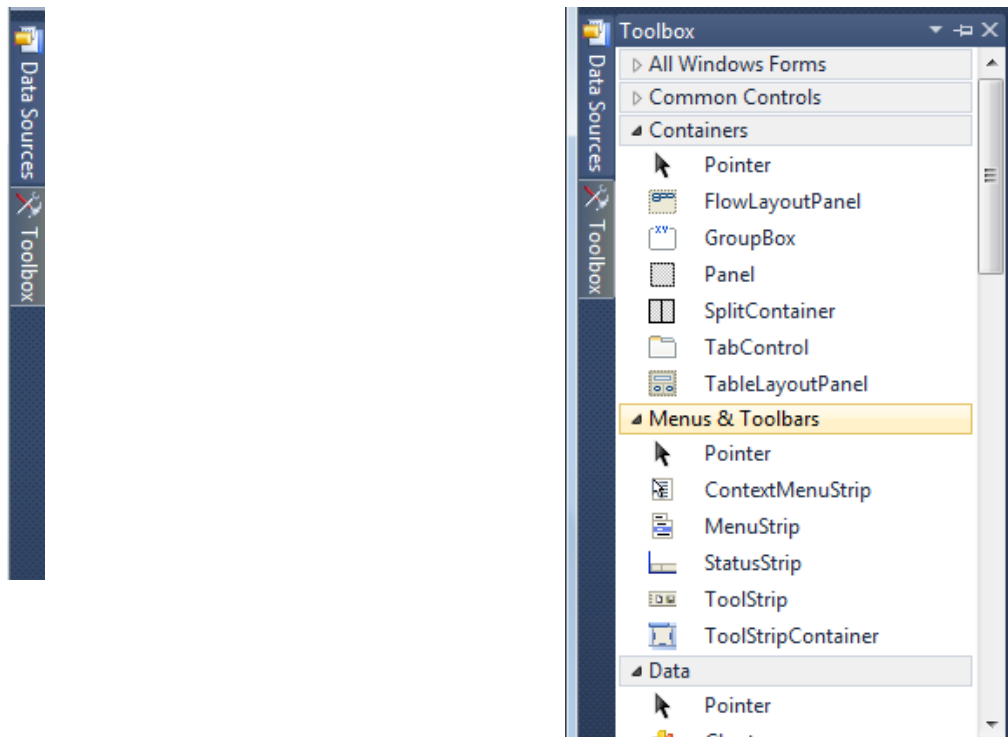
### 3 - الخصائص Properties

يعرض خصائص و مميزات الكائن (سواء زر ، قائمة ... إلخ) و يكون في الجانب الأيمن في الأسفل



### 4 - صندوق الأدوات

و هو صندوق العدد حيث يوفر لك جميع الأدوات التي ستحتاجها خلال التصميم المرئى و يكون في الجانب الأيسر



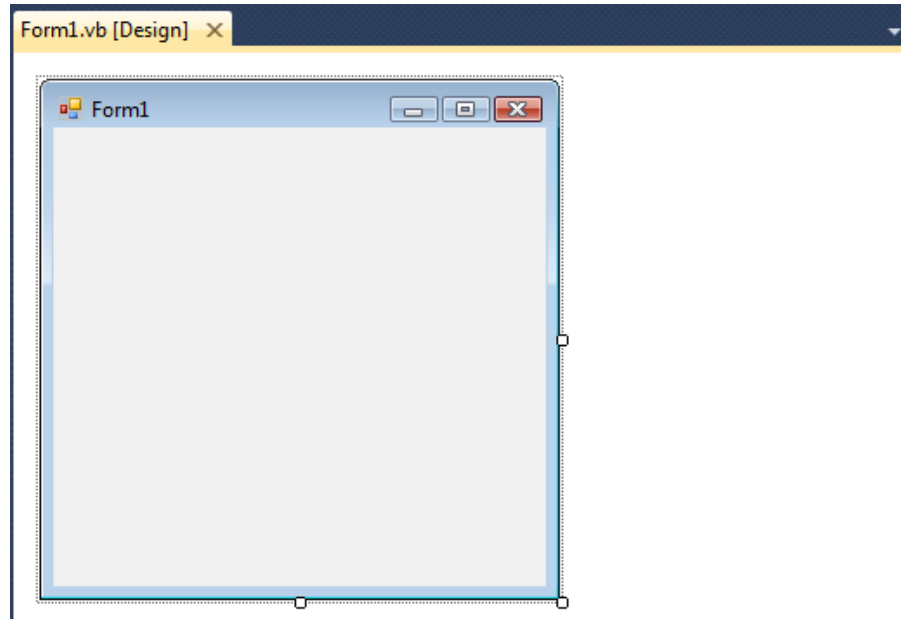


عند الاشارة بالماوس على صندوق الأدوات Toolbox سيظهر أوتوماتيكيا و سيختفى عند ابتعاد الماوس عنه كما هو موضح بالشكل الأعلى



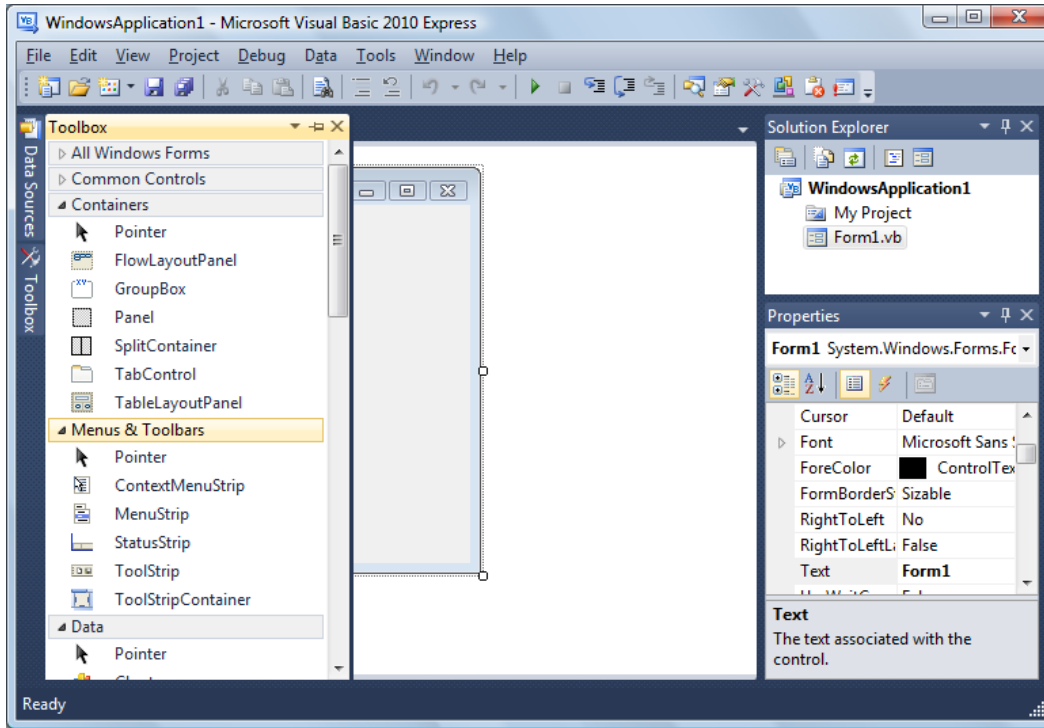
5 - شاشة التصميم (مرئى \ عن طريق كود البرمجة)

و هو النافذة الرئيسية فى البرنامج



## 3. أول برنامج Hello world

سننعمق أكثر في مكونات الـ Visual Basic و ذلك من خلال برنامج افتتاحي عن طريقه سنستكشف بعض المهارات المعرفية.

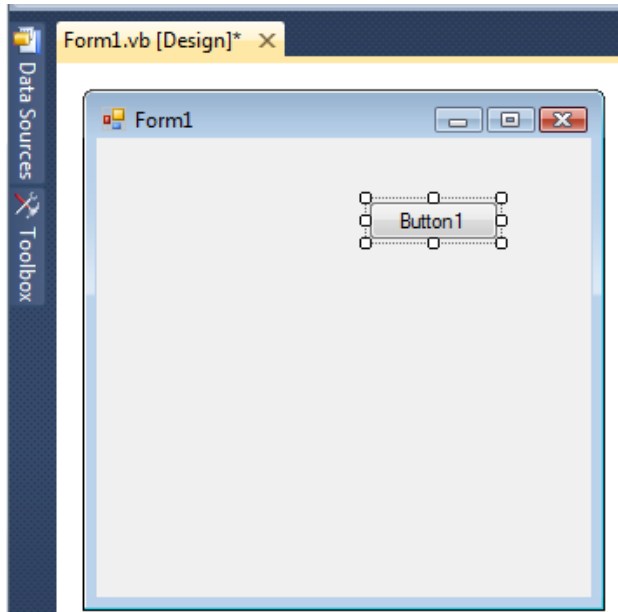


1 - اشر (شاور) بالماوس على صندوق الأدوات Toolbox .

Button  Button

2 - من Common Controls اضغط واستمر في الضغط على

3 - اسحبه و ضعه في النافذة الرئيسية.



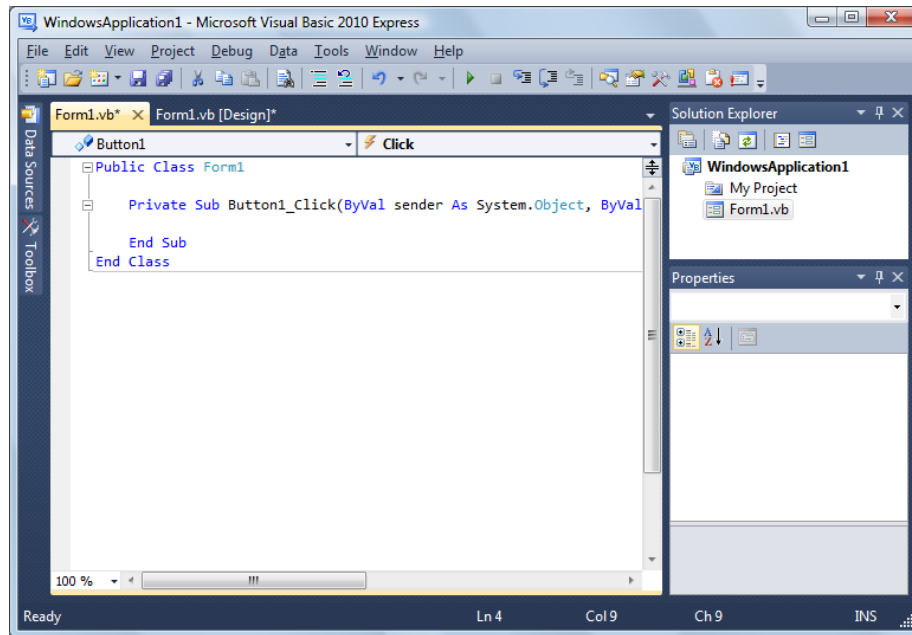
يمكنك ادخال Button عن طريق ضغطتين متتاليتين على Double click رمز Button.

 Button



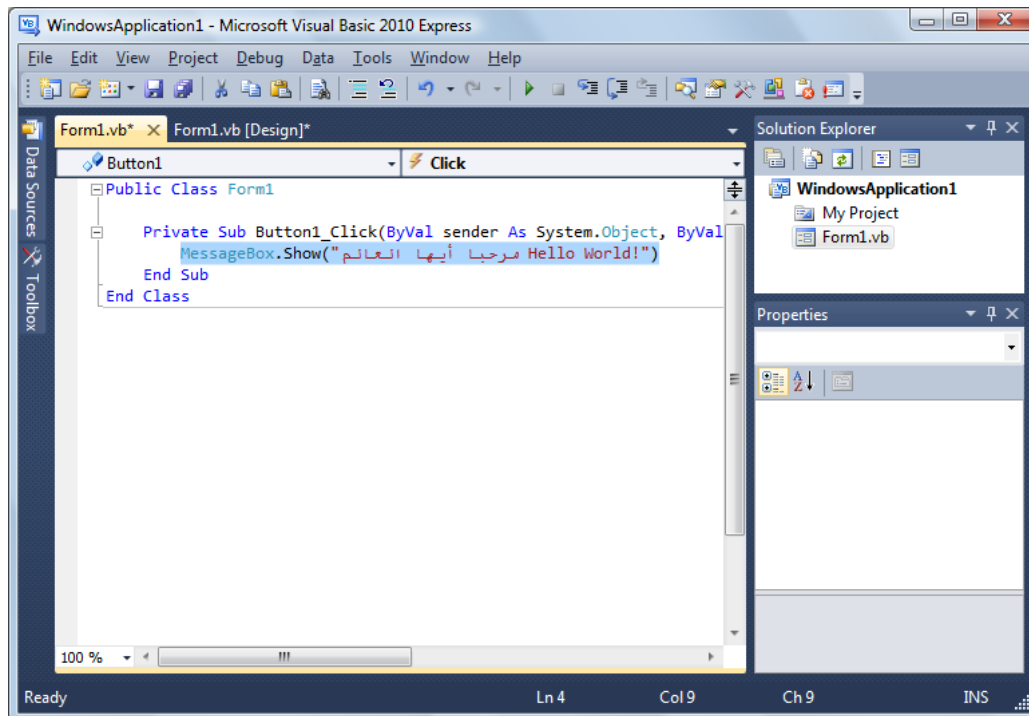


## 4 - اضغط عليه Double Click لتظهر لك نافذة التكويد

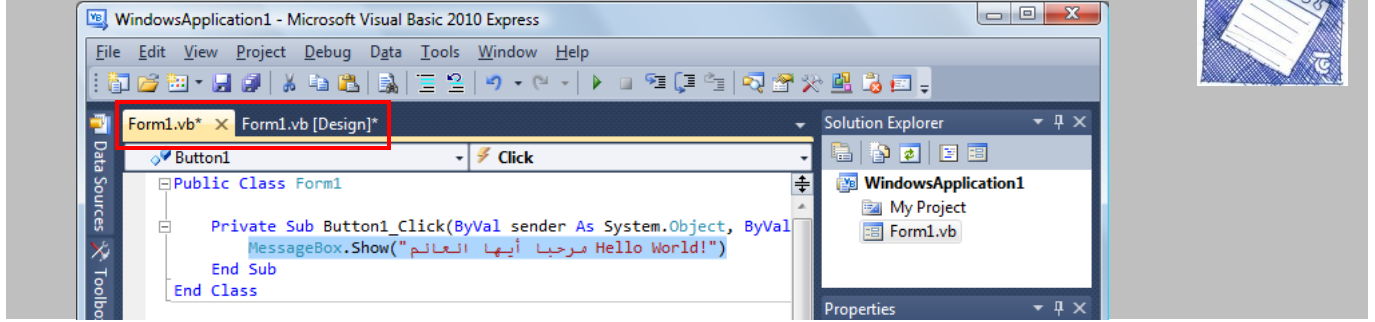


## 5 - اكتب في مكان كتابة الكود

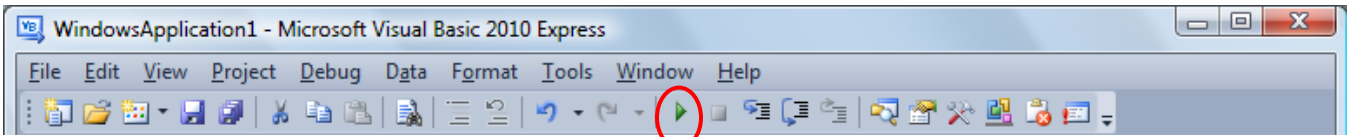
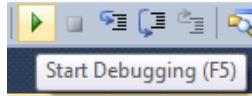
MessageBox.Show("Hello World! العالم أيها مرحبا")



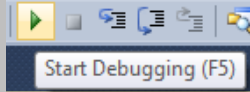
يمكنك الانتقال بين نافذة التصميم المرئي و نافذة التكويد من أعلى الصفحة



6 - اضغط على زر Debug في شريط الأدوات.



- زر Debug عمله هو معاينة كيفية سير عمل البرنامج و مراجعة إذا كان فيه أخطاء برمجية.
- يمكنك الضغط على F5 كوصلة سريعة لـ Debug.



7 - ستظهر هذه النافذة التي صممتها، ثم اضغط على Button 1 ستظهر لك رسالة كما هو موضح.

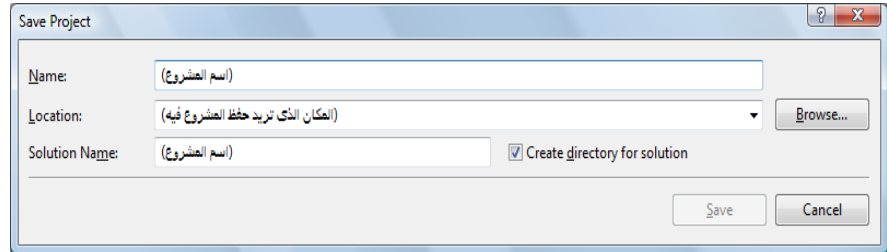
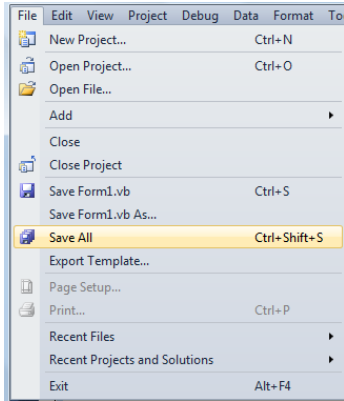


8 - و عند الانتهاء اضغط على X و ستعود إلى نافذة التصميم.





## 9 - لقد انتهينا من أول برنامج و نريد حفظ المشروع File >> Save All



10 - اضغط Save.



## 4. تدريب

---

- اصنع برنامج مكون من ثلاثة ازرار 3 Buttons و بحيث عندما تضغط على الأول يظهر لك (لقد ضغط الأول) و عندما تضغط على الثانى يظهر لك (لقد ضغطت الثانى) و الثالث كذلك.
- عاين عمل البرنامج.
- احفظ المشروع باسم Hello World فى مجلد المستندات.

---

# Microsoft VB Express للمبتدئين 2010

---

الدرس الثانى

---

[algoharism.blogspot.com](http://algoharism.blogspot.com)

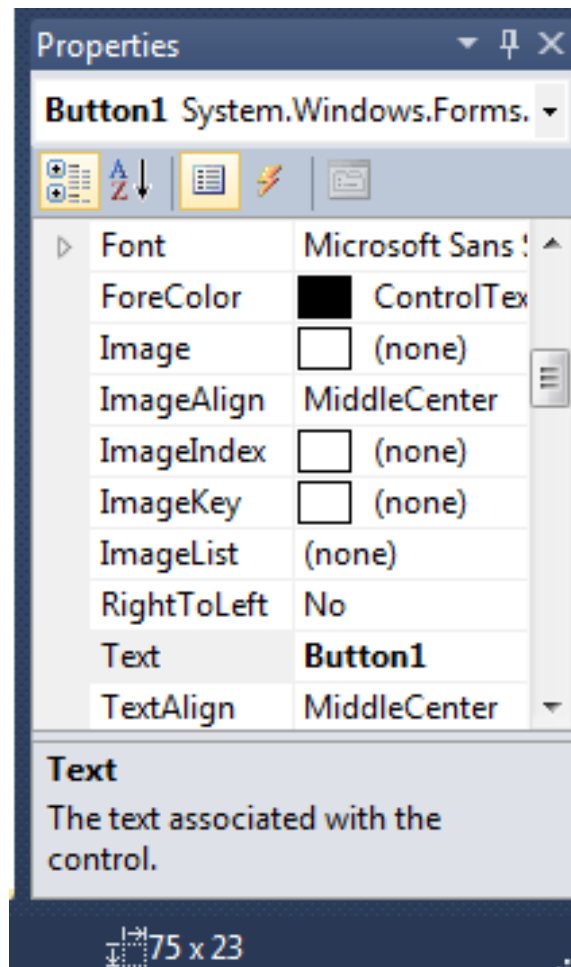
---

الدرس الثانى سنتعرف على بعض أهم العناصر فى  
صندوق الخصائص و أفضل طرق تصميم واجهة  
الاستخدام .

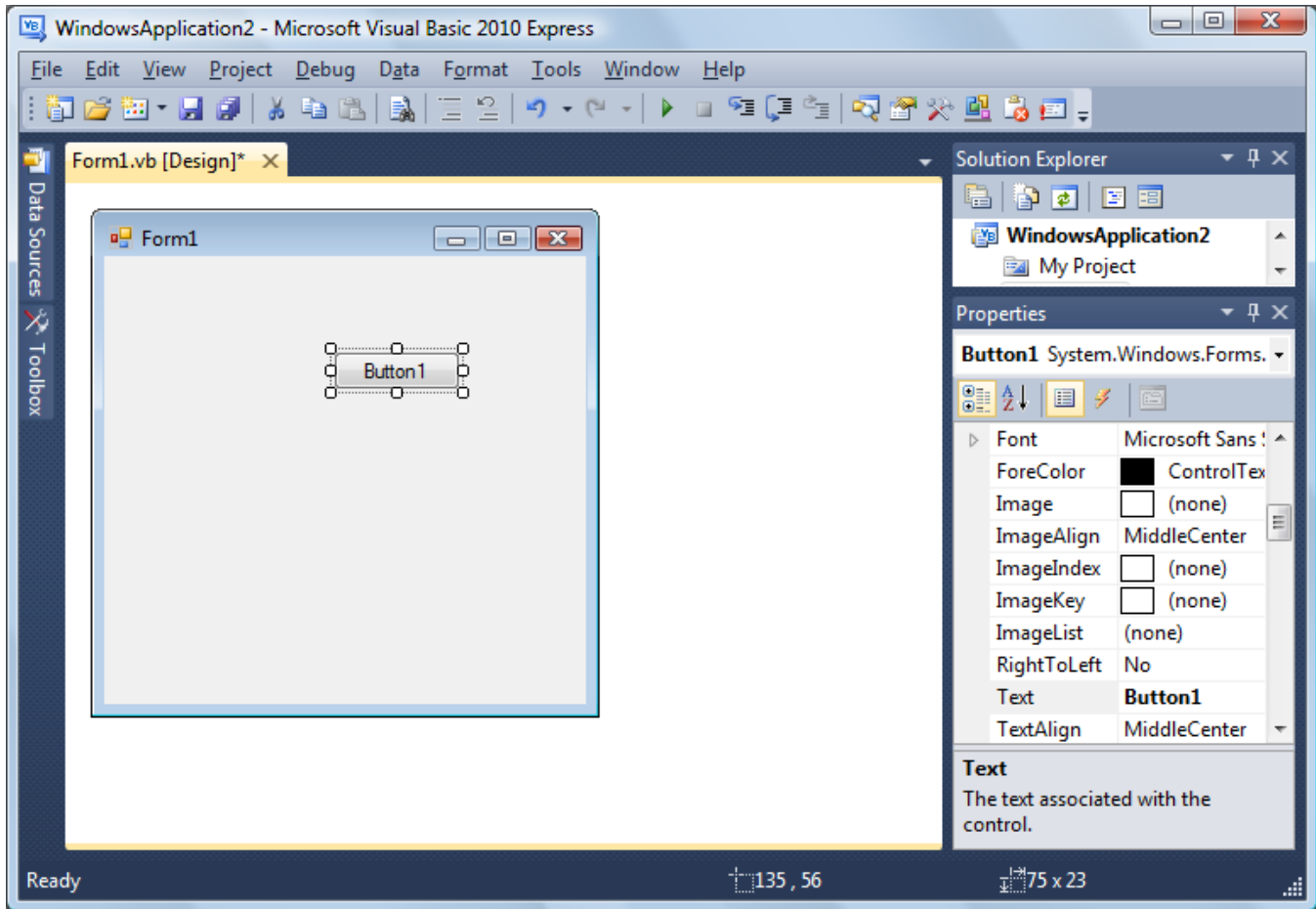


# 1. الخصائص Properties

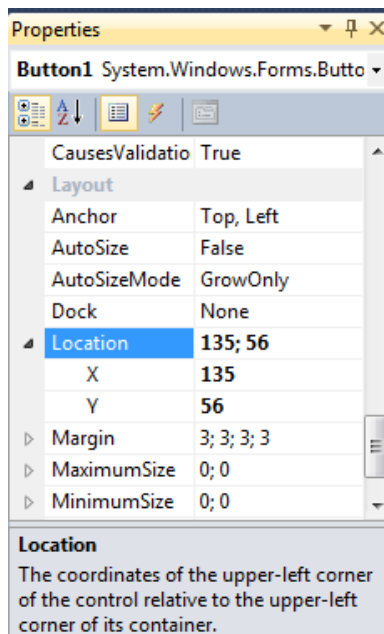
تحدثنا باختصار عن صندوق الخصائص و هو خاص بتجهيز الاعدادات المناسبة لكل أداة مستخدمة في المشروع و الخصائص تعنى أساسا بالصفات الشكلية للأداة المستخدمة.



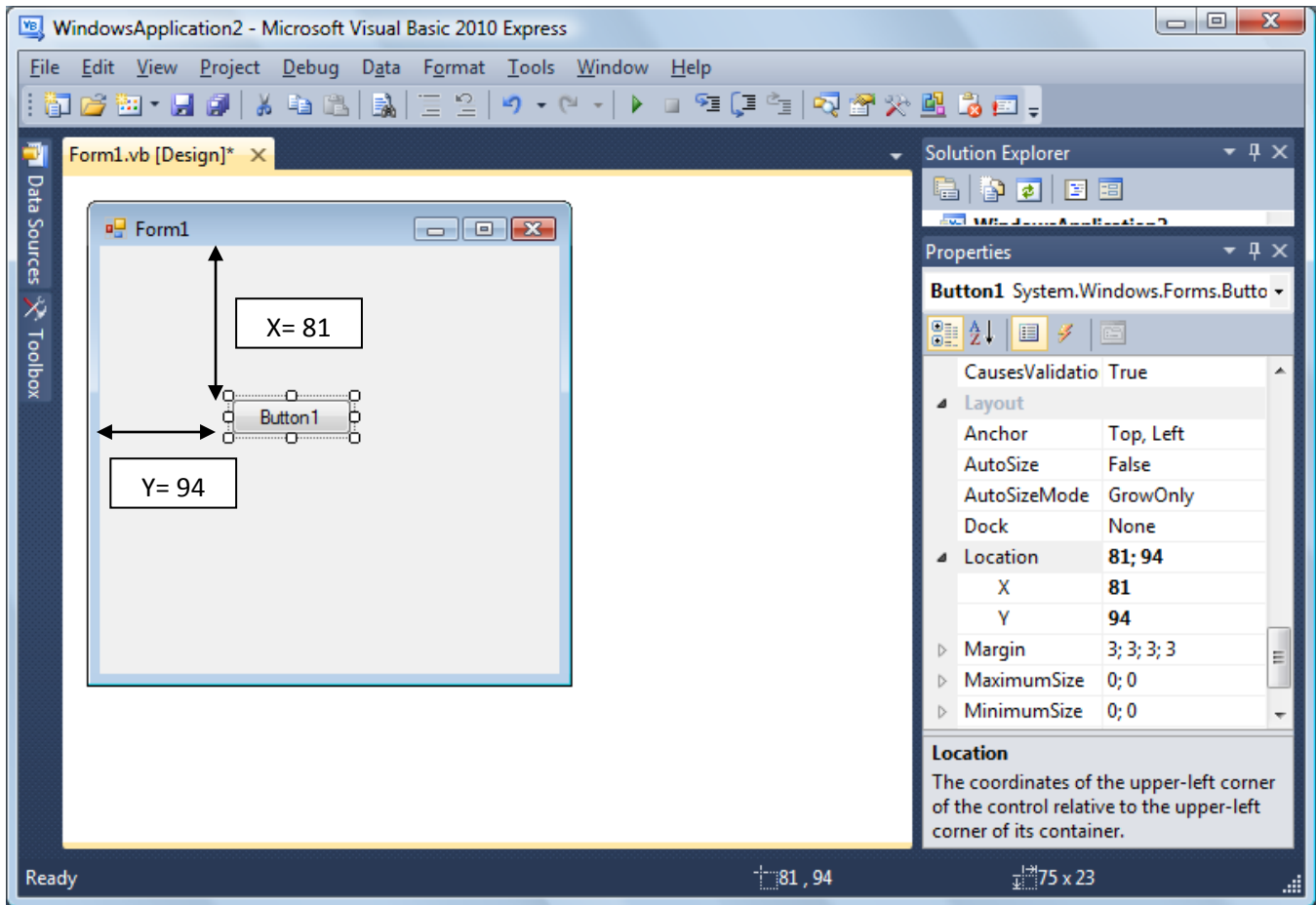
لقد قمنا في الدرس الأول بوضع زر Button و كتابة كود بسيط له، الآن سنتعامل مع بعض الخصائص و التي تعتبر عامة في كل العناصر التي سنستخدمها فيما بعد.



### 1- الموقع Location:

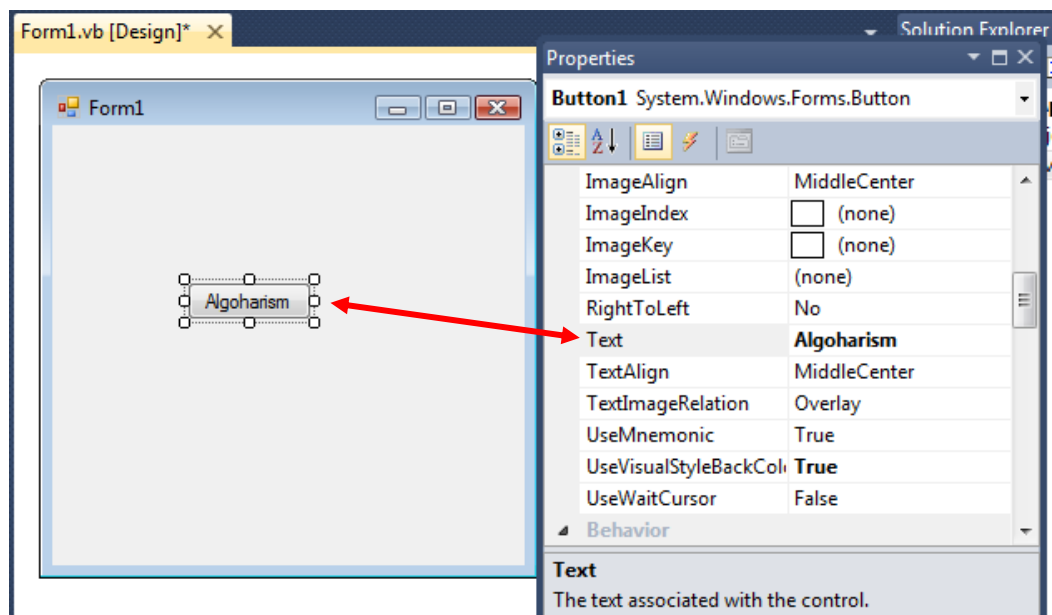


و هي الخاصية التي تتعامل مع موقع (مكان) العنصر المستخدم ، و تتكون من خاصيتين فرعيتين :  $X$  و  $Y$  ( يتعامل مع المسافة الرأسية و  $Y$  يتحكم في المسافة الأفقية من الجانب الأيسر) (انظر الشكل التالي).



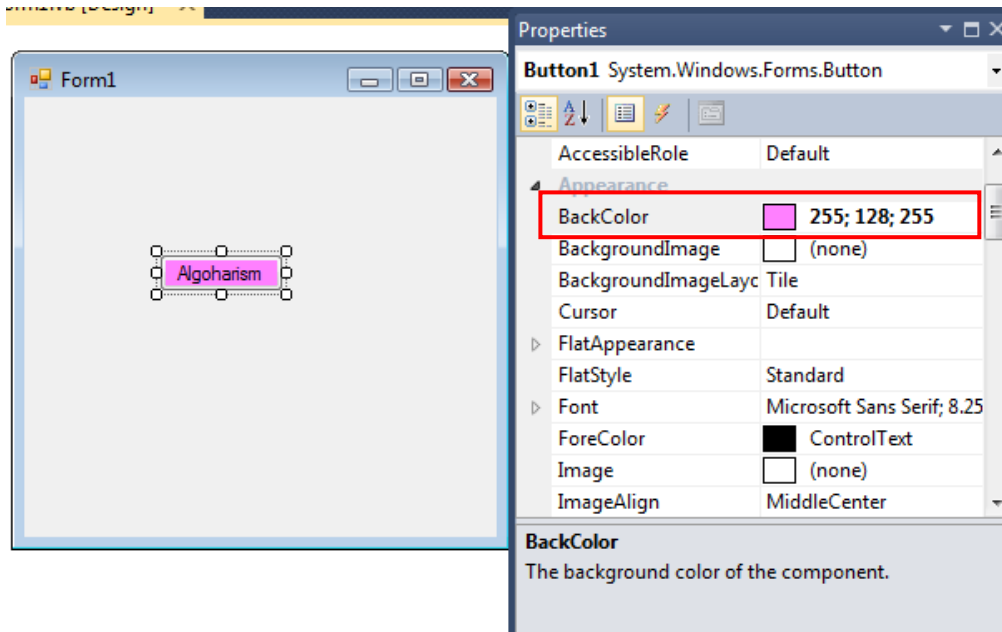
## 2- النص Text:

و هو خاص بتغيير النص المعرض على الزر.

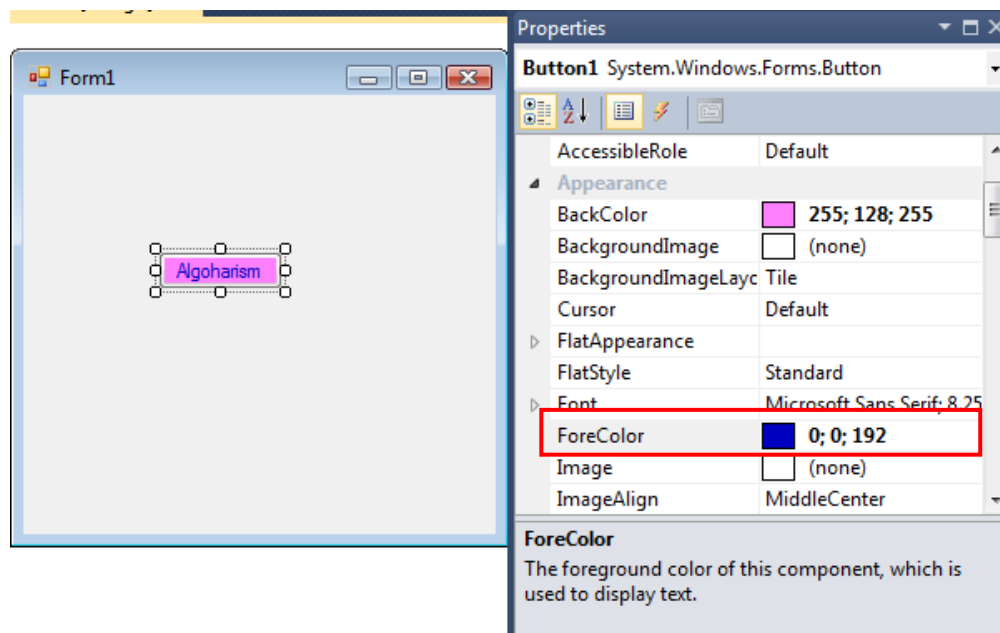




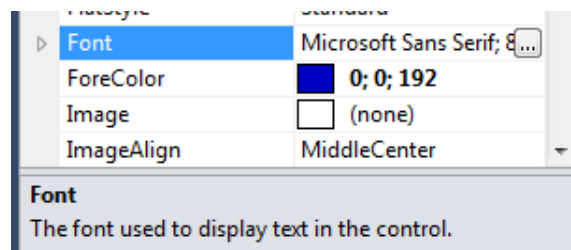
### 3 لون الخلفية :Backcolor

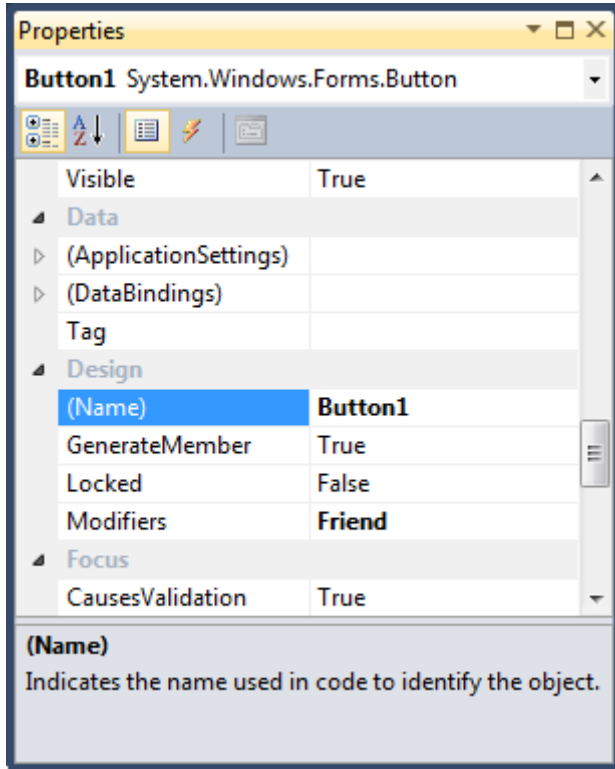


### 4 لون النص :Forecolor



### 5 الخط :Font





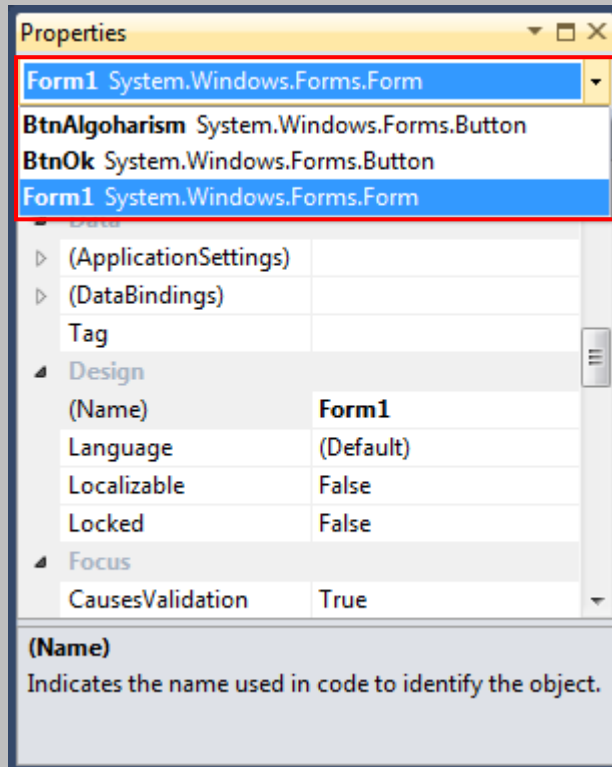
## 6- اسم العنصر (Name):

و هو عنوان العنصر فى لغة البرمجة حيث يجب أن يحصل كل عنصر على اسم حصرى.

- يمكنك ان تسمى العنصر كالاتى  
BtnOk  
حيث يعنى أن هذا العنصر هو الزر الخاص بـ OK و ذلك لتسهيل معرفة نوع العنصر و وظيفته.



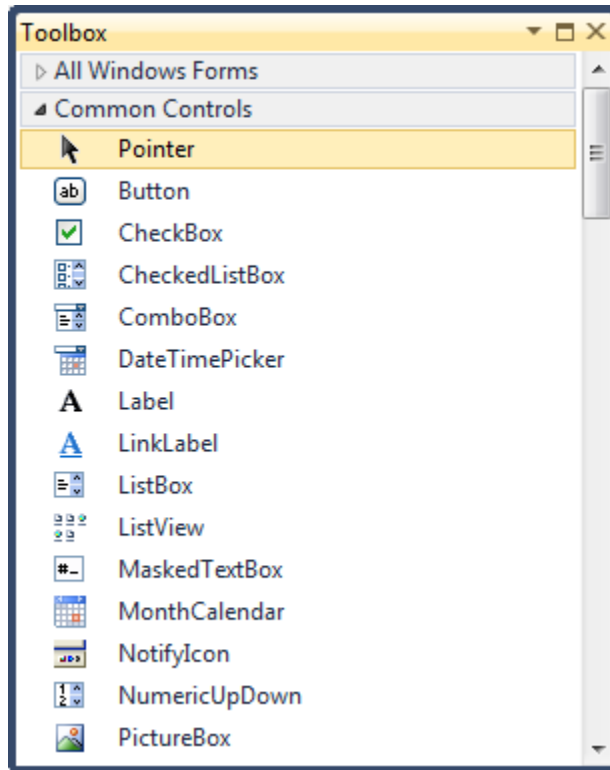
- لو كنت تملك أكثر من عنصر و تريد التنقل بينهم يمكنك استخدام قائمة العناصر من صندوق الخصائص.







## 2. أهم العناصر فى صندوق الأدوات (الأدوات الشائعة)

سنعرض أهم الأدوات و أكثرها شيوعا الموجودة و وظائفها من خلال واجهة برنامج خدمة عملاء .

أولا هذا هو صندوق الأدوات و فيه أكثر الأدوات شيوعا فى برامج النوافذ.



يستخدم لعمل زر أمر لتنفيذ وظيفة معينة يحددها المبرمج	 Button	زر Button
أسئلة نعم/لا و يستخدم لتحديد عدد من الوظائف إما كل على حدة و أما معا	 CheckBox	صندوق التعليم (الاختيار)
يستخدم لعرض نصوص على النماذج لا يستطيع المستخدم تعديلها أثناء تشغيل البرنامج	 Label	label
يستخدم لإدخال نصوص من لوحة و هو يمكن تعديل محتوياته بمعرفة المستخدم أثناء التشغيل	 TextBox	TextBox

يستخدم للاختيار بين عدة اختيارات و يصلح عندما يكون عدد الاختيارات محدودا	RadioButton	<b>Radio button</b>
و هو يستخدم للاختيار بين عدة اختيارات و يصلح عندما يكون عدد الاختيارات كبيرا	ListBox	<b>List Box</b>
و هو يجمع بين الـ ListBox و الـ Textbox و هو يستخدم للاختيار بين متعدد و يصلح عندما يكون عدد الاختيارات كبيرا	ComboBox	<b>Combo Box</b>

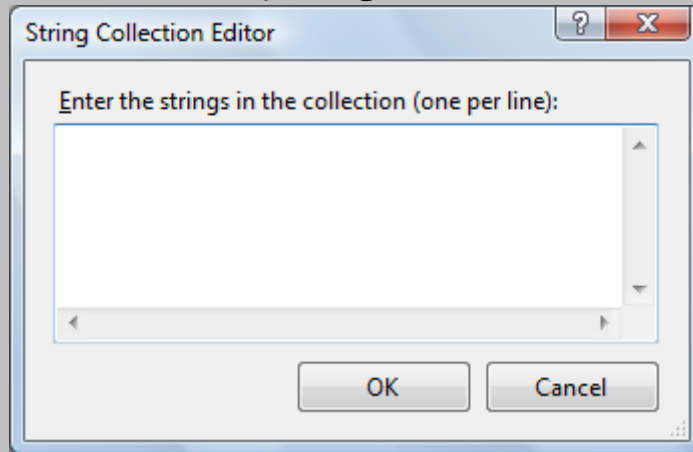
• مثال على هذه الخيارات: مثال مشروع "خدمة العملاء":  
1. سنضيف العناصر السابقة كالتالى:

2. اضافة معلومات إلى الـ Combo box.

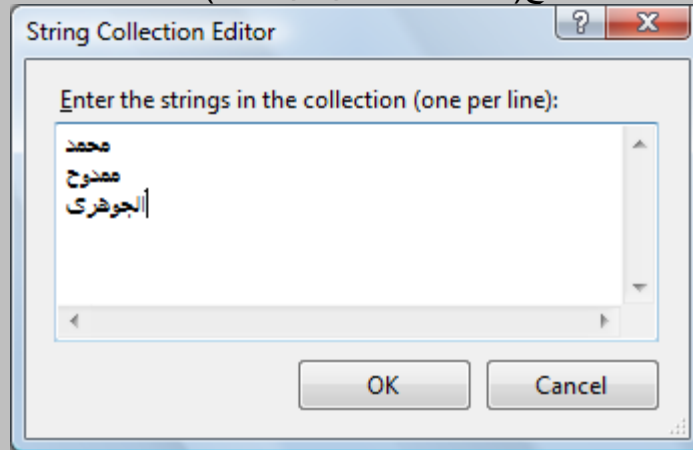
• اضغط على Combo box فتظهر هذه العلامة

• ستظهر نافذه فيها Edit items

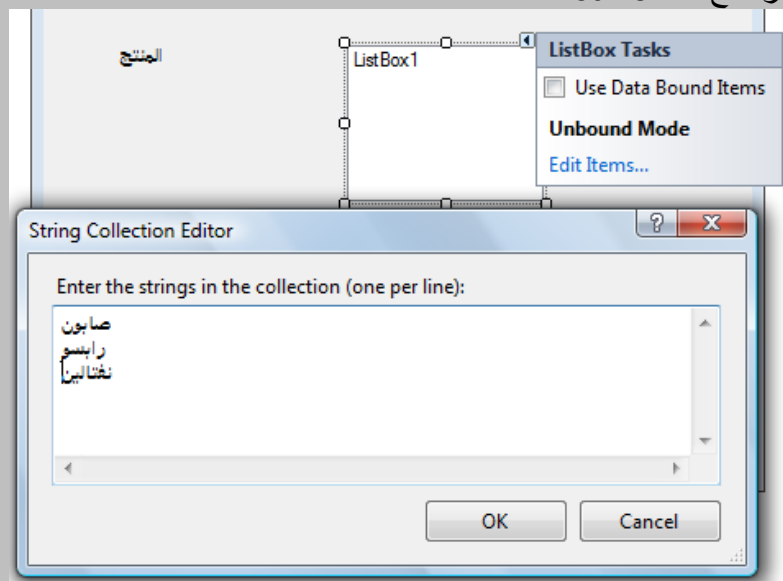
- ثم تظهر النافذة التالية لكتابة المعطيات في العنصر:



- سنكتب فيها كما هو موضح (اكتب كل معطى في سطر):



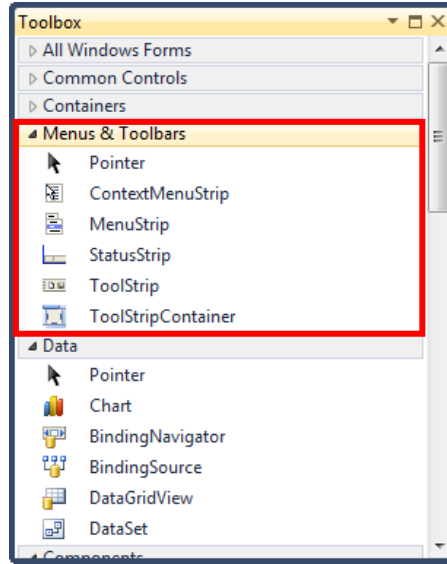
- سنعيد الكرة مع الـ List Box





## 3. أهم العناصر في صندوق الأدوات (أدوات القائمة و صندوق الأدوات)

سنعرض بعض أهم الأدوات و هي كيفية اضافة قائمة menu و صندوق أدوات Toolbox للمشروع الخاص بك.






و صلنا في مشروعنا السابق إلى هذا التصميم:

- راعى بين الفترة و الأخرى أن تحفظ المشروع حتى لا تضيع البيانات عند انقطاع الكهرباء  
بشئ مفاجئ  
اضغط فقط Ctrl+S



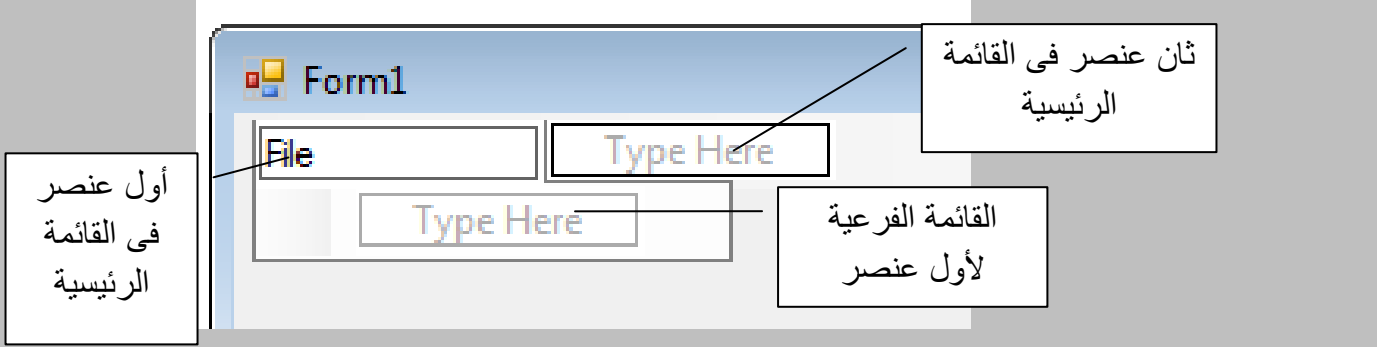
الآن مختصر لأهم الأدوات فى أدوات القائمة ثم سنعرض مثال على كل عنصر :

تعرض أوامر البرنامج فى مجموعات مرتبًا وظيفيًا	 MenuStrip	<b>Menu strip</b>
لعرض معلومات عن البرنامج و تقدم عمله	 StatusStrip	<b>Status Strip</b>
لإنشاء صندوق أدوات	 ToolStrip	<b>Tool Strip</b>

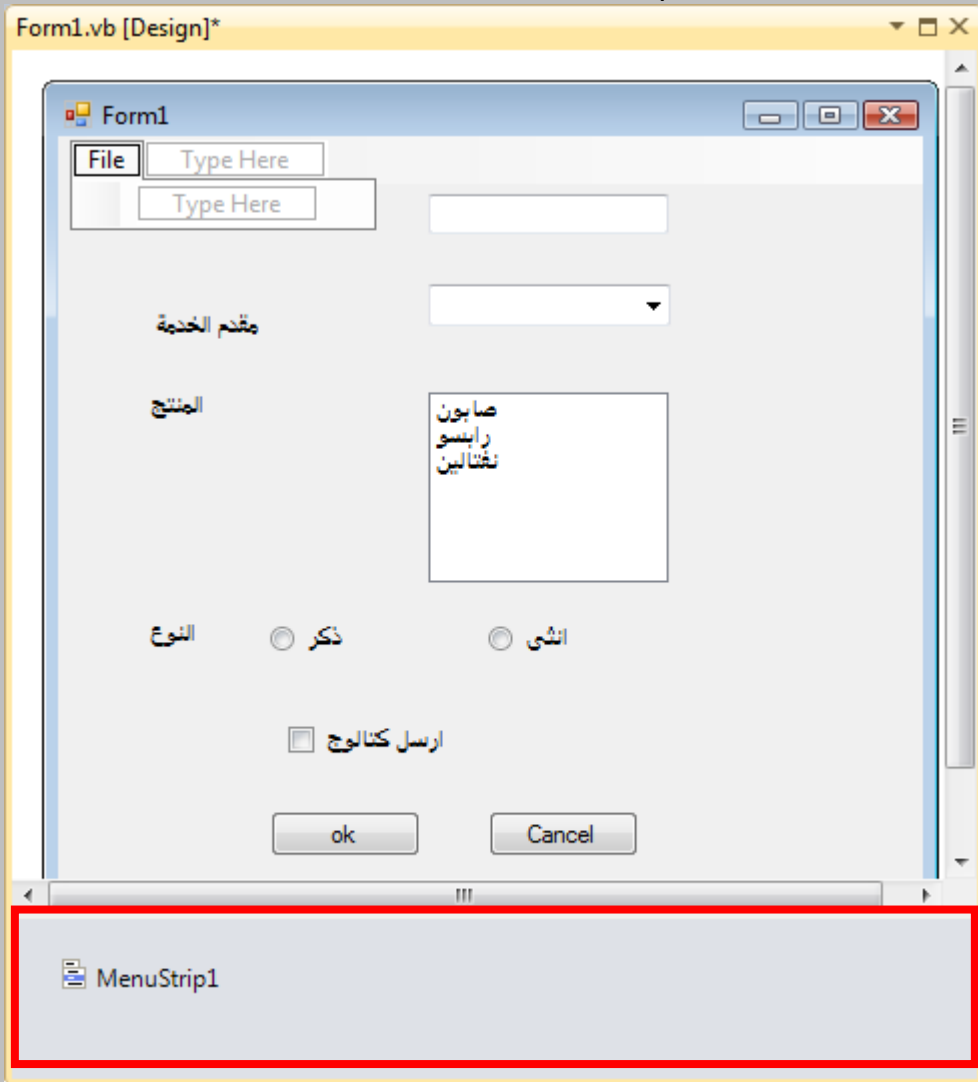
- مثال على هذه الخيارات: مثال مشروع "خدمة العملاء" قائمة Menu strip:  
1 - سنضيف Menu Strip من صندوق الأدوات



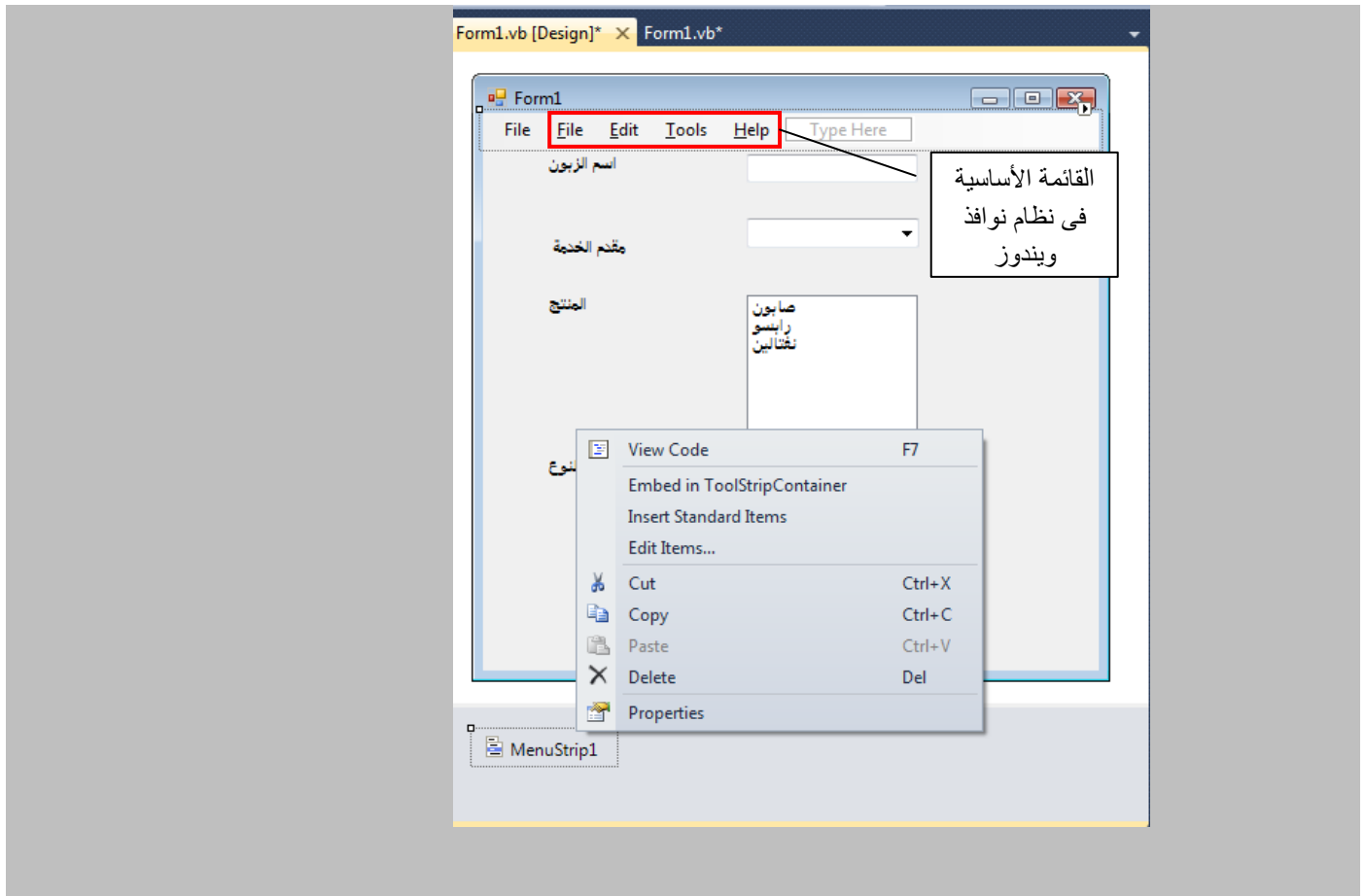
- 2 - يمكنك كتابة عناصر القائمة عن طريق الضغط على Type Here
- 3 - ستظهر 3 مربعات كالآتى:



4 - ستجد في أسفل الصفحة MenuStrip1

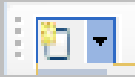
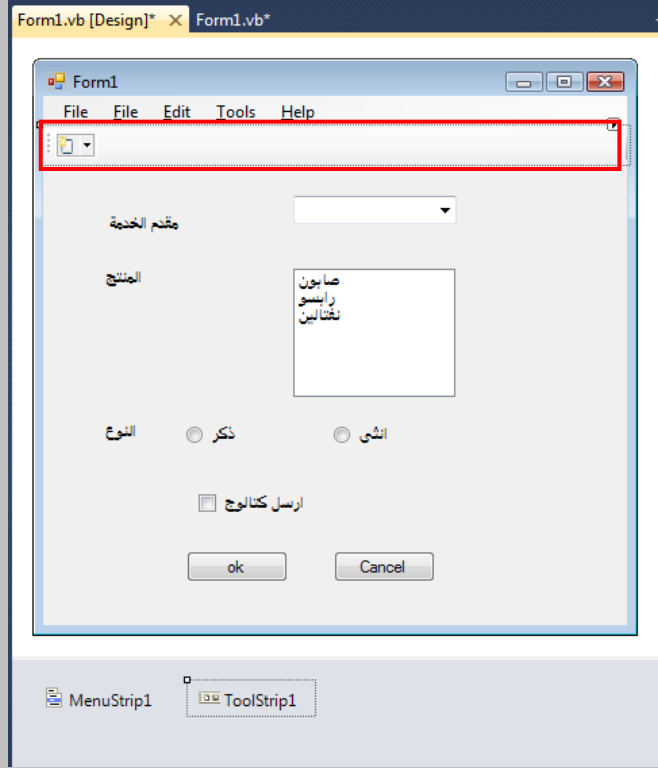


5 - اضغط عليها بالماوس بالـ Right Click وبعدين Insert Standard Items و ذلك و انشاء قائمة كما قوائم الويندوز الأساسية كما هو موضح بالشكل

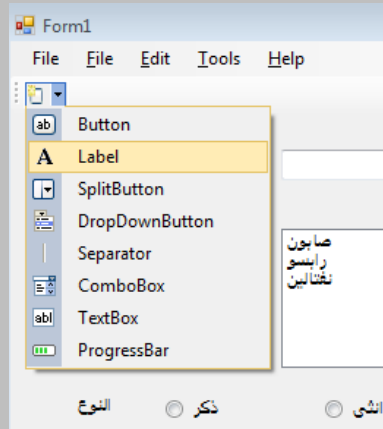


- مثال على هذه الخيارات: مثال مشروع "خدمة العملاء" قائمة tool Strip:

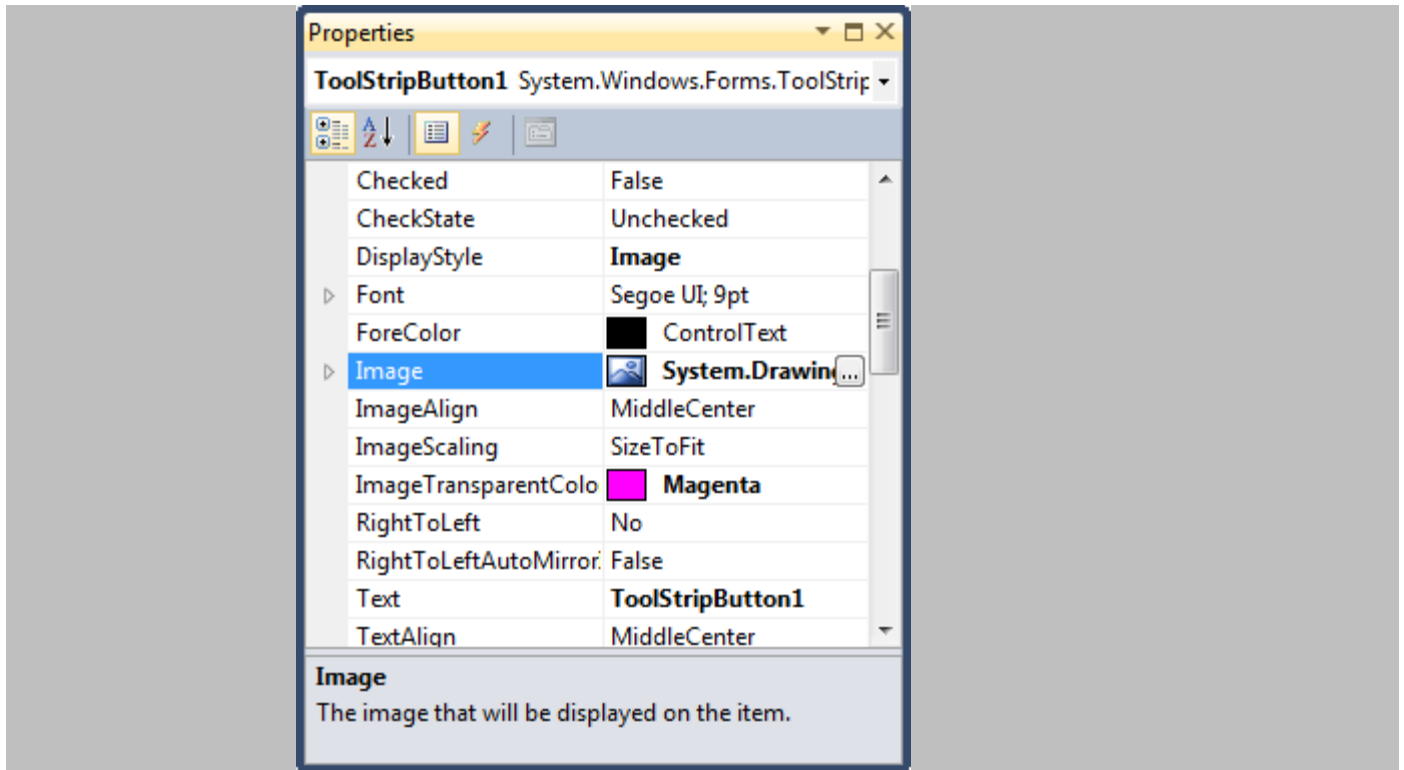
1 - سنضيف Tool Strip من صندوق الأدوات



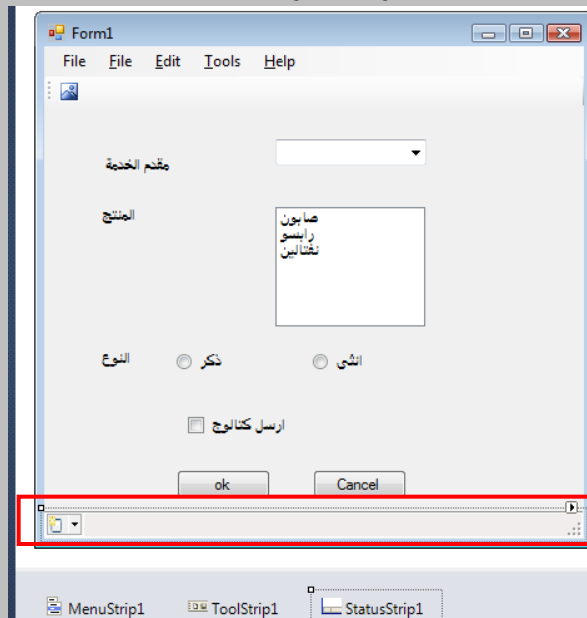
- 2 - إذا أردت إضافة عنصر للشرط الأدوات اضغط على السهم المشير لأسفل
- 3 - ستظهر قائمة كالآتى:



- 4 - إذا أردت إضافة صورة (رمز) اضغط على و غير اعدادات الصورة من صندوق الإعدادات:



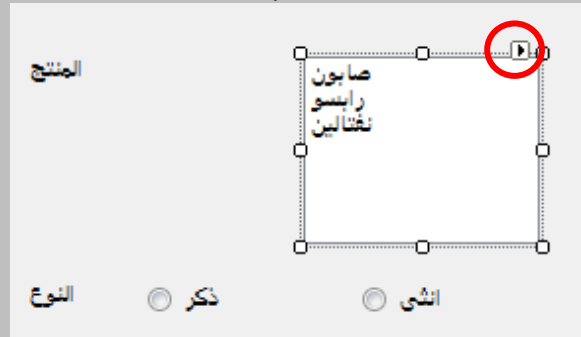
- مثال على هذه الخيارات: مثال مشروع "خدمة العملاء" قائمة Status Strip:  
1 - سنضيف Status Strip من صندوق الأدوات



- 2 - سيتم التعامل مع الـ Status Strip كما الآخرين.



- عند إضافة أى بيانات لأى عنصر ( مثل الـ List Box , Combo Box , Menu Strip , Status strip ) ابحث عن هذه العلامة

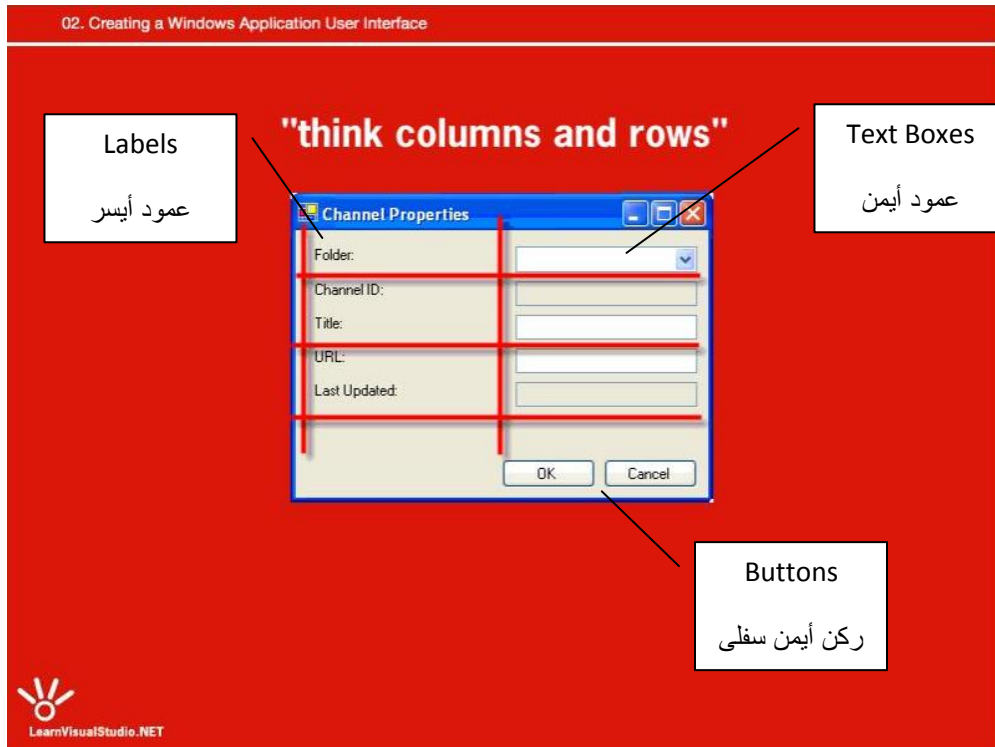


ثم اضغط عليها و اختر Edit Items و أضف كل ما تريد.

## 4. نصائح فى تصميم واجهة المشاريع

سنقدم هنا بعض النصائح البسيطة يجب مراعاتها فى تصميم واجهة المشروع ، سنقسم هذه النصائح إلى أفضل الطرق المتبعة فى التصميم و طرق تسهل الوصول إلى معلومات و تسهل قراءة واجهة البرنامج على المستخدم النهائى (المعنى باستخدام البرنامج).

- سنبدأ بأفضل الطرق لتصميم واجهة البرنامج:  
1 فكر بطريقة "الأعمدة و الصفوف":



أى من الأفضل أن تكون Labels فى عمود واحد على اليسار و العناصر التى تحتوى على إدخال بيانات فى عمود على اليمين و الأزرار Buttons التى ستقوم بفعل أو وظيفة تكون فى الركن اليمين السفلى. (كما هو موضح بالشكل).

- هذه المواصفات و النصائح للبرامج المكتوبة باللغة الانجليزية لكن إذا كانت واجهة البرنامج مكتوبة بالعربية راعى تغيير اليسار إلى اليمين و العكس فى الاتجاهات.







- 2- عند كتابة البيانات فى واجهة البرنامج راعى سهولة و وضوح هذه البيانات (مثال إذا أردت أن تجعل button لتأكيد الاختيار فلا تجعل تسميته "اذهب" فلتجعله بسيطاً كـ "موافق") ، لا تجعل المستخدم النهائى يختار فى معانى و وظائف العناصر المعروضة.
- 3- استخدم الألوان و الأساليب المتعارف عليها فى تصميم البرامج و فى إدخال البيانات حتى لا يختار المستخدم النهائى و لا تصعب عليه الموقف (مثال لا تختار مثلاً للزر button لون أسود).

- انتهينا من بعض النصائح فى تصميم، و سنعرض فى التالى نصائح تسهل للمستخدم النهائى الوصول إلى المعلومة و ادخال البيانات:  
1- Tab key:

هل جربت استخدام زر Tab من لوحة المفاتيح للتنقل بين الـ Text Boxes فى البرامج دون استخدام الماوس، لاحظ أنه يتم الانتقال بينهم بترتيب معين.

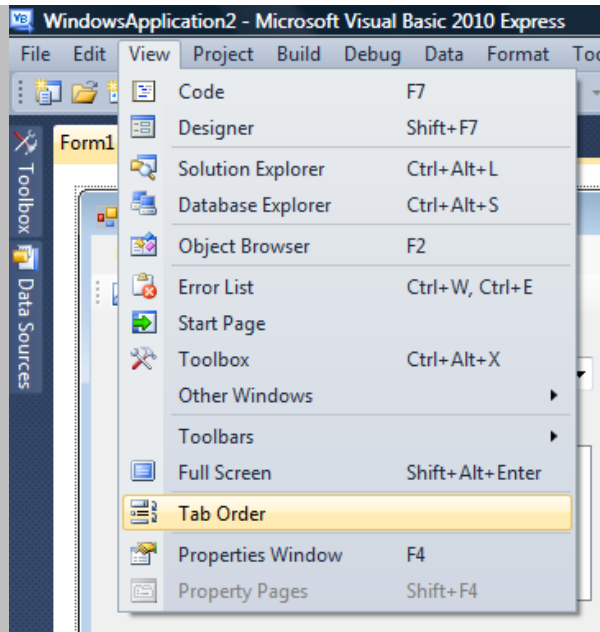
- مثال على استخدام Tab فى الانتقال بين عناصر الإدخال المختلفة:

- 1- اضغط على زر Debug (ستجده فى شريط الأدوات فى الأعلى) أو اضغط على F5 فى لوحة المفاتيح.
- 2- سيعمل البرنامج كالتالى:



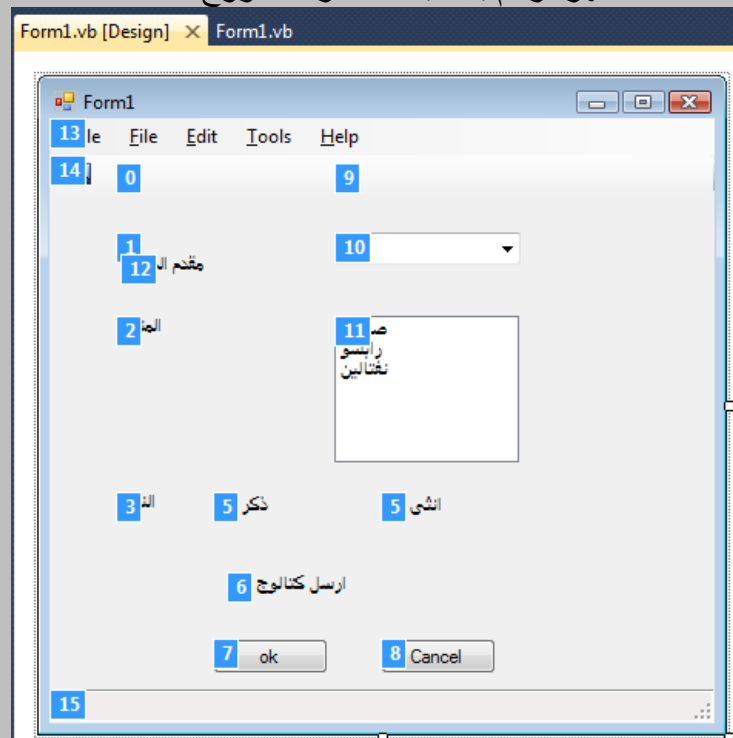
3 - إذا ضغطت على Tab فتتلقى مؤشر الإدخال (الذى يضيء و يطفى عند الكتابة و شكله | ) سينتقل بين صناديق الإدخال بترتيب عشوائى:

4 - لتعديل هذا الترتيب سنذهب إلى قائمة view ثم Tab order

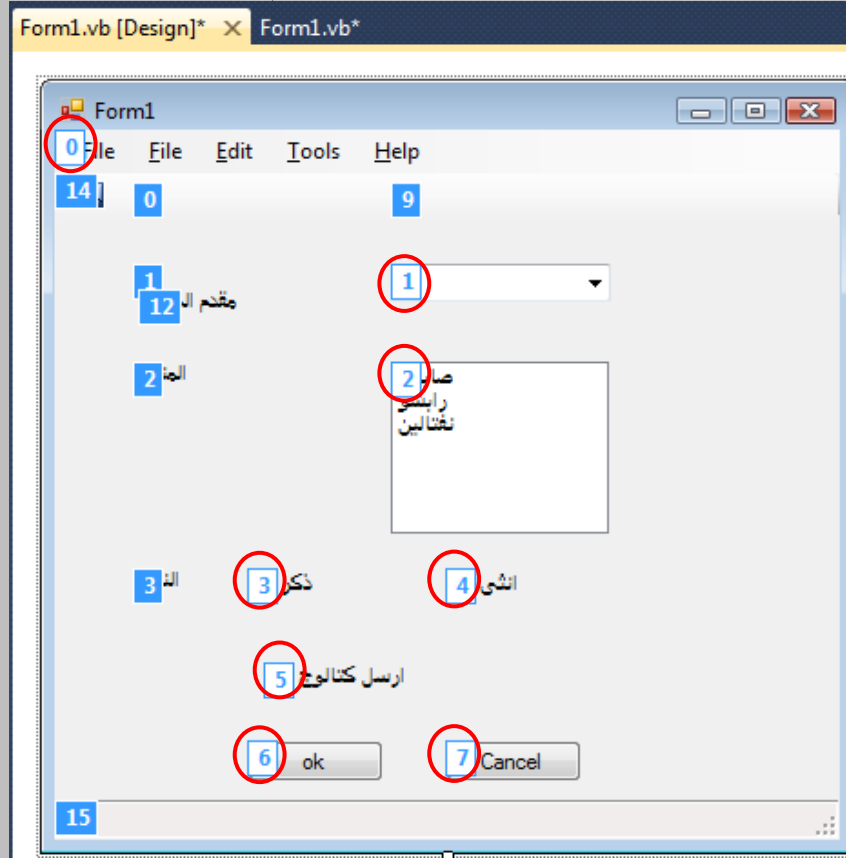


لو لم تجد Tab order افتح قائمة tools ثم Settings ثم expert settings ثم اذهب إلى قائمة view فـ tab order

### 5 - ستظهر أرقام بجانب عناصر المشروع



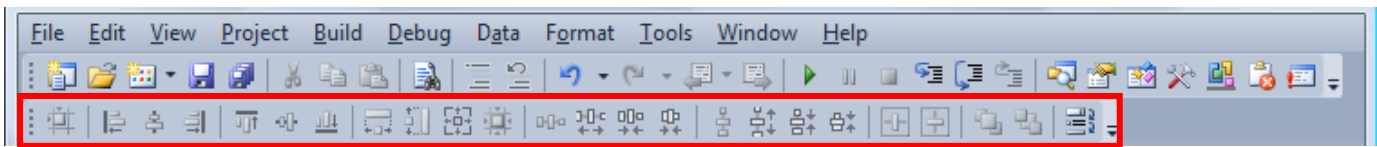
6 - لتغيير ترتيب العناصر اضغط على الأرقام بالترتيب الذى تريده للانتقال بين العناصر.



## 2 -LayoutToolbar:

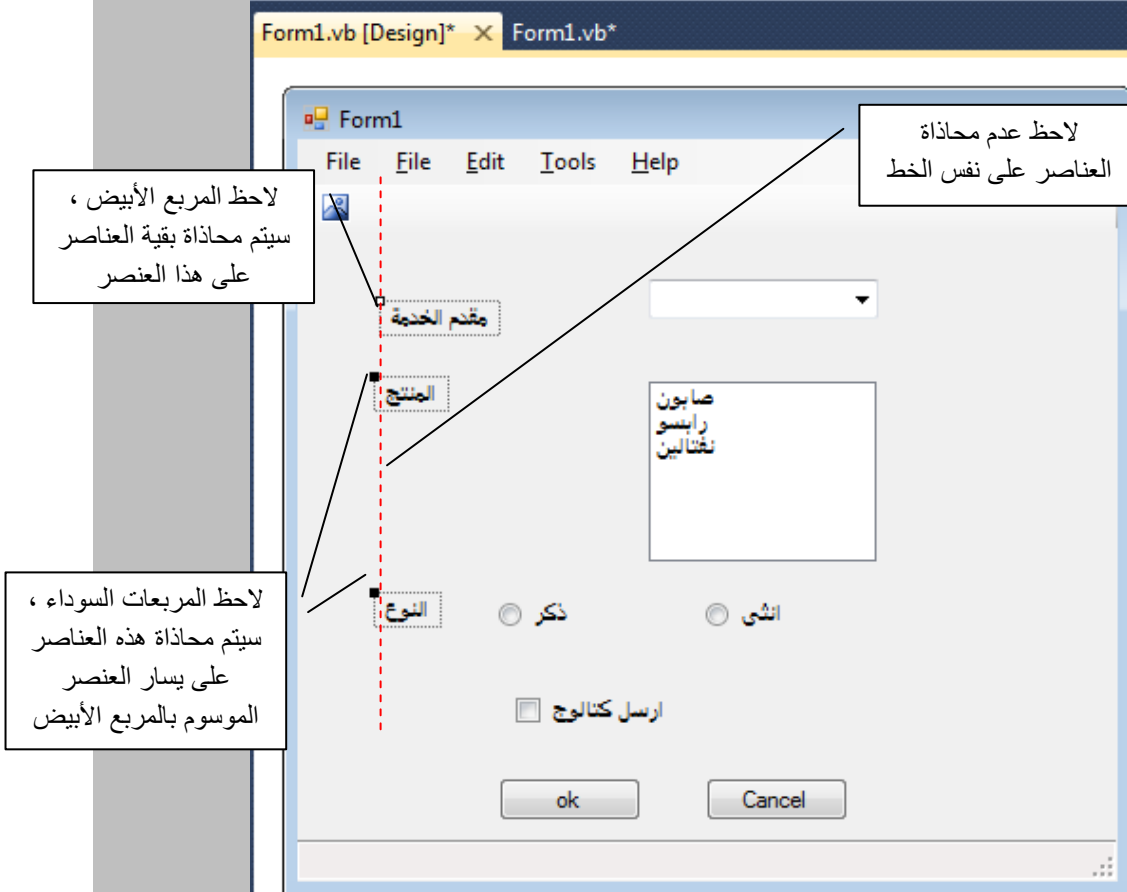
و هو شريط أدوات يتيح لك تنظيم عناصر العرض و يسهل عملية الترتيب المرئى و محاذاتها.

لإظهار شريط أدوات layout افتح قائمة view - toolbars ف layout



## • كيفية استخدام Layout Toolbar فى تنظيم العناصر المعروضة:

1 - Aligns left : لمحاذاة العناصر على اليسار  
- اختر أكثر من عنصر:



يمكنك اختيار أكثر من عنصر بالضغط على Ctrl و اختر بالمواس على العناصر المرادة.

- اضغط على محاذاة على اليسار ، سيتم محاذاة العناصر المختارة على يسار العنصر الموسوم بالمربع الأبيض



Form1.vb [Design]\* x Form1.vb\*

Form1

File File Edit Tools Help

مقدم الخدمة

المنتج

صابون  
راينسو  
نفتالين

النوع  ذكر  انثى

ارسل كتالوج

ok Cancel

لاحظ محاذاة جميع العناصر على يسار العنصر الموسوم بالمربع بالأبيض

Aligns right - 2 : للمحاذاة على اليسار.

Aligns top - 3 : للمحاذاة بالأعلى.



## 5. تدريب

أنشأ واجهة برنامج لتأجير سيارات ، حيث يتيح اختيار البرنامج للموظف ادخال اسم المستأجر و عنوانه و طريقة الدفع ( credit card ، كاش ، شيك ) ، و اجعل الترتيب الانتقال بين الخانات ترتيبيا منطقيا (Tab order).

---

# Microsoft VB Express للمبتدئين 2010

---

الدرس الثالث

---

[algharism.blogspot.com](http://algharism.blogspot.com)

---

الدرس الثالث ستعرف فيه على مقدمة لكيفية كتابة الكود  
و ما هي الـ Events و أهم خصائص الـ Events

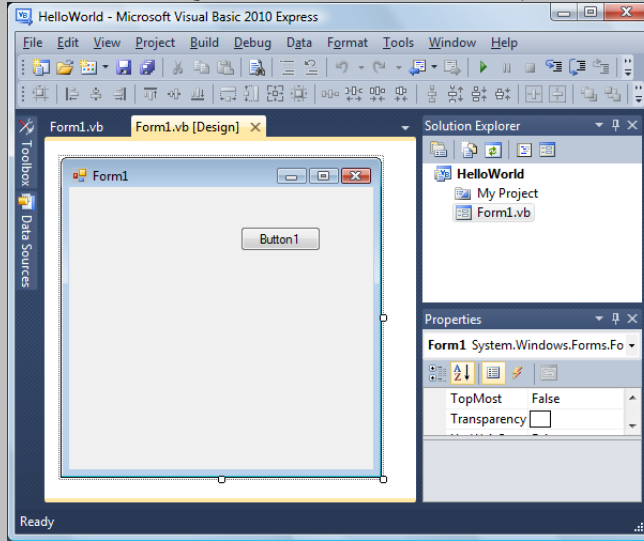


# 1. مقدمة عن الـ Events

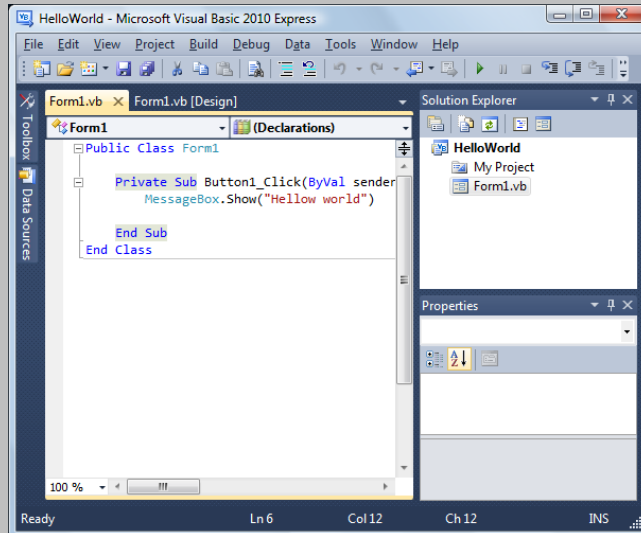
سنتعرف أكثر في هذا الدرس عن أول خطوة في كتابة الشفرة البرمجية ( الكود ) الخاص بالبرنامج ، حيث يصمم Visual Basic Express برامج تعرف بـ " برامج تعتمد على حدث " Event Driven "Programs".

يكون هذا الحدث Event مثلا ضغطة زر أو تغيير من برنامج آخر أو من الويندوز نفسه ، و يكون البرنامج مصمم إما أن يستجيب أو يتجاهل هذا الحدث.

- مثال: مشروع "مرحبا أيها العالم Hello World" المشروح في الدرس الأول:



- 1 - ضغطة مزدوجة Double Click على Button 1 لتظهر نافذة التكويد:



- 2 - سنجد الكود الآتي: و من خلاله سنعرف على بعض الأساسيات في التكويد



Button 1: أى أن العنصر Button 1 هو الذى سيتم التعامل معه  
Click : هو الحدث (الضغط على الزر) الذى سيتم تنفيذ الرد

من Private Sub إلى End Sub  
يسمى Code Block أو جزء كودى  
و هو البناء المتخصص فى بناء الكود

هم الاجراء الذى سيتم تنفيذه كرد على Click (الضغط)  
على Button 1

من Public Class إلى End Class  
يسمى Method و هو  
الوحدة الأساسية فى  
بناء الكود

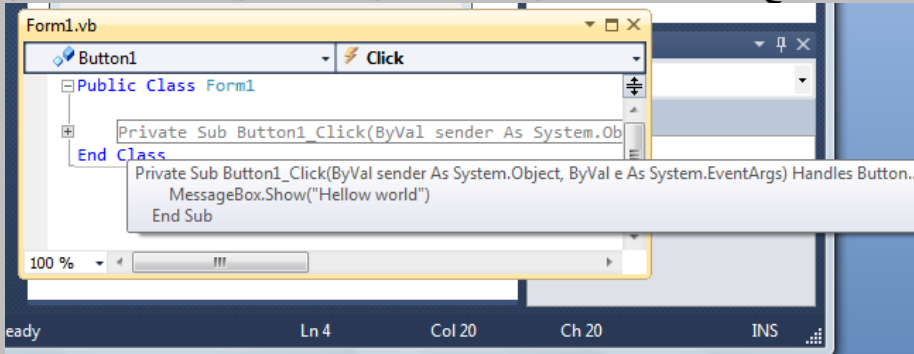
لتوضيح العملية أكثر: هي أن يوجد وحدة بنائية أساسية في تصميم الكود و هي Method و يبدأ من Public class و ينتهي عند End Class  
وهو يحتوي على وحدات أكثر تخصص و تسمى بـ Code Block و هي الخاصة بعنصر أو كائن محدد و تحتوي هي بدورها على أصغر وحدة بنائية للكود و هي في مثالنا الحالي Button1\_Click و تعنى أنني أريد عند الضغط بالماوس ( و هو الحدث Click ) على Button 1 سيتم إظهار رسالة MessageBox تقول "Hello World" ( و هو الرد Response على الحدث Click )

- يمكنك التحول من نافذة التصميم إلى نافذة الكود عن طريق الضغط على F7 و العودة من نافذة الكود إلى نافذة التصميم عن طريق Shift+F7



- يمكنك تقليص عرض الكود عن طريق الضغط على علامة (-) بجانب الكود الذى تريد إخفائه.

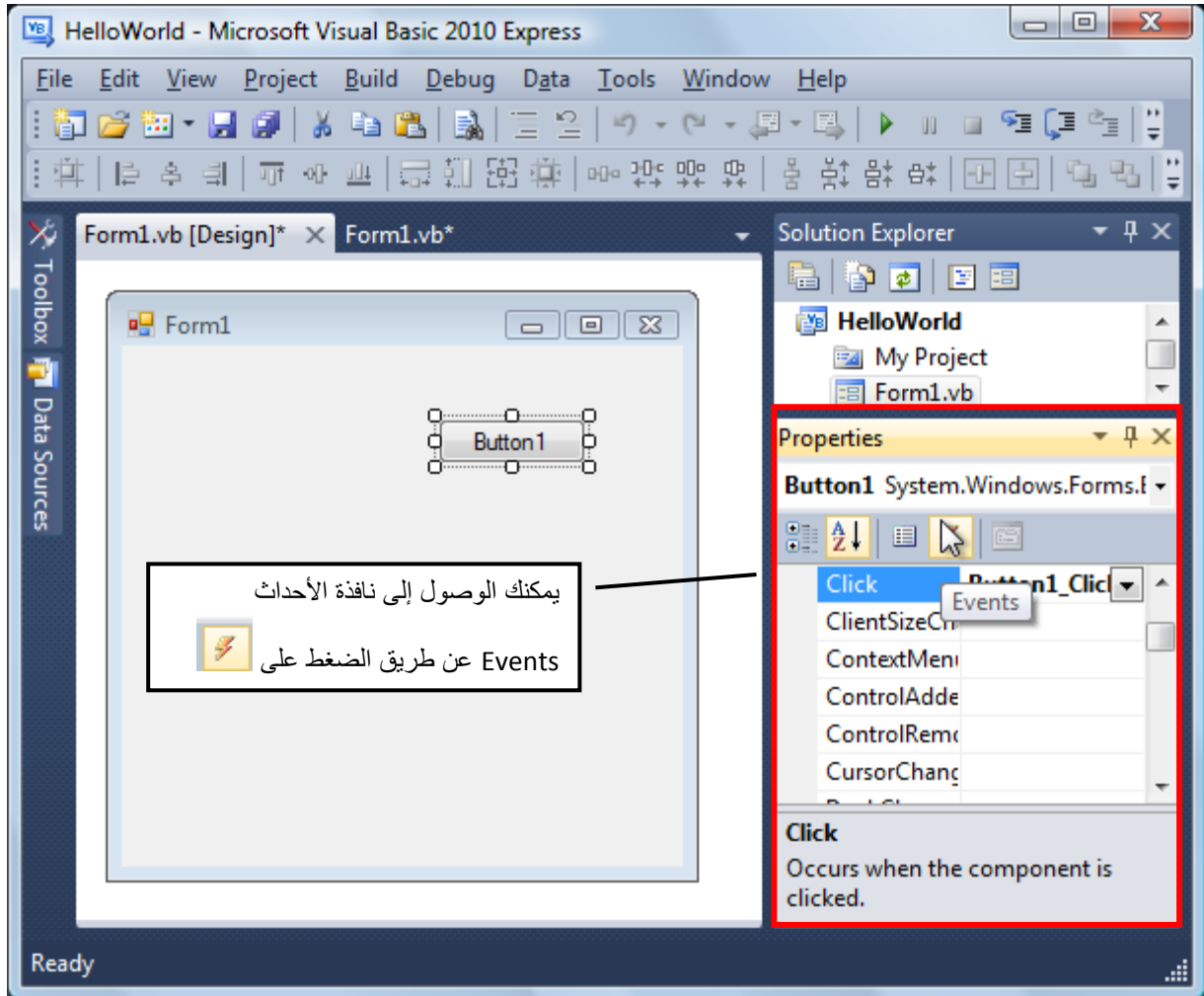
- بوضع مؤشر الماوس على الكود المخفى سيظهر في نافذة فرعية كالاتي:



## 2. كيف أكتب الكود؟

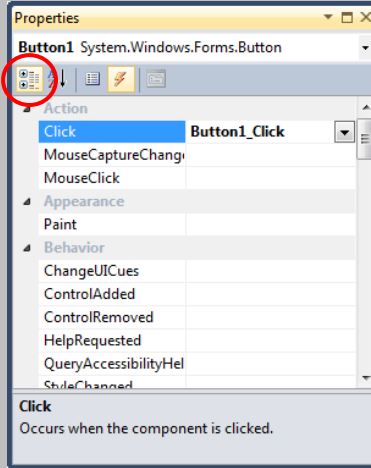
بعد أن وضعنا أين كتابة الكود و بعض أساسيات نافذة التكويد ، سنوضح كيفية كتابة كود برمجى و ما هى الأوامر التى تستخدم فى كتابته.

يجب أن تتذكر انه ليس عليك أن تحفظ المئات من الأوامر البرمجية للوصول إلى الكود المراد إنما يمكنك أن تصل و تكتب هذه الأوامر عن طريق طرق كثيرة منها نافذة الخصائص **properties**



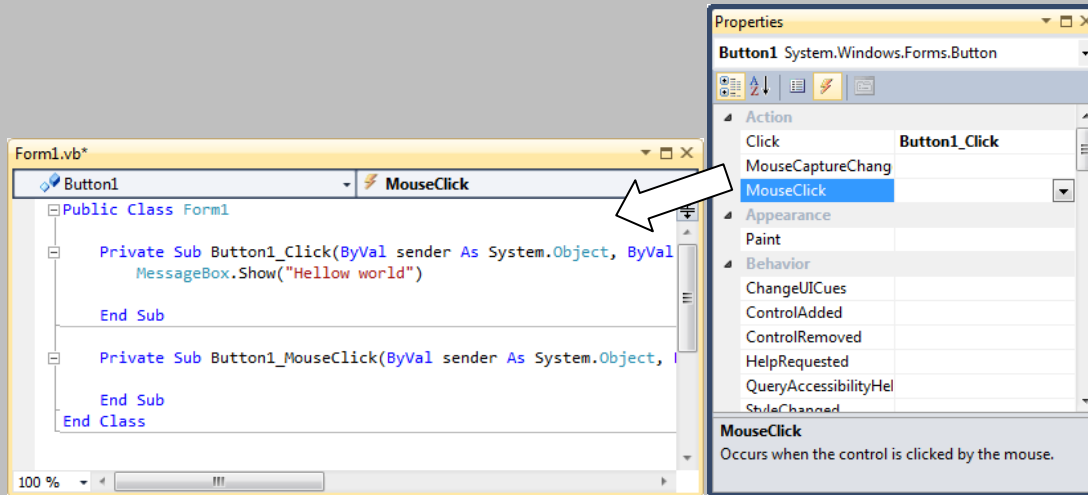
و تحتوى نافذة Events فى نافذة Properties على جميع الأحداث التى يمكن للعنصر الاستجابة له للقيام برد برمجى معين.

- لسهولة الوصول إلى الحدث Event المعين يمكنك الضغط على



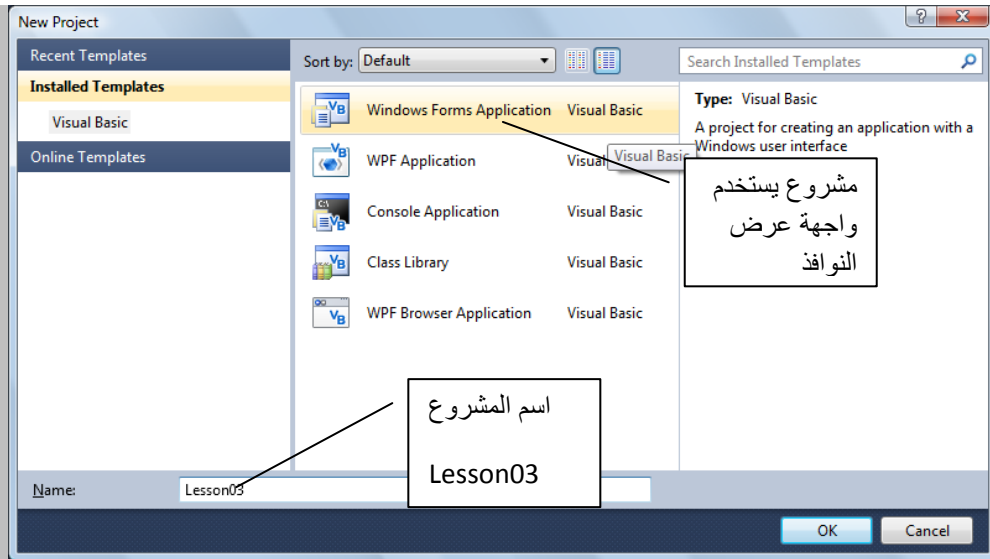
حيث يرتب الـ Events فى مجموعات تسهل لك الوصول إلى الحدث المرغوب

- عند الوصول إلى الحدث المرغوب يمكنك الضغط المزدوج Double Click للوصول إلى النافذة البرمجية لهذا الحدث

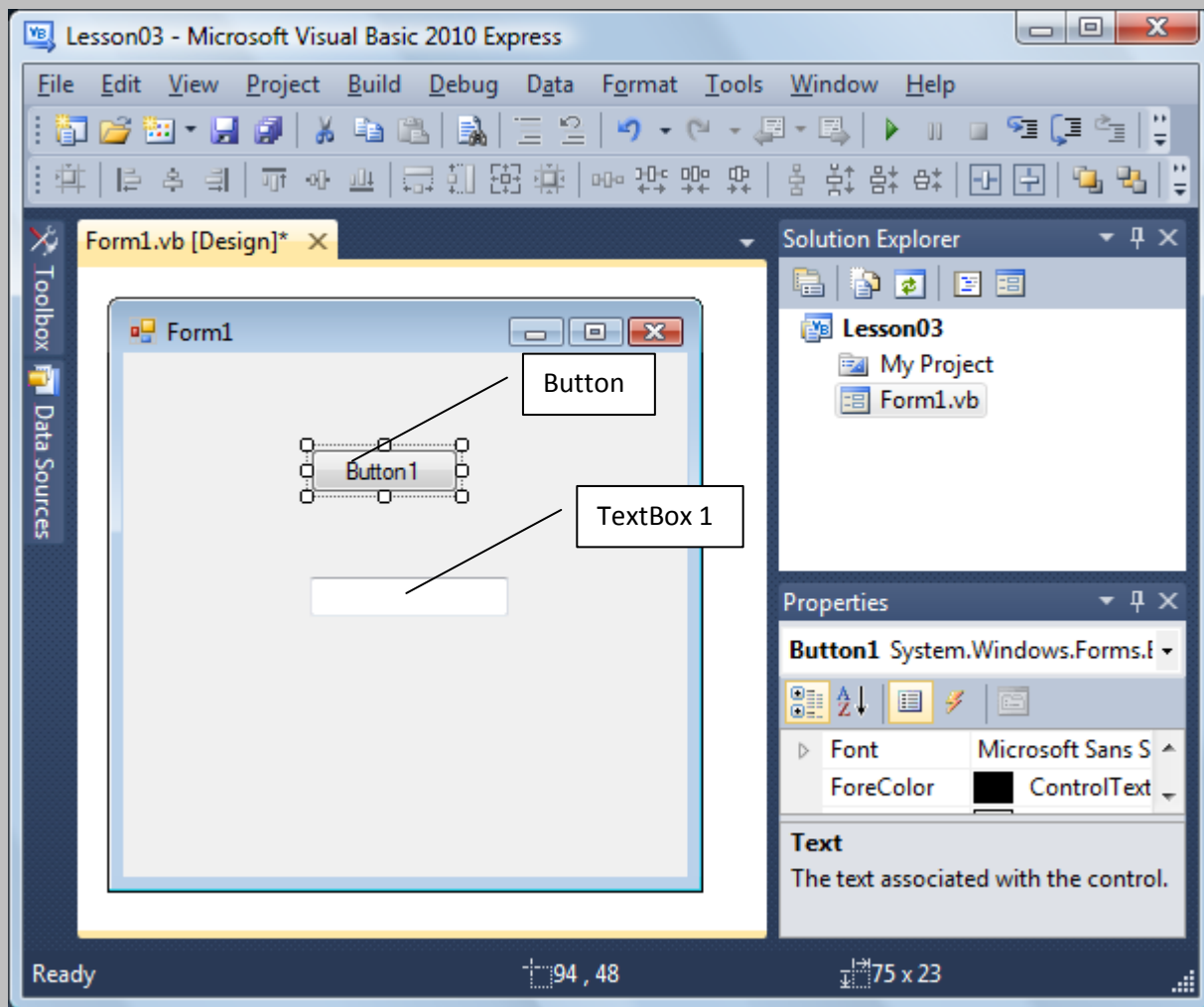


- مثال كيفية كتابة كود لحدث و له رد برمجى ( سيكون الهدف من البرنامج إنه عند الضغط على Button 1 سيكتب داخل TextBox1 "أهلا بأياها العالم!"):  
1 - أنشأ مشروع جديد:

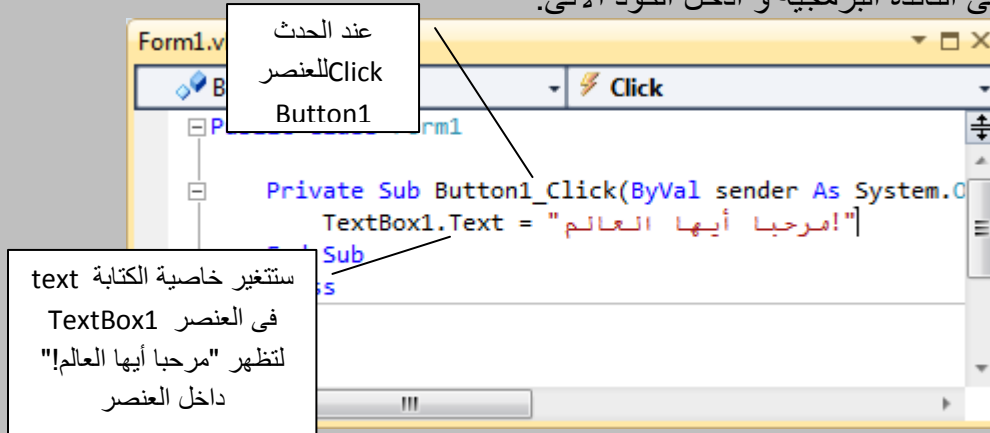




2 - تصميم الواجهة التالية:



3 - اذهب إلى النافذة البرمجية و أدخل الكود الآتي:

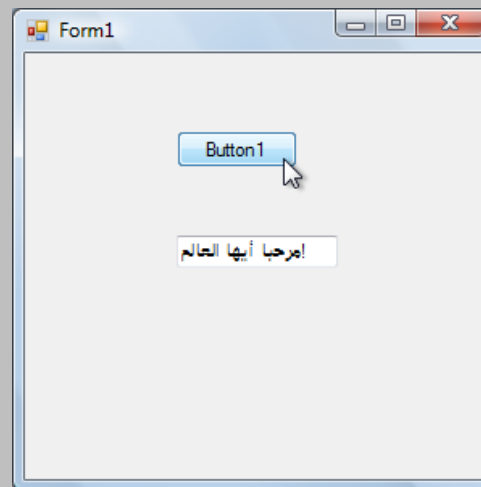


- و خطواته أنه عند الحدث Click للعنصر Button1 ستتغير خاصية الكتابة text في العنصر TextBox1

4 - عند الضغط على زر Debug

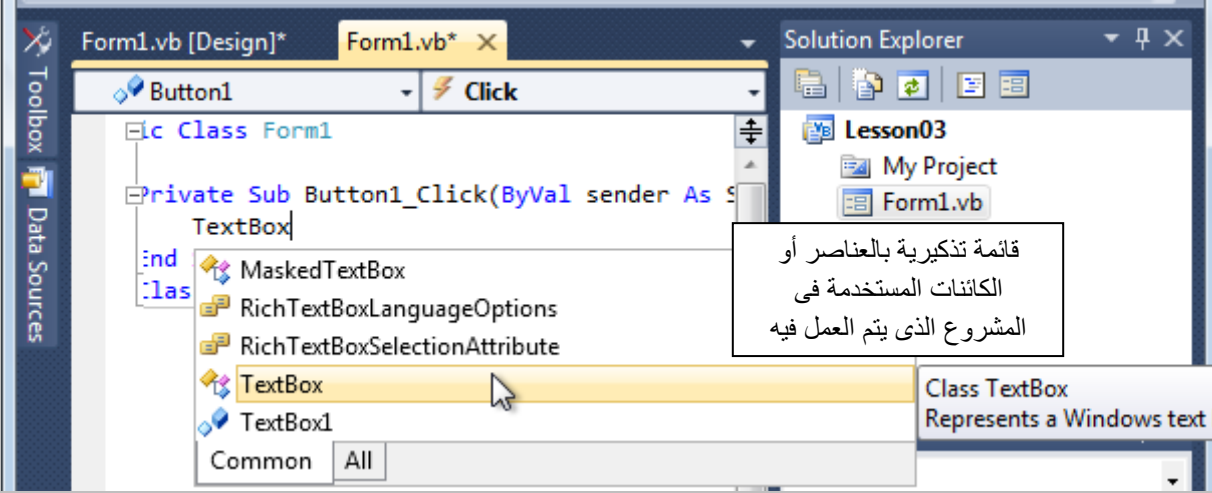


سيعمل البرنامج كالتالي:



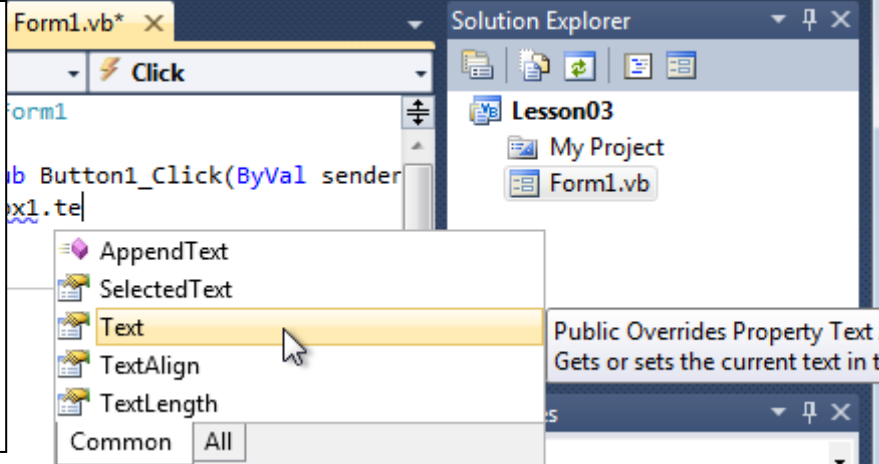


- ستلاحظ عند كتابة الكود السابق ستظهر قائمة كالتالى:



قائمة تذكيرية بالخصائص التى يتم التحكم فيها للعنصر المحدد و تظهر عند كتابة النقطة "■" و يمكنك الاختيار أيا من هذه الخصائص عن طريق الاختيار بالماوس أو الاختيار بالاسهم او حتى الاستمرار فى الكتابة و اختيار الكلمة بالمسافة

سلاحظ عند الوقوف على الخاصية المرادة سيظهر مربع يشرح لك أكثر عن الخاصية



حيث تمثل الصورة الأولى اقتراحات(تذكيرية) للعناصر المستخدمة فى المشروع و الصورة الثانية مرجع للخصائص التى يمكن تغييرها كرد للحدث Event .



## 3. أدوات تساعدك في كتابة الكود

يوجد بعض الأدوات التي ستساعدك في عملية كتابة الكود و مراجعته.

### 1- كتابة تعليق أو ملاحظة على الكود:

يمكنك من خلال كتابة ملاحظات (ليس لها دلالة برمجية) داخل الكدة حتى تتمكنك من معرفة وظيفة الكود أو تخبرك عن أهمية جزء معين من الكود.

يمكنك كتابة ملاحظة عن طريق " ' " و ستظهر باللون الأخضر كالأتي:

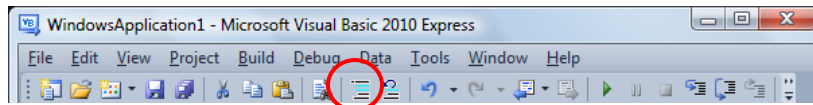
```

Form1.vb*
Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        TextBox1.Text = "مرحبا أيها العالم"
        ' هنا يمكنني كتابة تعليق
        ' او ملاحظة على جزء معين
        ' من الكود و ليس له أي وظيفة
        ' عند تنفيذ البرنامج
    End Sub
End Class

```

يمكنك أن تجعل جزء من الكود في ملاحظة أو تعليق حتى يمكنك تذكره فيما بعد ، و لكن تذكر أنه لن يعمل في تنفيذ البرنامج سيتحول إلى مجرد تعليق، و هذا سيساعدك إذا أردت كتابة كود لا تحتاجه في الوقت الحالي و لكن لا تريد حذفه مجرد تذكره.

ستعلم على الجزء المطلوب من الكود و ثم ابحث عن الزر التالي في شريط الأدوات في الأعلى:



و سيظهر الكود كالأتي:

```

Form1.vb*
Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        TextBox1.Text = "مرحبا أيها العالم"
        ' هنا يمكنني كتابة تعليق
        ' او ملاحظة على جزء معين
        ' من الكود و ليس له أي وظيفة
        ' عند تنفيذ البرنامج
    End Sub
End Class

```





يمكنك ادخال موقع داخل التعليق حيث ينفع فى ارفاق الكود مع موقع:

```

Form1.vb*
Button1 Click
Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        TextBox1.Text = "!مرحبا أيها العالم"
        ' هنا يمكننى كتابة تعليق
        ' او ملاحظة على جزء معين
        ' من الكود وليس له أى وظيفة
        ' عند تنفيذ البرنامج
        ' http://algharism.blogspot.com
    End Sub
End Class

```

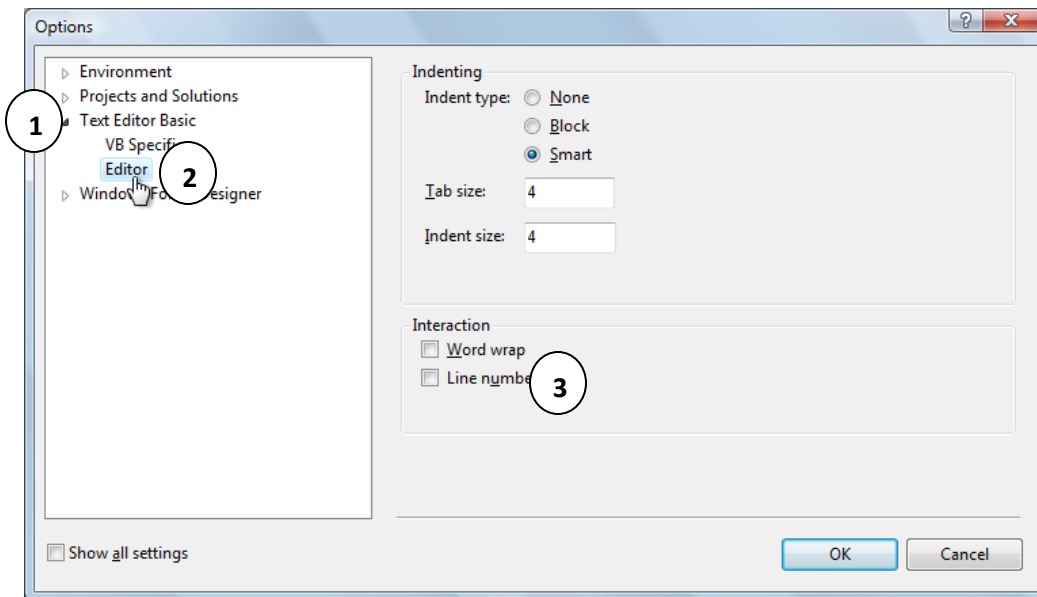
فقط عليك كتابة (داخل التعليق): http:// ثم اسم الموقع.

2 - اظهار أرقام السطو:

يتيح لك إظهار أرقام السطور طريقة سهلة للوصول إلى سطر معين داخل الكود.

يمكنك إظهار الأرقام كالتالى:

Tools >> Options :



ستظهر الأرقام بجانب السطور كالتالى:



```
Form1.vb*
Button1 Click
1 Public Class Form1
2
3 Private Sub Button1_Click(ByVal sender As System.Object,
4     TextBox1.Text = "مرحبا أيها العالم!"
5     ' هنا يمكننى كتابة تعليق
6     ' او ملاحظة على جزء معين
7     ' من الكود و ليس له أى وظيفة
8     ' عند تنفيذ البرنامج
9     ' http://algharism.blogspot.com
10 End Sub
11 End Class
```



## 4. تدريب

1- استكشف الأحداث events للعنصر button عن طريق ادخال Button1 داخل المشروع ، و اكتب حسب الحدث الذى تم اختياره داخل كود البرمجة:

MessageBox.show("Name of Event")

و ذلك للأحداث التالية:

- Click
- Enter
- Leave
- MouseHover

لو لم تفهم وظيفة الـ Event يمكنك اختيار الـ Event ثم اضغط F1 فى لوحة المفاتيح لتعرف أكثر عن الوظيفة.



2- جرب ان تستخدم textbox و تجرب الكود:

MessageBox.show("Name of Event")

مع الـ Events التالية:

- keyPress
- KeyDown
- KeyUp
- TextChanged

---

# Microsoft VB Express للمبتدئين 2010

---

الدرس الرابع

---

[algharism.blogspot.com](http://algharism.blogspot.com)

---

ما هي المتغيرات و ما جدواها و كيفية استخدامها و  
بعض التعريفات المهمة في المتغيرات ... كل هذا  
سنتعلمه في هذا الدرس



## 1. مقدمة عن الـ Variables المتغيرات

سنبدأ بتعريف المتغير Variable و هو يشير إلى موقع تخزينى فى الذاكرة و يمكن أن يحتوى على بيانات (عددية أو حرفية) ، و يتم تعريف المتغير أثناء كتابة البرنامج ، و يمكن تغيير محتوى هذا المتغير أثناء سير البرنامج ، و كل متغير له اسم وحيد يعرف به و لا يمكن وجود متغيرين أثناء سير البرنامج ، و كل متغير له اسم وحيد يعرف به و لا يمكن وجود متغيرين أو أكثر بنفس الاسم فى مكان واحد بالذاكرة.

عند تحديد اسم المتغير يجب اتباع شروط محددة:

1. أن يبدأ بحرف هجائى (انجليزى) : A,B, .....Y,Z or a,b,.....,y,z
2. ألا يحتوى على علامات خاصة (!،؟، :، ، ،....)
3. ألا يكون هذا السم من الكلمات المحفوظة فى اللغة البرمجية مثل End , Byte
4. السك قد يتضمن أرقام و حروف.

سنحدث عن أنواع المتغيرات :

أنواع البيانات	Data types	البيانات	الطول بالبايت
عدد صحيح	Integer	يتضمن أعداد صحيحة	2 byte
عدد صحيح طويل	Long	يتضمن أعداد صحيحة	4 byte
نعم لا	Boolean	يتضمن القيمة 1 true أو 0 false	1 byte
مفرد	Single	يتضمن أعداد بها كسور	4 byte
مزدوج	double	يتضمن أعداد بها كسور	8 byte
عملة	Currency	يتضمن أعداد بها كسور و رمز العملة	8 byte
سلسلة حرفية	String	يتضمن بيانات حرفية	1 byte
تاريخ	Date	يتضمن بيانات تاريخ	8 byte
كائن	Object	يتضمن كائن مثل صورة أو فيلم .... إلخ	
عام	variant	يتضمن بيانات من نوع أرقام أو حروف	

نأتى لجزء مهم فى تكوين المتغيرات و هو تعريف المتغير و هو الإعلان عن اسم المتغير و نوعه داخل نافذة البرمجة ، و من صور المتغير داخل نافذة البرمجة:

```
Dim VariableName As DataType
```



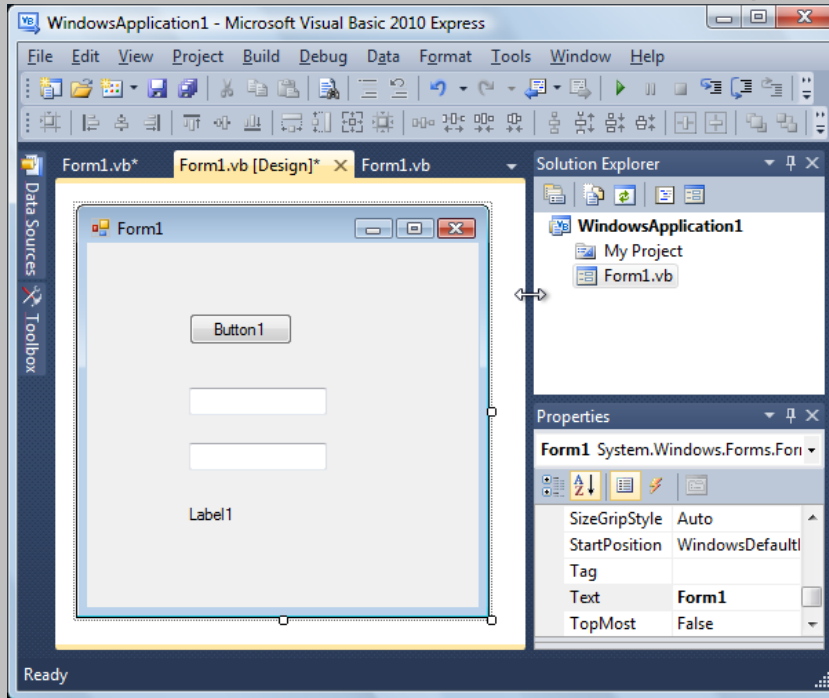
أمثلة تعريف على أنواع المتغيرات السابقة:

التعريف	Data types	أنواع البيانات
<code>Dim myIntegr As Integer = 3</code>	Integer	عدد صحيح
<code>Dim myBoolean As Boolean = true</code>	Boolean	نعم/لا
<code>Dim myDouble As Double = 3.14</code>	double	مزدوج
<code>Dim myString As String = "Algharism"</code>	String	سلسلة حرفية

- لاحظ في الأمثلة السابقة عند كتابة اسم المتغير Variable اتباعه للشروط المتبعة.
  - بالنسبة للتعريف للمتغير String نكتب الكلمات بين علامتي تنصيص " " .
  - لا تخلتف الكلمات المكتوبة بحروف capital عن المكتوبة بحروف small
- `myIntegr=myintegr`



- مثال: على كيفية كتابة متغير و تعريفه في نافذة البرمجة -  
-انشأ مشروع جديد باسم Lesson04  
- صممه كالتالى:



- اضغط Double Click على Button 1
- اكتب الآتى:



```

Form1.vb*
Button1 Click
1 Public Class Form1
2
3 Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
4
5     ' Example 1a
6     Dim myString As String = "Bob"
7
8
9     ' Example 1b
10    Dim myInteger As Integer = 3
11    Dim myDouble As Double = 3.14
12    Dim myBoolean As Boolean = True
13
14
15 End Sub
16 End Class
17

```

### - ملاحظة فى تعريف String

يتم كتابة التعريف string فى علمتى تنصيص " و عند عدم كتابة علامات التنصيص سيتعرف البرنامج عليهم على إنه اسم لمتغير آخر و ليس ما يتضمنه المتغير  
مثال:

```

Form1.vb*
Button1 Click
13
14     ' Example 1c
15     Dim hello As String
16     hello = "hello world"
17
18     ' سيظهر هنا ما يتضمن المتغير
19     ' hello
20     ' و هو ما يوجد بين علامتى التنصيص
21     ' hello world
22     MessageBox.Show(hello)
23
24     ' ستظهر هنا الرسالة
25     ' فقط hello
26     MessageBox.Show("hello")
27
28 End Sub
29 End Class

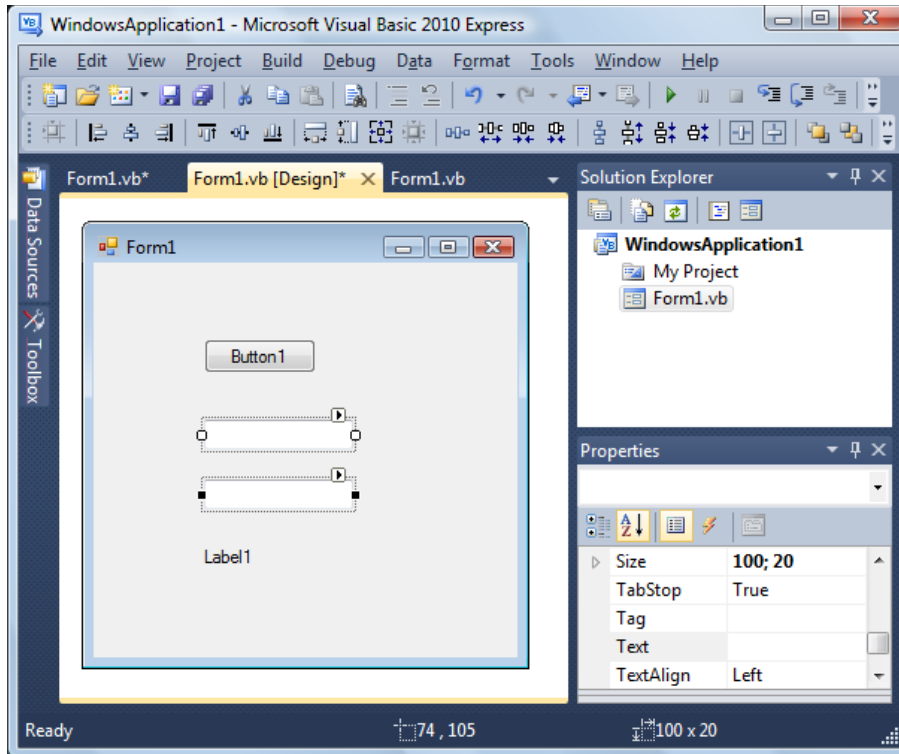
```

علامتى التنصيص هى كوعاء hello و يتضمن رسالة hello world



## 2. كيفية تعريف المتغيرات

تعلمنا ما هي المتغيرات و أنواعها و كيفية تعريفها ، و سنتعلم الآن كيفية استخدامها داخل الكود.  
سنستخدم المثال السابق و الذي تصميحه :



سنذهب إلى نافذة التكويد الخاصة بـ Button1 و نكتب الكود التالي:

```

Form1.vb*
Button1 Click
19     'hello
20     ' هو ما يوجد بين علامتي التنصيص
21     'hello world
22     MessageBox.Show(hello)
23
24     ' ستظهر هنا الرسالة
25     ' فقط hello
26     MessageBox.Show("hello")
27
28     ' Example 2
29     Dim firstTextBox As String = TextBox1.Text
30     Dim secondTextBox As String = TextBox2.Text
31
32     Label1.Text = firstTextBox & " " & secondTextBox
33 End Sub
34 End Class
35
100 %

```





سنكتب الكود من السطر 28 حتى السطر 32:

```
' Example 2
Dim firstTextBox As String = TextBox1.Text
Dim secondTextBox As String = TextBox2.Text

Label1.Text = firstTextBox & " " & secondTextBox
```

في هذه السطور سنجد أنه تم تعريف المتغير الأول و اسمه firstTextBox متغير نصي String ومحتواه هو النص Text المكتوب في العنصر TextBox1

و أيضا تم تعريف المتغير الثاني و اسمه secondTextBox كمتغير نصي String و محتواه هو النص المكتوب Text المكتوب في العنصر TextBox2

و تم تكوين العنصر Label1 ليظهر النص داخله Text:

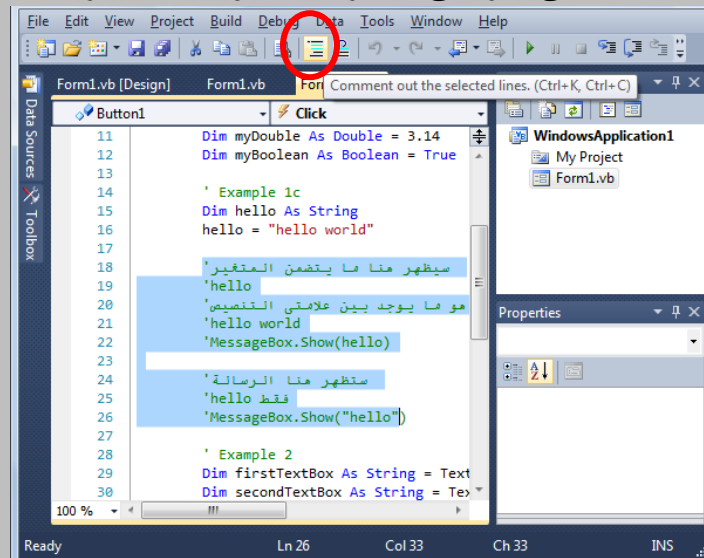
ما يحتويه المتغير firstTextBox (و هو النص المكتوب في Textbox1)

+ ( و هو ما تمثله علامة & ) ما هو مكتوب بين علامتي التنصيص ( المسافة )

+ ما يحتويه المتغير secondTextBox

لفهم الهدف من التكويد السابق اضغط على Debug في صندوق الأدوات في الأعلى أو اضغط F5 ليظهر البرنامج كالتالي:

- إذا كنت تعمل على نفس المشروع السابق في نفس الدرس قبل البدء في تنفيذ البرنامج يجب إيقاف العمل في الكود في السطرين 22 و 26 عن طريق التعليق على الكود و اضغط على





اكتب "المتغير 1" في المربع  
النصي 1 و اكتب "المتغير  
2" في المربع النصي 2 ثم  
اضغط على Button1

## 3. كيفية التحويل بين أنواع المتغيرات

تعلمنا فى الجزء الأول من هذا الـ درس أن المتغيرات هى مساحات يتم شغلها فى الذاكرة (مثل وعاء يحتوى بيانات) و أيضا هذه المتغيرات تختلف فى أنواعها و بالتالى تختلف حسب كل نوع فى المساحة التى تشغلها.

- ارجع لجدول أنواع المتغيرات و مساحاتها فى الجزء الأول من الـ درس الرابع.



الغرض من التحويل بين أنواع البيانات هو الترشيد فى المساحات المشغولة بسبب البيانات التى تحويها .

يوجد طريقتان للتحويل بين أنواع المتغيرات ، سنتناولهما ثم سنعرض مثال على كل طريقة:

### 1- Implicit تحويل مطلق:

- و هى تتضمن تعريف المتغير أثناء شحنه بالقيمة أو البيانات أى تعريفه و شحنه بالبيانات و حجز مساحته فى الذاكرة فى نفس الوقت.
- يعيب هذه الطريقة عدم إمكانية التأكد من صحة أنواع البيانات المدخلة مسبقا.

### 2- Explicit تحويل محدد واضح:

- و تتضمن تعريف المتغير و تحديد نوعه قبل شحنه بالبيانات.
- تتميز هذه الطريقة أنه يتم اكتشاف المتغيرات بسهولة و التأكد أن جميع العمليات الحسابية تتم بشكل صحيح.

الآن سنعرض مثال لكل طريقة:

• مثال: على التحويل بين أنواع البيانات باستخدام طريقة Implicit:



- 1 - حول الكود السابق فى البرنامج الحالى إلى تعليقات
- 2 - اكتب الكود التالى:





```

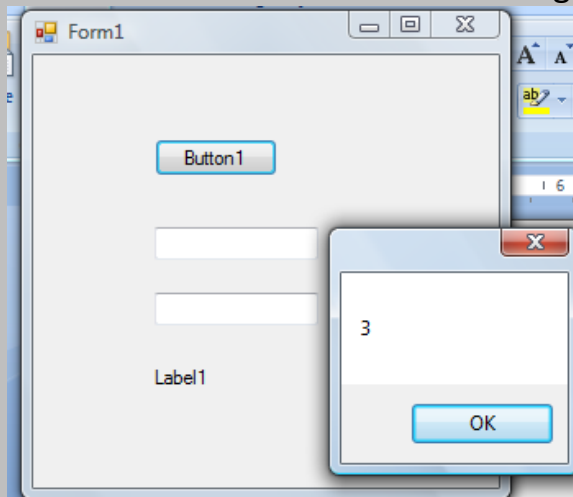
Form1.vb*
Button1 Click
33
34 ' Implicit conversion - dangerous
35 Dim myValue As Integer
36 Dim myOtherValue As Double = 3.14
37 myValue = myOtherValue
38
39 MessageBox.Show(myValue)
40 End Sub
41 End Class

```

- إذا لاحظنا السطر 35 تم تعريف المتغير myValue كـ Integer (يشغل 2 بايت في الذاكرة)
- السطر 36 : تم تعريف المتغير myOtherValue كـ Double (يشغل 8 بايت في الذاكرة) و يحتوي على البيان 3.14
- السطر 37: تم تحويل المتغير myOtherValue إلى myValue (تصغير المساحة التي يشغلها من 8 إلى 2 بايت)
- السطر 39: سنجد اننا قمنا بعملية تحويل أخرى للمتغير myValue من Integer (2 بايت) إلى String (1 بايت) حيث الحدث Show في MessageBox لا يعرض إلى متغير String
- لتلخيص هذه العملية من التحويل بين أنواع المتغيرات

السطر 39 MessageBox.Show	السطر 37
myValue → (message)	myOtherValue → myValue
Integer (2Bytes) → String (1Byte)	Double (8 Bytes) → Integer(2 bytes)

3 - عند تشغيل البرنامج Debug



- عند الضغط على Button 1 ستظهر رسالة "3"

- على الرغم من أن المتغير الأساسي كان 3.14 إلا أنه تحول إلى "3" فقط ، فقد حدث فقد في البيانات عند التحويل من Double (يحتوي أعداد فيها كسور) إلى Integer (يعرض أعدادا دون الكسور) و لذلك تعتبر عملية التحويل بين المتغيرات بطريقة Implicit خطيرة لأنها قد تؤدي إلى فقد بيانات.





- ملخص عملية التحويل :

Double 3.14 (8 Bytes)	-→	Integer 3 (2 bytes)	Line 37
Integer 3 (2bytes)	→	String "3" (1 byte)	Line 39

• مثال: على التحويل بين أنواع البيانات باستخدام طريقة Explicit:

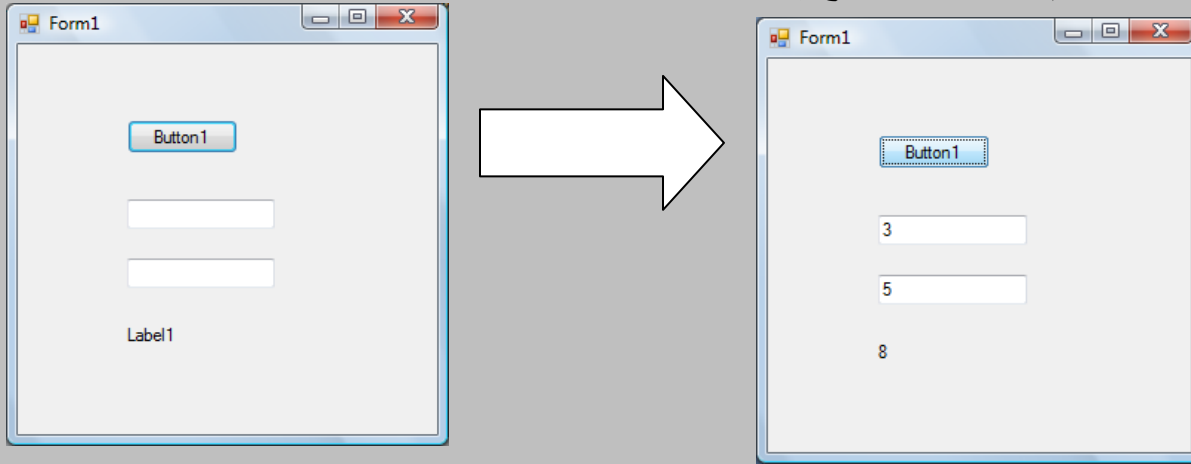


- 1 - حول الكود السابق في البرنامج الحالي إلى تعليقات
- 2 - اكتب الكود التالي:

```

Form1.vb*
Button1 Click
39     'MessageBox.Show(myValue)
40
41     Dim firstTextBox As Integer = 0
42     Dim secondTextBox As Integer = 0
43     Dim result As Integer = 0
44
45     firstTextBox = TextBox1.Text
46     secondTextBox = TextBox2.Text
47
48     result = firstTextBox + secondTextBox
49     Label1.Text = result
50
51     End Sub
52 End Class
    
```

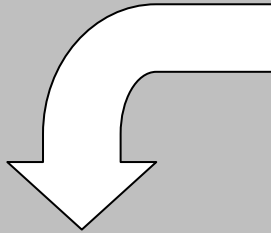
- في السطور 41 و 42 و 43 تم تعريف 3 متغيرات ك Integer
  - في السطرين 45 و 46 : سيحتوي المتغير firstTextBox على البيانات المدخلة في TextBox1 وكذلك بالنسبة secondTextBox
  - في السطرين 48 و 49 : جعل المتغير result يحتوى على ما يحتويه المتغير firstTextBox (أرقام و كسور Integer) + secondTextBox (أرقام و كسور Integer) ، و العنصر Label1 سيعرض ما يحتويه المتغير result
- 3- لفهم أكثر شغل البرنامج Debug



سنجد أنه جمع محتوى textbox1 و textbox2 مثل الآلة الحاسبة أي أنه لم يحول integer إلى string لتحديد أنواع المتغيرات سابقا و عدم ترك عملية التحويل للبرنامج



- عند ادخال حروف بدلا من الأرقام سيحدث خطأ:



- و ذلك دليل على عدم تغيير نوع المتغيرات ،حيث "five" هو متغير نصي لا يقبله المتغير Integer



## 4. بعض التعريفات في المتغيرات

سنعرض بعض التعريفات التي سنحتاجها في عملية بناء و تحديد المتغيرات ، و أهمها تعريفين: statement و expression ، سنعرضهما من خلال المثال التالي:

إذا كتبنا الكود التالي و قمنا بتشغيله:

```
Dim x As Integer
x + 3
```

عند تشغيله سيحدث خطأ ، و ذلك لأن السطر الثاني من الكود السابق هو Expression و ليس Statement.

سيوضح الجدول التالي أكثر الفرق بينهما:

جملة Statement	تعبير Expression
هو إفادة أو تصريح مجرد إفادة أو تصريح بعملية برمجة	هي عبارة جبرية أو مقدار جبرى يمكن تقييمها بالأرقام لأنها تتعلق بالعمليات الحسابية
تتضمن تعريف المتغير Dim x as Integer أو شحن المتغير بالبيانات MyInteger=3 أو إظهار رسالة نصية "Hello world" MessageBox.Show أو حتى إعلان عنصر جديد	تتضمن عمليات طرح و قسمة و سائر العمليات الحسابية أو حتى كالمثال التالي: 2 > 3 صح أم خطأ؟ ..... خطأ

- في تعريف آخر للـ Expression:  
هي هبارة عن ربط مجموعة من العناصر (سواء كانت هذه العناصر ثوابت أو متغيرات أو معاملات) و يشترط في ذلك أن تكون مكونات التعبير expression الواحد متجانسة مع بعضها في نوع البيان المرتبط في هذا التعبير (data type) .





## 5. تدريب

قم ببناء البرنامج الخاص بخدمة العملاء المشروح فى الدرس الثانى ، مع إضافة زر OK و الذى سيقوم بتسجيل جميع البيانات المدخلة إلى متغيرات قم بتحديد ما مع مراعاة تعريف الأنواع الصحيحة للمتغيرات المستخدمة.

The screenshot shows a form titled 'Form1' with the following components and labels:

- Text box:** A label 'اسم الزبون' points to a text input field.
- Combo Box:** A label 'مقدم الخدمة' points to a dropdown menu. A callout box indicates it contains 'علي (أحمد - محمد - ممدوح)'.
- List box:** A label 'المنتج' points to a list box labeled 'ListBox1'. A callout box indicates it contains 'علي (خضر - فاكهة - بقالة - ألبان)'.
- Radio Box:** A label 'النوع' points to two radio buttons labeled 'ذكر' and 'انثى'.
- Check Box:** A label 'ارسل كتالوج' points to a checkbox.
- Button:** Two buttons labeled 'Button1' and 'Button2' are shown. A separate callout box labeled 'Button' contains the text 'زر ok و الآخر cancel'.
- Labels:** A box labeled 'Labels' has lines pointing to the labels 'اسم الزبون', 'مقدم الخدمة', 'المنتج', and 'النوع'.

---

# Microsoft VB Express للمبتدئين 2010

---

الدرس الخامس

---

[algharism.blogspot.com](http://algharism.blogspot.com)

---

شرح لبعض أنواع الجمل statements وكيفية  
استخدامها و أنواعها



# 1. مقدمة

في هذا الدرس سنتعمق أكثر في عملية بناء الكود ، فسنوضح أمثلة على الـ Statements، فقد قلنا في الدرس السابق أن الـ Statements الجمل ما هي إلا إفادات أو تصريحات.

- راجع الجزء الخاص بالـ Statements و Expressions في الدرس الرابع



نذكر من هذه الأمثلة :

1. Branching Statements الجمل التشعبية.
2. Recursion Statements الجمل الحلقية التكرارية.
3. Arrays متغيرات متعددة المحتوى.

## Recursion Statements

### الجمل الحلقية التكرارية

أما هذا النوع فيسمح لنا التكرار في تنفيذ الكود داخل مجموعة من البيانات و الهدف منه كمثال مراجعة البيانات خلال عملية التنفيذ  
أمثلة:

1. For Each
2. For ...Next
3. Do While...Loop

## Branching Statements

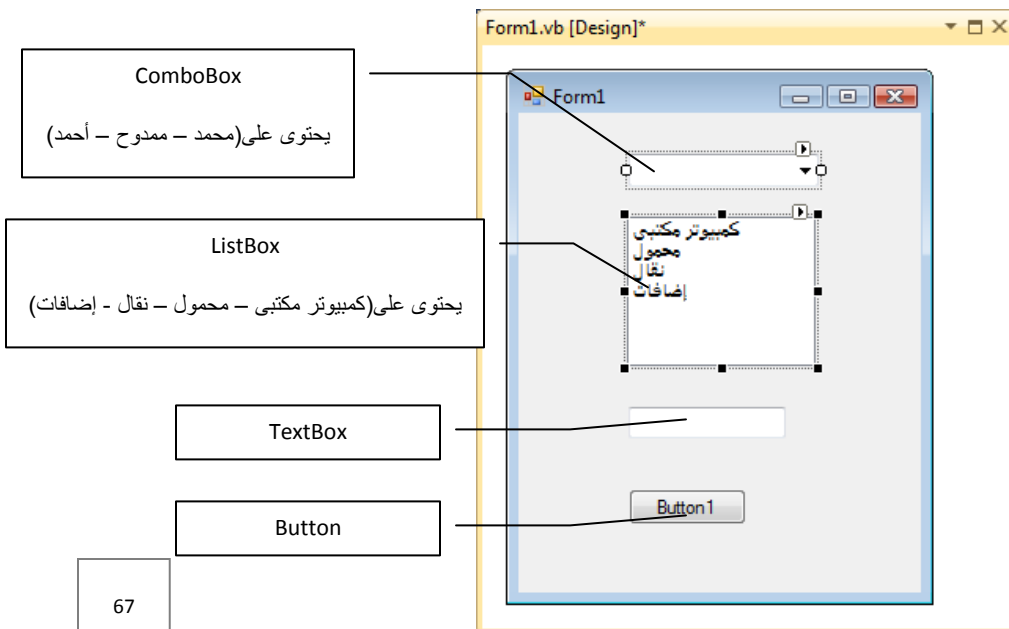
### الجمل التشعبية

يسمح هذا النوع من الجمل بتغيير أو تفريع تنفيذ الكود حسب تقييم المقدار الجبري expression

أمثلة من الإفادات:

1. If....Then
2. Select Case

لمعرفة المزيد عن هذه الجمل تابع الشرح التالي. قم بإنشاء التصميم التالي و سمي lesson5:





## 1.2. الجمل التشرعية: If...Then

هي جملة تشعبية وظيفتها تقرير ما إذا سيتم تنفيذ أو عدم تنفيذ مجموعة من الأوامر حسب تقييم مقدار جبرى expression.

اكتب الكود التالي في Button1

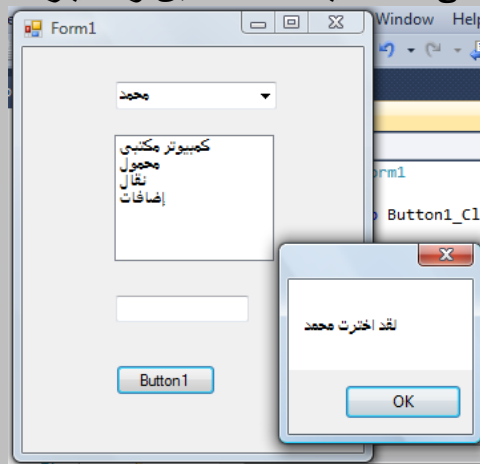
```

Form1.vb*
Button1 Click
1 Public Class Form1
2
3     Private Sub Button1_Click(ByVal sender As System.Object, By
4
5         ' Example 1 - First 'if/then' statement syntax
6         If ComboBox1.Text = "محمد" Then
7             MessageBox.Show("نقد اخترت محمد")
8             ComboBox1.Text = ""
9         End If
10
11     End Sub
12 End Class
13
    
```

و معنى الكود If...Then أنه في حالة اختيار "محمد" من العنصر ComboBox1 ستظهر رسالة "لقد اخترت محمد"، لكن في حالة عدم اختيار "محمد" لن يتم تنفيذ الكود بعد Then، و وظيفة End If لتقرير نهاية الجملة statement ليعلم الـ VB أنها نهاية جملة If..Then.

### • مثال: على If...Then

- في حالة تنفيذ المثال السابق و اختيار "محمد":



أما في حالة عدم الاختيار لن يحدث شيء.



- سلاحظ أنه تم تنفيذ الكود أو عدم تنفيذه حسب تقييم التعبير expression (نعم أو لا) للجملة اختيار محمد:

أى فى حالة اختيار محمد ← تقييم التعبير بـ "نعم" ← سيتم تحقيق الكود  
و فى حالة عدم اختيار محمد ← تقييم التعبير بـ "لا" ← لن يتم تحقيق الكود

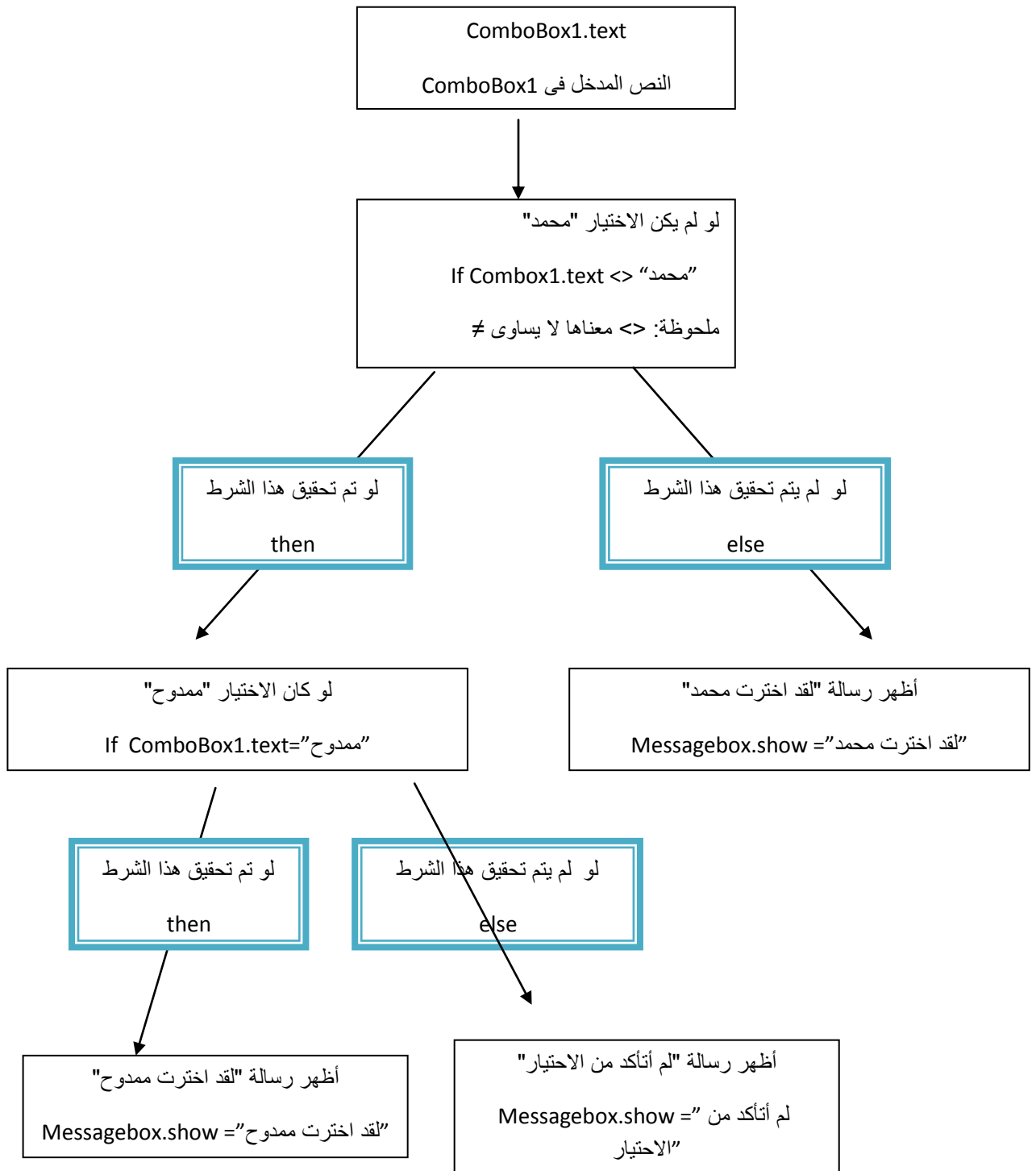


لننتقل إلى مثال أصعب عن جملة If..Then وهى أن تكون فى سياق الكود:

```

Form1.vb*
Button1 Click
5      ' Example 1 - First 'if/then' statement syntax
6      'If ComboBox1.Text = "محمد" Then
7      '    MessageBox.Show("لقد اخترت محمد")
8      '    ComboBox1.Text = ""
9      'End If
10
11     ' Example 2 - Embedded 'if/then' statements,
12     'else' statement
13     If ComboBox1.Text <> "محمد" Then
14         If ComboBox1.Text = "ممدوح" Then
15             MessageBox.Show("لقد اخترت ممدوح")
16         Else
17             MessageBox.Show("لست متأكدا من اختيارك")
18         End If
19     Else
20         MessageBox.Show("لقد اخترت محمد")
21     End If
22
23     End Sub
24 End Class
  
```

خطوات تنفيذ هذا الكود ستكون كالاتى:





عند تنفيذ البرنامج ، عند اختيار ممدوح ستظهر رسالة "لقد اخترت ممدوح" ، و أيضا عند اختيار "محمد" ستظهر رسالة " لقد اخترت محمد" ، أى اختيار آخر ستظهر رسالة " لست متأكدا من اختيارك".



## 2.2. الجمل التشعبية: Select Case

تمثّل الجملة التشعبية If..Then .

اكتب الكود التالي :

```

Form1.vb*
(General) (Declarations)
21      'End If
22
23      ' Example 3 - case/select branching statement
24      Select Case ListBox1.SelectedItem.ToString()
25          Case "محمول"
26              MessageBox.Show("لقد اخترت محمول")
27          Case "نقال"
28              MessageBox.Show("لقد اخترت نقال")
29          Case Else
30              MessageBox.Show("لقد اخترت شيئاً آخر")
31      End Select
32
33      End Sub
34      End Class
35
100 %
  
```

- خصائص listbox في الكود (ListBox1.SelectedItem.ToString()) سنتطرق له  
الدروس القادمة.



و معنى هذا الكود أنه عند اختيار "محمول" من ListBox1 ستظهر رسالة "لقد اخترت محمول" و هكذا عند اختيار "نقال" ، أما عند اختيار آخر ستظهر "لقد اخترت شيئاً آخر".

في نهاية شرح الجمل التشعبية نذكر أن لكل نوع له نقاط قوته و لا نستطيع تفضيل إحداهما على الأخرى ، ستعرف أكثر ما هو الأنسب مع برنامجك مع التدريب أكثر و التوغل في عملية التكويد أكثر.





## 3. متغيرات متعددة المحتوى Arrays

هي طريقة لجمع مجموعة من المتغيرات مترابطة معا بطريقة ما ، أو هو متغير واحد يحتوى على عدة بيانات مترابطة ( وعاء له أكثر من جزء لحمل الأشياء) لكل بيان داخله له رقم للوصول إليه ، و يمكن استخدام Array مثلا لحفظ درجات امتحان لفصل محدد أو قائمة بأسعار و هكذا.

لتعلم كتابة array انظر إلى الكود التالي:

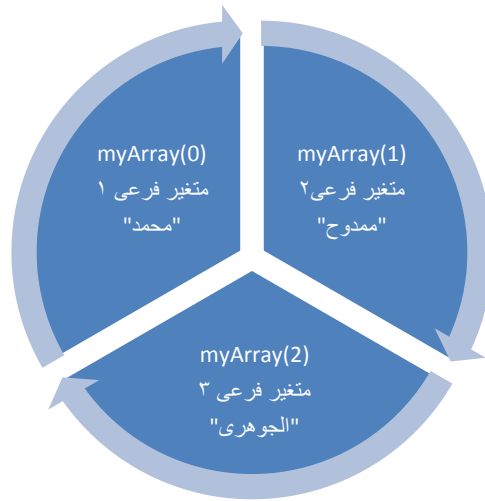
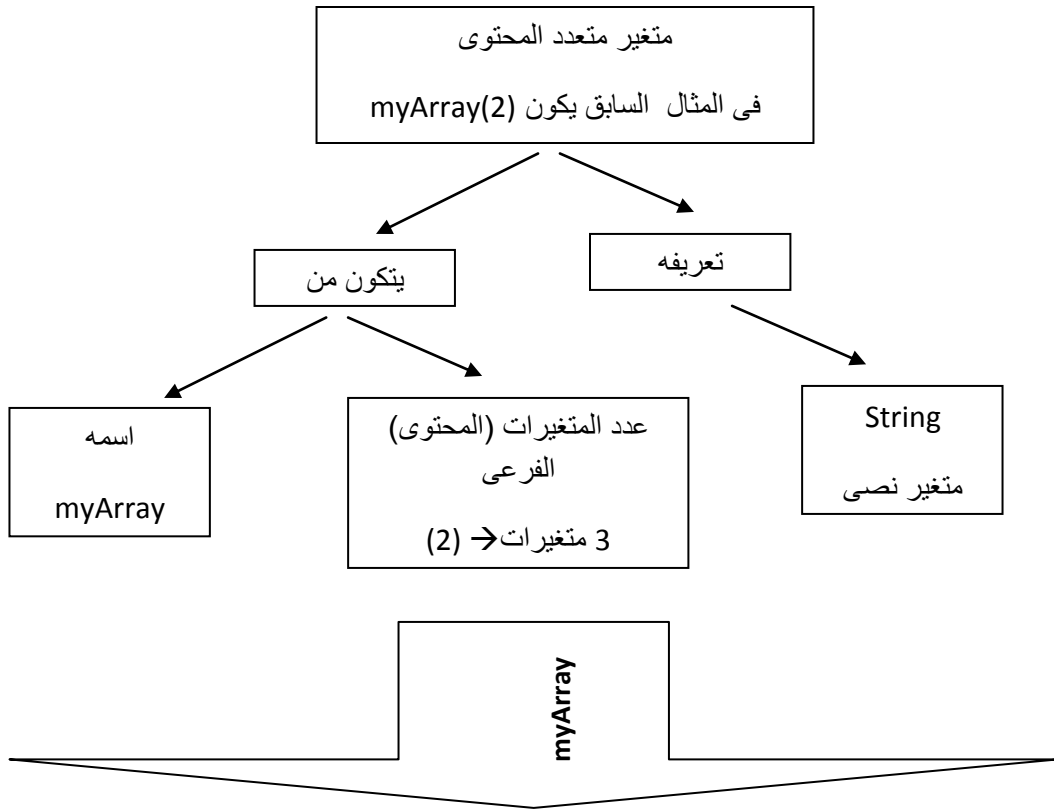
```

Form1.vb*
Button1 Click
32
33 ' Example 4 - Sized Array
34 Dim myArray(2) As String
35 myArray(0) = "محمد"
36 myArray(1) = "ممدوح"
37 myArray(2) = "الجمهرى"
38 MessageBox.Show(myArray(2))
39
40
41 End Sub
42 End Class
43
100 %

```

السطر 34	<b>Dim:</b> هي لإعلان المتغير <b>myArray(2):</b> myArray: اسم المتغير (2) عدد البيانات التي سيحويها المتغير المتعدد المحتوى <b>As String:</b> تعريف المتغير كمتغير خطى (يحتوى على نصوص)
السطور 35 و 36 و 37	محتويات المتغير "محمد" = هي المتغير الفرعى رقم 1 (المحتوى رقم 1) <b>myArray(0):</b> المتغير الفرعى 2 <b>myArray(1):</b> المتغير الفرعى 3 <b>myArray(2):</b>
السطر 38	إفادة بإظهار رسالة بمحتويات المتغير الفرعى 3 ( <b>MessageBox.Show(myArray(2))</b> )

لفهم أكثر للمتغير متعدد المحتوى: انظر الشكل التالي

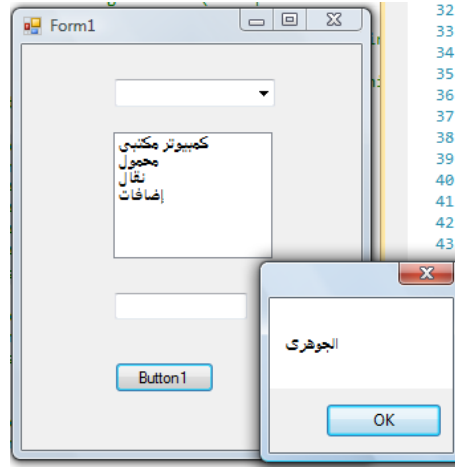


- نلاحظ على الرغم من أننا حددنا عدد المتغيرات الفرعية برقم 2 إلى إنه العدد الفعلي للمتغيرات الفرعية كان 3 لأن العد يبدأ من الرقم 0





إذا قمنا بتشغيل debug البرنامج ، سنجد أنه أظهر رسالة بالمتغير الفرعي 3:



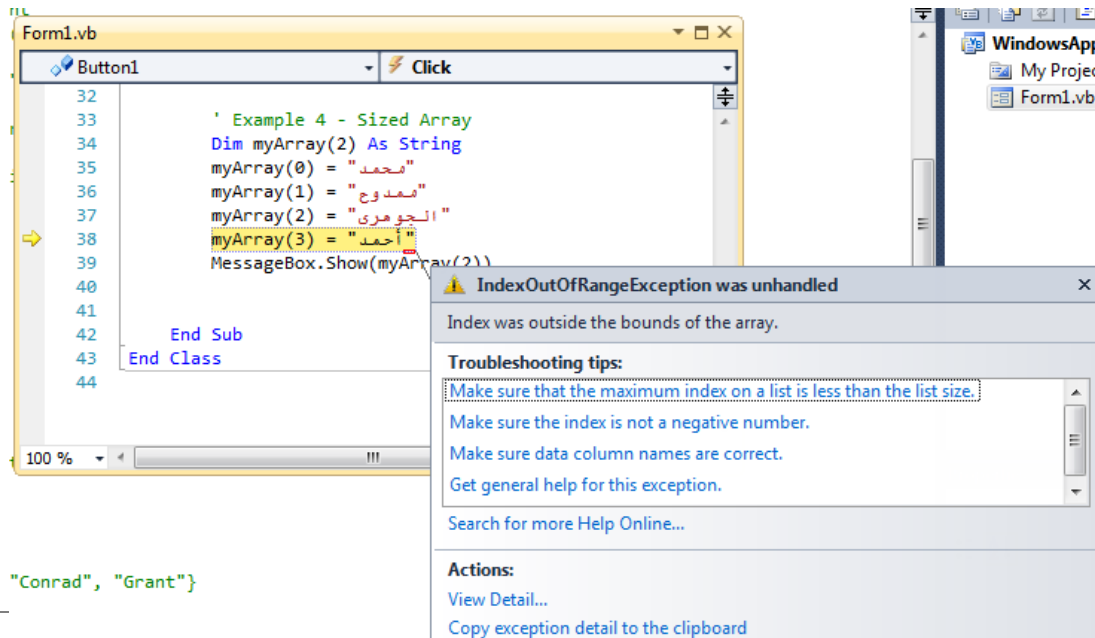
إذا غيرنا الكود بإضافة متغير فرعي رابع مع عدم تحديد عدده في المتغير الأساسي:

```

Form1.vb*
Button1 Click
32
33 ' Example 4 - Sized Array
34 Dim myArray(2) As String
35 myArray(0) = "محمد"
36 myArray(1) = "مدوح"
37 myArray(2) = "الجوهري"
38 myArray(3) = "أحمد"
39 MessageBox.Show(myArray(2))
40
41
42 End Sub
43 End Class

```

ثم قمنا بتشغيل البرنامج ، ستظهر رسالة خطأ تفيد بأن يوجد متغير فرعي خارج نطاق المتغير الأساسي:



"Conrad", "Grant"}



طريقة أخرى لكتابة كود المتغير المتعدد المحتوى بدون تحديد عدد معين للمتغيرات الفرعية منه:

```

Form1.vb*
Button1 Click
41 ' Example 5 - Initialized Array
42 Dim myOtherArray() As String = {"محمد", "معدوح", "الجمهورى", "أحمد"}
43 MessageBox.Show(myOtherArray(2))
44
45
46 End Sub

```

حيث يمكنك اى عدد من المتغيرات الفرعية أو تغيير عددها خلال عملية التكويد ، لاحظ أنه تمت كتابة المتغيرات الفرعية داخل { } ، و عند تشغيل debug الكود ستظهر رسالة نصية بمتوى المتغير الفرعى 3.

## 1.4. الجملة الحلقية التكرارية: for/each

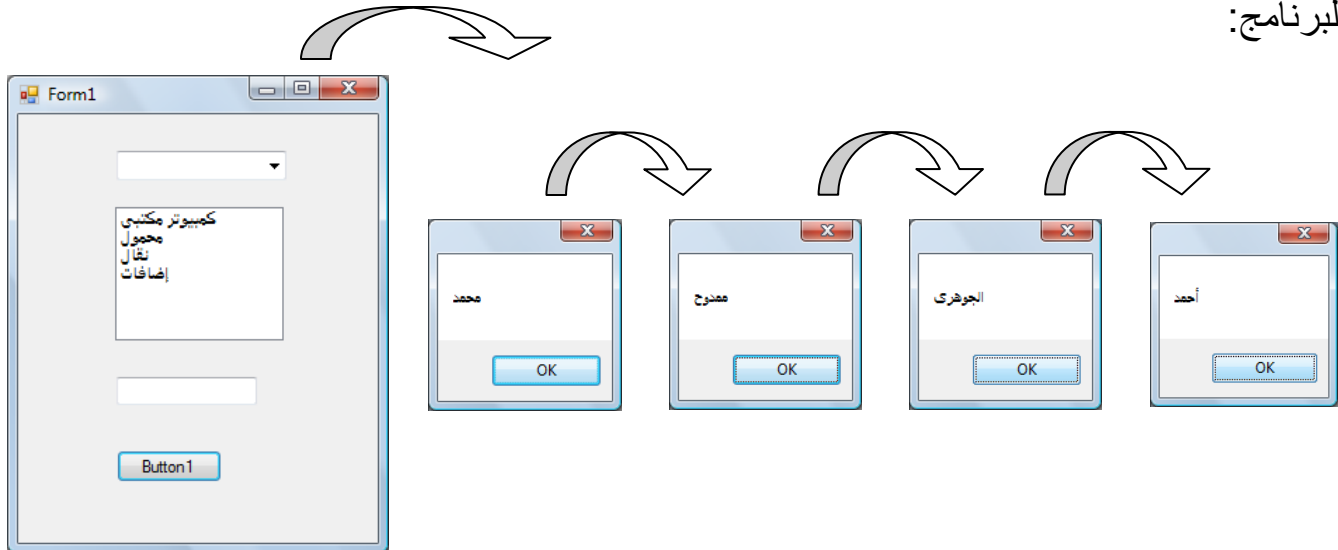
كما عرفنا في الجزء الاول من هذا الدرس أن الجمل التكرارية تتيح لك التكرار داخل جزء من الكود أو يمر على كل متغير فرعى في متغير متعدد المحتوى، لنر ذلك في المثال التالي:

```

Form1.vb*
Button1 Click
44
45
46 ' Example 6 - for each مع array
47 Dim myArray() As String = {"محمد", "ممدوح", "الجومرى", "أحمد"}
48
49 Dim person As String
50 For Each person In myArray
51     MessageBox.Show(person)
52 Next
53
54 End Sub
55 End Class
56
100 %

```

في هذا الكود تم تعريف myArray كمتغير نصى متعدد المحتوى ، و المتغير person تم تعريفه أيضا كمتغير نصى و تم استخدامه في السطر 50 ليحتوى متغير فرعى واحد في كل مرة من خلال الكود for each لتظهر رسالة نصية في كل خطوة بمحتوى المتغير person، لفهم أكثر قم بتشغيل البرنامج:



أى أنه ينتقل كل خطوة (بشكل تكرارى) في الكود المحدد له (هنا المتغير متعدد المحتوى) حتى نهاية التكرار.



## 2.4. الجملة التكرارية: for/next

تستخدم لتكرار جزء من الكود عدد محدد من المرات يتم تحديدها مسبقا ، سيتم شرحها من خلال الكود التالي:

```

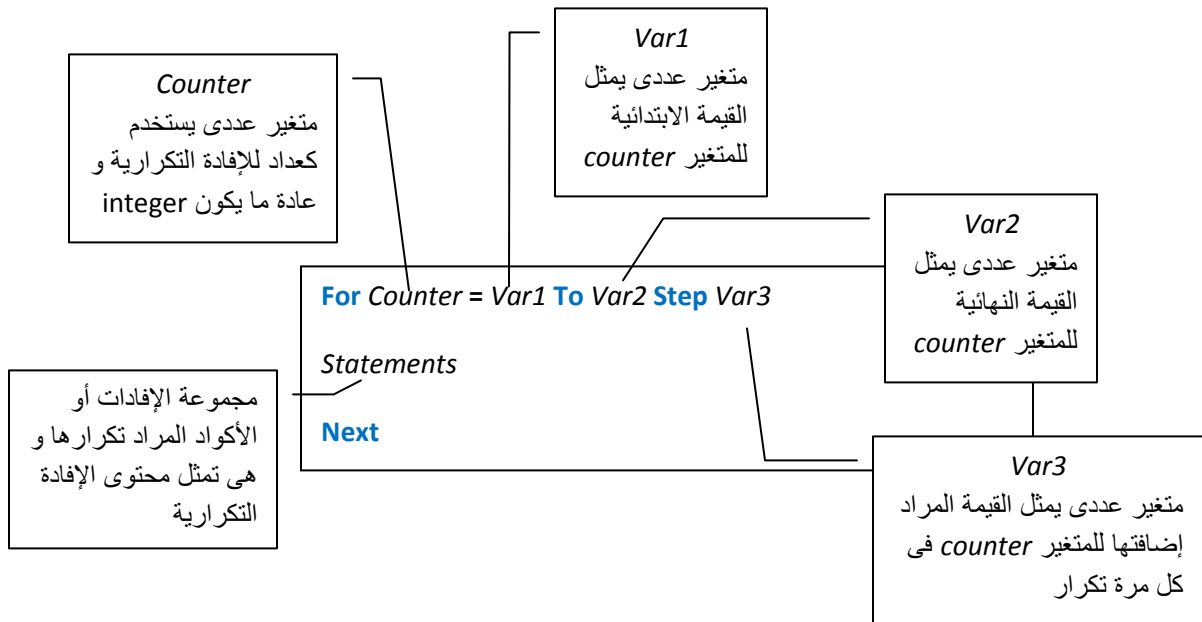
Form1.vb*
Button1 Click
53
54     ' Example 7 - 'for/next' recursion statement
55
56     Dim i As Integer = 0
57     For i = 0 To 5
58         MessageBox.Show(i.ToString())
59     Next
60
61 End Sub
62 End Class
63
100 %

```

سطر 56: تم تعريف المتغير  $i$  كـ integer و شحنه بالقيمة 0 .

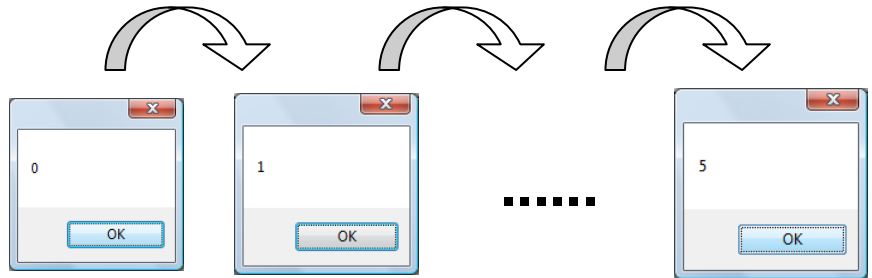
سطر 57 و 58: تكويد المتغير  $i$  بأنه يبدأ من 0 حتى يصل إلى رقم 5 ، فيما معناه أنه سيتم إعادة الكود 5 مرات و في كل مرة سيظهر رقم من 0 إلى 5 بالترتيب في رسالة سطر 59: نهاية الإفادة.

لفهم أكثر انظر الشكل التالي للكود for/next:





عند تشغيل البرنامج:





## 5. تركيبات بين الجمل التكرارية و التشعبية

بعض التركيبات بين الجمل لتوضيح بعض المجالات التي يمكن استخدامها مجتمعة، و سنبدأ بالجمع بين for/next و if ...then

```

Form1.vb*
Button1 Click
69
70 ' Example 9 - combining array and branching statements
71 Dim myArray() As String = {"محمد", "ممدوح", "الجوهري", "أحمد"}
72
73 Dim i As Integer
74 For i = 0 To myArray.Length - 1
75     If myArray(i) = "الجوهري" Then
76         MessageBox.Show("تم إيجاد الجوهري")
77         Exit For
78     End If
79 Next
80
81 End Sub
82 End Class
  
```

السطر 71: تم تعريف متغير متعدد المحتوى myArray و شحنه بالبيانات المذكورة في الكود.

السطر 73: تم تعريف المتغير i ك integer

السطر 74: أصبح المتغير i هو العداد للإفادة for/next و قيمته الابتدائية 0 و قيمته النهائية هي عدد القيم في المتغير متعدد المحتوى (myArray.length) - 1 ، حيث سينتقل في كل تكرار إلى قيمة من قيم المتغير متعدد المحتوى بالترتيب.

- راجع Arrays لمعرفة الفرق بين عدد القيم الفعلي داخل المتغير myArray و عدد القيم التي نحددها بين أقواس المتغير myArray() لمعرفة سبب الكود (ناقص 1)



السطر 75 و 76: هي محتوى إفادة for/next و يتكون من الإفادة if...then لترشد الكود أنه إذا وصلت الإفادة التكرارية في المتغير متعدد المحتوى (myArray(i)) إلى القيمة "الجوهري" ستظهر رسالة نصية "تم اختيار الجوهري".

السطر 77: Exit for لتفيد الخروج من الإفادة التكرارية و عدم اكمالها و ذلك من خلال الإفادة .if..next



السطر 78 و 79: هى نهايات الجمل if و for.



اقرأ الجدول من هذا الاتجاه

لفهم أكثر عن عملية تنفيذ البرنامج ، سنشرح كيف سينفذ الكمبيوتر الكود لا تقم

بتشغيل البرنامج فى هذه اللحظة (don't debug):

	الجملة for	الجملة if
	ستكرر حسب محتوى المتغير i والذي سيبدأ من 0 حتى عدد قيم المتغير myArray ناقص 1 = من 0 حتى (1-4) = من 0 حتى 3	شرطها أن يكون قيمة "الجوهري" myArray="
myArray(i)	من الجملتين السابقتين سيستنتج الكمبيوتر أن المطلوب منه الآتى: لو وصلت للاختيار "الجوهري" فى المتغير myArray حسب العداد i الخاص بالإفادة for :	
خطوة 0 myArray(0)	سيقع الاختيار على القيمة رقم 1 و هى "محمد"	مما يتنافى مع if ولن يتم تنفيذ و سينتقل إلى الخطوة التالية
خطوة 1 myArray(1)	سيقع الاختيار على القيمة رقم 2 و هى "ممدوح"	مما يتنافى مع if ولن يتم تنفيذ و سينتقل إلى الخطوة التالية
خطوة 2 myArray(2)	سيقع الاختيار على القيمة رقم 3 و هى "الجوهري"	يتناسب مع شرط if وسيتم تنفيذ الكود التالى: إظهار رسالة نصية و الخروج من الإفادة التكرارية و عدم إكمالها

لكن عند تشغيل البرنامج debug سنجد أن الكمبيوتر ينتقل إلى خطوة 2 على الفور لأنه لم يأخذ تعليمات أخرى (كود لينفذه) فى الخطوة 0 و 1 لينفذها لذلك لم يكن واضحا الانتقال بين الخطوتين 0 و 1 إنما الخطوة 2 كانت الواضحة.

- دائما ما يفكر الكمبيوتر (برنامج الـ Visual Basic) بهذه الطريقة ، مما يمكنك من معرفة كيفية التصميم المثلى للبرنامج و كيفية اكتشاف الأخطاء.





## التركيبة الثانية هي بين for/next و select case

```

Form1.vb*
Button1 Click
81 ' Example 10 - recursion and branching combined
82 Dim i As Integer
83 For i = 0 To ListBox1.SelectedItems.Count - 1
84     Select Case ListBox1.SelectedItems(i).ToString()
85         Case "كمبيوتر مكتبي"
86             MsgBox.Show("كمبيوتر")
87         Case "نقال"
88             MsgBox.Show("نقال")
89         Case "محمول"
90             MsgBox.Show("محمول")
91         Case "إضافات"
92             MsgBox.Show("إضافات")
93     End Select
94 Next
100 %

```

الهدف من الكود أنه عند ضغط button1 سيتم تنفيذ إفاذة تكرارية عدادها i وقيمتة الابتدائية 0 و النهائية هو عدد العناصر داخل listBox1 ناقص 1 ، و في كل تكرار سيتحدد الاختيارات التي قام بها المستخدم لتظهر له رسائل نصية بمحتوى الاختيار ، لفهم أكثر قم بتشغيل البرنامج و قم بالتعليم على أكثر من اختيار في listBox1 ( وذلك من خلال التعليم بالماوس مع الضغط على Ctrl ) و اضغط button1 عندما تفرغ.



## 6. تدريب

1 – ابدأ فى تنفيذ برنامج آلة حاسبة بسيطة مكون من :

- 2 textboxes : لإدخال الأرقام

- 1 listbox منسدل: يحتوى على هذه القيم : جمع ، طرح ، قسمة ، ضرب

اكتب الكود حيث حسب الاختيار فى الـ listbox ستتم عملية الحساب بين الأرقام فى textbox

ستستخدم العمليات الحسابية التالية:

+ الجمع

- الطرح

\* الضرب

/ القسمة



راع التعريفات الازمة للمتغيرات المستخدمة و طرق التحويل بينها و ذلك لأن القيم كلها رقمية.

2- اجعل قائمة بأسماء أفلام Dvd فى متغير متعدد المحتوى Array و هذه القائمة هى المراد تأجيرها ، بفرض أن كل فيلم سعره 3 جنيهات ، اعرض المجموع الكلى لسعر الأفلام و عدد الأفلام فى القائمة فى label ، و اعرض اسم كل فيلم و سعره فى label آخر و حاول أن يكون اسم كل فيلم و سعره فى سطر خاص به.

---

# Microsoft VB Express للمبتدئين 2010

---

الدرس السادس

---

[algharism.blogspot.com](http://algharism.blogspot.com)

---

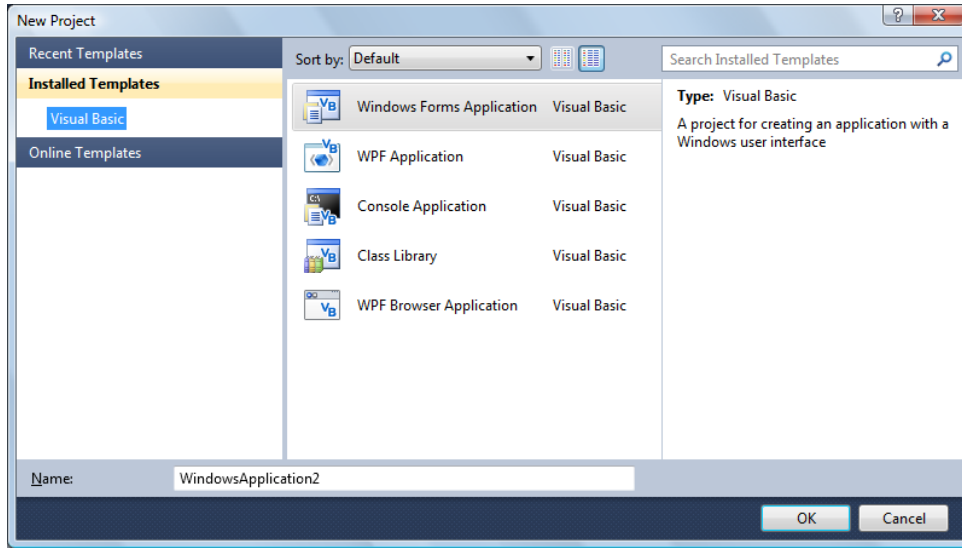
درس تلخيصي لكل الكتاب حيث سيتم صناعة أول  
برنامج متكامل في لغة الفيجوال بيسك

## 1. إنشاء مشروع جديد

سنحاول التعرض لبعض العناصر التي تم شرحها سابقا من خلال مشروع تصفح صور.

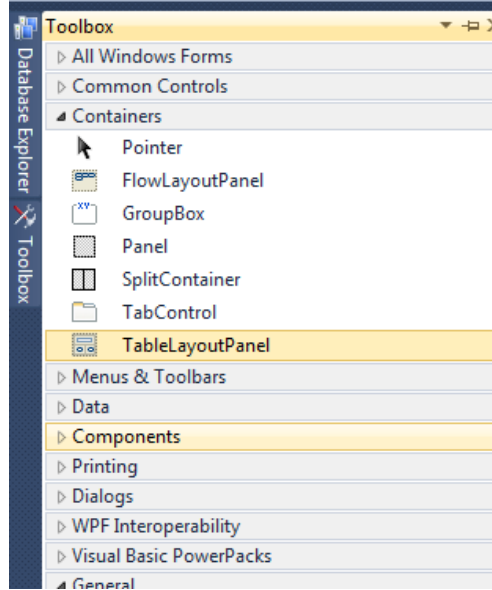
انشأ مشروعاً جديداً ، افتح البرنامج من قائمة البداية الخاصة بـ Windows Start menu ثم أكثر من  
Microsoft Visual Basic 2010 Express برنامج

اختر New Project ثم Windows Forms Application واسم البرنامج الجديد MyFirstVBApp



## 2. أبدأ في التصميم

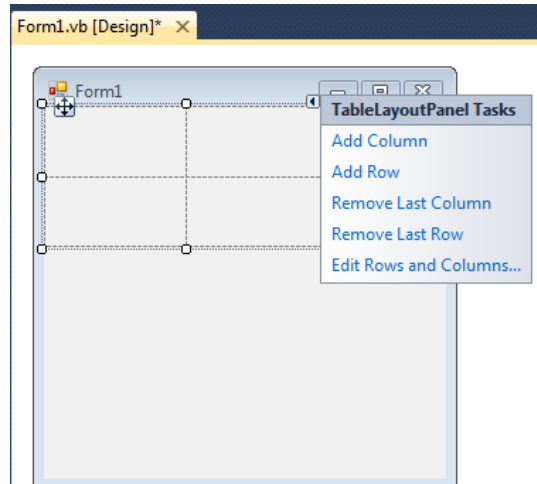
1. اذهب إلى Toolbox على يسار النافذة لتظهر لك ، اختر من قسم Containers العنصر TableLayoutPanel



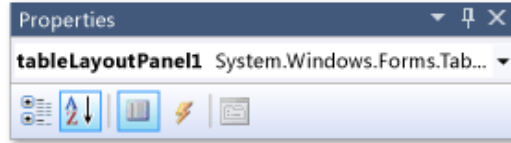
- يمكنك تثبيت ظهور نافذة الـ Toolbox عن طريق الضغط على رمز الدبوس في الجزء الأعلى منها



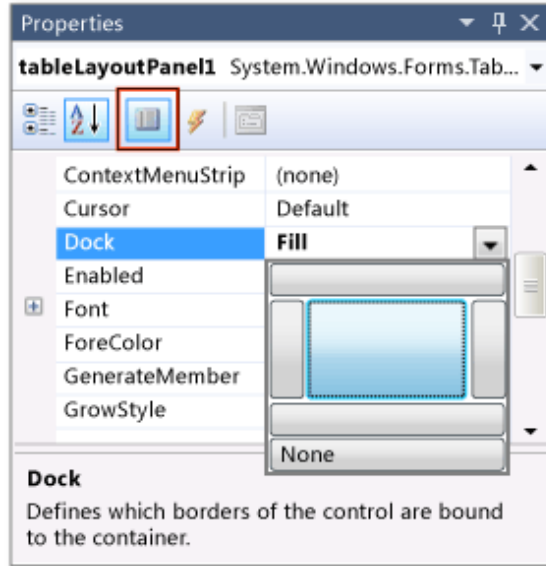
عند إضافة العنصر المراد ستظهر نافذة التصميم كالاتى:



2. اذهب إلى نافذة properties فى الجانب الأيمن من الشاشة و تأكد من اختيار TableLayoutPanel كما فى الصورة

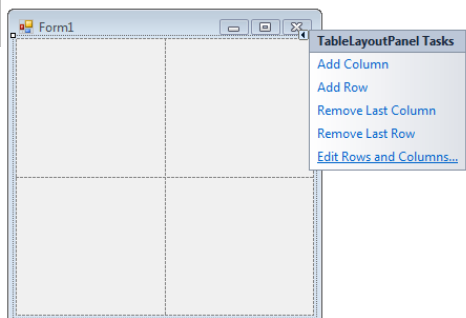


إذهب إلى الخاصية Dock فى نافذة الـ properties و تأكد أنها غير مفعلة **None** ثم اضغط على السهم الظاهر لتظهر لك قائمة بها مبرعات ، اختر المربع الأوسط ليظهر لك الاختيار fill كما فى الصورة:



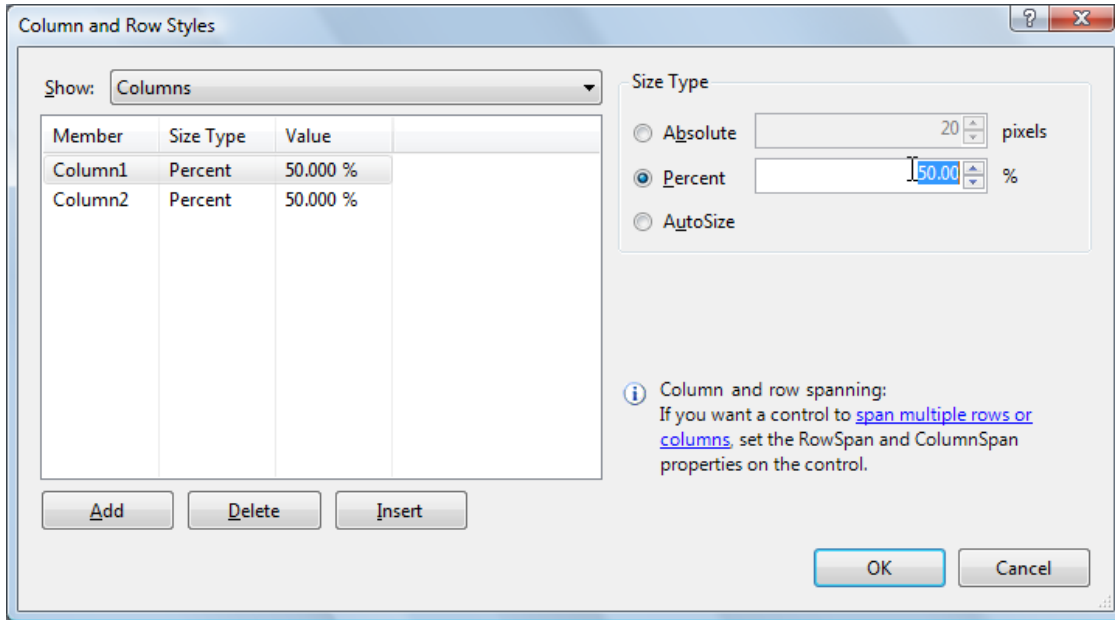
عند اتمام الاختيارات ستجد أن TableLayoutPanel قد ملاً نافذة التصميم.

وظيفة TableLayoutPanel كما وظيفة الجدول فى الورد حيث تتكون من أعمدة و سطور ، و كل خلية ممكن أن تحتوى عنصر



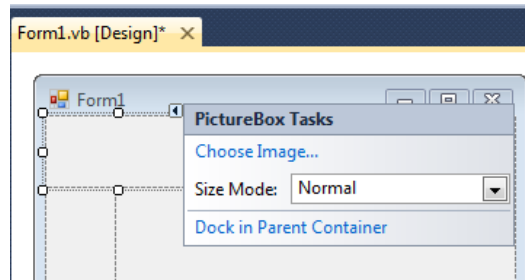
3. اذهب إلى نافذة التصميم و اختر الـ TableLayoutPanel لتجد سهم على يمين العنصر فى الأعلى اضغط عليه ثم اختر Edit Columns and Rows كما فى الشكل:

ستظهر لك نافذة Column and Row Styles، تأكد من التعليم على Column1 و اضبط حجمه بـ 15% و column2 بـ 85% ، و كذلك Row1 بـ 90% و Row2 بـ 10%، ثم اضغط ok



4. اذهب إلى نافذة التصميم و اختر من الـ Toolbox العنصر PictureBox (عن طريق الضغط عليها مرتين) ، سيتم إضافة العنصر في أول خانة على اليسار في الأعلى.

اضغط على السهم الموجود في أعلى العنصر المضاف PictureBox لتظهر لك قائمة اختر Dock in Parent Container



تأكد أن الخاصية dock في نافذة الـ Properties محتواها Fill، ثم اذهب إلى ColumnSpan و غيره إلى 2 ، و غير BorderStyle لـ Fixed3d لإظهار العنصر حتى لو فارغ.



5. أضف checkbox من الـ toolbox عن طريق ضغطتين متتاليتين بالماوس على العنصر لتضاف إلى أول خانة فارغة و هي الخانة السفلى على اليسار (لأن الخانتين العلويتين مشغولتين بالـ picturebox).

قم بتغيير خاصية text ألى "مَدَد".

6. اذهب إلى الـ toolbox مرة أخرى و اختر من مجموعة containers العنصر FlowLayoutPanel لتضاف إلى آخر خانة ، ثم اضغط على السهم و اختر Dock in parent container

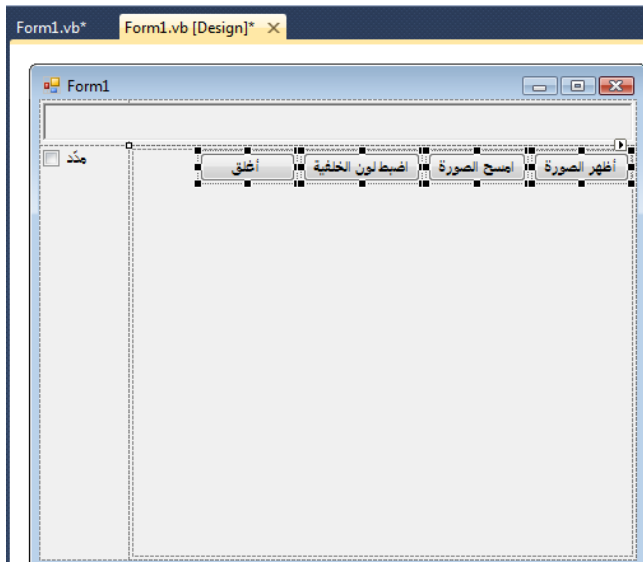
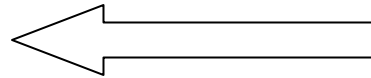
وظيفة FlowLayoutPanel أنها من الممكن أن تحتوى على عناصر أخرى لترتيبها فى صفوف منظمة، و عند تغيير مساحتها يمكنها أن ترتب محتوياتها فى سطور ، سنستخدمها هنا فى احتواء 4 ازرار.



7. لإضافة الأزرار:

- اختر FlowLayoutPanel ثم اذهب إلى toolbox و أضف 4 أزرار ( button1 و button2 و button3 و button4 ) ليتم إضافتهم فى FlowLayoutPanel .
- غير الـ text فى نافذة property لزرار الأول لـ " أظهر الصورة " و الثانى لـ "امسح الصورة" و الثالث لـ "اضبط لون الخلفية" و الرابع " أغلق".
- يمكنك تغيير ترتيب أماكن الأزرار ، اذهب إلى خاصية flowdirection الخاصة بـ FlowLayoutPanel و اختر RightToLeft لتكون محاذاة الأزرار من اليمين إلى اليسار.
- علم بالماوس على زر "أغلق" ، ثم اضغط على زر Ctrl فى لوحة المفاتيح باستمرار ثم علم على بقية الـ 3 أزرار ، ثم اذهب إلى نافذة properties ثم autosize و غيرها لـ True ليتم ضبط حجم الأزرار تلقائياً.

ليكون نتيجة ذلك :





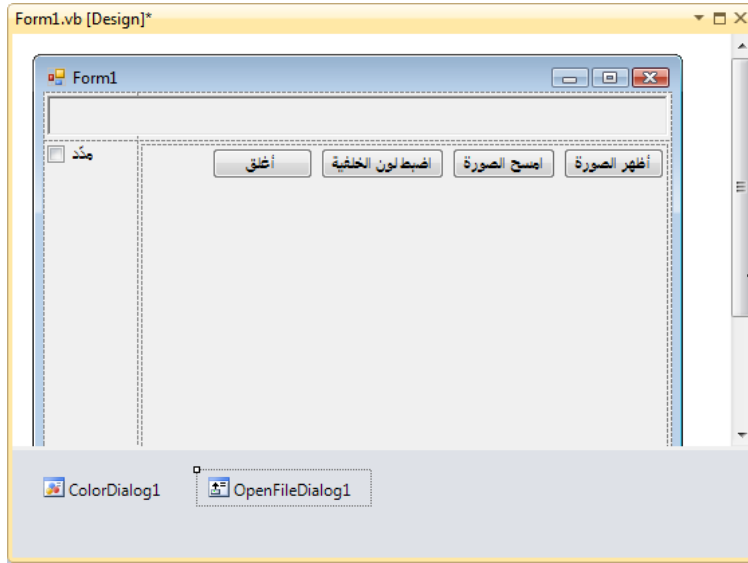
## 3. غير أسماء العناصر

سنبدأ فى عملية تغيير أسماء العناصر الموجودة فى التصميم حتى تسهل علينا تمييزهم فى عملية الكود، سنبدأ فى تغيير أسماء الأزرار ، اختر زر "أغلق" بالماوس ثم اذهب إلى خاصية (Name) و غيرها إلى `closeButton` و قم بالسابق مع زر "أظهر الصورة" ل `showButton` و "امسح الصورة" ل `clearButton` و "اضبط لون الخلفية" ل `backgroundButton` .

## 4. أضف مربعات الحوار

مربعات الحوار DialogBoxes سنستخدمها لفتح ملف الصورة الذى نريد عرضه بالإضافة لاختيار لون الخلفية ، و هذا سنقوم بإنشائه الآن.

1 - اذهب إلى toolbox و اختر من مجموعة Dialogs العنصر و أضف OpenFileDialog و ColorDialog ليظهروا كالاتى:

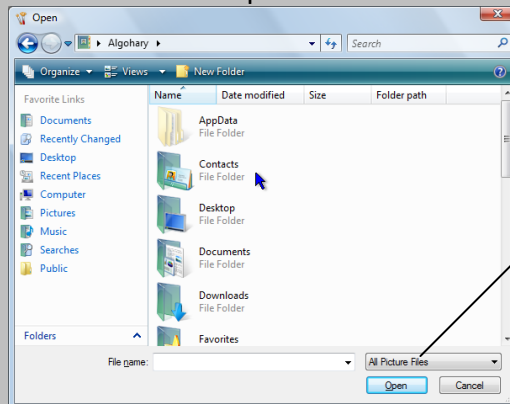


اختر OpenFileDialog1 و فى properties ستجد Filter اكتب السطر التالى فيها:

\*.\*| كل الملفات|.bmp|.png|.png| ملفات PNG|.jpg|.jpg| ملفات JPEG

و فى خاصية Title: اختر ملف صور

كمثال توضيحي على OpenFileDialog اذهب إلى برنامج Paint فى ويندوز و شغله و اذهب إلى قائمة file ثم اختر open



يمثله خاصية  
filter  
فى  
التكويد

## 5. ابدأ في كتابة الكود

هنا سيت كتابة الكود الخاص بالعناصر:

- 1- سنبدأ من showButton ، سنذهب إلى نافذة التصميم و نضغط بالماوس ضغطتين على showButton "أظهر الصورة" لننتقل إلى نافذة التكويد، سنكتب الكود التالي:

```

Form1.vb*
(General) (Declarations)
3 Private Sub showButton_Click() Handles showButton.Click
4
5     ' Show the Open File dialog. If the user clicks OK, load the
6     ' picture that the user chose.
7
8     If OpenFileDialog1.ShowDialog() = DialogResult.OK Then
9
10        PictureBox1.Load(OpenFileDialog1.FileName)
11
12    End If
13
14
15 End Sub
16
17 End Class
18
100 %

```

اي في حالة أن المستخدم اختار ملف من خلال OpenFileDialog و ضغط زر OK سيتم تحميل هذا الملف في مربع PictureBox1.

- 2- اذهب إلى نافذة التصميم و اضغط على زر clearButton "امسح الصورة" ثم اضغط ضغطتين متتاليتين لتذهب إلى نافذة الكود و اكتب:

```

Private Sub clearButton_Click() Handles clearButton.Click
    ' Clear the picture.
    PictureBox1.Image = Nothing
End Sub

```

و معناه أن يتم عرض لاشئ في مربع الصورة عند الضغط على الزر.

- 3 - و في backButton " اضبط لون الخلفية":

```

Private Sub backButton_Click() Handles backButton.Click
    ' Show the color dialog box. If the user clicks OK, change the
    ' PictureBox control's background to the color the user chose.
    If ColorDialog1.ShowDialog() = DialogResult.OK Then
        PictureBox1.BackColor = ColorDialog1.Color
    End If
End Sub

```



يمكنك من اختيار لون خلفية مربع الصورة حسب اختيارك في ColorDialog1.  
4 - و في closeButton "أغلق":

```
Private Sub closeButton_Click() Handles closeButton.Click
    ' Close the form.
    Close()
End Sub
```

5 - و في checkbox :

```
Private Sub CheckBox1_CheckedChanged() Handles CheckBox1.CheckedChanged
    ' If the user selects the Stretch check box, change
    ' the PictureBox's SizeMode property to "Stretch". If the user
    ' clears the check box, change it to "Normal".
    If CheckBox1.Checked Then
        PictureBox1.SizeMode = PictureBoxSizeMode.StretchImage
    Else
        PictureBox1.SizeMode = PictureBoxSizeMode.Normal
    End If
End Sub
```

و وظيفته عند التعليم فيه سيتم تمديد الصورة المعروضة بحجم النافذة.

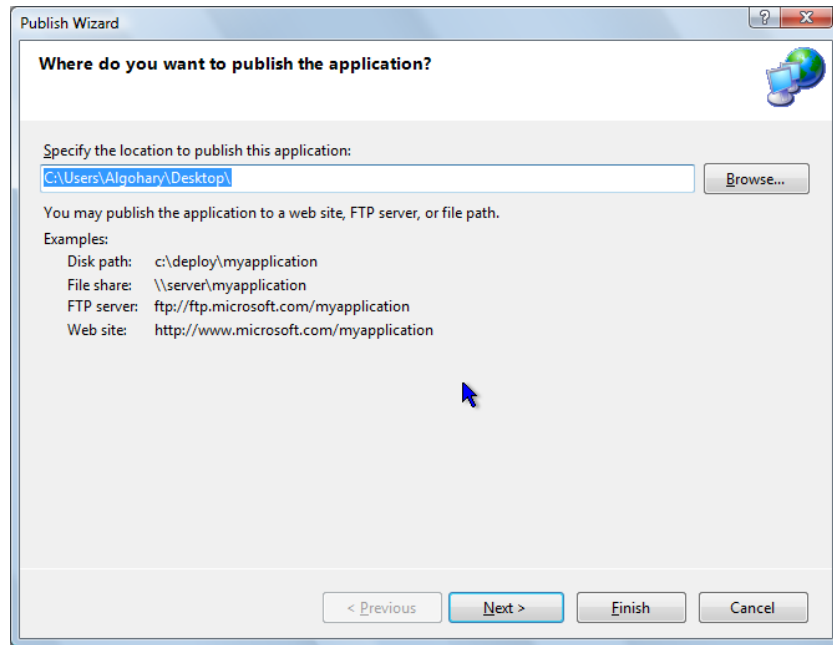
الآن انتهينا من كتابة الكود للبرنامج الأول في الفيچوال بيسك ابدأ في تجريب البرنامج من خلال F5



## 6. لإخراج البرنامج مستقل

فى هذه الخطوة سنستكشف كيفية سنتعلم كيفية صنع برنامج مستقل يمكننا مشاركته مع الآخرين دون الحاجة إلى الفيجوال بيسك.

أسرع طريقة هى بعد الانتهاء من البرنامج نذهب إلى قائمة Build ثم Publish اتظهر لنا نافذة



اكتب المسار الذى تريد فيه نشر البرنامج.

يمكنك أن تغير اسم البرنامج و بعض الاعدادات الأخرى الخاصة بالبرنامج من خلال القائمة  
Project ثم Properties

اسم البرنامج بعد تجميعه

شكل الأيقونة الخاصة بالبرنامج

عنوان و وصف و نسخة و شركة البرنامج

بعد الانتهاء يمكنك الذهاب إلى هذه القائمة لنشر البرنامج. اذهب إلى مسار الملف الذى حددته سيكون هناك جميع ملفات البرنامج

---

# الملحقات

---

الملحقات الخاصة بالدليل

---

[algharism.blogspot.com](http://algharism.blogspot.com)

---



## المراجع:

- كتاب الحاسب الآلى – للصف الأول الثانوى – وزارة التربية و التعليم – مصر طبعة 2003/2002.
- البرمجة بلغة ال Visual Basic – المستوى الأول – هيئة قناة السويس- مصر 2001.
- فيديوهات لتعليم Visual Basic Express 2005 موقع مايكروسوفت -
- 2010 - <http://msdn.microsoft.com/en-us/beginner/bb308891.aspx>
- موقع MSDN من مايكروسوفت -- <http://msdn.microsoft.com/en-us/library/dd492135%28VS.100%29.aspx> - 2010.
- وزارة التربية و التعليم – محاضرة أساسيات البرمجة - <http://knowledge.moe.gov.eg/NR/ronlyres/327B0AF2-677C-459F-B6C7-6F2A6708E9DD/15719/1672009.ppt> - 2010.



## Toolbar Buttons

	New Project		Start Debugging
	Open File		Break All
	Add New Item		Stop Debugging
	Save Selected Item		Step Into
	Save All		Step Over
	Cut		Step Out
	Copy		Solution Explorer
	Paste		Properties Window
	File		Object Browser
	Comment out selected lines		Toolbox
	Uncomment selected lines		Error List
	Undo		
	Redo		

## File Commands

Project	
	Add Windows Form...
	Add User Control...
	Add Module...
	Add Class...
	Add New Item...
	Add Existing Item...
	Show All Files
	Add Reference...
	Add Web Reference...
	WindowsApplication1 Properties...

## Edit Commands

Edit			
	Undo		Ctrl+Z
	Redo		Shift+Alt+Escape
	Cut		Ctrl+X
	Copy		Ctrl+C
	Paste		Ctrl+V
	Delete		Del
	Select All		Ctrl+A
	Find Symbol		Alt+F12
	Quick Find		Ctrl+F
	Quick Replace		Ctrl+H
	Go To...		
	Insert File As Text...		
	IntelliSense		
	Next Method		Ctrl+FgDn
	Previous Method		Ctrl+FgUp

## View Commands

View			
	Code		F7
	Designer		Shift+F7
	Database Explorer		Ctrl+Alt+S
	Solution Explorer		Ctrl+R
	Object Browser		F2
	Error List		Ctrl+W, Ctrl+E
	Properties Window		F4
	Toolbox		Ctrl+Alt+X
	Other Windows		
	Toolbars		
	Full Screen		Shift+Alt+Enter
	Property Pages		Shift+HF4

## Project Commands

File			
	New Project...		Ctrl+Shift+N
	Open Project...		Ctrl+Shift+O
	Open File...		Ctrl+O
	Close		
	Close Solution		
	Save ConsoleApplication1		Ctrl+S
	Save ConsoleApplication1 &s...		
	Save All		Ctrl+Shift+S
	Export Template...		
	Page Setup...		
	Print...		Ctrl+P
	Recent Projects		
	Exit		

## Debug Commands

Debug			
	Start Debugging		F5
	Step Into		F8
	Step Over		Shift+F8
	Exceptions...		Ctrl+Alt+E
	Toggle Breakpoint		F9
	Delete All Breakpoints		Ctrl+Shift+F9

## Window Commands

Window			
	New Window		Shift
	Floating		
	Dockable		
	Tabbed Document		
	Auto Hide		
	Hide		
	Auto Hide All		
	New Horizontal Tab Group		
	New Vertical Tab Group		
	Close All Documents		
	Reset Window Layout		
	1 Form1.vb*		
	2 Form1.vb [Design]*		
	3 Start Page		
	Windows...		



### Toolbar Buttons

	New Project		Start Debugging
	Open File		Break All
	Add New Item		Stop Debugging
	Save Selected Item		Step Into
	Save All		Step Over
	Cut		Step Out
	Copy		Solution Explorer
	Paste		Properties Window
	Find		Object Browser
	Comment out selected lines		Toolbox
	Uncomment selected lines		Error List
	Undo		
	Redo		

### File Commands

Project	
	Add Windows Form...
	Add User Control...
	Add Module...
	Add Class...
	Add New Item...      Ctrl+Shift+A
	Add Existing Item...      Ctrl+D
	Show All Files
	Add Reference...
	Add Web Reference...
	WindowsApplication1 Properties...

### Edit Commands

Edit	
	Undo      Ctrl+Z
	Redo      Shift+Alt+Escape
	Cut      Ctrl+X
	Copy      Ctrl+C
	Paste      Ctrl+V
	Delete      Del
	Select All      Ctrl+A
	Find Symbol      Alt+F12
	Quick Find      Ctrl+F
	Quick Replace      Ctrl+H
	Go To...
	Insert File As Text...
	IntelliSense
	Next Method      Ctrl+FgDn
	Previous Method      Ctrl+FgUp

### View Commands

View	
	Code      F7
	Designer      Shift+F7
	Database Explorer      Ctrl+Alt+S
	Solution Explorer      Ctrl+R
	Object Browser      F2
	Error List      Ctrl+W, Ctrl+E
	Properties Window      F4
	Toolbox      Ctrl+Alt+X
	Other Windows
	Toolbars
	Full Screen      Shift+Alt+Enter
	Property Pages      Shift+F4

### Project Commands

File	
	New Project...      Ctrl+Shift+N
	Open Project...
	Open File...      Ctrl+O
	Close
	Close Solution
	Save ConsoleApplication1      Ctrl+S
	Save ConsoleApplication1 As...
	Save All      Ctrl+Shift+S
	Export Template...
	Page Setup...
	Print...      Ctrl+P
	Recent Projects
	Exit

### Debug Commands

Debug	
	Start Debugging      F5
	Step Into      F8
	Step Over      Shift+F8
	Exceptions...
	Toggle Breakpoint      Ctrl+Alt+E
	Delete All Breakpoints      Ctrl+Shift+F9

### Window Commands

Window	
	New Window
	Split
	Floating
	Dockable
	Tabbed Document
	Auto Hide
	Hide
	Auto Hide All
	New Horizontal Tab Group
	New Vertical Tab Group
	Close All Documents
	Reset Window Layout
	1 Form1.vb*
	2 Form1.vb [Design]*
	3 Start Page
	Windows...

## استخدام الخوارزم في حل المشكلات:

- (١) تحديد المشكلة.
- (٢) تحديد المخرجات.
- (٣) تحديد المدخلات.
- (٤) اهمال البيانات الغير مفيدة في حل المشكلة.
- (٥) تحديد خطوات الحل (العمليات).
- (٦) كتابة Algorithm الحل.

## مثال: حل معادلة من الدرجة الثانية:

$$a x^2 + b x + c = 0$$

معادلة من الدرجة الثانية

**المطلوب:**

تحديد خطوات حل المعادلة باستخدام الخوارزم

علما بأن جذري المعادلة:

$$x_{1,2} = \frac{-b \pm (b^2 - 4ac)^{.5}}{(2 a)}$$



**الحل:**

- (١) تحديد المشكلة: ايجاد جذري المعادلة.
- (٢) تحديد المخرجات: طباعة  $X1, X2$
- (٣) تحديد المدخلات:  $a, b, c$ .
- (٤) إهمال البيانات الغير مفيدة في حل المشكلة:  
لا يوجد.

**تابع الحل:**

- (٥) تحديد خطوات الحل:
- حساب قيمة المميز أسفل الجذر التربيعي.  
$$M = b^2 - 4 a c$$
- اختبار قيمة  $M$ :
- إذا كانت سالبة إذا المعادلة ليس لها حل.
- إذا كانت تساوي صفر إذا يوجد جذر واحد.
- إذا كانت القيمة موجبه إذا هناك جذرين للمعادلة.

## تابع الحل:

(٦) كتابة Algorithm الحل:

- ادخال قيم المعاملات  $a, b, c$ .
- حساب المميز  $m$ .
- اختبار قيمته.
- حساب قيمة جذري المعادلة  $x_1, x_2$ .
- طباعة الناتج  $x_1, x_2$ .

## أشهر أساليب حل المشكلات:

هناك طرائق عديدة نذكر منها:

■ نموذج IPO.

■ خرائط التدفق.

■ سودوكود.

## نموذج IPO:

هو جدول أو نموذج يضم تحليل أي مشكلة إلى مدخلات ومعالجة ومخرجات.

مثال: حل معادلة من الدرجة الثانية:

$$a x^2 + b x + c = 0$$

معادلة من الدرجة الثانية

المطلوب:

إعداد نموذج IPO الذي يوضح المدخلات والمعالجة والمخرجات.  
علما بأن جذري المعادلة:

$$x_{1,2} = \frac{-b \pm (b^2 - 4ac)^{.5}}{2 a}$$



المدخلات (Input)	المعالجة (Processing)	المخرجات (Output)
a, b, c	<p>حساب قيمة المميز: <math>M = b^2 - 4ac</math></p> <p>هل <math>M</math> أقل من صفر إذا لا يوجد حل للمعادلة.</p> <p>هل <math>M</math> تساوي صفر إذا هناك جذر واحد للمعادلة: <math>X = -b / (2a)</math></p> <p>هل <math>M</math> أكبر من صفر إذا يوجد للمعادلة جذرين:</p> <p><math>X1 = (-b + (b^2-4ac)^{0.5}) / (2a)</math></p> <p><math>X2 = (-b - (b^2-4ac)^{0.5}) / (2a)</math></p>	<p>طباعة الناتج وقد يكون:</p> <p>-رسالة "لا يوجد حل".</p> <p>- طباعة <math>X</math>.</p> <p>-طباعة الجذرين <math>X1, x2</math></p>

## خرائط التدفق: Flow Chart

هي عبارة عن تمثيل تخطيطي، يوضح بسهولة ترتيب خطوات حل المشكلة، بدءاً من إدخال البيانات، ثم تحديد العمليات الحسابية والمنطقية، وصولاً للمخرجات التي تمثل حل المشكلة.



## بعض الأشكال المستخدمة في رسم خرائط التدفق:



الرمز الطرفي Terminal



رمز الإدخال والإخراج Input / Output



رمز العمليات Process



رمز اتخاذ القرار Decision

## تتابع بعض الأشكال المستخدمة في رسم خرائط التدفق:



خط الاتجاه Flow Line



رمز الربط أو الاتصال Reference



رمز إضافة تعليق Annotation

### الاعتبارات الواجب مراعاتها عند رسم خرائط التدفق:

- بساطة ووضوح الخريطة لسهولة تتبع خطواتها.
- الاتجاه الافتراضي لأشكال الخريطة من اليسار إلى اليمين ومن أعلى إلى أسفل.
- رمز العمليات يخرج منه خط اتجاه واحد فقط.
- شكل اتخاذ القرار يدخل له خط اتجاه واحد ويخرج منه خطان (نعم / لا).

### تابع الاعتبارات الواجب مراعاتها عند رسم خرائط التدفق:

- الرمز الطرفي للبداية له خط اتجاه واحد خارج والعكس في الرمز الطرفي للنهاية.
- يفضل استخدام رمز اضافة تعليق مع أشكال الخريطة التي تحتاج توضيح.
- يستخدم رمز الربط أو الاتصال إذا كانت خريطة التدفق كبيرة وتحتاج أكثر من صفحة.
- بعد رسم خريطة التدفق يفضل تتبع جميع مساراتها واختبارها بقيم افتراضية معروف نتائجها.

## مميزات استخدام خرائط التدفق:

- تمثل ضرورة قبل كتابة البرامج الكبيرة.
- تمثل أحد أشكال توثيق البرنامج.
- تضع تصورا كاملا لحل المشكلة وتساعد في تتبع مسارها.
- تساعد في عدم تكرار أجزاء معينة في الرسم أو البرمجة.
- الأشكال المستخدمة في رسم خرائط التدفق لها مدلول واحد لدى جميع المتعاملين معها.
- تساعد في تصحيح الأخطاء بسهولة.
- تساعد في تطوير وصيانة البرامج.

## عيوب استخدام خرائط التدفق:

- قد تبدو الخريطة معقدة للمشكلات الكبيرة.
- بعض التعديلات في البرنامج قد يؤدي لإعادة رسم الخريطة.
- أحيانا تشكل نسخ خريطة التدفق صعوبة كبيرة.
- الوقوع في بعض التفاصيل التي تبعدنا عن الحل.



## مثال: حل معادلة من الدرجة الثانية:

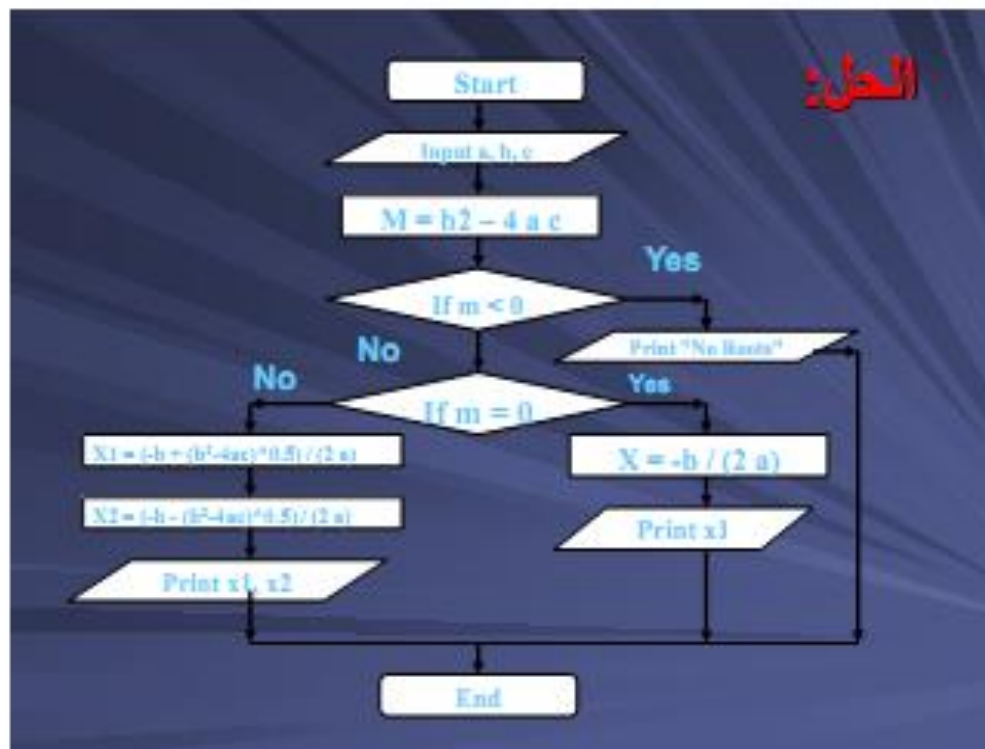
$$a x^2 + b x + c = 0$$

معادلة من الدرجة الثانية

**المطلوب:**

رسم خريطة التدفق التي توضح خطوات حل معادلة من الدرجة الثانية. علما بأن جذري المعادلة:

$$x_{1,2} = \frac{-b \pm (b^2 - 4ac)^{.5}}{2 a}$$

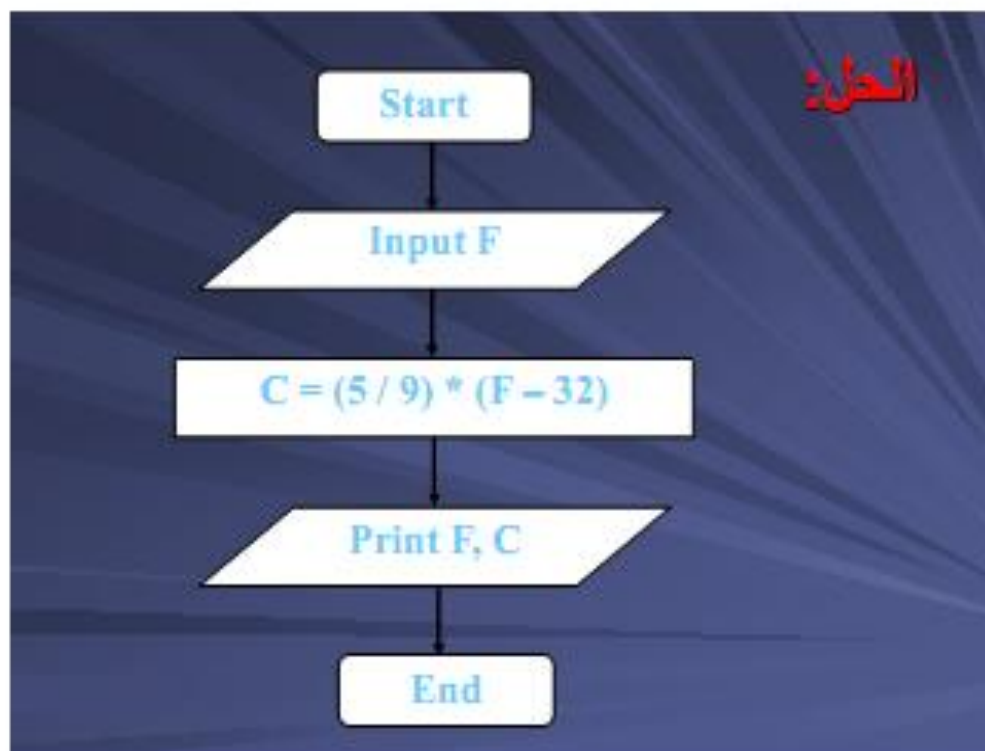


**مثال: تحويل درجة الحرارة من فهرنهايت إلى مئوية:**

ارسم خريطة التدفق التي ندخل لها درجة الحرارة بالفهرنهايت فيتم تحويلها إلى ما يناظرها بدرجة الحرارة المئوية.

علما بأن قانون التحويل هي:

$$C = (5 / 9) * (F - 32)$$

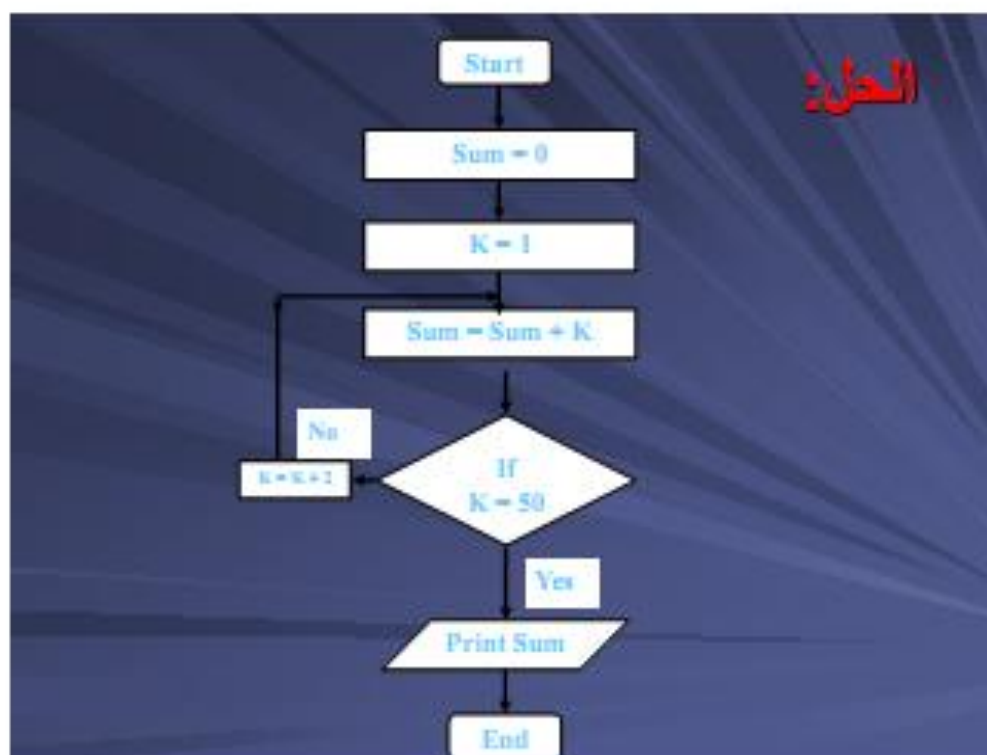


## مثال: حساب مجموع بعض الأعداد:

ارسم خريطة التدفق لحساب حاصل جمع أول ٢٥ عدد فردي من الأعداد الطبيعية.

وذلك كما يلي:

$$\text{Sum} = 1 + 3 + 5 + \dots + 49$$



## سودوكود : Pseudocode

طريقة لعرض خطوات حل  
المشكلة بلغة بسيطة مختصرة  
قريبة من لغات البرمجة.

## مكونات سودوكود:

- الكلمات: وهي تمثل الأفعال المطلوب تنفيذها مثل: (Calculate - Print - Read - Stop).
- الجمل: تشرح المطلوب القيام به مثل:  
**Calculate Sum of two Numbers**
- مقاطع: والتي تمثل وحدة واحدة تضم عدة جمل مثل: **Block IF**.



## قواعد وقيود استخدام سودوكود:

- اختيار أسماء ذات معنى واضح للمتغيرات بحيث تعبر عن محتواها.
- كتابة الأوامر في صورة خطوات سهلة وبسيطة وواضحة.

## قواعد وقيود استخدام سودوكود:

- تتميز "سودوكود" بأنها:
  - سهلة الفهم.
  - لا تستخدم رموزا خاصة.
  - لا تخضع لقواعد صعبة أو معقدة.
  - سهلة التحويل لبرنامج بأي لغة برمجة.
- يؤخذ على "سودوكود":
  - قد تكون طويلة بعض الشيء خاصة في المشاكل المعقدة.



## مثال: حل معادلة من الدرجة الثانية:

$$a x^2 + b x + c = 0$$

معادلة من الدرجة الثانية

**المطلوب:**

كتابة سودوكود يوضح خطوات حل معادلة من الدرجة الثانية.

علما بأن جذري المعادلة:

$$x_{1,2} = \frac{-b \pm (b^2 - 4ac)^{.5}}{2 a}$$

## الحل:

- (1) Input Values: read a, b, c
- (2) Calculate m, Where it given by:  
calculate  $m = b^2 - 4 a c$
- (3) Detect m value:
  - If m less than Zero then  
print "No Roots"
  - Else if m Equal 0 then  
calculate  $x = - b / (2 a)$   
Print " x1 = x2 = ", x
  - else  
calculate  $x_1 = (-b + (b^2 - 4ac)^{.5}) / (2 a)$   
 $x_2 = (-b - (b^2 - 4ac)^{.5}) / (2 a)$   
Print x1, x2
  - end if
  - end if
- (4) Stop processing

بعض خدمات المدونة:

<a href="http://algoharism.blogspot.com/search/label/كتب">http://algoharism.blogspot.com/search/label/كتب</a>	كتب	
<a href="http://store.ovi.com/content/94223">http://store.ovi.com/content/94223</a>	تطبيق نوكيا	
<a href="http://twitter.com/algoharism">http://twitter.com/algoharism</a>	تغريدات Twitter	
<a href="http://algoharism.blogspot.com/search/label/فيديو">http://algoharism.blogspot.com/search/label/فيديو</a>	قناة فيديو	
<a href="http://www.facebook.com/algoharism">http://www.facebook.com/algoharism</a>	صفحة الفييس بوك	
<a href="http://algoharism.blogspot.com/search/label/راديو">http://algoharism.blogspot.com/search/label/راديو</a>	اسمع الراديو	