

:

: •

: •

: •

: -



: -

: •

: -

: -





:

.A

-

:

-

:

-

:



:

**.B**

:

:

•

.I

.

.

.II

.III

.

/

:

•

:

•

.

:

•

-

."Fat Client"

-

.

-

.

:

**.C**

:

•

:

. ( - )

Web

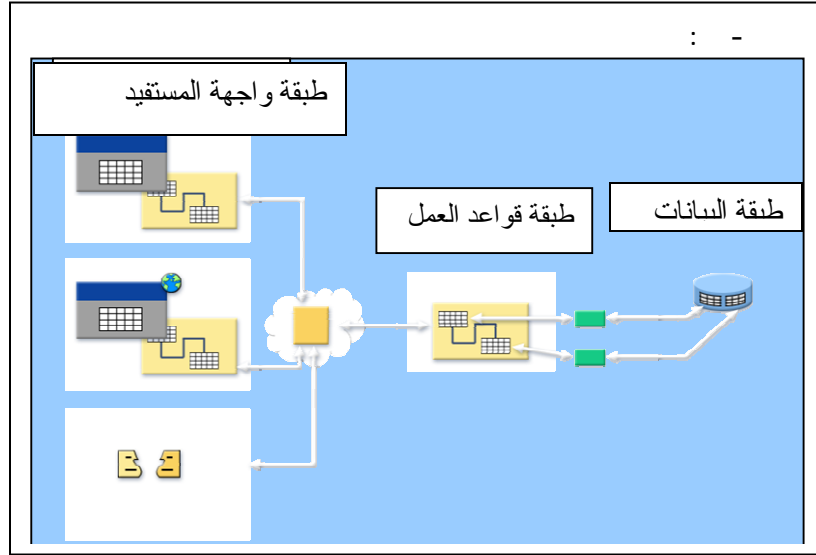
:

•

=

:

•



.D

:

•

:

•

:

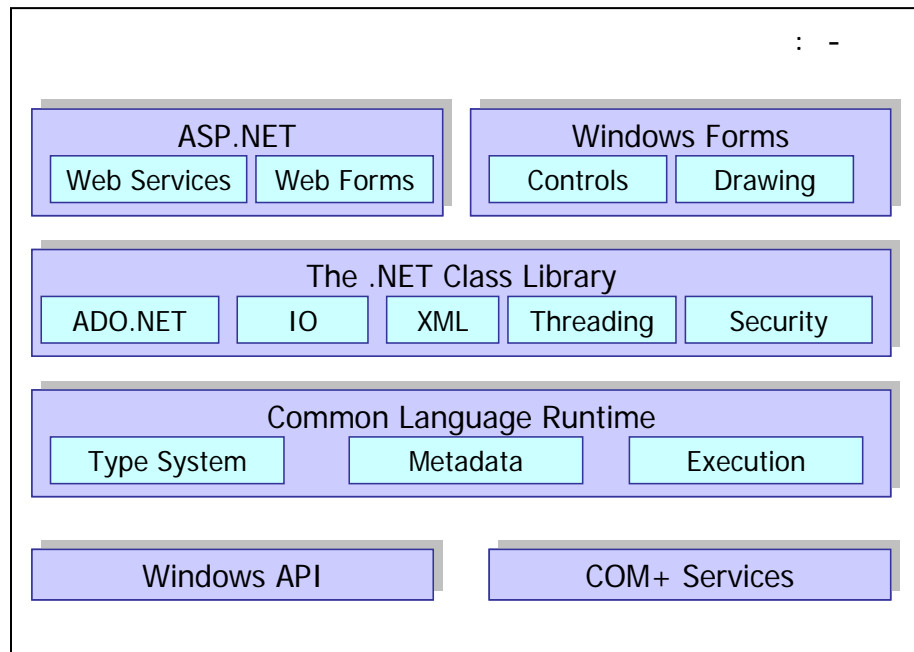
•

-

-

:  
( - ) (CLR, FCL, ASP, WinForms) -  
.vs.net,office win.net -  
-  
-

:  
**Common Language Runtime** .I  
**Common Language Specification** .II



# (CLR) .I

CLR CLR

CLR CLR VB,C#,C++,Delphi,COBOL

:

(

: - Debugging- (

.DLL Hell (

(

.Thread (

# CLS .II

CLR CLR

-Integers,Long- CLS

CLS

: .CLR

-

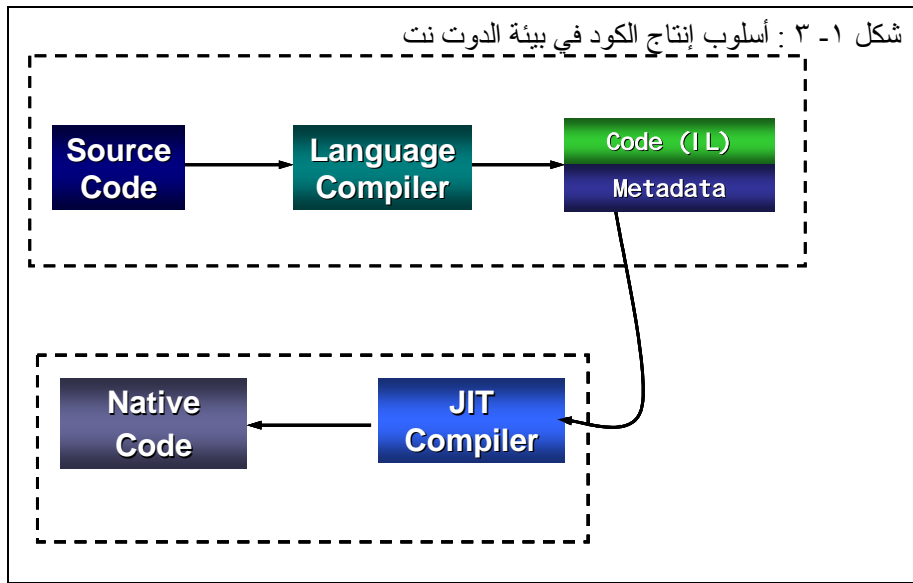
-

-

-







Type Library

IDL

**JIT**

IL

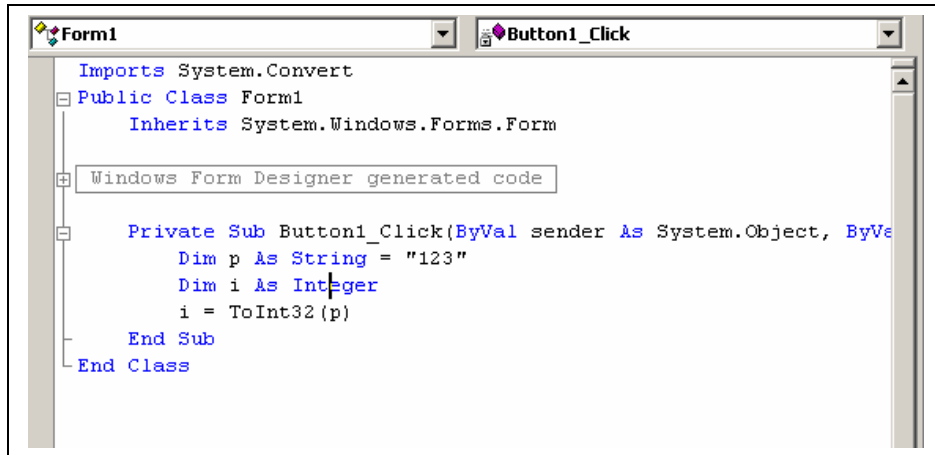
IL

**NameSpaces**

NameSpaces

.(System.IO)

:  
: Imports -  
Imports



```
Imports System.Convert  
Public Class Form1  
    Inherits System.Windows.Forms.Form  
    Windows Form Designer generated code  
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click  
        Dim p As String = "123"  
        Dim i As Integer  
        i =.ToInt32(p)  
    End Sub  
End Class
```

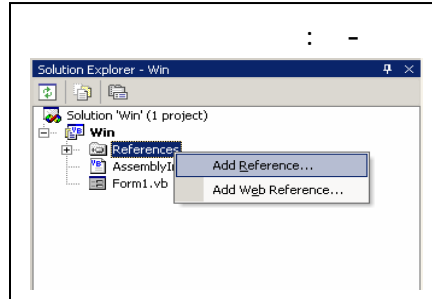
: - -  
System.Convert .I  
 .ToInt32 .II  
 . System.Convert  
System.Convert .III  
: ToInt32  
System.convert.ToInt32



References

.I

( - )

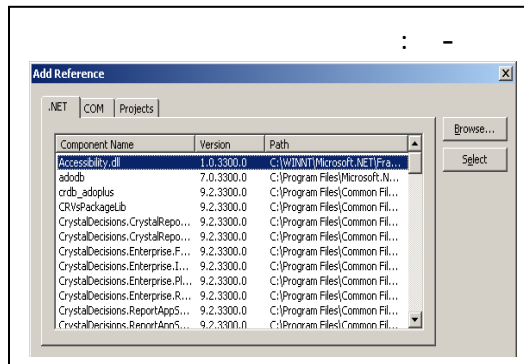


( - )

Add Reference

.II

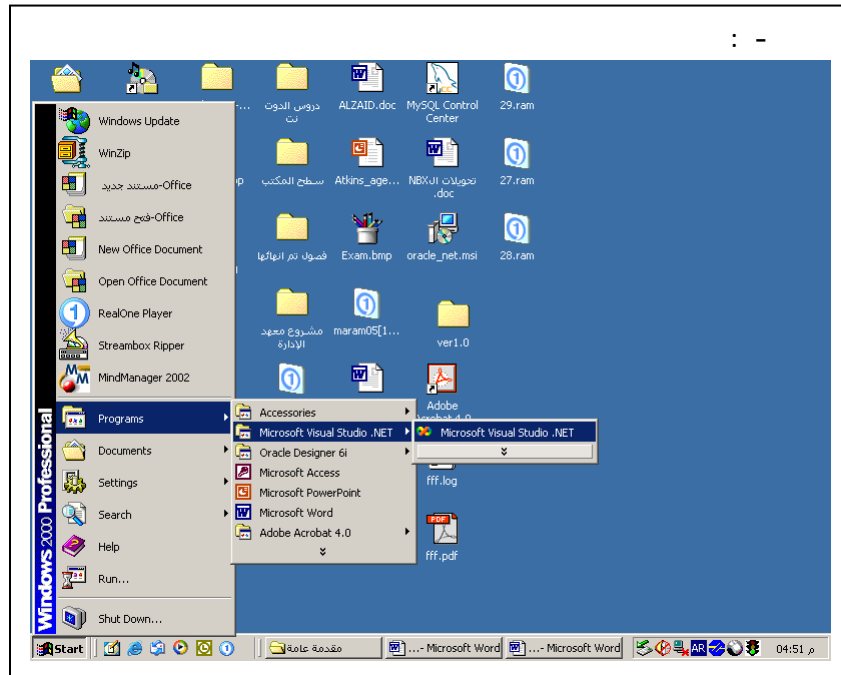
.III

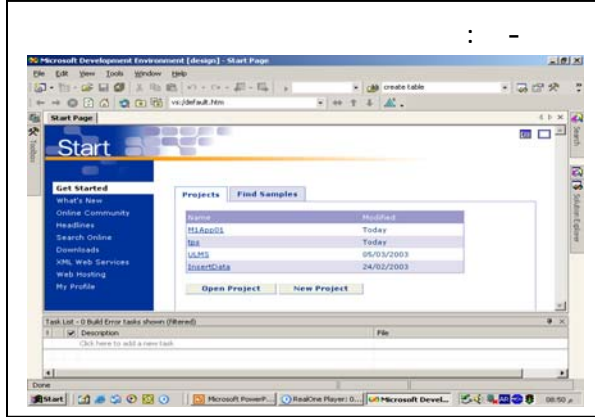


queues, linked lists, hash	<b>System.Collections</b>
.sorted lists	
	<b>System.Data</b>
	<b>System.Drawing</b>
	<b>System.IO</b>
	<b>System.XML</b>

## Visual Studio.Net

- .Windows Start -
- .Programs -
- Microsoft Visual Studio.Net -
- .( - ) Microsoft Visual Studio -

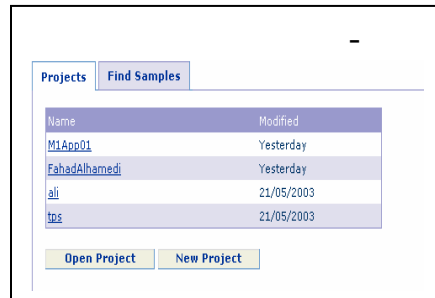




Projects

.I

.( - )



: .II

.( - )

Projects Find Samples

Samples Profile: Filter by:  Keyword  Type

Visual Basic Developer Tier Go

Help Samples

[Duwamish 7.0: Visual Studio Samples: Duwamish 7.0](#)

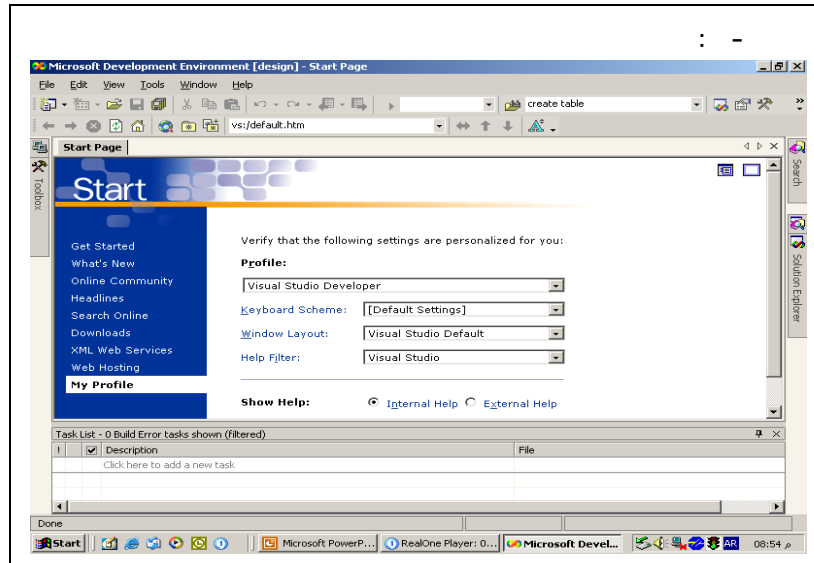
: -

## My Profile

.( - )

:

( - )



	WindowsApplication
	ClassLibrary
	Windows Control Library

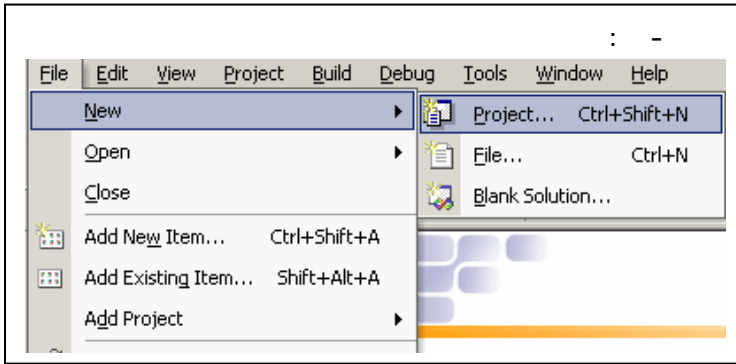
Web	ASP.NET Web Application
.XML Web .XML	ASP.NET Web Service
.DOS	Consol Application
	Windows Service
	Web Control Library



( - )

File → New → Project

-



:

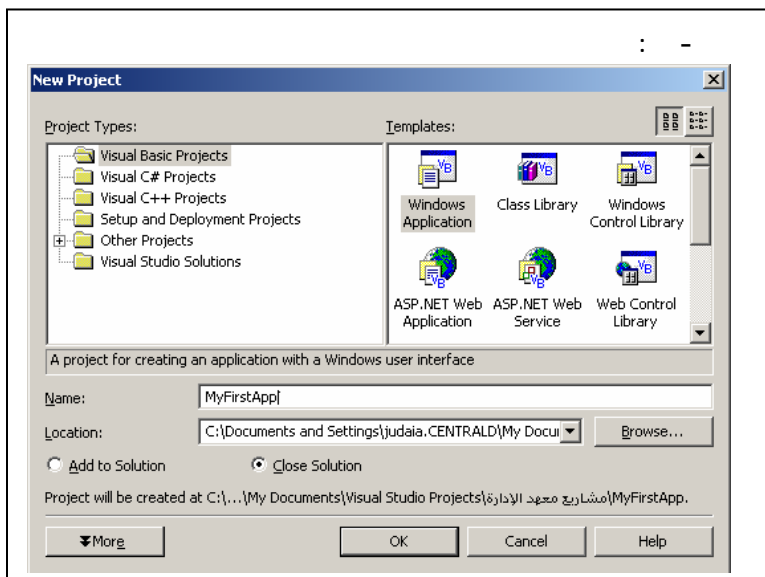
( - )

-

- 
- 
- 

## Windows Application

## MyFirstApplication



Form1.vb

MyFirstAppl

vb



C#

.

.C Sharp

CS

Project Window

- ( - )

-

Properties "

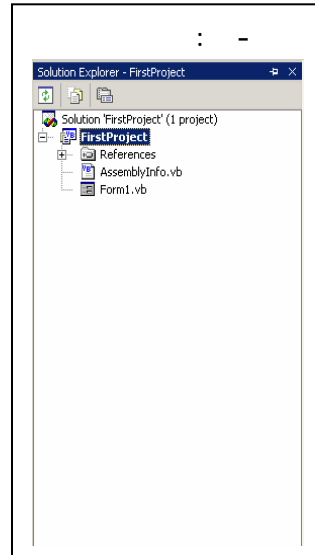
"

Set As StartUp Object "

"

Form1.vb

( - )



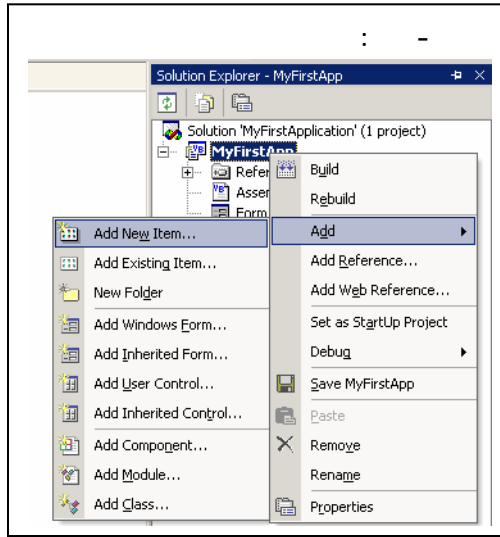
Ctrl+Alt+L

View → Solution Explorer

( - )

Add New Item

Add

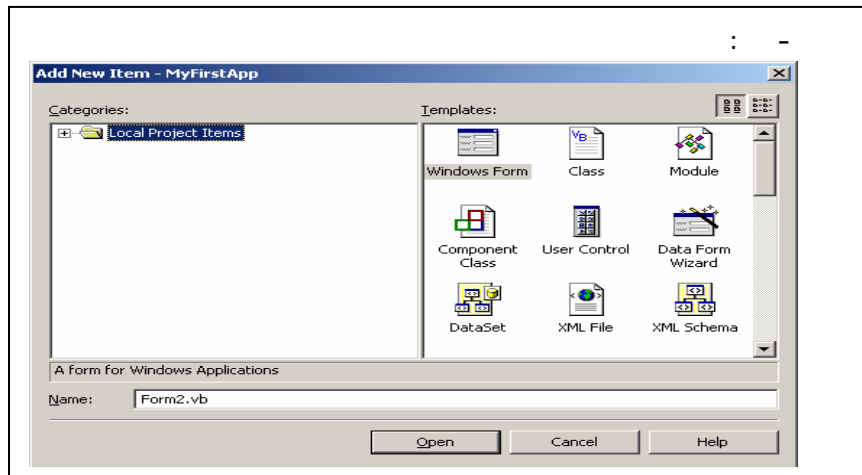


( - )

HTML

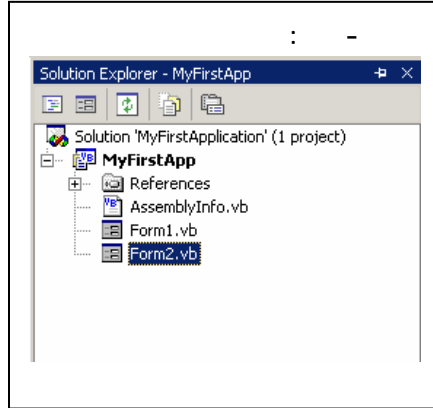
Window Form

Open



.( - )

:

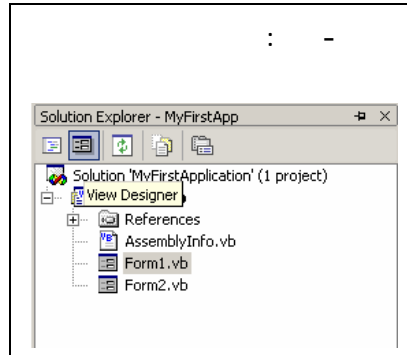


.I

.II

( - )

View Designer Form

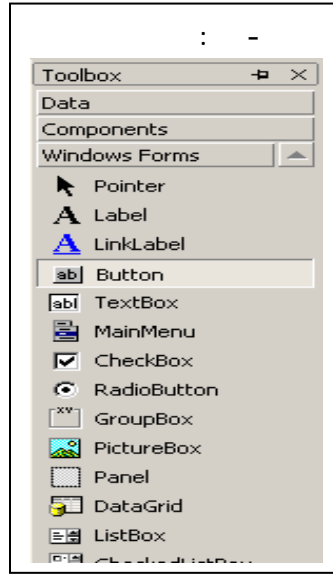


Exclude From Project



.Remove " "

. ( - )

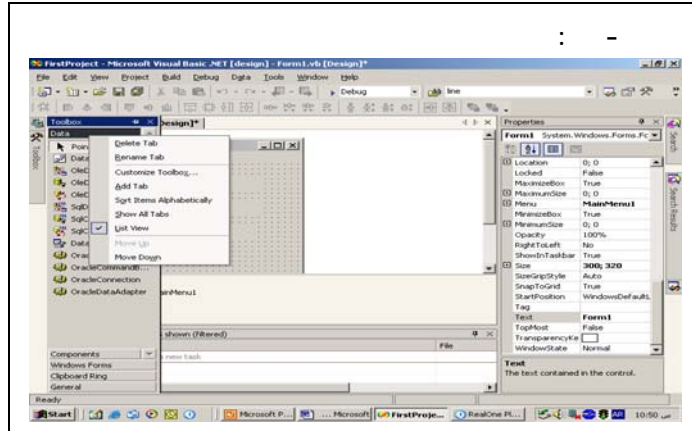


. ( - )

: -Tab-

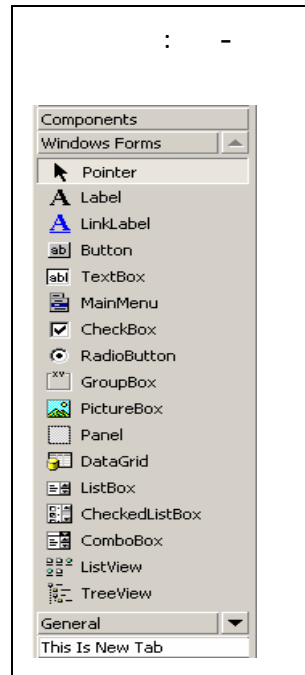
( - )

Add Tab



)( - )

( This Is New Tab



:

**:Crystal Reports**

**:Data "**

**:Windows Forms "**

**:Web Forms "**

Web Control

**:HTML "**



.( )

:

:



:

:

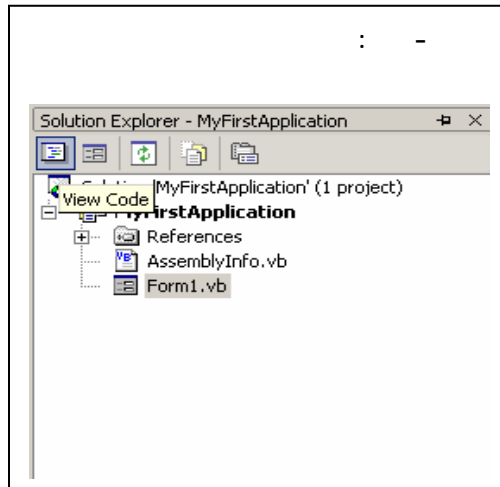
:

-

:

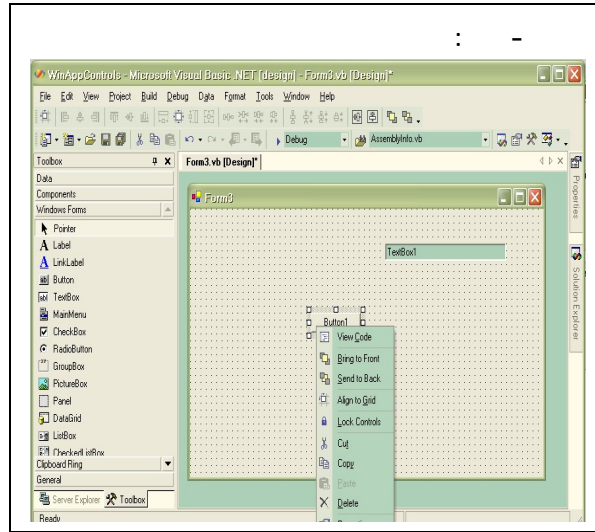
-

( - )

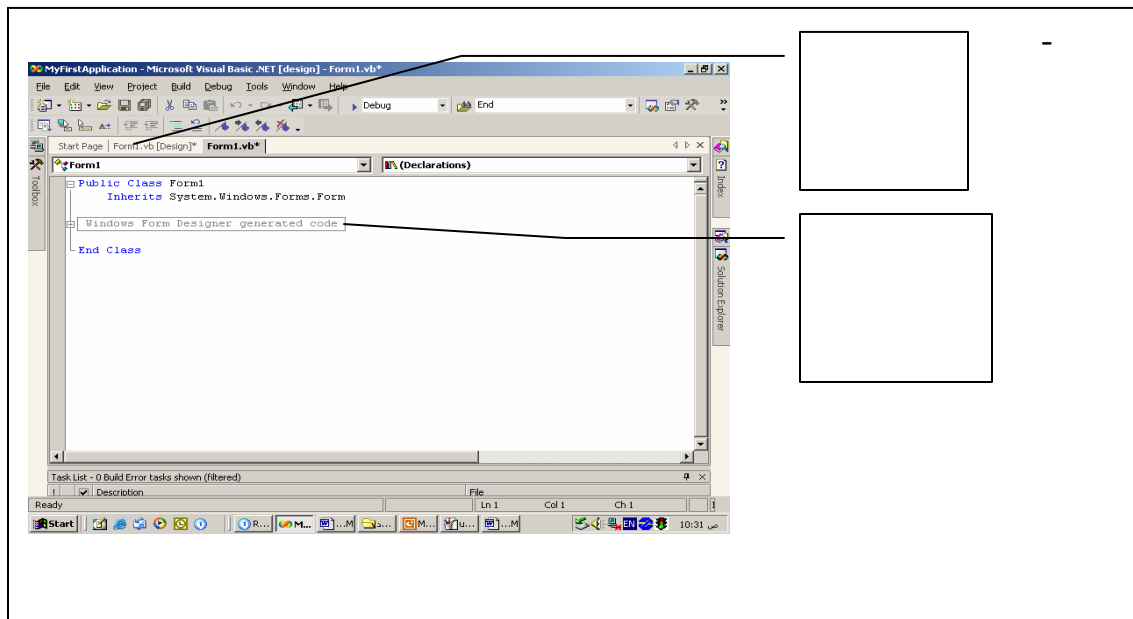


# ViewCode

( - )



( - )



- .I
- .II
- .III
- .IV
- .V

**Tool – Option – Text Editor - General – All Language – Line Numbers**

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal
Dim ControlVar As Control
For Each ControlVar In Me.Controls
If TypeOf ControlVar Is TextBox Then
If Trim(ControlVar.Text) = "" Then
erReq.SetError(ControlVar, "بي إدخال البيانات")
Else
erReq.SetError(ControlVar, "")
End If
End If
Next
End Sub
    
```

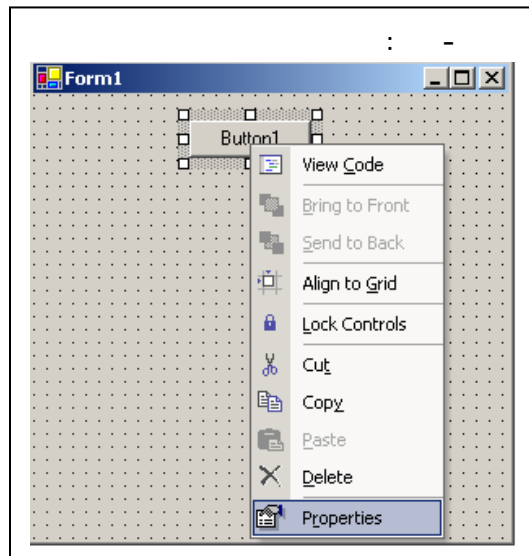
# Properties



-Property Window- ( - )

Properties

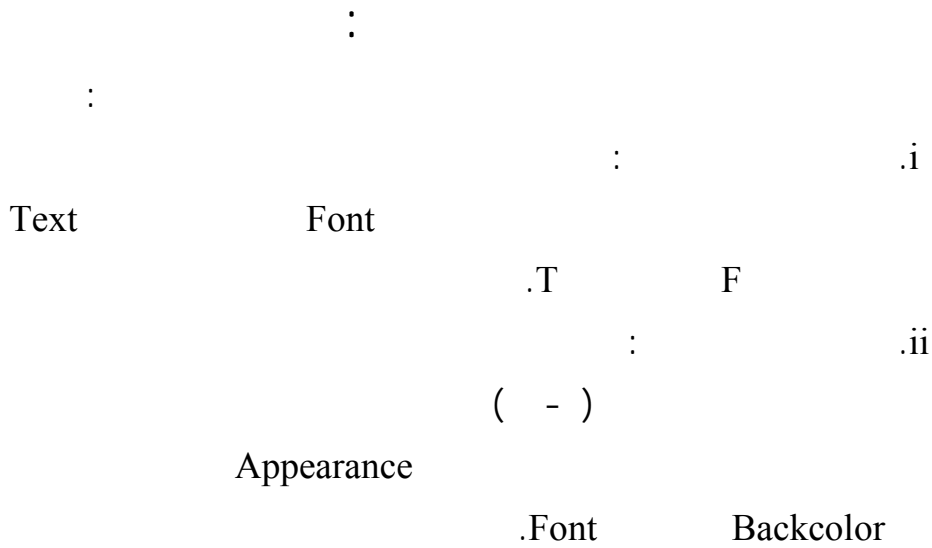
( - )

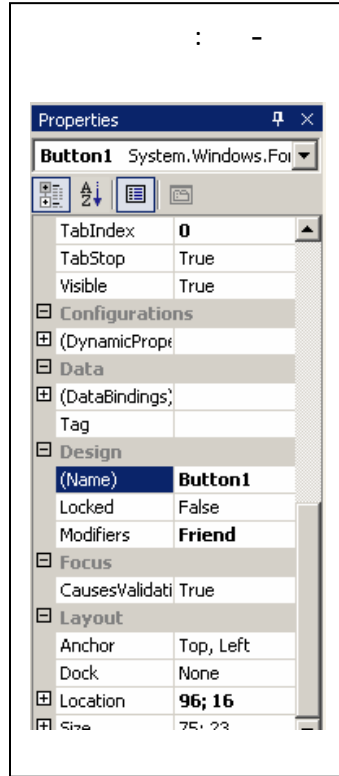


View → Property Window



.F4



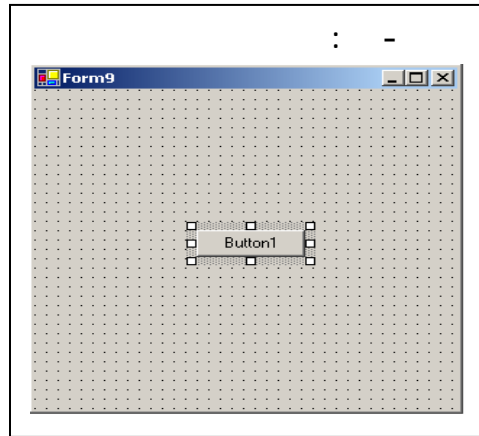


Properties

Text

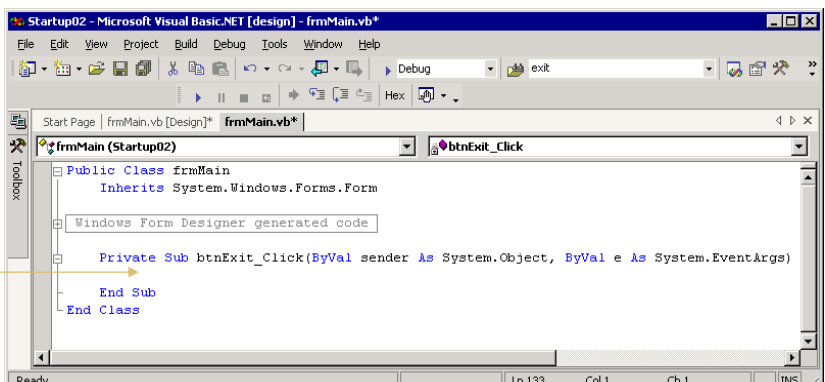
# Events

( - )



( - )

تقوم بيئة الدوت نت  
بانشاء معلومات  
الحدث

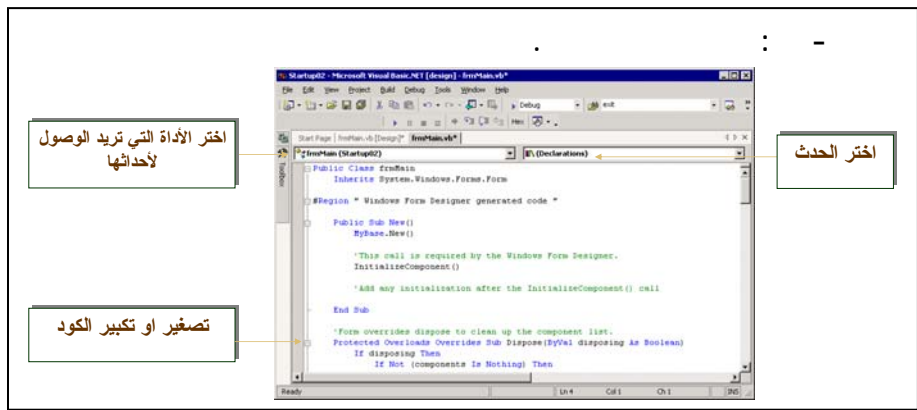
A screenshot of the Visual Basic .NET IDE. The code editor shows the following code:

```
Public Class frmMain
    Inherits System.Windows.Forms.Form

    Windows Form Designer generated code

    Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    End Sub
End Class
```

The IDE title bar shows 'Startup02 - Microsoft Visual Basic.NET [design] - frmMain.vb\*'. The menu bar includes File, Edit, View, Project, Build, Debug, Tools, Window, and Help. The status bar at the bottom shows 'Ready', 'Ln 133', 'Col 1', 'Ch 1', and 'INS'. A callout box on the left points to the 'btnExit\_Click' event handler.



End Class

Public Class Form1

:

.I

-Class-

Private Sub Button1\_Click

:

.II

" "

VB.NET

Horizontal ScrollBar

Underscore

VB.NET

.Syntax





Windows Form Designer generated code : -III

. + :

	العبارة
WinForms.Form Form1 WinForms.Form	Inherits System.WinForms.Form
Form_Initialize . . . .InitializeComponent()	Public Sub New() MyBase.New()  'This call is required by the Windows Form Designer. InitializeComponent()  'Add any initialization after the InitializeComponent() call  End Sub
Form_Terminate	Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean) If disposing Then If Not (components Is Nothing) Then components.Dispose() End If End If MyBase.Dispose(disposing) End Sub
	Private Sub InitializeComponent() Me.Button1 = New System.Windows.Forms.Button() إلى نهاية هذا الإجراء End Sub

--	--

--

```
Dim a As String  
Dim b As Integer  
ctyp(a, b)
```

```
Dim a As Integer  
If a > 10 Then  
    a = a + 10
```

.If end if



.Try Catch



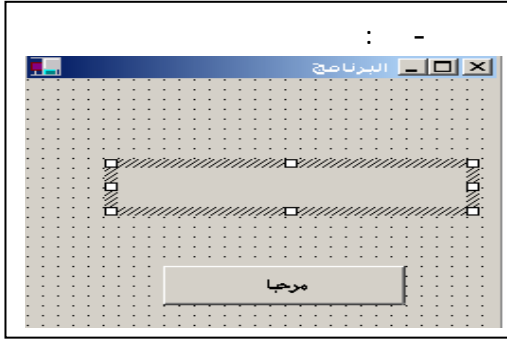
```
Dim a As Integer
a = 5
Do While a < 5
    a = a - 5
Loop
```




vb.net :

":

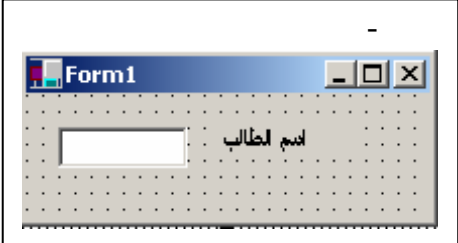
:

Form1 -	-
.( - ) 	-
Edit → Advance → Word Wrapping	-

:	-																					
<table border="1"> <tr style="background-color: #000080; color: white;"> <td></td> <td></td> <td></td> </tr> <tr> <td>CmdWelcome</td> <td>Name</td> <td>Button1</td> </tr> <tr> <td></td> <td>Text</td> <td></td> </tr> <tr> <td>LblHello</td> <td>Name</td> <td>Label1</td> </tr> <tr> <td></td> <td>Text</td> <td></td> </tr> <tr> <td></td> <td>Text</td> <td>Form1</td> </tr> <tr> <td>Yes</td> <td>RightToLeft</td> <td></td> </tr> </table>				CmdWelcome	Name	Button1		Text		LblHello	Name	Label1		Text			Text	Form1	Yes	RightToLeft		
CmdWelcome	Name	Button1																				
	Text																					
LblHello	Name	Label1																				
	Text																					
	Text	Form1																				
Yes	RightToLeft																					
.cmdWelcome  <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre> Private Sub CmdWelcome_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CmdWelcome.Click     End Sub </pre> </div> lblHello.Text = "	-  Text  -																					

Form1.vb	-	-
Form1.vb[Design]*		
Debug	F5	-
	.Start	- Click
	-	-
	.( - )	
		

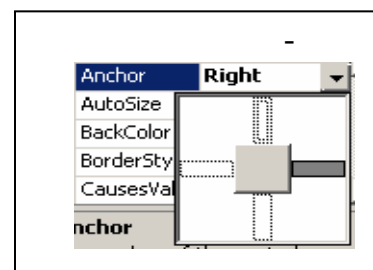
A Label

 ( - )	

		Name



( - ) :		Anchor
( - ) .		
-	-	



True

Label1

Label1.AutoSize = True

	True	AutoSize
( )	False	

		BackColor

```
Label1.BackColor = Color.Blue
```

lblHello

```
lblHello.BorderStyle = BorderStyle.Fixed3D
```

( - ) Fixed3D	Fixed3D	BorderStyle
	FixedSingle	
	None	



True	True	CauseValidate
.Validate		
	False	

		Contextmenu

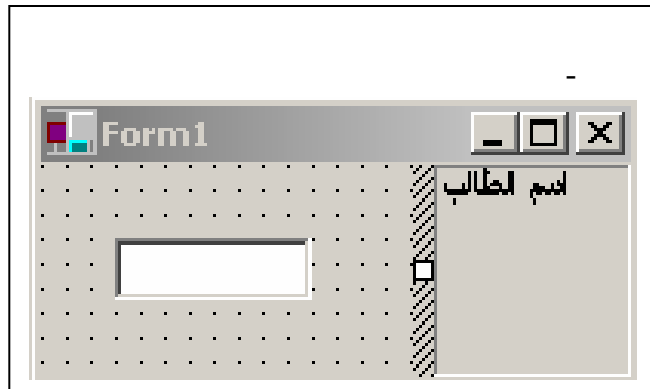
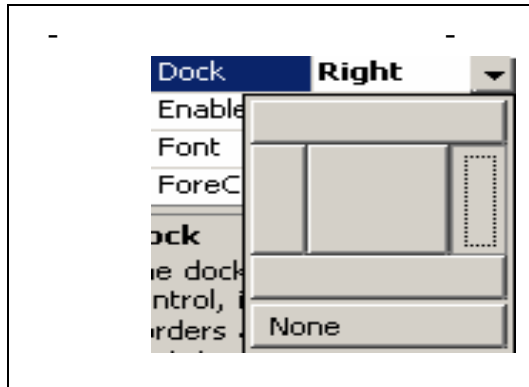
		Curser

Windows

.Explorer

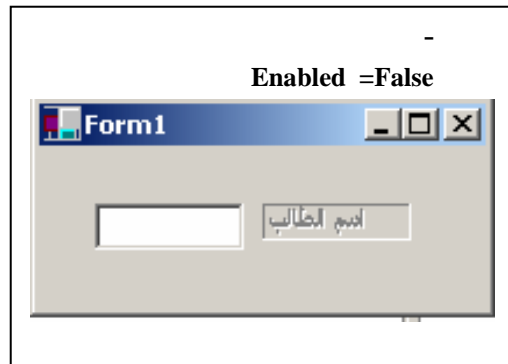
.( - ) ( - )

.	Right	Dock
.( - ) ( - )		
.	Left	
.	Top	
.	Bottom	
.	Fill	
.	Non	



.False

.	True	Enabled
( - ) . False	False	



.		FontName	Font
	True False	FontBold	
		FontSize	
	True False	FontStrikThru	
	True False	FontItalic	

		ForeColor

	True	Locked
	False	

	Yes	RightToLeft
	No	

		Height	Size
		Width	

Tab		TabIndex



Label1

```
Label1.Text = " "
```

		Text

.Label1

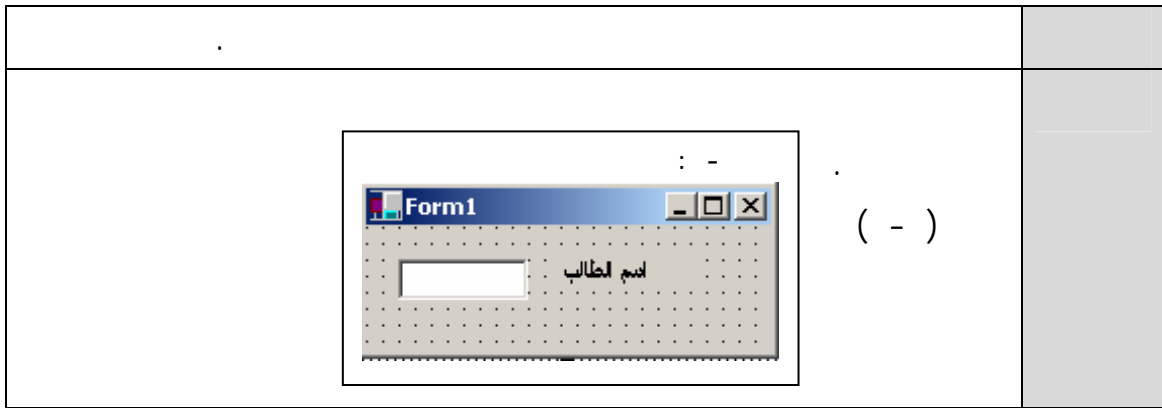
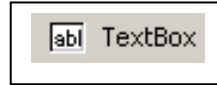
```
Label1.TextAlign = Label1.HorizontalAlignment.Center
```

	Right	TextAlign
	Left	
	Center	

& : .  
True .  
+ Alt Tab  
. Text &

&	True Alt	True/False UseMnoneic





:  
CauseValidate Borderstyle Backcolor Autosize Anchor  
Locked ForeColor Font Enabled Dock Cursor Contextmenu  
TextAlign Text TabIndex Size

:

:CharacterCasing -

Capital

Upper

( - )

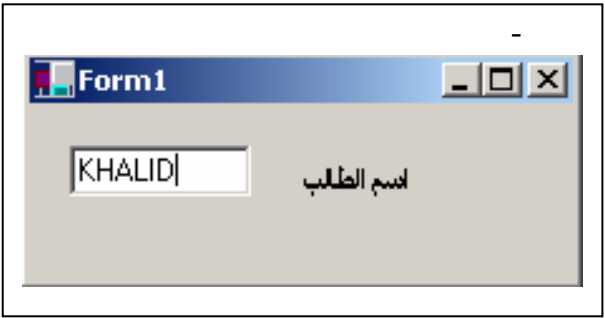
TxtUpper

:

```
TxtUpper.CharacterCasing = CharacterCasing.Upper
```

b :  
.Upper B

	Upper	CharacterCasing
	Lower	
	Normal	



( - )

	True	HideSelection
	False	



: (Lines) -

) :

Y X ( Y X



	X	Location
	Y	

: -

		MaxLength

: -

	True	MultiLine
	False	

:

: PasswordChar-

( - )

	*	PasswordChar



:

	True	ReadOnly
	False	

.Tab

:

. Tab	True	TabStop
. Tab	False	

True :  
(TabIndex )

txtName

txtName.visible = False

	True	Visible
	False	

MultiLine : -

	True	WordWrap
	False	

:

```

'عرض نص مرحبا بكم على مربع النص
  TextBox1.Text = "بكم مرحبا"
'
'سوف يمكن الاستفادة من ضغط زر الرجوع مما يؤدي إلى فتح سطر جديد
  TextBox1.AcceptsReturn = True
'
'سوف يمكن من قبول الانتقال إلى مربع النص بعد ضغط زر تاب
  TextBox1.AcceptsTab = True
'
'سوف يتم دمج عبارة مرحبا بكم مرة أخرى مع العبارة الأولى في سطر
'جديد
  TextBox1.AppendText(vbCrLf & "أخرى مرة بكم مرحبا")
'
'تغيير لون الخلفية إلى أزرق
  TextBox1.BackColor = Color.Blue
'
'تغيير لون النص إلى أسود
  TextBox1.ForeColor = Color.Black
'
'جعل خلفية مربع النص الصورة
  TextBox1.BackgroundImage =
Image.FromFile("C:\MyTextBoxBackground.gif")
'
'تغيير نمط الحدود
  TextBox1.BorderStyle = BorderStyle.FixedSingle
'
'مسح النص الموجود في مربع النص
  TextBox1.Clear()
'
'تغيير مؤشر الفأرة إلى يد عند مروره على مربع النص
  TextBox1.Cursor = Cursors.Hand
'
'تغيير الخط لمربع النص
  TextBox1.Font = New Drawing.Font("Comic Sans MS", 12,
FontStyle.Bold, GraphicsUnit.Pixel)
'
'تعيين أقصى عدد من الحروف
  TextBox1.MaxLength = 3000
'
'إرجاع النص إلى القيمة الافتراضية
  TextBox1.ResetText()
'
'إضافة شريط انزلاق إلى يمين مربع النص
  TextBox1.ScrollBars = ScrollBars.Vertical
'
'سوف يتم تظليل النص من البداية بطول 12 حرفا
  TextBox1.Select(0, 11)
'
'سوف يتم تظليل كل النص في مربع النص
  TextBox1.SelectAll()
'
'تعريف متغير وتعيين قيمته تساوي النص المظلل
Dim SelectedText As String = TextBox1.SelectedText
'
'بدء التظليل من أول حرف
  TextBox1.SelectionStart = 0
'
'طول النص المظلل
  TextBox1.SelectionLength = 11
'
'تغيير حجم مربع النص
  TextBox1.Size = New Size(300, 400)
'
'عرض النص في منتصف مربع النص
  TextBox1.TextAlign = HorizontalAlignment.Center
'
'عرفنا متغير قيمته طول النص الموجود في مربع النص
Dim i As Integer = TextBox1.TextLength
'
'التراجع عن آخر تحرير لمربع النص
  TextBox1.Undo()
'
'إخفاء مربع النص
  TextBox1.Visible = True
'
'اختيار كل النص ثم نسخه
  TextBox1.SelectAll()

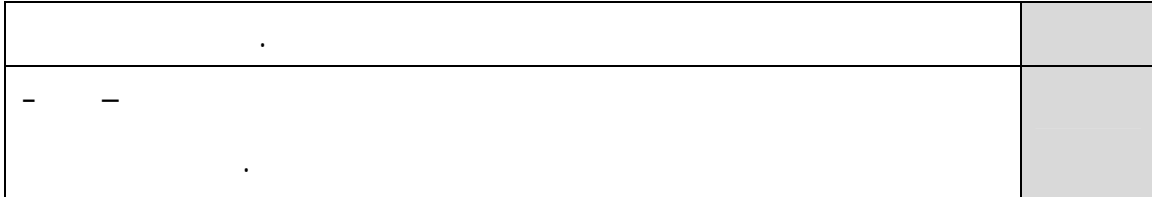
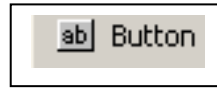
```

TextBox1.Copy()

اختيار كل النص ثم قصه

TextBox1.SelectAll()

TextBox1.Cut()



:

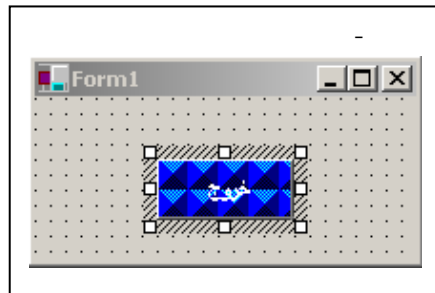
Dock Cursor Contextmenu CauseValidate Backcolor Anchor  
Text TabStop TabIndex Size Locked ForeColor Font Enabled  
Visible TextAlign

:

( - )

:

-





**: DialogResult**

)

Cancel

Ok

( /

ShowDialog	:
------------	---

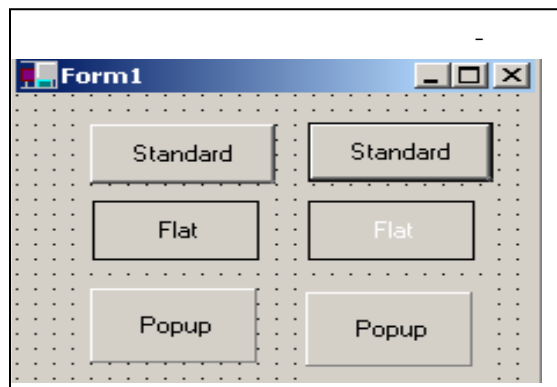
	None	DialogResult
	Ok	
	Cancel	

**:FlatStyle**

1.12

1.12

.( - )	Flat	FlatStyle
.( - )	PopUP	
.( - )	Standard	

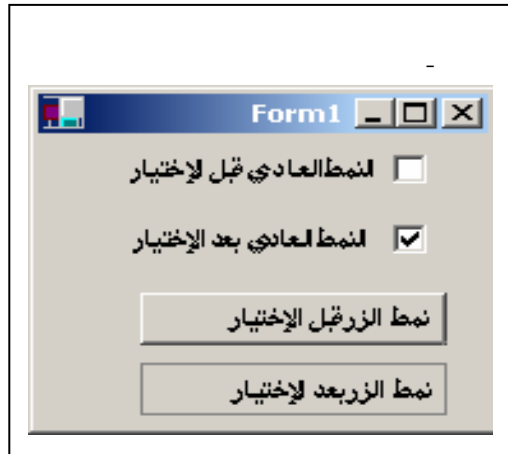




( / ) ( / )	
✓	
✓	
.	

Contextmenu CauseValidate Background Backcolor Anchor  
 Size Locked ForeColor Font FlatStyle Enabled Dock Cursor  
 Visible TextAlign Text TabStop TabIndex

.( - )	Normal	Appearance
.( - )	Button	

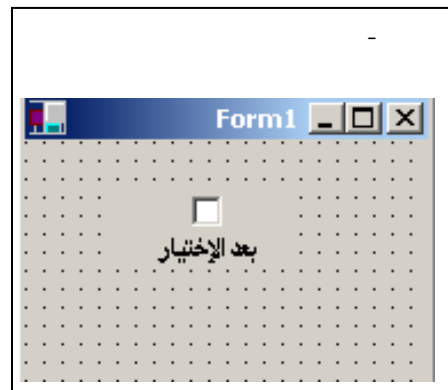
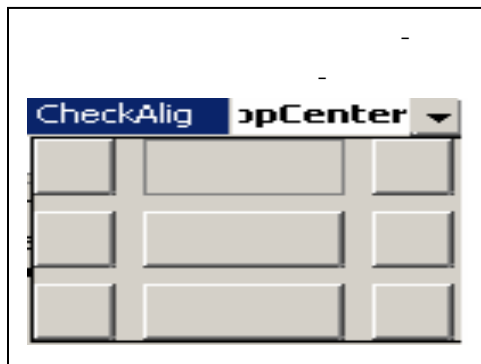


✓	True	AutoCheck
	False	

( ) TopCenter

( - )

	( - )	CheckAlign



(✓)

	True	Checked
	False	

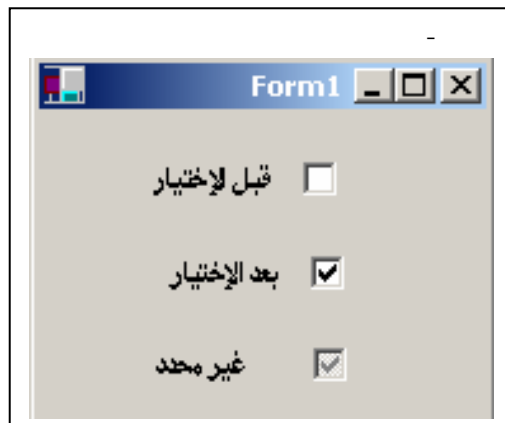
	Unchecked	CheckState
	Checked	
	Indeterminate	

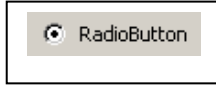
ThreeState = True :  
CheckState

)

( - ) (

	True	ThreeState
✓	False	





### (Radio Button)

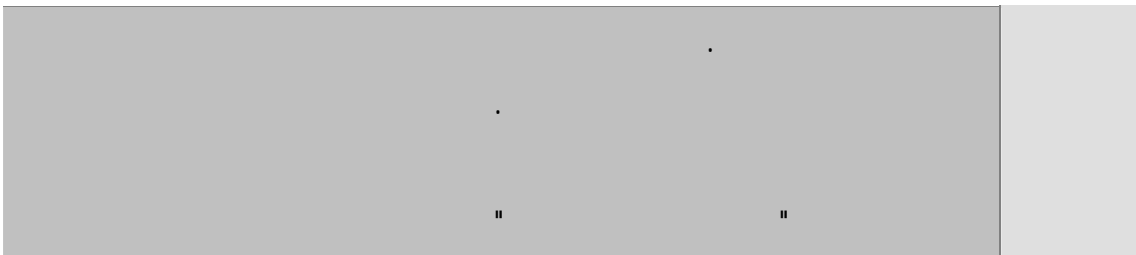
	( - )

: :

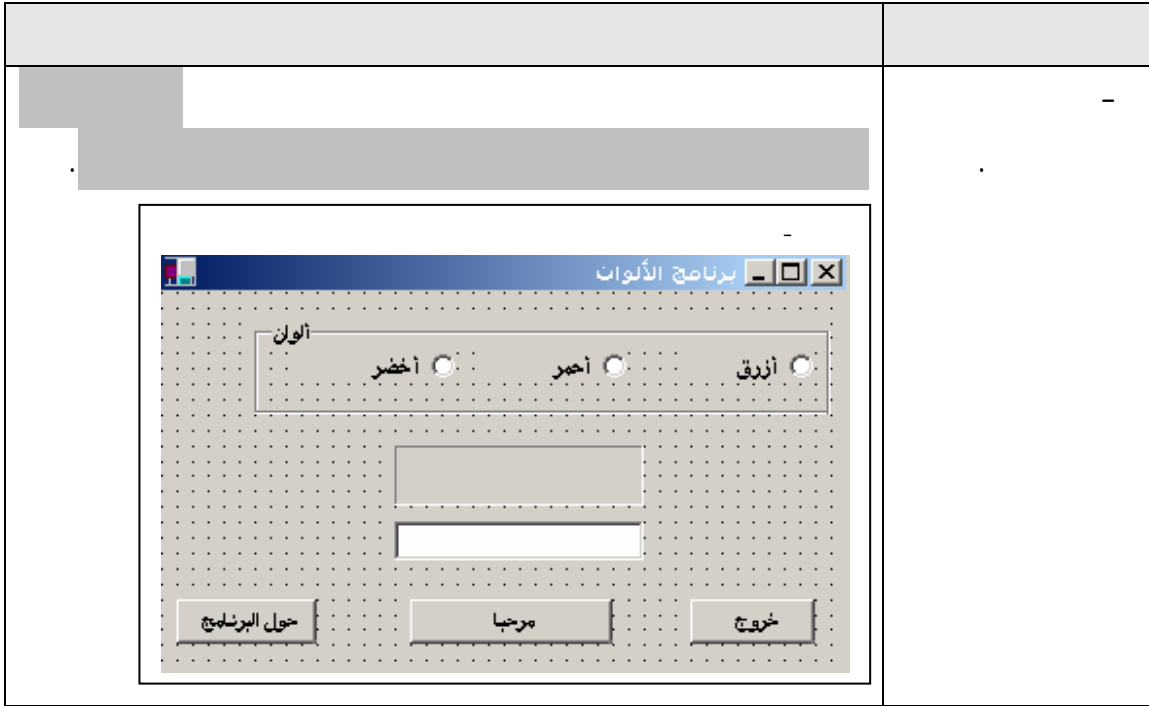
Background Backcolor AutoCheck Appearance Anchor  
Dock Curser Contextmenu Checked CheckAlign CauseValidate  
TabIndex Size Locked ForeColor Font FlatStyle Enabled  
Visible TextAlign Text TabStop

ThreeState :





```
txtName.text = " "
TxtName = "
vb.net Text
```



( - ) .

( ) .

"	Text	Form1
"		
Right	RightToLeft	
rdbBlue	Name	RadioButton1
	Text	
rdbRed	Name	RadioButton2
	Text	
rdbGreen	Name	RadioButton3
	Text	
GrbColors	Name	GroupBox1
	Text	
LblColor	Name	Label1
	Text	
txtColor	Name	Text1

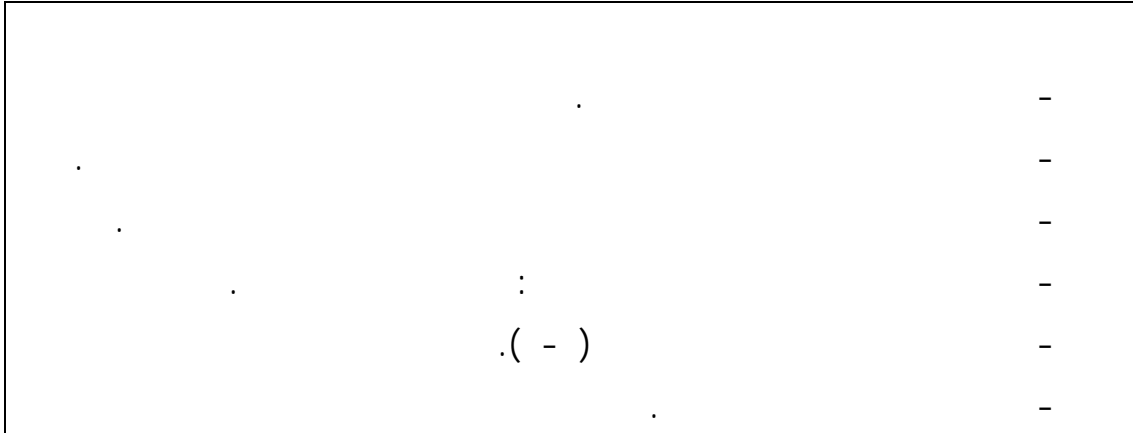
		Text	
cmdMove	Name	Button1	
		Text	
cmdExit	Name	Button2	
		Text	
CmeAbout	Name	Button3	
		Text	

<pre> .I lblColor.ForeColor = color.Blue ) ( ForColor  .ForeColor      BackColor .II lblColor.ForeColor = color.red .III <i>lblColor.ForeColor = color.Green</i> </pre>	<pre> - ) ( ) ( </pre>
<pre> "      " <i>lblcolor.Text = txtColor.Text</i> txtColor.Text  lblcolor.Text  txtColor.Text =  lblcolor.Text = txtColor.Text  .      lblcolor.Text </pre>	<pre> - Text Text </pre>



	<p>cmdMove</p>
<p>cmdAbout</p> <p><i>MessageBox.Show("الألوان برنامج هو هذا", "ألوان", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)</i></p> <p><i>MessageBox.Show</i></p> <p><i>MessageBox.Show (</i></p> <p><i>Information</i></p>	<p>-</p> <p><i>MessageBox.Show</i></p> <p><i>.cmdAbout</i></p>
<p>End</p>	<p>-</p> <p><i>.cmdExit</i></p>





### (List Box)

( )	( - )
.	



:

Contextmenu CauseValidate BorderStyle Backcolor Anchor  
Size Locked Location ForeColor Font Enabled Dock Curser  
Visible TabStop TabIndex

:

:

-

MultiColumn = True :

		ColumnWidht

: (DataSource) -

--

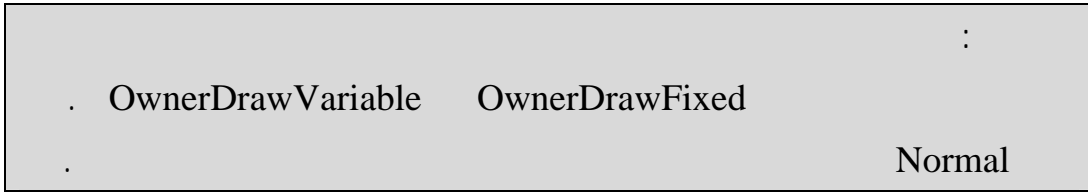
:(DisplayMember) -

( )  
 ( )VauleMember

: -

( )	Normal	DrawMode
( ) .Height	OwnerDrawFixed	
	OwnerDrawVariable	

: (ItemHeight) -



OwnerDrawFixed .I

OwnerDrawVariable .II

: (Items) -

( - )

Ok

.( - )



: -

.

	True	MultiColumn
	False	

ColumnWidth	True	:
-------------	------	---

: -

.

	True	ScrollAlwaysVisible
	False	

: -

.

	None	SelectionMode
	One	
( - )	MultiSimple	
( - )	MultiExtended	

) MultiExtended (	) MultiSimple	:
Ctrl Shift	(	



	True	Sorted
	False	

:ValueMember -





( - )

:

-

	Simple	Style
	DropDown	
	DropDownList	

The screenshot shows a Windows form titled "Form1" with three dropdown menus. The first menu, labeled "ابسيط" (Simple), shows a list of names: محمد, خالد, محمد, and سلمان. The second menu, labeled "منسدلة" (DropDown), shows a list of names: خالد, محمد, سلمان, and أحمد. The third menu, labeled "قائمة منسدلة" (DropDownList), shows a list of names: خالد, محمد, سلمان, and أحمد.



### (CheckedListBox)

( - )	
✓ ( - )	



:  
( - )

Style =      Style = DropDown      :  
DropDownList

.		MaxDropDownItems



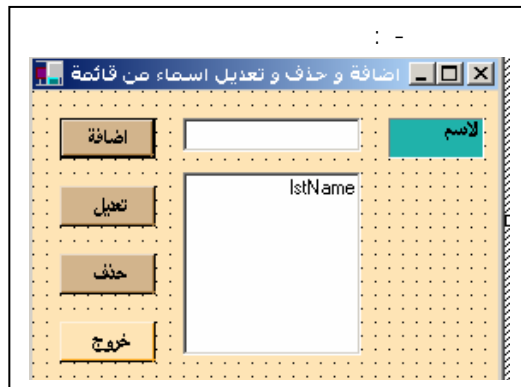
vb.net

**-ListBox-**  
( - )

-Enter -  
No Yes  
( - ) ( - )

( - )

LblName	Name	Label1
	Text	
txtName	Name	Text1
	Text	
cmdAdd	Name	Button1
	Text	
cmdUpdate	Name	Button2
	Text	
CmdDelete	Name	Button3
	Text	
CmdExit	Name	Button4
	Text	
LstName	Name	ListBox
Right	RightToLeft	Form1
cmdAdd	AcceptButton	
Moccasin	BackColor	



-

-





<p data-bbox="236 577 268 616">&amp;</p> <p data-bbox="1013 577 1093 616">Text</p> <p data-bbox="973 761 1045 801">Yes</p>	
--	--

AcceptButton

:

cmdAdd

-cmdAdd-

-Enter-

AcceptButton

:

-

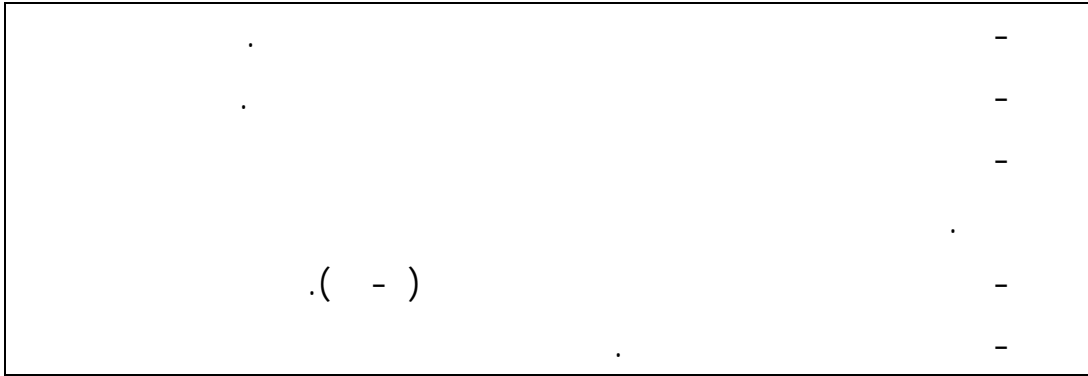
-

-

-

.( - )





vb.net

-ListBox-

- )

(

-Enter -

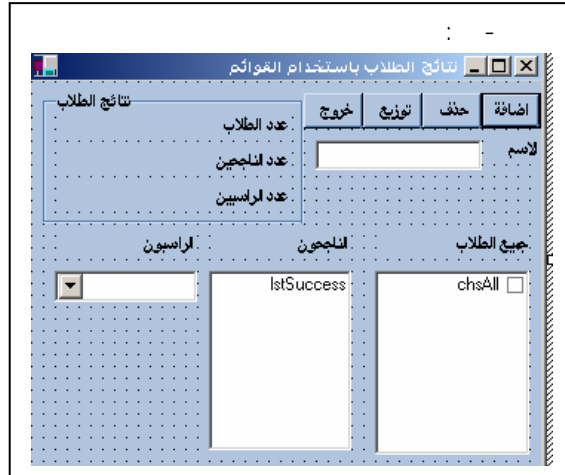
No

Yes

. ✓

( - )

( - )



-

( - )

LblName	Name	Label1
	Text	
lblSuccess	Name	Label2
	Text	
lblFail	Name	Label3
	Text	
lblAll	Name	Label4
	Text	
Label5	Name	Label5
	Text	
Label6	Name	Label6
	Text	
Label7	Name	Label7
	Text	
txtName	Name	Text1
	Text	
cmdAdd	Name	Button1

-

		Text	
cmdUpdate	Name	Button2	
		Text	
CmdMove	Name	Button3	
		Text	
CmdExit	Name	Button4	
		Text	
LstSuccess	Name	ListBox1	
cboFail	Name	ComboBox1	
chsAll	Name	CheckedListBox	
Right	RightToLeft	Form1	
cmdAdd	AcceptButton		
Moccasin	BackColor		

.I

```

If txtName.Text <> "" Then
  chsAll.Items.Add(txtName.Text)
  txtName.Focus ()
  txtName.ResetText ()
Else
  MessageBox.Show( " " " "
MessageBox.OK)
End If

```

.II

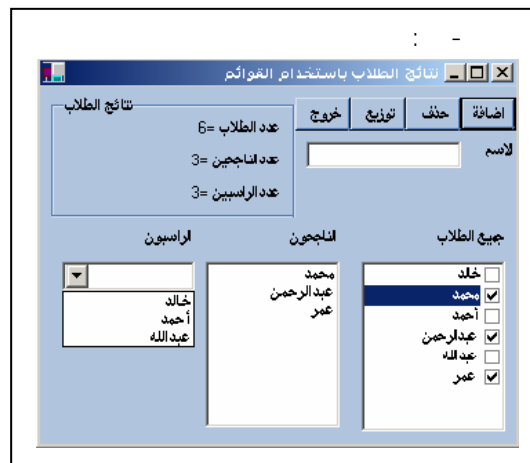
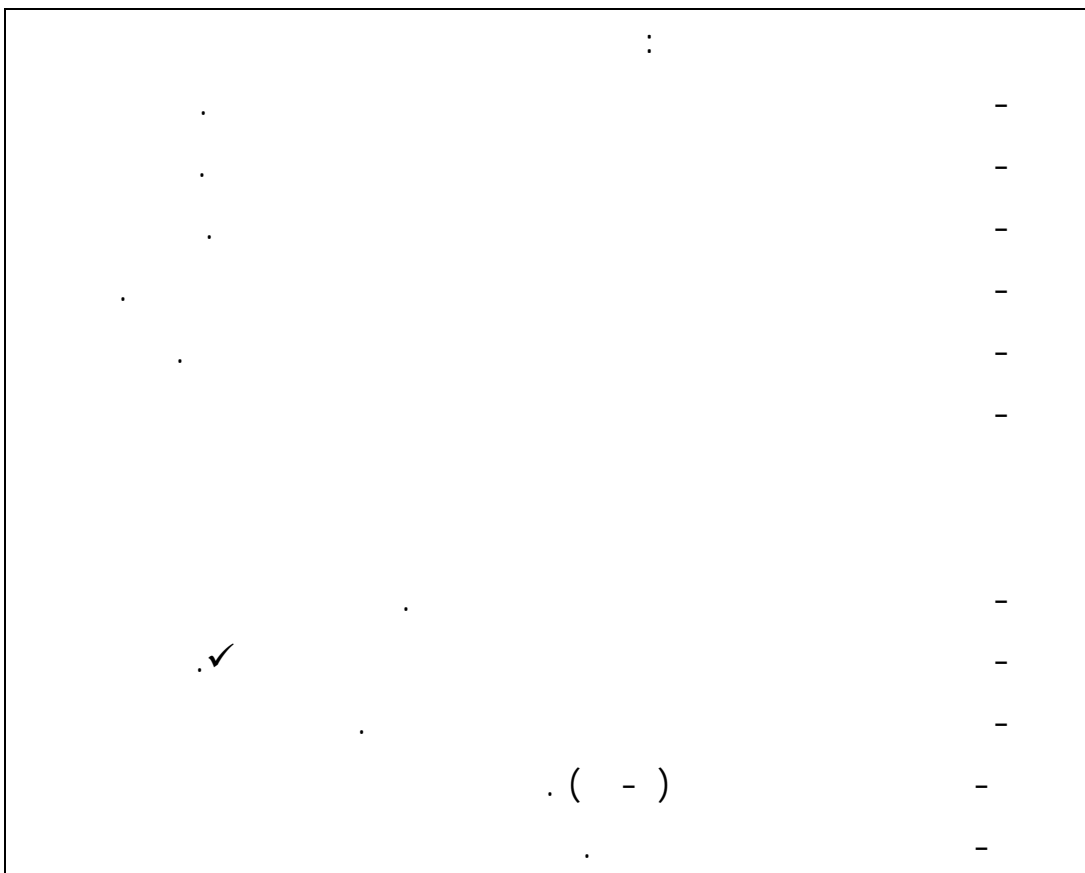
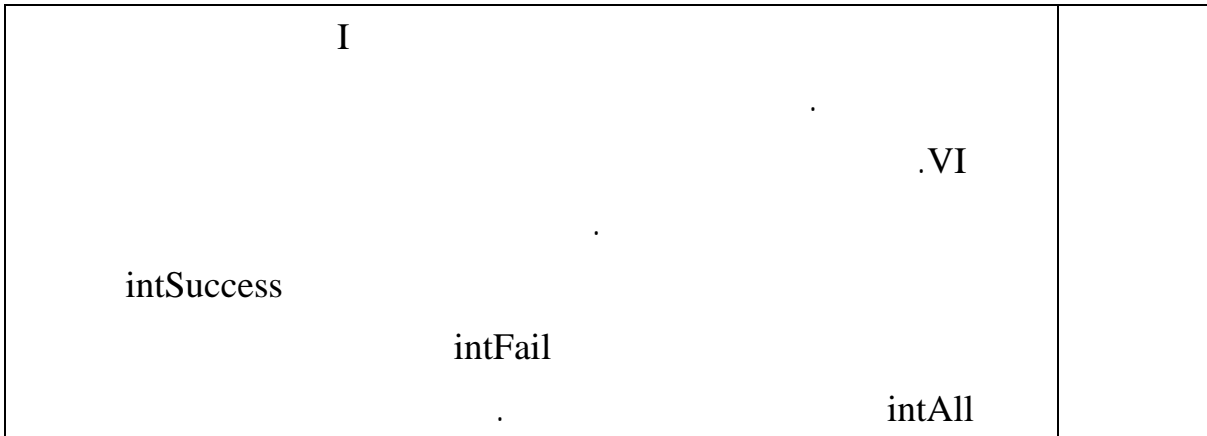
Add

Text

chsAll.Items.Add(txtname.Text)

<p style="text-align: center;">Else</p>	<p style="text-align: center;">.III If</p>
<div style="border: 1px solid black; padding: 10px; margin-bottom: 10px;"> <pre> Dim s As String If chsAll.SelectedIndex &lt;&gt;-1 Then     s = "          " &amp; chsAll.SelectedItem.ToString     If MessageBox.Show(s_ " : "     MessageBox.YesNo + MessageBox.IconQuestion) _ = DialogResult.Yes Then         chsAll.Items.Remove(chsAll.SelectedIndex)     End If End If </pre> </div> <p style="text-align: center;">s</p> <p style="text-align: center;">chsAll.SelectedIndex</p> <p style="text-align: center;">s = "          " &amp; chsAll.SelectedItem.ToString</p> <p style="text-align: center;">Remove</p> <p style="text-align: center;">chsAll.SelectedIndex</p>	<p style="text-align: center;">.I</p> <p style="text-align: center;">.II</p> <p style="text-align: center;">.III</p> <p style="text-align: center;">.IV</p> <p style="text-align: center;">.V</p>

<pre> Dim i As Integer Dim intSuccess As Integer Dim intFail As Integer Dim intAll As Integer cboFail.Items.Clear ()  lstSuccess.Items.Clear () For i = 0 To (chsAll.Items.Count - 1)   If chsAll.GetItemChecked(i) = True Then     lstSuccess.Items.Add(chsall.Items(i))     intSuccess = intSuccess + 1     intAll = intAll + 1   Else     cboFail.Items.Add(chsall.Items(i))     intFail = intFail + 1     intAll = intAll + 1   End If Next lblSuccess.Text = lblSuccess.Text &amp; " = " &amp; intSuccess lblFail.Text = lblFail.Text &amp; " = " &amp; intFail lblAll.Text = lblAll.Text &amp; " = " &amp; intAll </pre>	<p>.I</p>	<p>-</p> <p>X</p>
<p style="text-align: center;">I</p>	<p>.II</p>	
<p style="text-align: center;">intAll    intSuccess    intFail    .</p>		
	<p>.III</p>	
	<p>.IV</p>	
	<p>.V</p>	
<p>If chsAll.GetItemChecked(i) = True Then</p>		
<p style="text-align: center;">chsAll.GetItemChecked(i)</p>		
<p style="text-align: center;">True</p>		<p>✓</p>



Form1 : -

.Form2

:

```
Private Sub Button1_Click(ByVal Sender As Object, ByVal e As EventArgs)
Handles Button1.Click
    Dim frm2 As New Form2()
    frm2.Show()
End Sub
```

:

.Form2 .I

.Form2 .II

.. .III

: -

:

```
Private frm2 As Form2

    Private Sub Button1_Click(ByVal Sender As Object, ByVal e As
EventArgs) Handles Button1.Click
        If IsNothing(frm2) = False Then
            If frm2.IsDisposed = False Then
                Else
                    frm2 = New Form2()
                    frm2.Show()
                End If
            Else
                frm2 = New Form2()
                frm2.Show()
            End If
        End Sub
```



.Instance Form2 -  
.Isnothing -  
.Isdisposed -  
. -

Module

Form1

```
Public PassingVar As String
```

Form2

```
Dim myForm As Form1  
myForm.myVar = "هذه القيمة تم تعيينها في نموذج ٢"
```

DesktopLocation

```
Me.DesktopLocation = New Point(0, 0)
```

( - ) .

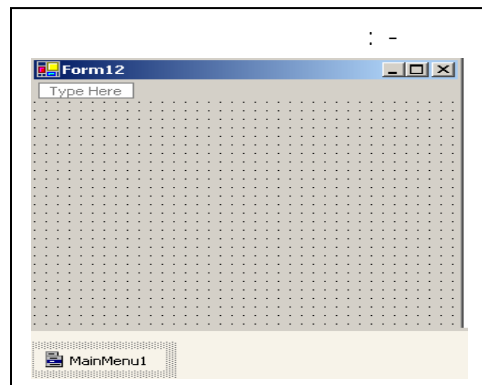


( - ) .

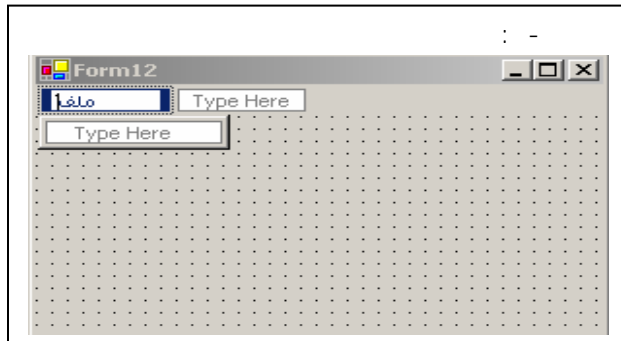


:

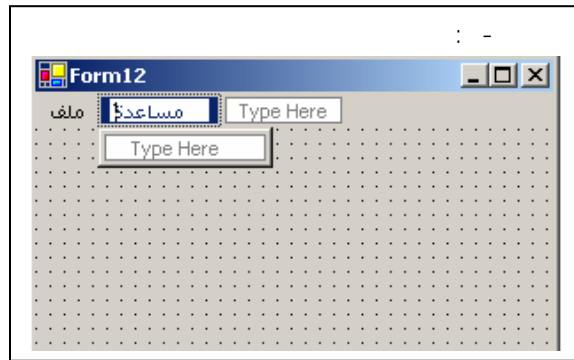
( - )



( - )



.( - )



```
Dim frm2 As New Form2()  
frm2.Show()
```

. Form2

## أداة التوقيت Timer

(Millisecond)

Interval

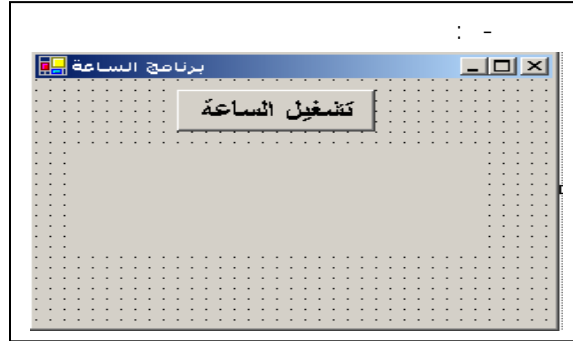
Tick()

Start

.Stop



( - )



١- أعط الأدوات الخصائص التالية :

cmdStartStop	Name	Button1
	Text	
lblClock	Name	Label1
	Text	
Timer1	Name	Timer
1000	Interval	Timer
برنامج الساعة	Text	Form



Interval  
Tick

```
If Not Timer1.Enabled Then  
    Timer1.Start()  
    Button1.Text = "    "  
Else  
    Timer1.Stop()  
    Button1.Text = "    "  
End If
```

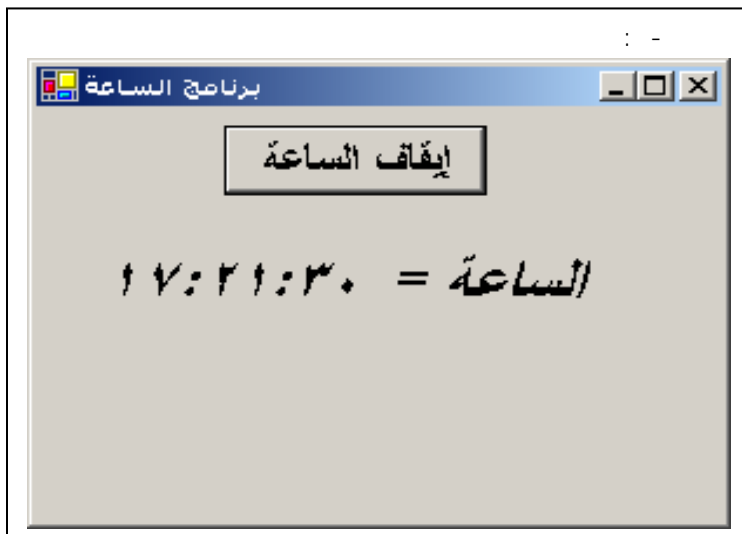
Stop            Start            Enabled

Tick

```
Label1.Text = "الساعة = " & TimeString
```

Tick

( - )



## DateTimePicker

تستخدم هذه الأداة لعرض الوقت بشكل جميل على شكل تقويم كما تقوم بحفظ التاريخ الذي تم اختياره من قبل المستخدم. أهم خصائص هذه الأداة هي :

الخاصية	الشرح
MinDate	أقل قيمة يمكن اختيارها من التقويم.
MaxDate	أكبر قيمة يمكن اختيارها من التقويم.
ShowCheckBox	إذا أردت عرض مربع اختيار في الأداة.
Value	التاريخ الذي تم اختياره.



```
DateTimePicker1.MinDate = New DateTime(1985, 6, 20)
DateTimePicker1.MaxDate = DateTime.Today
DateTimePicker1.ShowCheckBox = True
MessageBox.Show(DateTimePicker1.Value)
```

( - ) .



:

- .
- .
- .
- .
- .
- .
- .

PictureBox



:

cmdOpenPicture	Name	Button1
	Text	
pb	Name	PictureBox
ofd	Name	OpenFileDialog

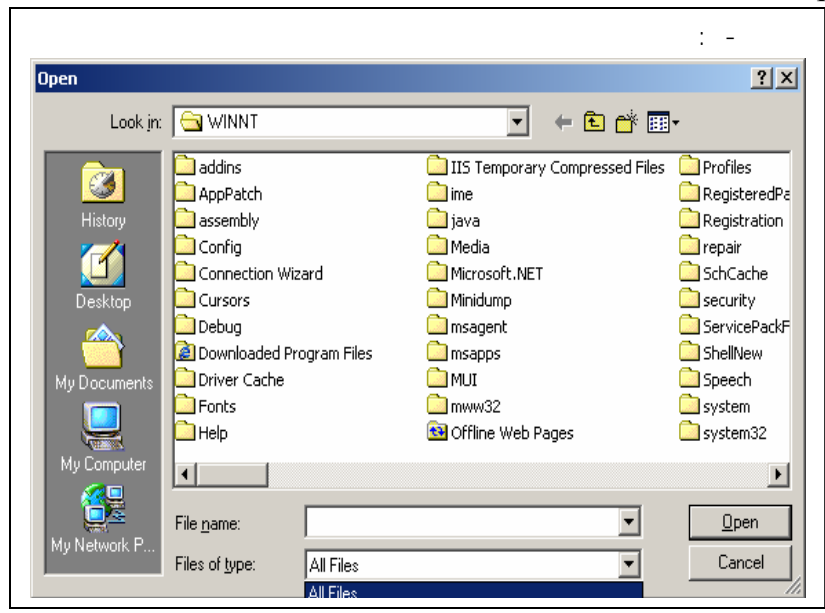
```
With ofd
    .CheckFileExists = True
    .ShowReadOnly = False
    .Filter = "All Files|*.*|Bitmap Files (*)|*.bmp;*.jpg"
    .FilterIndex = 1
    If .ShowDialog = DialogResult.OK Then
        pb.Image = Image.FromFile(.FileName)
    End If
End With
```

With -  
-  
-  
Bmp -  
.All Files -  
Jpg -  
Ok -  
( - )

File Of Type



\* | File Name :  
Bmp





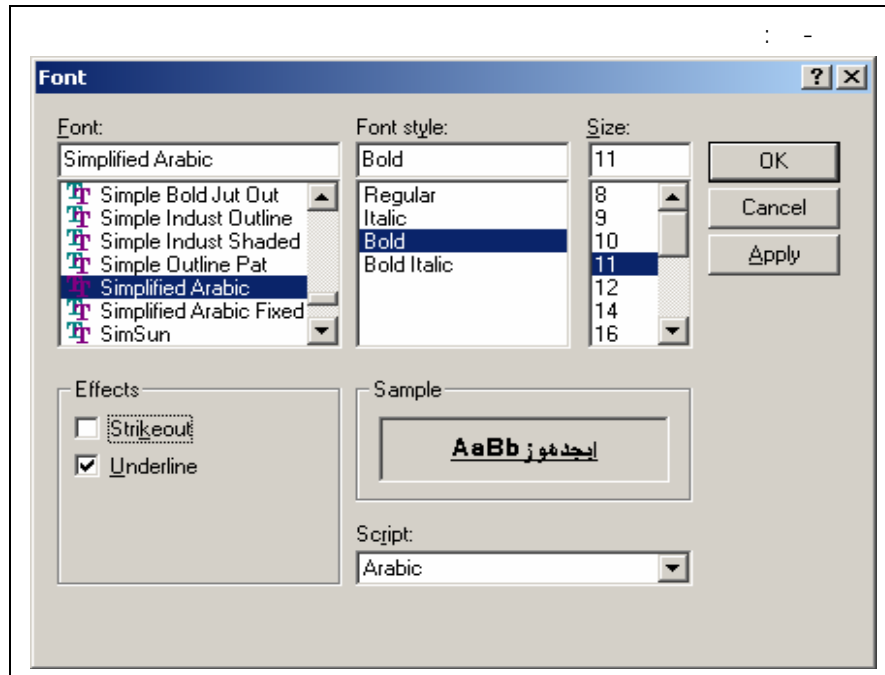


```
With FontDialog1
    .ShowApply = True
    If .ShowDialog = DialogResult.OK Then
        Button1.Font = .Font
        Label1.Font = .Font
    End If
End With
```

Apply

ShowApply

.Ok



.BackColor

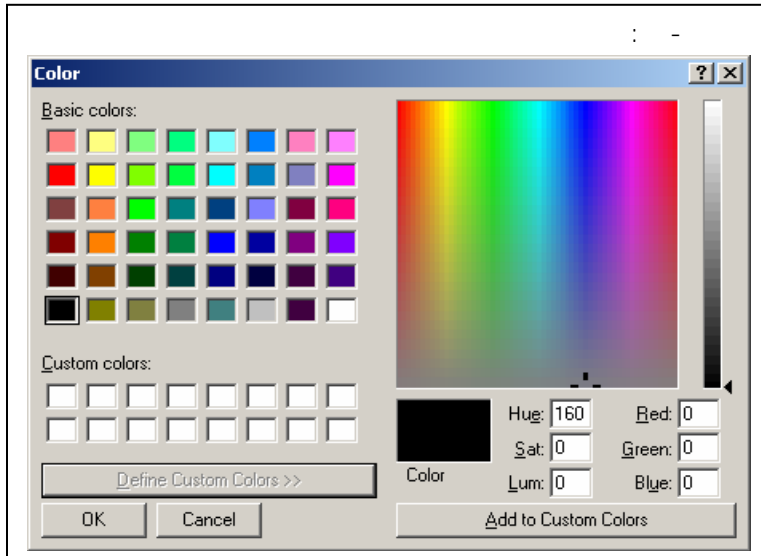
ForeColor



ColorDialog1		
cmdChangeColor	Name	Button1
	Text	
Label1	Name	Label1
ColorDialog1	Name	ColorDialog

:( - )

```
With ColorDialog1
    .SolidColorOnly = True
    If .ShowDialog = DialogResult.OK Then
        Button1.BackColor = .Color
        Label1.ForeColor = .color
    End If
End With
```



:  
:  
-Front End Validation-

:  
•  
•

:  
-Back End Validation-

:  
•  
•

:

- 
- 
- 
- 

:

الكائن	أسلوب التدقيق
مربع النص	تحديد أو تعديل البيانات المدخلة.
مربع الاختيار	حصر القيم باختيار أو عدمه.
زر الاختيار	اختيار قيمة واحدة.
القائمة	تقديم مجموعة من البيانات الصحيحة.
أداة التاريخ	حصر المدخلات بالتاريخ.

:

- PasswordChar :

( - ) .



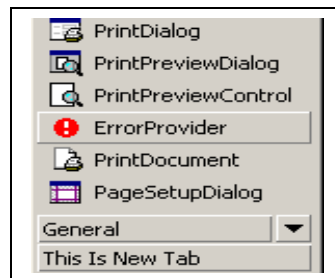
-٢ MaxLength :

-٣ ReadOnly :

-٤ CharcterCasing :

## ErrorProvider

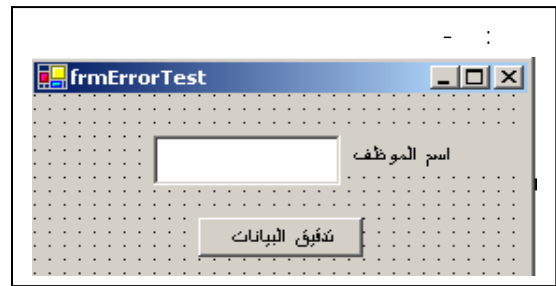
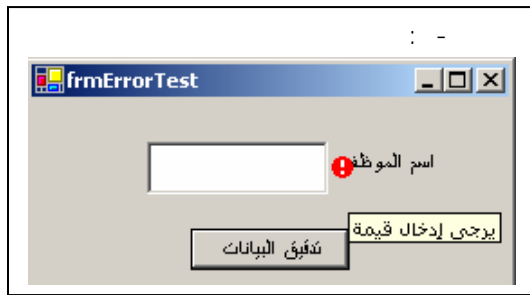
setError





( - )

cmdValidate	Name	Button1	
		Text	
Textbox1	Name	Textbox	
		Text	
Label1	Name	Label1	
		Text	
er	Name	ErrorProvider	
frmErrorTest	Name	Form	



:

```

If Trim(TextBox1.Text) = "" Then
    er.SetError(TextBox1, "قيمة إدخال يرجى")
Else
    er.SetError(TextBox1, "")
End If
    
```

:

.er  
.( - )

:

```
Dim ControlVar As Control
For Each ControlVar In Me.Controls
    If TypeOf ControlVar Is TextBox Then
        If Trim(ControlVar.Text) = "" Then
            erReq.SetError(ControlVar, " ")
        Else
            erReq.SetError(ControlVar, "")
        End If
    End If
Next
```

:

-

-

Dim . Visual Basic .NET

:

Dim Variable as DataType

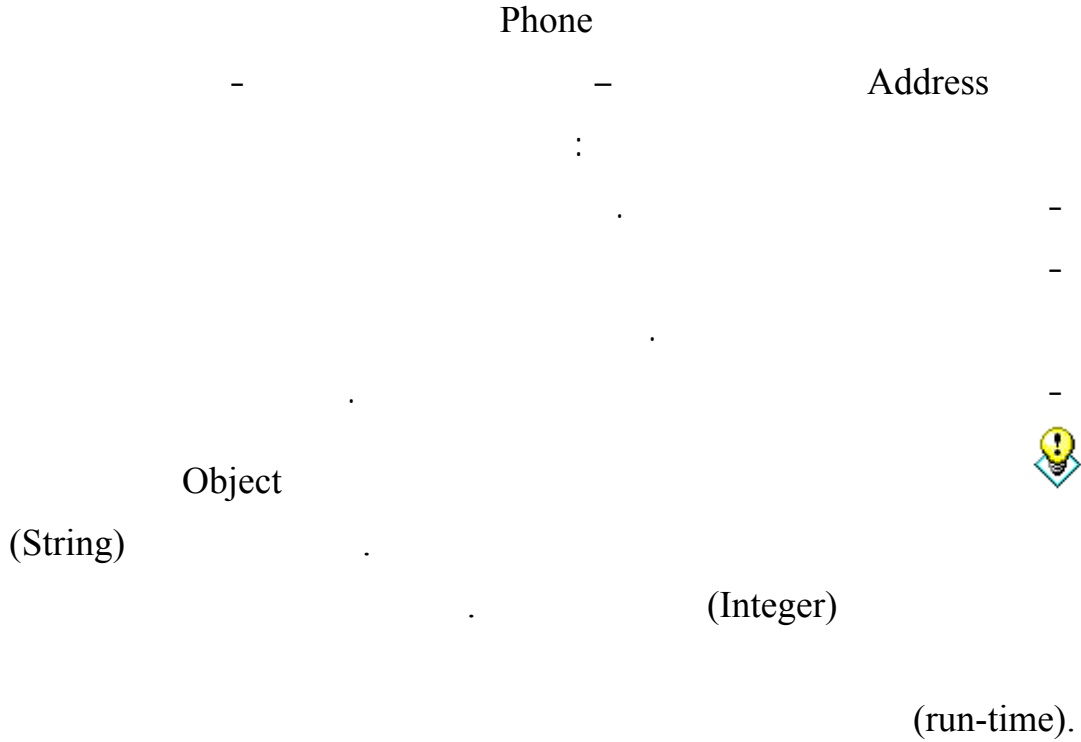
:

Dim -  
Variable -  
As -  
DataType -

Address . Phone ,

```
Dim Phone As Integer  
Dim Address
```





نوع البيانات	المساحة التخزينية	القيم المسموح بها
Boolean	4 bytes	صحيحة (True) خاطئة (False)
Byte	1 byte	يمثل عدد بين ٠ و ٢٥٥ (قيم موجبة فقط)
Char	2 bytes	يمثل حرف واحد فقط
DateTime	8 bytes	قيم تبدأ من ١-١-١ م إلى ٣١-ديسمبر-٩٩٩٩
Decimal	12 bytes	يمثل الأرقام العشرية ( ١٦ byte-وهي من ٠ الى موجب أو سالب 79,228,162,514,264,337,593,543,950,335)
Int32	4 bytes	من - 2,147,483,648 إلى 2,147,483,647
Int64	8 bytes	من -9,223,372,036,854,775,808 إلى 9,223,372,036,854,775,807
Object	4 bytes	أي نوع يمكن أن يخزن داخل هذا النوع
Int16	2 bytes	من - 32,768 إلى 32,767
String	10 bytes + (2 طول السلسلة *)	يمثل قيمة حرفية تتسع إلى ٢ بليون حرف



int32 int16

## Integer

int64 int16 Int32

```
Dim Phone As Integer  
Dim Phone As Int32
```

.int32

Integer

.Int64 int16

Int32

int32

Int32 Long

## String

. System.Class

Left,Right,Mid

## Variant

Object

:

:

-

:

-

:

-

-

-

Static

-

-

Public

-

Friend -

. Private -

Dim -

.Static

## Variable Scope



```
Sub Test1()
    Dim L As Integer ' مجال الاجراء
    If L = 5 Then
        Dim B As Integer ' مجال المقطع
        B = L
    End If
End Sub
```

:

			L	.I
B	.If		B	.II
				If

:

```
Dim myVar As Integer = 9
```

:

```
Dim I, J As Integer
Dim L As Integer, M As Integer
Dim N As Integer, X As Double
```

Integer

.Dim



Integer

Variant

.Integer

:

-

:

ReadOnly x As Integer



:

-

-

-

-

	Class
&	Phone&Number
	18Width
	First Name

Var=Value

=



Dim NumberOfStudent as integer  
NumberOfStudent =15

:

-

-



Dim Name as String  
Name="Khalid"

:

-

Khalid

-



Dim Result as Boolean  
Result=True

:

.Boolean

-

-

```
Dim a As Integer = 5
Dim b As Integer
b = a
```

```

:
Integer a -
.Integer b -
. a b -

```

```
Dim a As String
a = TextBox1.Text
```

```

:
.Integer a -
Text a -
. TextBox1

```

### Explicit

Explicit

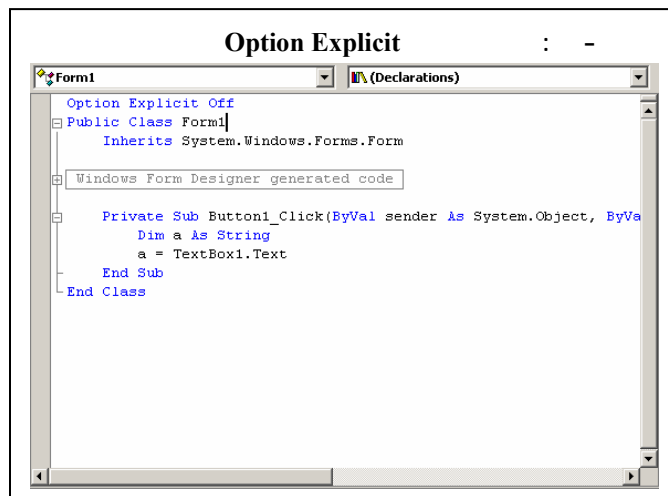
.Off On

Explicit

On

. ( - )

Off



## Constants

:

<b>Const MyConst = 3</b>
--------------------------

:

**MyConst**

-

-



. String



```
Dim Name As String
Dim HouseNumber As String
Name = "أحمد"
HouseNumber = "12345"
```

HouseNumber      Name

.HouseNumber

Name



```
Dim str1 As String = "B"
Dim str2 As String = "A"
```

A

B

: . Compare

str2

str1

A

B

```
Dim str1 As String = "AA"
Dim str2 As String = "AA"
```

. str2

str1



:

.res

.Integer

Compare

-  
-  
-  
-

```
Dim str1 As String = "AA"  
Dim str2 As String = "AA"  
Dim res As Integer  
res= String.Compare(str1, str2)  
Label1.Text = "نتيجة المقارنة:" + res.ToString()
```

:




:

```
Dim FirstName As String = "Khalid"  
Dim LastName As String = "AL-Jduaia"
```

FirstName

. Concat

.LastName

```
Label1.Text = String.Concat(FirstName, LastName)
```



.Insert



:Insert

```
Dim str1 As String = "Saat"  
Dim strRes As String  
strRes = str1.Insert(2, "lam")  
Label1.Text = strRes.ToString()
```

:

.saat

-

-

lam

Insert

-

strRes

str1

.Salamat



saat

## Remove

: Remove



```
Dim s As String = "salamat"  
Label1.Text = s.Remove(5, 2)
```

.salamat

s

-

Remove

-

: Remove

Remove

.I

.II

.salam

-



: Substring



```
Dim s As String = "salamat"  
Label1.Text = s.Substring(5)
```

.salamat

.I

s

.II

.at

-

```
Dim s As String = "salamat"  
Label1.Text = s.Substring(3, 2)
```

.salamat

:

substring

-

-

.am

s



Replace

: Replace



```
Dim str1 As String = "ppp"  
Label1.Text = str1.Replace("p", "k")
```

:

.ppp

-

p

Replace

-

.k

.kkk

:

	.ToLower()
	ToUpper()



```
Dim aStr As String = "ahmed"
Dim bStr As String = "OMAR"
Label1.Text = aStr.ToUpper()
Label2.Text = bStr.ToLower()
```

:

OMAR

ahmed

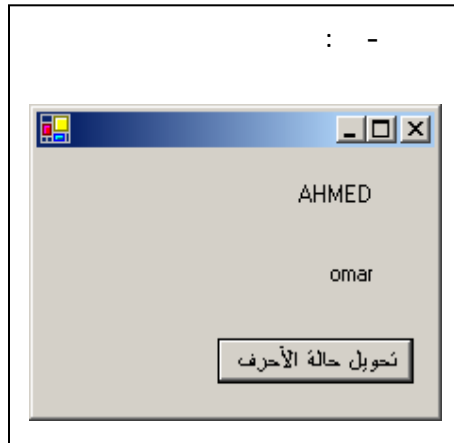
ahmed

ToUpper

OMAR

ToLower

( - )



:



:

```
Dim str As String = " ahmed "  
Label1.Text = str.Trim()
```

ahmed

str

-

Trim

-

:

PadLeft

:

-

:



```
Dim str1 As String = "MyString"  
Label1.Text = str1.PadLeft(10)
```

. MyString

-

PadLeft

-

MyString

"MyString "

```
Dim str1 As String = "MyString"  
Label1.Text = str1.PadRight(10)
```



.PadRight



```
Dim str1 As String = "MyString"  
Label1.Text = str1.PadRight(10)
```

. MyString

PadRight

MyString

" MyString"



## indexOf

	<pre> <b>IndexOf</b>( _     ByVal <u>value</u> As <u>Char</u> _ ) As <u>Integer</u>                     </pre>
	<pre> <b>IndexOf</b>( _     ByVal <u>value</u> As <u>String</u> _ ) As <u>Integer</u>                     </pre>
	<pre> <b>IndexOf</b>( _     ByVal <u>value</u> As <u>Char</u>, _     ByVal <u>startIndex</u> As <u>Integer</u> _ ) As <u>Integer</u>                     </pre>
	<pre> <b>IndexOf</b>( _     ByVal <u>value</u> As <u>String</u>, _     ByVal <u>startIndex</u> As <u>Integer</u> _ ) As <u>Integer</u>                     </pre>
	<pre> <b>IndexOf</b>( _     ByVal <u>value</u> As <u>Char</u>, _     ByVal <u>startIndex</u> As <u>Integer</u>, _     ByVal <u>count</u> As <u>Integer</u> _ ) As <u>Integer</u>                     </pre>
	<pre> <b>IndexOf</b>( _     ByVal <u>value</u> As <u>String</u>, _     ByVal <u>startIndex</u> As <u>Integer</u>, _     ByVal <u>count</u> As <u>Integer</u> _ ) As <u>Integer</u>                     </pre>



```
Dim str As String = "salamat"  
Label1.Text = str.IndexOf("a")
```

salamat

-I

a

-II

a

```
Dim str As String = "salamat"  
Label1.Text = str.IndexOf("a",4)
```

.salamat

-I

a

-II

a

```
Dim str As String = "salamat"  
Label1.Text = str.IndexOf("m",3,2)
```

.salamat

-I

m

-II

m

	-

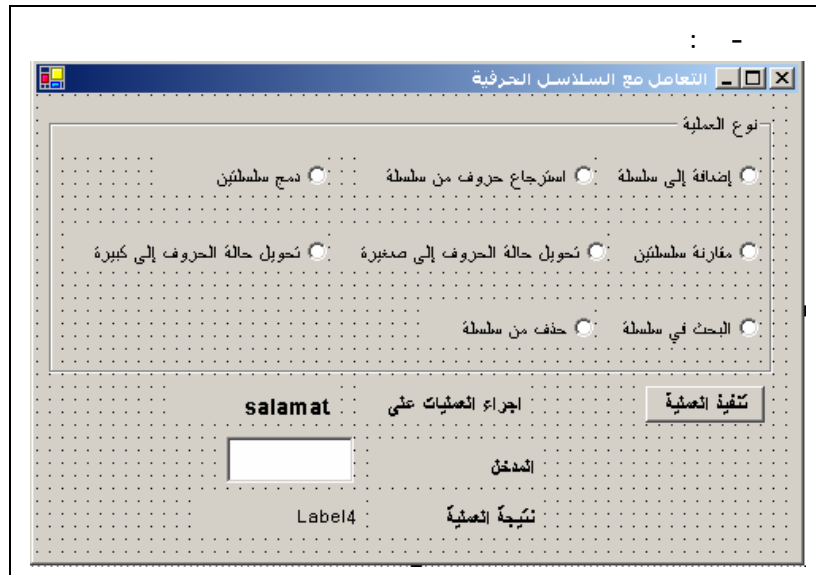
n



N

:

( - )



salamat

.salamat

-

:

القيمة	الخاصية	اسم الأداة
WorkWithString	Name	Form
التعامل مع السلاسل الحرفية	Text	
True	RightToLeft	
Simplified Arabic	Font	
rdoAdd	Name	RadioButton
إضافة إلى سلسلة	Text	
rdoSub	Name	RadioButton
استرجاع حروف من سلسلة	Text	
rdoCon	Name	RadioButton
دمج سلسلتين	Text	
rdoComp	Name	RadioButton
مقارنة سلسلتين	Text	
rdoSmall	Name	RadioButton
تحويل حالة الحروف إلى صغيرة	Text	
rdoCap	Name	RadioButton
تحويل حالة الحروف إلى كبيرة	Text	
rdoSearch	Name	RadioButton
البحث في سلسلة	Text	
rdoRemove	Name	RadioButton
حذف من سلسلة	Text	
Button1	Name	Button
تنفيذ العملية	Text	
Label2	Name	Label
اجراء العمليات على السلسلة	Text	
lblSal	Name	Label
salamat	Text	
Label3	Name	Label
المدخل	Text	
txtInput	Name	TextBox
	Text	
Label5	Name	Label
نتيجة العملية	Text	
lblRes	Name	Label
Label4	Text	

-

-

:



```

If rdoAdd.Checked Then
    If txtInput.Text.Trim <> "" Then
        lblRes.Text = lblSal.Text.Insert(2, txtInput.Text)
    End If
End If
If rdoSub.Checked Then
    If txtInput.Text.Trim <> "" And IsNumeric(txtInput.Text) Then
        lblRes.Text = lblSal.Text.Substring(txtInput.Text)
    End If
End If
If rdoCon.Checked Then
    If txtInput.Text.Trim <> "" Then
        lblRes.Text = String.Concat(lblSal.Text, txtInput.Text)
    End If
End If
If rdoComp.Checked Then
    If txtInput.Text.Trim <> "" Then
        lblRes.Text = String.Compare(lblSal.Text, txtInput.Text)
    End If
End If
If rdoSmall.Checked Then
    If txtInput.Text.Trim <> "" Then
        lblRes.Text = txtInput.Text.ToLower
    End If
End If
If rdoCap.Checked Then
    If txtInput.Text.Trim <> "" Then
        lblRes.Text = txtInput.Text.ToUpper
    End If
End If
If rdoSearch.Checked Then
    If txtInput.Text.Trim <> "" Then
        lblRes.Text = lblSal.Text.IndexOf(txtInput.Text)
    End If
End If
If rdoRemove.Checked Then
    If txtInput.Text.Trim <> "" And IsNumeric(txtInput.Text) Then
        lblRes.Text = lblSal.Text.Remove(txtInput.Text, 1)
    End If
End If
    
```

-  
-  
-  
-

( - )

m

.m



# ADO.NET

## ADO.NET

-CLASSES-

ADO.NET

.ADO.NET

ADO.NET

: (CSV)

.ActiveDirectory Excel

.XML

.MySQL SQLServer

**ADO.NET**

ADO.NET

-Framework-

OO -

- garbage collection-

. multi-language

-design

ADO.NET

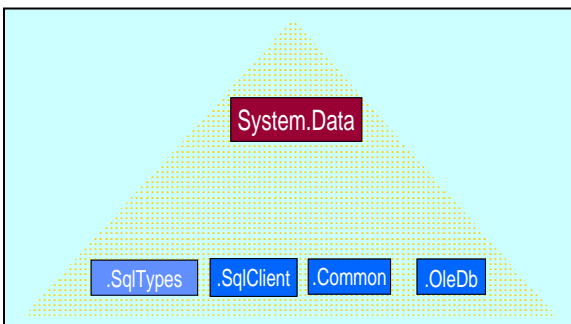
ADO

ADO

ADO

System.data

System.xml





## :ADO ADO.NET

ADO.NET                      ADO  
 .ADO                                      ADO.NET

ADO.NET	ADO	
		xml
DATASET XML	ADO	
DataSet	Recordset	
.DataRelation		
	.Recordset	
Dataset .DataAdaptor		

## : ADO.NET

ADO.NET -ObjectModel-

.Dataset -

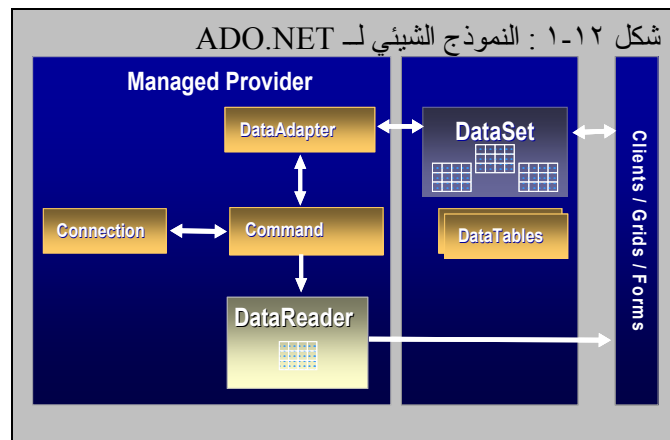
.net DataProvider Classes -

## : Dataset

Dataset

## : .net DataProvider Classes

( - ) .Sql Server .net Data Provider



## ADO.NET

oledb

: Oledb -

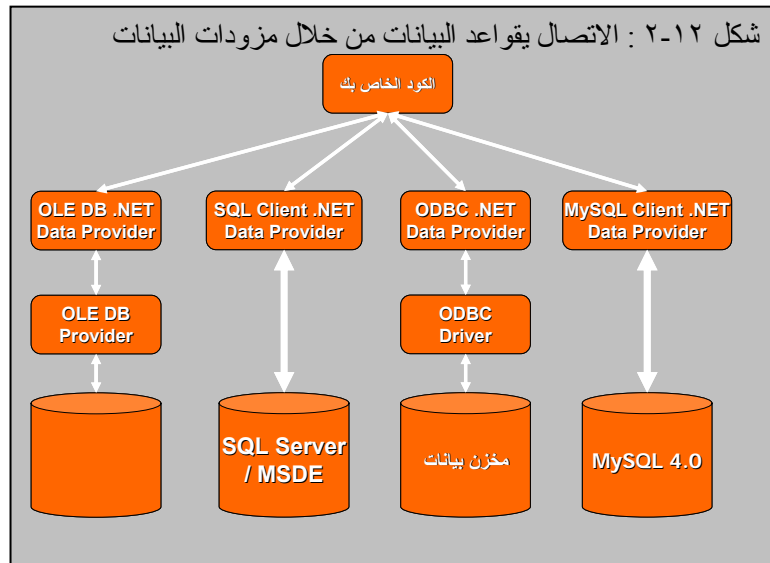
MySQL

: -Managed Data Provider-

Native -

-Provider

( - )



. ODBC Oledb

.Oledb -

.SQLServer -



System.Data.SqlClient  
System.Data.OleDb

SQLServer

-Classes-

: -Connection-

System.Data.OleDb.OleDbConnection  
System.Data.SqlClient.SqlConnection

Stored :

: -Command-

Procedure

System.Data.OleDb.OleDbCommand  
System.Data.SqlClient.SqlCommand

: -DataReader-

-  
System.Data.OleDb.OleDbDataReader  
System.Data.SqlClient.SqlDataReader

- : -DataAdaptor- -  
-Dataset  
:  
..

System.Data.OleDb.OleDbDataAdapter  
System.Data.SqlClient.SqlDataAdapter

.-Connection-

System.Data.OleDb

Oledb



-Connection String-

## : -ConnectionString-

.I

### -Parameters-

ole .OLE DB	Provider
	DataSource
	Intial Catalog
	Integrated Security
	ConnectionTimeout
	UserID/Password

ole

قاعدة البيانات	اسم مزود Ole
Microsoft Access	Microsoft.Jet.OLEDB.4.0
Microsoft SQL Server	SQLOLEDB.1
Oracle	MSDAORA.1

: .II

SQLServer -

Provider=sqloledb;Data Source=SQL Server DB;Initial Catalog=Northwind;User ID=UserName;Password=password

-

Provider=MSDAORA;Data Source=Oracle DB;User ID=UserName;Password=password

-

Provider=Microsoft.Jet.OLEDB.4.0;Data Source=c:\Mydb\db1.mdb;

-Directory-



.III

: .

.MyConnString.txt      TextFile      -

.(Universal Data Link      ) UDL      txt      -

( - )      -

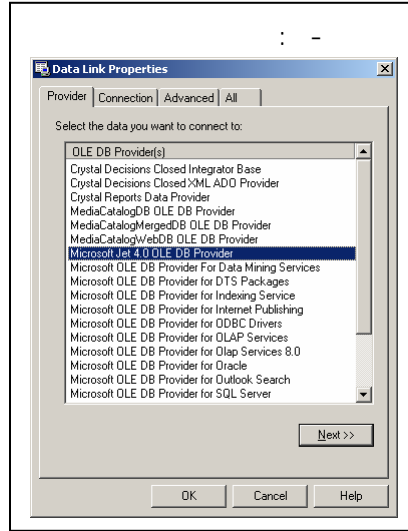
Provider      -

Microsoft.Jet.OLEDB.4

. Connection      -

-



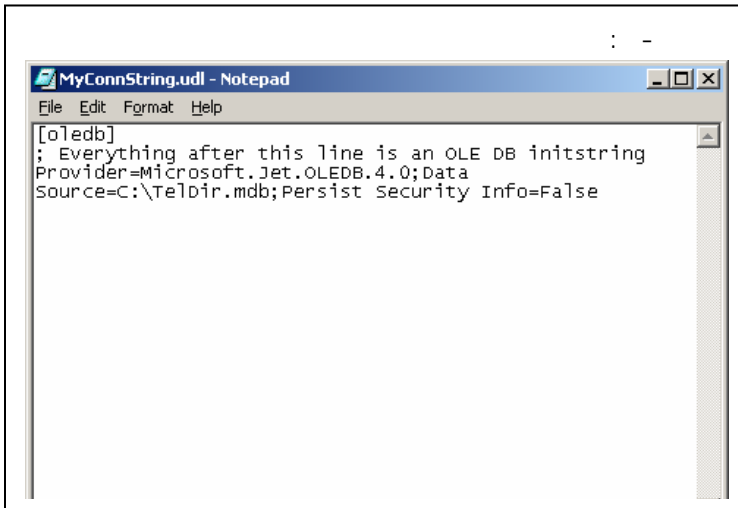


.Notepad

MyConnString.udl

Provider

( - )



Open

Dispose Close



```
if (ConnectionObject.State = ConnectionState.Open)
```

:

Update Fill



.Try Catch

-Exception-

.Onerror

.Onerror

:

```
Try
Catch
Finally
End Try
```

:

```
.Try -
```

```
. Catch -
```

```
.Catch -
```

```
Finally -
```



```
Dim con As New OleDb.OleDbConnection()
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source="
& _
"C:\TelDir.mdb;Persist Security Info=False"
Try
    con.Open()
Catch Exp As OleDb.OleDbException
    Labell1.Text = Exp.Message
Finally
    con.Close()
End Try
```

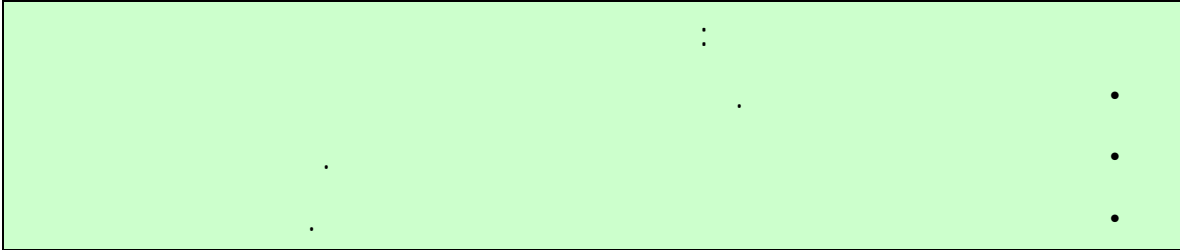
:

-

-

-

.Open -  
.Catch -  
Oledb Catch -



:

: .I

: .II

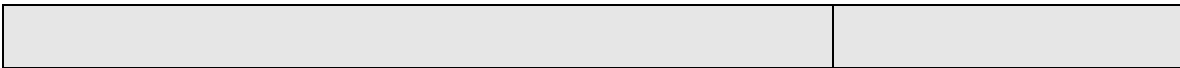
-

-

**.III** :

:  
:  
-  
-

The diagram is enclosed in a rectangular border. In the top right corner, there is a yellow lightbulb icon with a blue glow. The text 'Close' is positioned in the upper left area. Below it, 'Dispose' is centered. Further down, 'Max' and 'Min' are placed side-by-side. At the bottom right, the word 'Timeout' is written. A vertical line of dashes is on the right side, and two black dots are positioned to the right of the 'Dispose' and 'Max' lines.

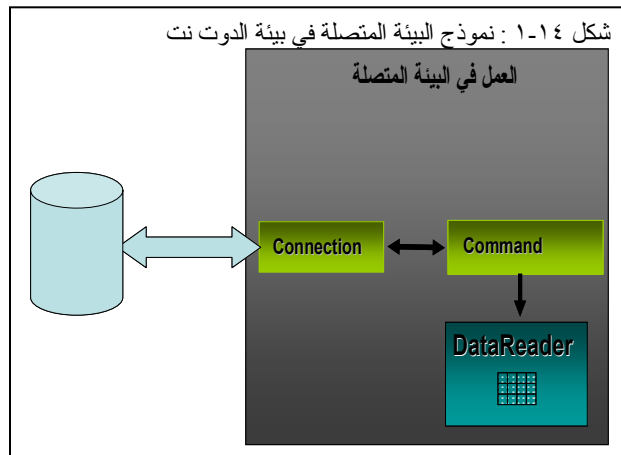


	Connection Lifetime
	Max Pool Size
	Min Pool Size

# ADO.NET

## ADO.NET

( - )





## : Command

: .I

: Command

.-SQL Statement -

-

.-Stored Procedure-

-

.DataReader

: .II

:

System.Data.OleDb.OleDbCommand
System.Data.SqlClient.SqlCommand

: .III

:

		Name
		Connection
	Text	CommandType
	StoredProcedur	
	TableDirect	

-	CommandType	CommandText
-		
-		

Command Text



.CommandType

:

.IV

:

	ExecuteReader
	ExecuteNonQuery
	ExecuteScalar



ADO ADO.NET

:

-

-

## SQL



ADO

ADO.NET



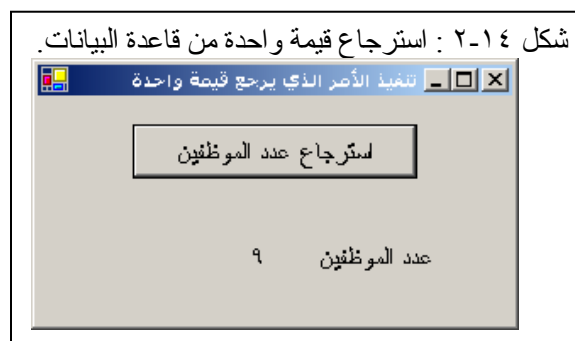
:

```
Dim con As New OleDb.OleDbConnection()  
Dim sql As String = "select count(*) from person"  
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _  
"Data Source=C:\TelDir.mdb;Persist Security Info=False"  
Try  
    con.Open()  
    Dim cmd As New OleDb.OleDbCommand(sql, con)  
    Label1.Text = cmd.ExecuteScalar().ToString  
Catch e1 As OleDb.OleDbException  
    Label1.Text = e1.Message  
End Try
```

:

## ExecuteScalar

.( - )





c:\

( - )



.ExecuteReader

:

System.Data.OleDb.OleDbDataReader System.Data.SqlClient.SqlDataReader
--

:

	-
False                      True	Read
	Item
	GetValues
Null	isDBNull
	close

sql	RecordsAffected
-----	-----------------

:

```

Dim con As New OleDb.OleDbConnection()
Dim rdName As OleDb.OleDbDataReader
Dim sql As String = "select * from person"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
Try
    con.Open()
    Dim cmd As New OleDb.OleDbCommand(sql, con)
    rdName = cmd.ExecuteReader()
    Do While rdName.Read()
        ListBox1.Items.Add(rdName.GetString(1))
    Loop
    rdName.Close()
Catch e1 As OleDb.OleDbException
    Label1.Text = e1.Message
End Try

```

:

		-
		-
		-
	ExecuteReader	-
False	Read	-
		.Read
		-

:

cmdExecute	Name	Button1
	Text	
Label1	Name	Label1
	Text	
ListBox1	Name	
	Text	Form1
Yes	RightToLeft	



```
rdName = cmd.ExecuteReader(CommandBehavior.CloseConnection)
```

•

:

-  
-  
-

## DDL

:

•  
•

-

View -

Trigger -

:

	Create
	ALTER
	DROP

## DCL

:

	Grant
	DENY

## DCL DDL

## ExecuteNonQuery

## DCL DDL

```
Dim con As New OleDb.OleDbConnection()  
Dim sql As String = "create table MyNewTable (t5 varchar(10))"  
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _  
"Data Source=C:\TelDir.mdb;Persist Security Info=False"  
Try  
    con.Open()  
    Dim cmd As New OleDb.OleDbCommand(sql, con)  
    Label1.Text = cmd.ExecuteNonQuery().ToString  
Catch e1 As OleDb.OleDbException  
    Label1.Text = e1.Message  
End Try
```

ExceuteNonQuery

## DML





	INSERT
	UPDATE
	DELETE

## DML

ExecuteNonQuery

:

:

cmdExecute	Name	Button1
	Text	
Label1	Name	Label1
	Text	
Label2	Name	Label2
	Text	
Label	Name	Label
	Text	
Textbox1	Name	
	Text	
Textbox1	Name	TextBox2
	Text	
Yes	RightToLeft	Form1

-2

```
Dim con As New OleDb.OleDbConnection()
Dim sql As String = "insert into person (name,ename) values " & _
"(' " & TextBox1.Text & "','" & TextBox2.Text & "')"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source" & _
```

```
"=C:\TelDir.mdb;Persist Security Info=False"  
Try  
    con.Open()  
    Dim cmd As New OleDb.OleDbCommand(sql, con)  
    Label1.Text = cmd.ExecuteNonQuery().ToString  
Catch e1 As OleDb.OleDbException  
    Label1.Text = e1.Message  
End Try
```

:

•

-

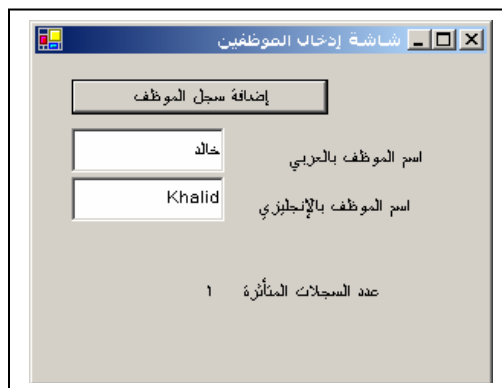
-

ExceuteNonQuery

-

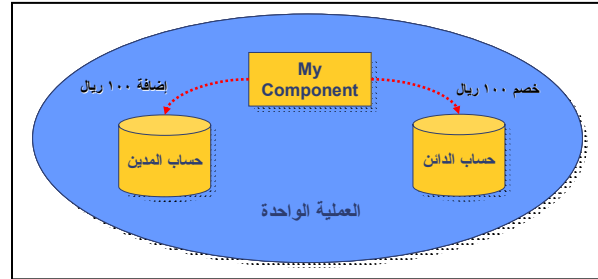
-

-



-Commit-

. - RollBack -



:

BeginTransaction

- Connection

Oletransaction

-

Transaction

-

Commit

-

Rollback

:



```
Dim con As New OleDb.OleDbConnection()
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\TelDir.mdb;Persist Security Info=False"
con.Open()
Dim cmd As New OleDb.OleDbCommand()
Dim trans As OleDb.OleDbTransaction = con.BeginTransaction
cmd.Connection = con
cmd.Transaction = trans
Dim sql1 As String = "insert into person (name,ename) values (" &
TextBox1.Text & "," & TextBox2.Text & ")"
Dim sql2 As String = "insert into Teltype (name,ename) values (" &
TextBox3.Text & "," & TextBox4.Text & ")"
Try
cmd.CommandText = sql1
cmd.ExecuteNonQuery()
cmd.CommandText = sql2
cmd.ExecuteNonQuery()
```

```
trans.Commit()  
Catch e1 As OleDb.OleDbException  
trans.Rollback()  
Label1.Text = e1.Message  
End Try
```

```
HRMTran=HRMCon.BeginTransaction(isolationLevel.Serializable
```

```
: Dirty Read -  
-Commit-
```

```
: Phantom Read -
```

IsolationLevel                      OLEDBTransaction  
 BeginTransaction

<b>.Phantom Read   Dirty Read</b>	<b>ReadUncommitted</b>
<b>Phantom Read   Dirty Read</b>	<b>Serializable</b>
<b>Dirty Read</b>  <b>Phantom Read</b>	<b>Readcommitted</b>

# DATASET

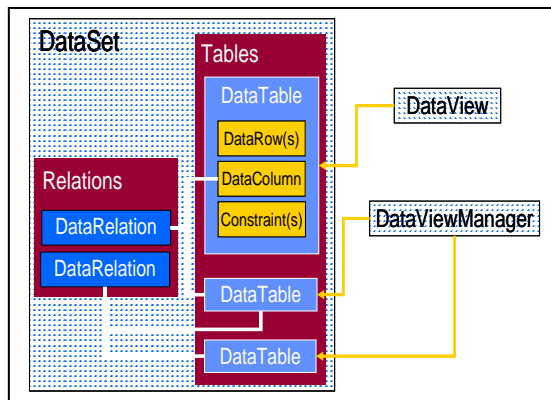
:

## ADO.NET

:

: Dataset

-



:

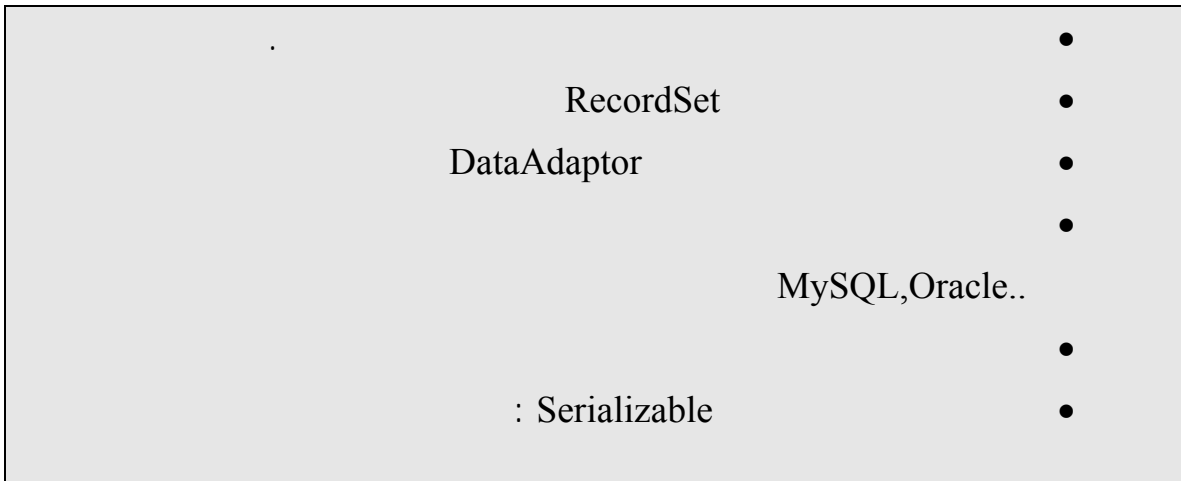
: DataTable

.a

: DataColumn

.b

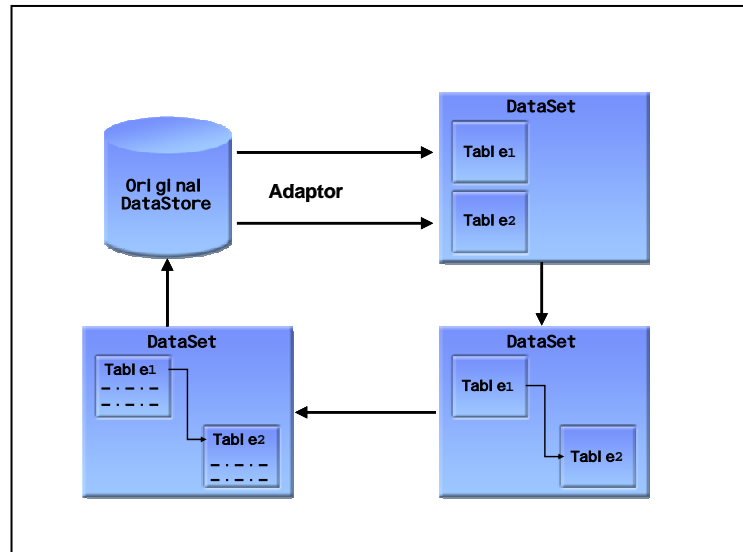
.c : DataRow  
.d : DataConstarint  
Unique Constraint  
.Null  
.e : DataRelation  
.f : DataView



- RecordSet
- DataAdaptor
- MySQL, Oracle..
- : Serializable

: Data Adaptor -





```
Dim dsTelDir As New DataSet()
```

```
.dsTelDir
```

```
Dim dtPerson As DataTable = dsTelDir.Tables.Add()
```

```
. dtPerson
```

```
TableX
```

```
Dim colName As DataColumn = dtPerson.Columns.Add("Name", GetType(System.String))
colName.AllowDBNull = False
colName.Unique = True
```

```
.String Name - - .a
```

```
Null .b
```

```
Unique .c
```



```
Dim dsTelDir As New DataSet()
```

```
Dim dtPerson As DataTable = dsTelDir.Tables.Add()
```

```
Dim colName As DataColumn = dtPerson.Columns.Add("Name",
GetType(System.String))
```

```
colName.AllowDBNull = False  
colName.Unique = True  
MessageBox.Show(dtPerson.TableName)  
MessageBox.Show(dsTelDir.Tables.Count.ToString())
```

:

-

-

:

-

-



:

```
Dim myTable As DataTable = New DataTable()  
Dim myColumn As DataColumn  
myColumn = New DataColumn()  
myColumn.DataType = System.Type.GetType("System.String")  
myColumn.ColumnName = "FirstName"  
myTable.Columns.Add(myColumn)  
myTable.PrimaryKey = New DataColumn() {myTable.Columns("FirstName")}
```

:

-

-

-

:

-

```
myTable.PrimaryKey = New DataColumn() {myTable.Columns("FirstName")}
```

Expression

.DataColumn



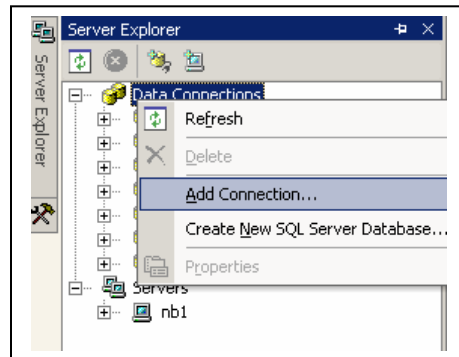
:

```
Dim ColUnitPrice As New DataColumn("Price", GetType(System.Decimal))
Dim ColQuantity As New DataColumn("Quantity", GetType(System.Decimal))
Dim TotalCost As New DataColumn("Total", GetType(System.Decimal))
Dim mytable As New DataTable()
TotalCost.Expression = "Price * Quantity"
mytable.Columns.Add(ColUnitPrice)
mytable.Columns.Add(ColQuantity)
mytable.Columns.Add(TotalCost)
Dim myRow As DataRow
myRow = mytable.NewRow()
myRow("Price") = 5
myRow("Quantity") = 1
mytable.Rows.Add(myRow)
Label2.Text = mytable.Rows(0).Item(2)
```

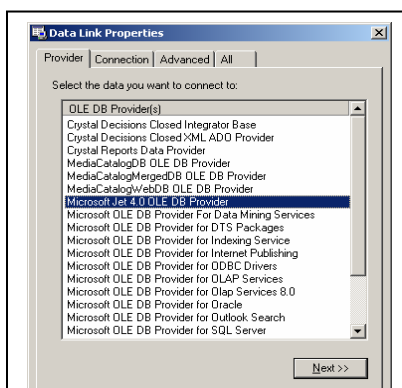
:

	TotalCost	ColQuantity	ColUnitPrice	-
				-
Price *			TotalCost	-
				-
			Quantity	-
				-
				-
				-
				-
				-
				-
				-
				-

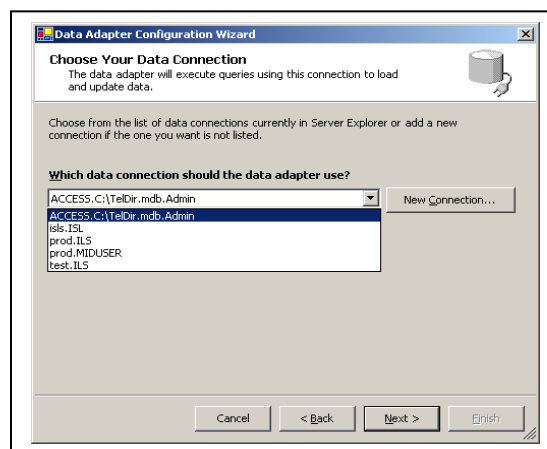
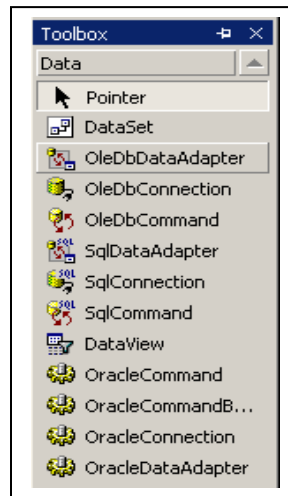
:  
.  
-  
-  
.  
.- Prototype -  
.  
.Teldir -

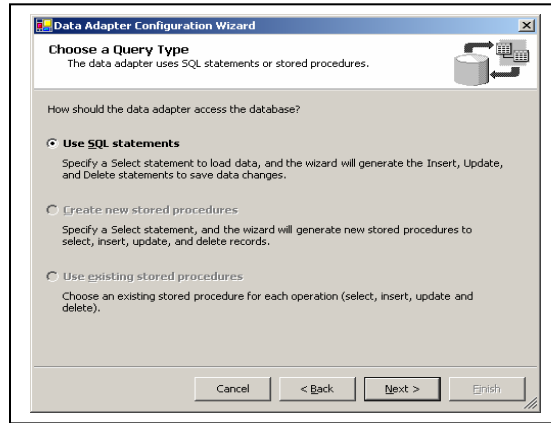


.Add Connection  
Data Connection  
Provider  
Connection Microsoft.Jet.OLEDB.4  
Teldir



Data



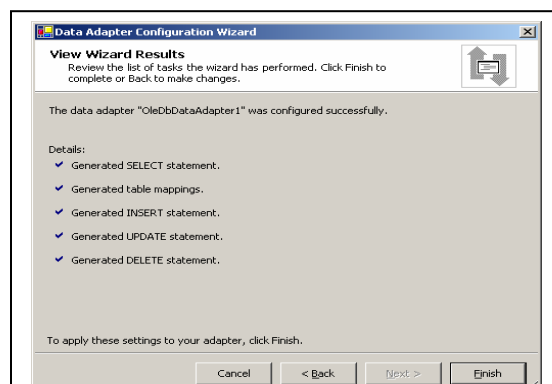
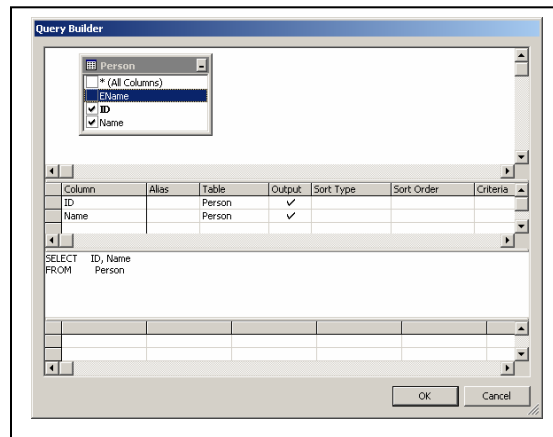


SQL

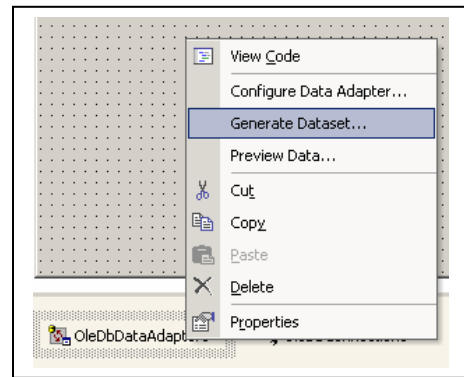
QueryBuilder

Person

Name ID



## Generate Data Set

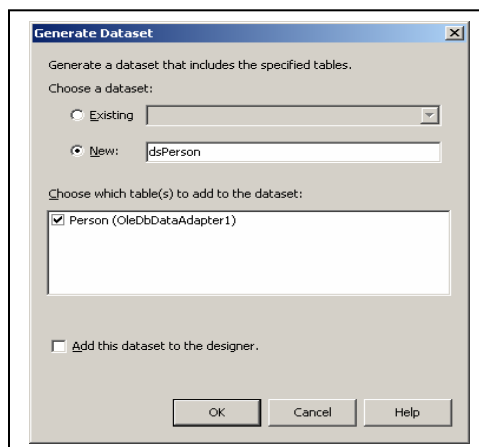


Add This

dsPerson

Dataset to The Designer

dsPerson.xsd



Datagrid



```
Dim ds As New DataSet()  
Me.OleDbConnection1.Open()  
Try  
    Me.OleDbDataAdapter1.Fill(ds)  
    DataGrid1.DataSource = ds.Tables(0)  
Catch Ex As OleDb.OleDbException  
    MessageBox.Show(Ex.Message)  
End Try  
Me.OleDbConnection1.Close()
```

:

Datagrid

ID	Name
1	خالد
2	احمد
3	حسام
4	دخيل

cmdFillGrid	Name	Button1
	Text	
Label1	Name	Label1
	Text	
DataGrid1	Name	dataGrid
	Text	Form1
Yes	RightToLeft	

```

Dim con As New OleDb.OleDbConnection()
Dim dsperson As New DataSet()
Dim sql As String = "select id,name from person"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
Try
    Dim da As New OleDb.OleDbDataAdapter(sql, con)
    da.Fill(dsperson, "Person")
    DataGrid1.DataSource = dsperson.Tables("Person")
Catch e1 As OleDb.OleDbException
    Label1.Text = e1.Message
End Try
    
```

sql

sql

.sql

Datagrid



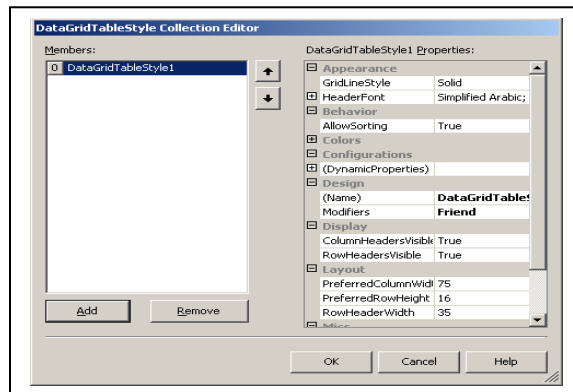
id

.Name

Person

: da

.TableStylas

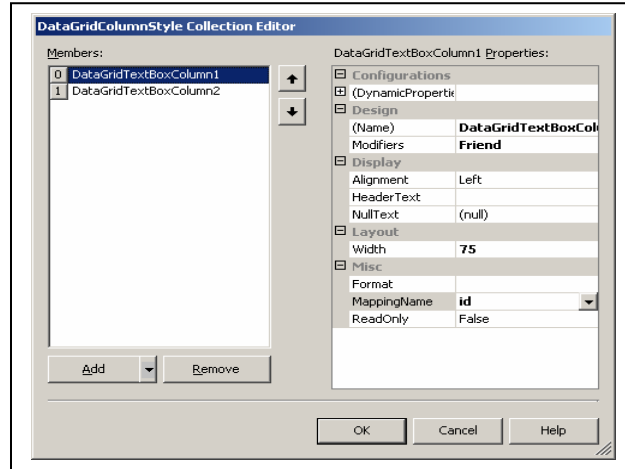


DataGridTableStyle1

.Add

Person

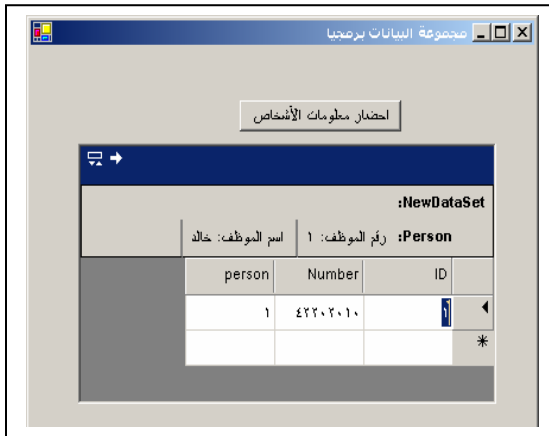
Person MappingName -  
..



GridColumnStyle Add  
id id MappingName  
HeaderText Person

HeaderText .name MappingName  
Ok Ok





:

```

Dim con As New OleDb.OleDbConnection()
Dim Myds As New DataSet()
Dim sql(1) As String
sql(0) = "select id,name from person"
sql(1) = "select ID,Number,person from Phone"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
    Try
        Dim daPerson As New OleDb.OleDbDataAdapter(sql(0), con)
        Dim daPhone As New OleDb.OleDbDataAdapter(sql(1), con)
        daPerson.Fill(Myds, "Person")
        daPhone.Fill(Myds, "Phone")
        Myds.Relations.Add("الموظف هاتف أرقام", " & _
" Myds.Tables("Person").Columns(0), Myds.Tables("Phone").Columns(2), True)
        DataGridView1.DataSource = Myds
    Catch e1 As OleDb.OleDbException
        Label1.Text = e1.Message
    End Try

```

:

..Myds

Person

Phone

Phone



```
Dim SearchRow() As DataRow =  
Myds.Tables("Person").Select("name='عبد الله'")  
    MessageBox.Show(SearchRow(0).Item(0))
```



```
Dim SearchRow() As DataRow =  
Myds.Tables("Person").Select("name='عبد الله'", "id DESC")  
    MessageBox.Show(SearchRow(0).Item(0))
```

DESC



:

```
Dim DV As New DataView(Myds.Tables("Person"))  
DV.RowFilter = "name='عبد الله'"  
DV.Sort = "ename"
```

:



cmdSearch	Name	Button1

	Text	
Label1	Name	Label1
	Text	
btnNewData	Name	RadioButton1
	Text	
rdoDSSearch	Name	RadioButton1
	Text	
DataGrid1	Name	dataGrid
	Text	Form1
Yes	RightToLeft	

```

Public Class SearchProj
    Inherits System.Windows.Forms.Form
    Dim Myds As New DataSet()
    Private Sub cmdSearch_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdSearch.Click
        Dim con As New OleDb.OleDbConnection()
        Dim sql As String = "select id,name,ename from person"
        con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
            "Data Source=C:\TelDir.mdb;Persist Security
Info=False"
        If rdoNewData.Checked Then
            Myds.Clear()
            Dim daPerson As New OleDb.OleDbDataAdapter(sql, con)
            daPerson.Fill(Myds, "Person")
        End If
        If Myds.Tables.Count = 0 Then Exit Sub
        Dim DV As New DataView(Myds.Tables("Person"))
        DV.RowFilter = "name Like '" & txtEmpName.Text & "''
        DV.Sort = "ename"
        DataGrid1.DataMember = ""
        DataGrid1.DataSource = Nothing
        DataGrid1.DataSource = DV
    End Sub
End Class

```

:

-

-



- 
- 
- 
- 

البحث عن الموظفين

خيارات البحث

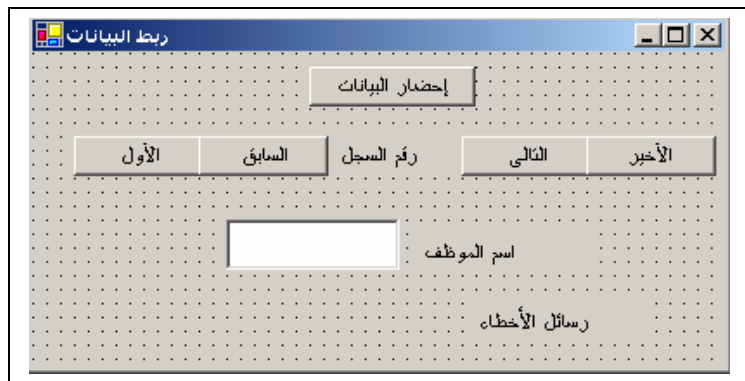
البحث بنسخة محدثة  البحث في مجموعة البيانات

اسم الموظف:

ename	name	id
أحمد	عبد الله	7
مديسون	عبد الله	36

\*

# Data Binding



cmdNext	Name	Button1
	Text	
cmdPerv	Name	Button
	Text	
cmdFirst	Name	Button
	Text	
cmdLast	Name	Button
	Text	
btnFetch	Name	Button
	Text	
Label1	Name	Label1
	Text	
Label1	Name	Label

	Text	
lblCurrent	Name	Label
	Text	
ListBox1	Name	
	Text	Form1
Yes	RightToLeft	

:

Edit → Advance → Word Wrap

:

```
Public Class DataBinding
    Inherits System.Windows.Forms.Form
    Dim WithEvents bmb As BindingManagerBase
    Dim Myds As New DataSet()

    Private Sub btnFetch_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnFetch.Click
        Dim con As New OleDb.OleDbConnection()
        Dim sql As String
        sql = "select id,name from person"
        con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
        Try
            Dim daPerson As New OleDb.OleDbDataAdapter(sql, con)
            daPerson.Fill(Myds, "Person")
        Catch e1 As OleDb.OleDbException
            Label1.Text = e1.Message
        End Try
        txtEmpName.DataBindings.Add("Text", Myds, "Person.name")
        bmb = Me.BindingContext(Myds, "Person")
        bmb.Position = bmb.Count
        bmb.Position = 0
    End Sub

    Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNext.Click
        bmb.Position += 1
    End Sub

    Private Sub cmdPrev_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdPrev.Click
        bmb.Position -= 1
    End Sub

    Private Sub cmdLast_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdLast.Click
```

```

        bmb.Position = bmb.Count - 1
    End Sub

    Private Sub cmdFirst_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles cmdFirst.Click
        bmb.Position = 0
    End Sub

    Private Sub bmb_PositionChanged(ByVal sender As Object, ByVal e As
System.EventArgs) Handles bmb.PositionChanged
        lblCurrent.text = String.Format("{0} of {1}", bmb.Position + 1,
bmb.Count)
    End Sub
End Class

```

:

```
Dim WithEvents bmb As BindingManagerBase
```

-

btnFetch

-

:

```

txtEmpName.DataBindings.Add("Text", Myds, "Person.name")
bmb = Me.BindingContext(Myds, "Person")
bmb.Position = bmb.Count
bmb.Position = 0

```

Person.name      txtEmpName

.Text

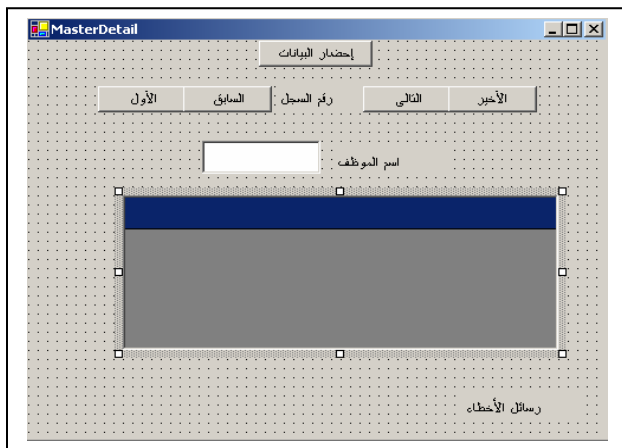
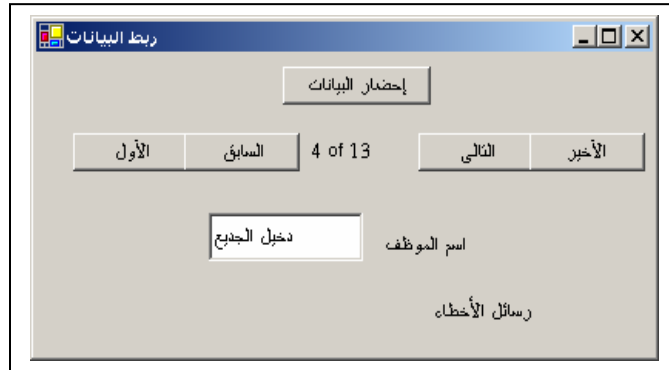
.Myds

.Checked

.Position

bmb\_PositionChanged -

-

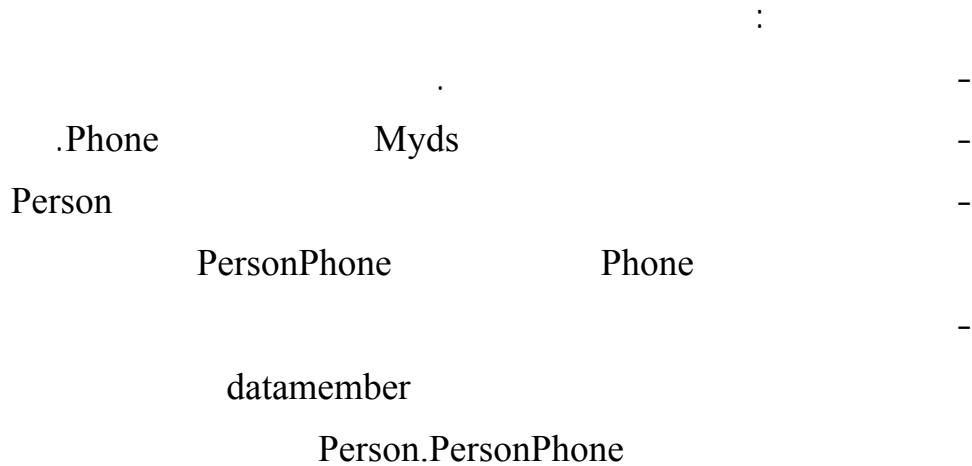


```
Dim con As New OleDb.OleDbConnection()
```

```

Dim Myds As New DataSet()
Dim sql As String
sql = "select id,name from person"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
Try
    Dim daPerson As New OleDb.OleDbDataAdapter(sql, con)
    daPerson.Fill(Myds, "Person")
    Dim daPhone As New OleDb.OleDbDataAdapter("select & _
ID,Num,phone from Phone", con)
    daPhone.Fill(Myds, "Phone")

Myds.Relations.Add("PersonPhone",Myds.Tables("Person").Columns(0), & _
Myds.Tables("Phone").Columns(2), True)
Catch e1 As OleDb.OleDbException
    Label1.Text = e1.Message
End Try
Myds.WriteXml("da.xml")
txtEmpName.DataBindings.Add("Text", Myds, "Person.name")
DataGrid1.DataSource = Myds
DataGrid1.DataMember = "Person.PersonPhone"
bmb = Me.BindingContext(Myds, "Person")
bmb.Position = bmb.Count
bmb.Position = 0
    
```









:

NewRow

-

-

-

:



:

```
Dim dtPerson As DataTable = Myds.Tables(0)
Dim dr As DataRow = dtPerson.NewRow()
dr("name") = "عبدالله"
dr("ename") = "Abduallah"
dtPerson.Rows.Add(dr)
MessageBox.Show(Myds.Tables(0).Rows.Count)
```

:

-

-

-

-



:

```
Myds.Tables("Person").Rows.Add(New Object() {Nothing, " ", "Abduallah"})
```

Nothing



:

```
Dim dtPerson As DataTable = Myds.Tables(0)
Dim dr As DataRow = dtPerson.Rows(dtPerson.Rows.Count - 1)
dr("name") = "عبدالله"
dr("ename") = "Abduallah"
```



```
Dim dtPerson As DataTable = Myds.Tables(0)
Dim dr As DataRow
For Each dr In dtPerson.Rows
    dr("ename") = ""
Next
```

:

Delete

-

:

```
Myds.Tables(0).Rows(1).Delete()
```

Delete

Remove

-

:

```
Dim dr As DataRow = Myds.Tables(0).Rows(1)
Myds.Tables(0).Rows.Remove(dr)
```

## .Rowstate

:

	Added
	Deleted
	Detached
	Modified
	Unchanged

## :Rowstate



```
Dim dtPerson As DataTable = Myds.Tables(0)
Dim dr As DataRow = dtPerson.NewRow()
MessageBox.Show(dr.RowState)
dr("name") = "عبدالله"
dr("ename") = "Abduallah"
dtPerson.Rows.Add(dr)
MessageBox.Show(dr.RowState)
```

```

Dim Myds As New DataSet()
Dim con As New OleDb.OleDbConnection()
Dim sql As String = "select id,name,ename from person"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
    Try
        Dim daPerson As New OleDb.OleDbDataAdapter(sql, con)
        daPerson.Fill(Myds, "Person")
        Myds.Tables("Person").Rows.Add(New Object() {Nothing,
"عبد الله", "Abdulallah"})
        Dim ComBuilder As New OleDb.OleDbCommandBuilder(daPerson)
        daPerson.InsertCommand = ComBuilder.GetInsertCommand
        daPerson.Update(Myds, "Person")
    Catch e1 As OleDb.OleDbException
        MessageBox.Show(e1.Message)
    End Try

```

.CommandBuilder

Update



```
daPerson.DeleteCommand = ComBuilder.GetDeleteCommand
daPerson.UpdateCommand = ComBuilder.GetUpdateCommand
```

:



:

```
Dim Myds As New DataSet()
    Dim con As New OleDb.OleDbConnection()
    Dim sql As String = "select id,name,ename from person"
    con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
        "Data Source=C:\TelDir.mdb;Persist Security
Info=False"
    Try
        Dim daPerson As New OleDb.OleDbDataAdapter(sql, con)
        daPerson.Fill(Myds, "Person")
        Myds.Tables("Person").Rows.Add(New Object() {Nothing,
"حارث", "Harith"})
        If Myds.HasChanges Then
            Dim AffectedDS As DataSet = Myds.GetChanges
            Dim ComBuilder As New
OleDb.OleDbCommandBuilder(daPerson)
            daPerson.InsertCommand = ComBuilder.GetInsertCommand
            daPerson.Update(AffectedDS, "Person")
        End If
        Catch e1 As OleDb.OleDbException
            MessageBox.Show(e1.Message)
        End Try
```

:

Haschanges

-

-

Getchanges

-

## ContinueUpdateOnError



### : ContinueUpdateOnError

```
daPerson.ContinueUpdateOnError = True  
daPerson.Update(Myds, "Person")  
If Myds.HasChanges Then  
End If
```

:

ContinueUpdateOnError

-

-

-



cmdGetData	Name	Button1
	Text	
cmdUpdate	Name	Button1
	Text	
ResolveConfilct	Name	Form

```

Public Class ResolveConfilct
    Inherits System.Windows.Forms.Form
    Dim Myds As New DataSet()
    Dim con As New OleDb.OleDbConnection()
    Dim sql As String = "select id,name,ename from person"
    Dim daPerson As New OleDb.OleDbDataAdapter(sql, con)
    Dim dtPerson As DataTable
    Dim dr As DataRow
    Private Sub cmdUpdate_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles cmdUpdate.Click
    Try
        If Myds.HasChanges Then
            Dim AffectedDS As DataSet = Myds.GetChanges
            Dim ComBuilder As New OleDb.OleDbCommandBuilder(daPerson)
            daPerson.ContinueUpdateOnError = True
            daPerson.UpdateCommand = ComBuilder.GetUpdateCommand
            daPerson.Update(AffectedDS, "Person")
            Dim ConflictRow As DataRow
            Dim ConfilctRowCount As Integer =
AffectedDS.Tables("Person").GetChanges.Rows.Count
            For Each ConflictRow In AffectedDS.Tables("Person").Rows
                If ConflictRow.RowState = DataRowState.Modified Then
                    MessageBox.Show("تحديث تعارض", "تعارض",
                    MessageBoxButtons.YesNo)
                End If
            Next
        End If
    Catch e1 As OleDb.OleDbException
        MessageBox.Show(e1.Message)
    End Try
End Sub

```

```
End Try
Private Sub cmdGetData_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdGetData.Click
    con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
        "Data Source=C:\TelDir.mdb;Persist Security
Info=False"
    daPerson.Fill(Myds, "Person")
    dtPerson = Myds.Tables(0)
    dr = dtPerson.Rows(dtPerson.Rows.Count - 1)
    dr("name") = "عبدالله"
    dr("ename") = "Abduallah"
End Sub
```

:

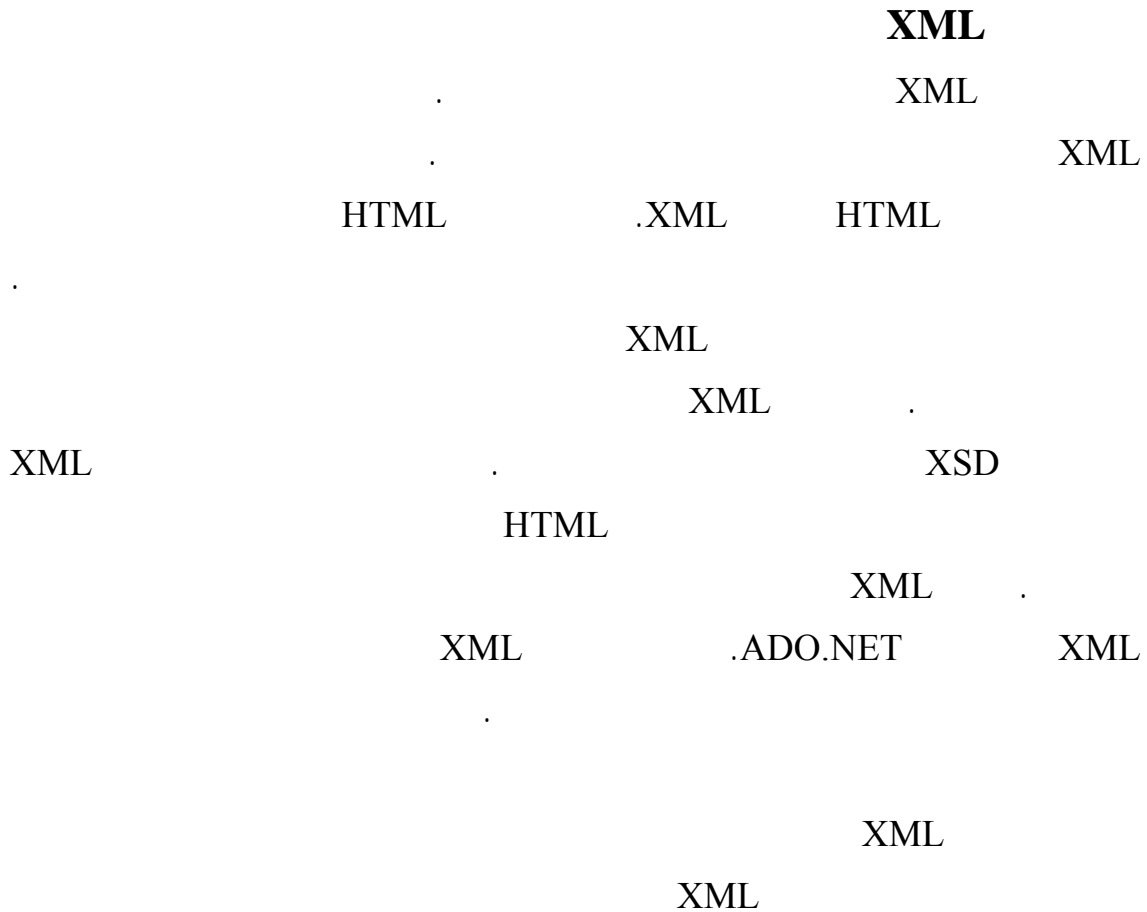
-

-

-



# XML ADO.NET



```
<?xml version="1.0" standalone="yes"?>  
<NewDataSet>  
  <Person>  
    <id>1</id>  
    <name>خالد</name>  
  </Person>  
  <Person>  
    <id>2</id>  
    <name>احمد</name>  
  </Person>  
</NewDataSet>
```

id	name
Person	

## XML

: XML

.XML

-

XML

-

## XML

.WriteXML XML

:

```

Dim con As New OleDb.OleDbConnection()
Dim Myds As New DataSet()
Dim sql As String = "select id,name from person where id=1"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
Try
    Dim daPerson As New OleDb.OleDbDataAdapter(sql, con)
    daPerson.Fill(Myds, "Person")
    Myds.WriteXml("dsPerson.xml")
Catch e1 As OleDb.OleDbException
    Label1.Text = e1.Message
End Try

```

:

-

-

-

XML

WriteXml

-

Bin

dsPerson.xml

.

XML

-

```

<?xml version="1.0" standalone="yes"?>
<NewDataSet>
  <Person>
    <id>1</id>
    <name>خالد</name>
  </Person>
</NewDataSet>

```

## XML

.ReadXml

XML

```
Myds.WriteXml("dsPerson.XML")  
Dim ds As New DataSet()  
ds.ReadXml("dsPerson.XML")
```

:

XML

-

-

-

## XML

.Getxml

XML

XML

.

xml

Text

XML

.

```
TextBox1.Text = Myds.GetXml
```

XML

GetXml

.

## XML

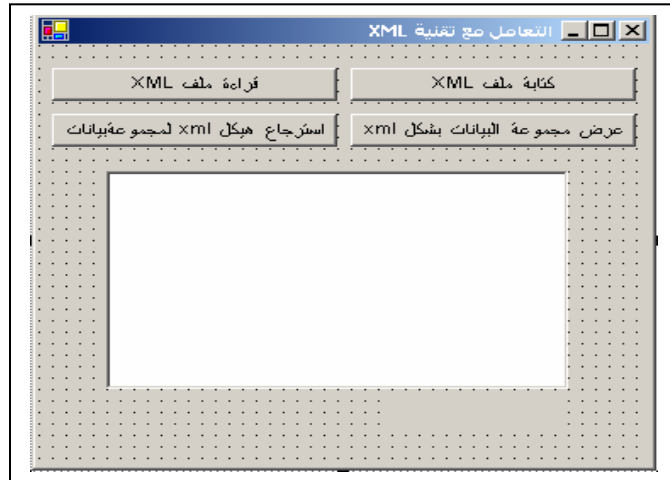
.GetXmlSchema

XML

```
TextBox1.Text = Myds.GetXmlSchema
```

# XML

:



:

cmdWritexml	Name	Button1
XML	Text	
cmdGetXml	Name	Button1
xml	Text	
cmdReadXml	Name	Button1
XML	Text	
cmdGetSchema	Name	Button1
xml	Text	
Label1	Name	Label1
	Text	
TextBox1	Name	TextBox
True	MultiLine	
Both	Scrollbars	
XML	Text	Form1
Yes	RightToLeft	

```

Public Class WriteXml
    Inherits System.Windows.Forms.Form
    Dim Myds As New DataSet()
    Private Sub cmdWritexml_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdWritexml.Click
        Dim con As New OleDb.OleDbConnection()
        Dim sql As String = "select id,name from person where id=1"
        con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
        Try
            Dim daPerson As New OleDb.OleDbDataAdapter(sql, con)
            daPerson.Fill(Myds, "Person")
            Myds.WriteXml("dsPerson.xml")
        Catch e1 As OleDb.OleDbException
            Labell.Text = e1.Message
        End Try
    End Sub
    Private Sub cmdReadXml_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles cmdReadXml.Click
        Myds.WriteXml("dsPerson")
        Dim ds As New DataSet()
        ds.ReadXml("dsPerson")
    End Sub
    Private Sub cmdGetXml_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles cmdGetXml.Click
        TextBox1.Text = Myds.GetXml()
    End Sub
    Private Sub cmdGetSchema_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles cmdGetSchema.Click
        TextBox1.Text = Myds.GetXmlSchema()
    End Sub
End Class

```

:

-

.XML

-

cmdWritexml\_Click

-

.XML

Myds

cmdReadXml\_Click

-

XML

XML

.

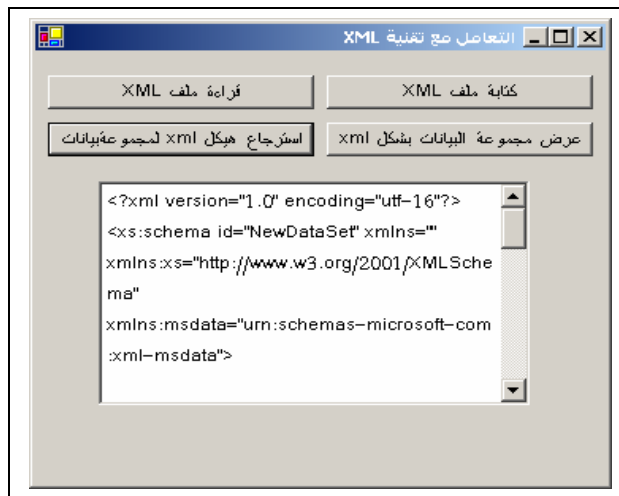
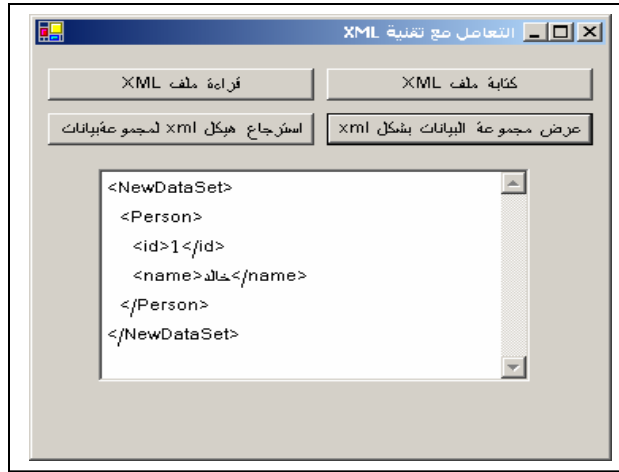
cmdGetXml\_Click

-

XML

cmdGetSchema\_Click -

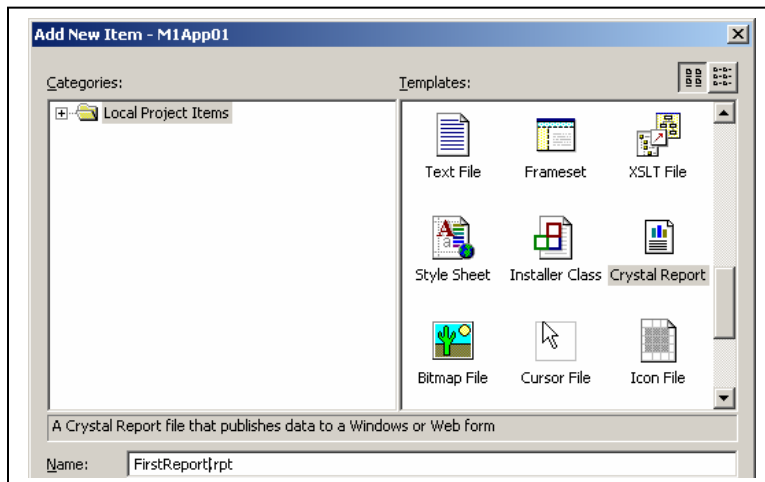
XML



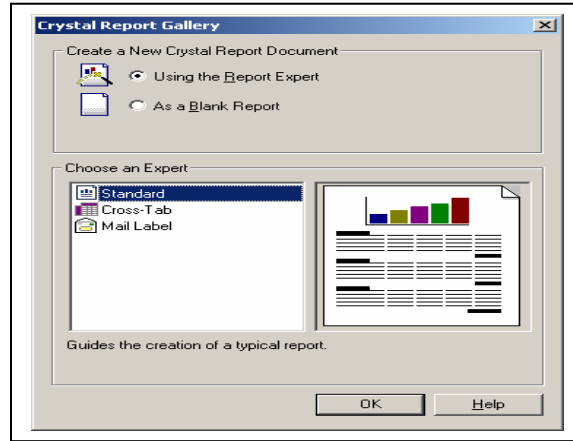
.CrystalDemo

Add

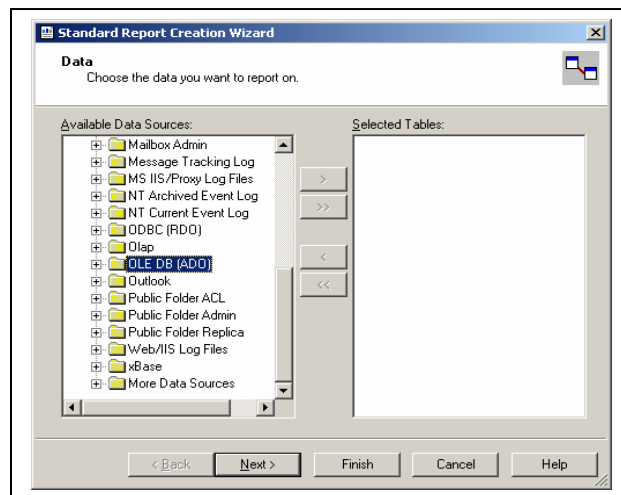
.FirstReport



Ok



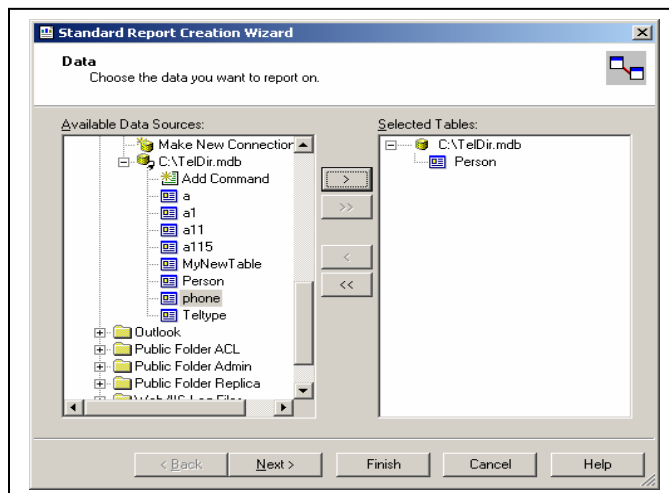
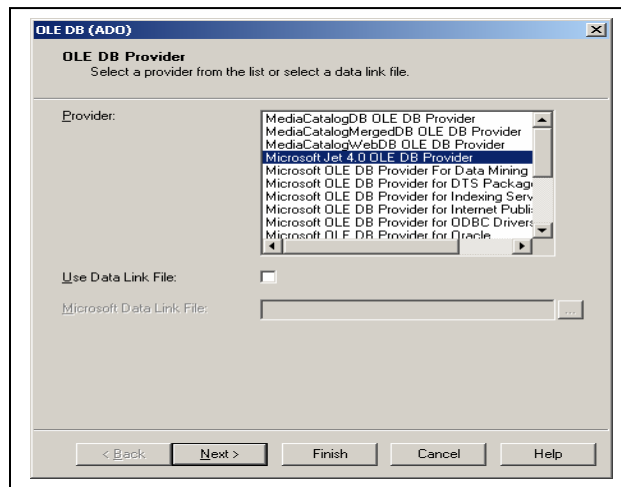
OLE DB ADO

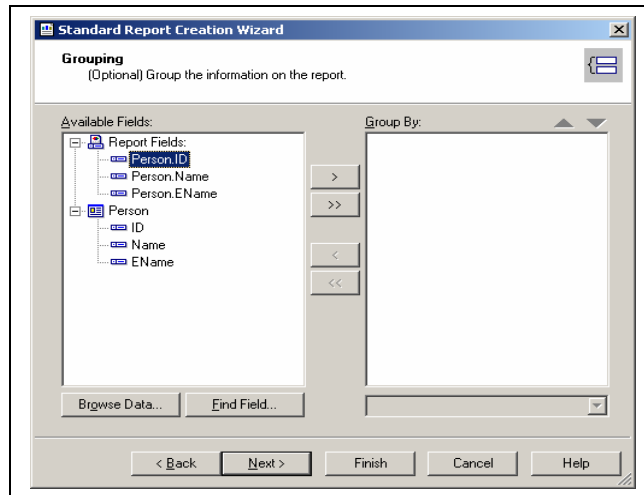
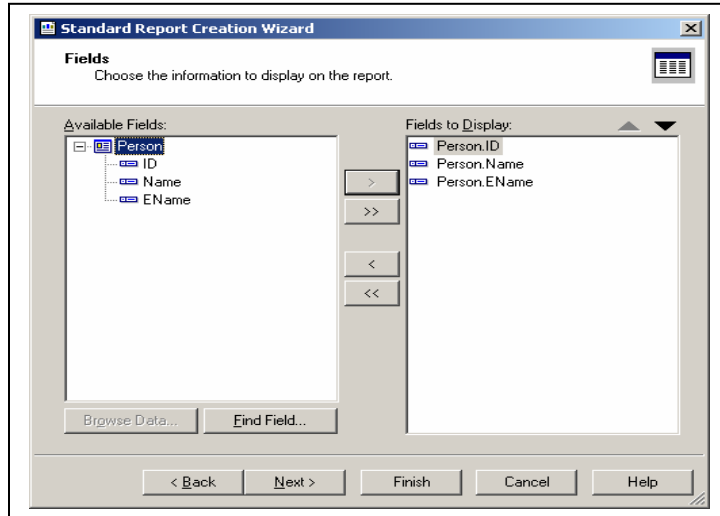




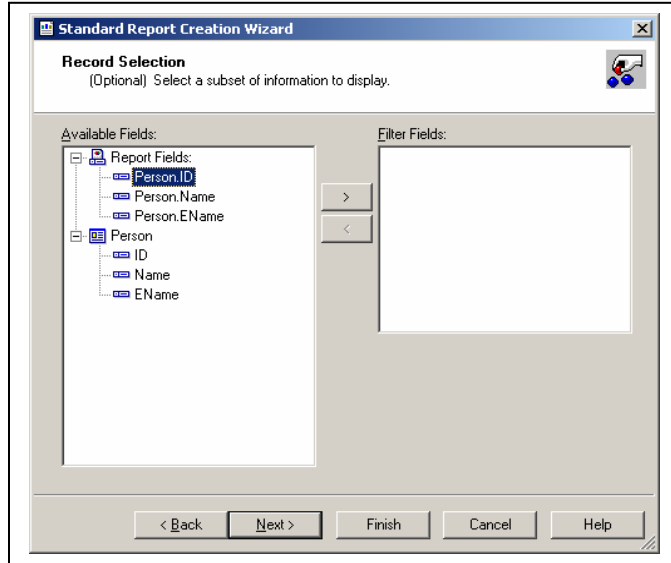
Next

.Ok



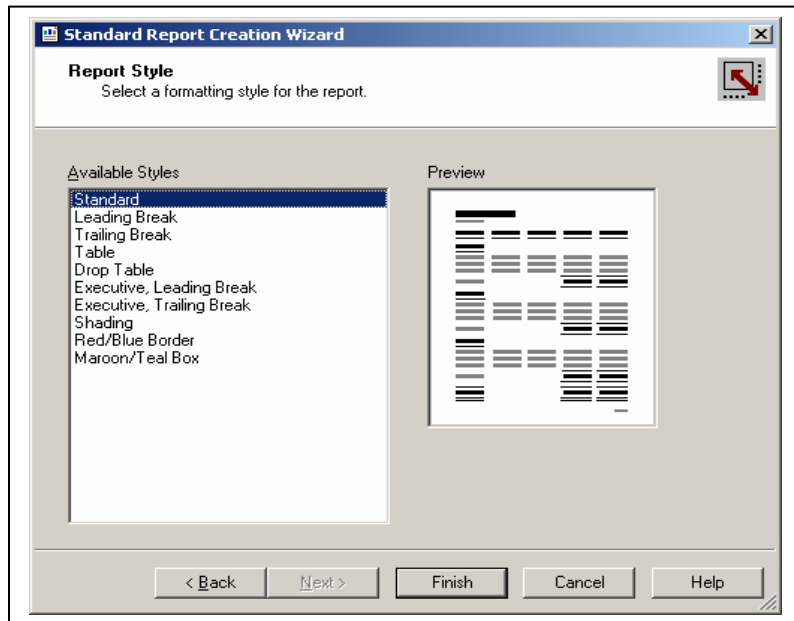


: -

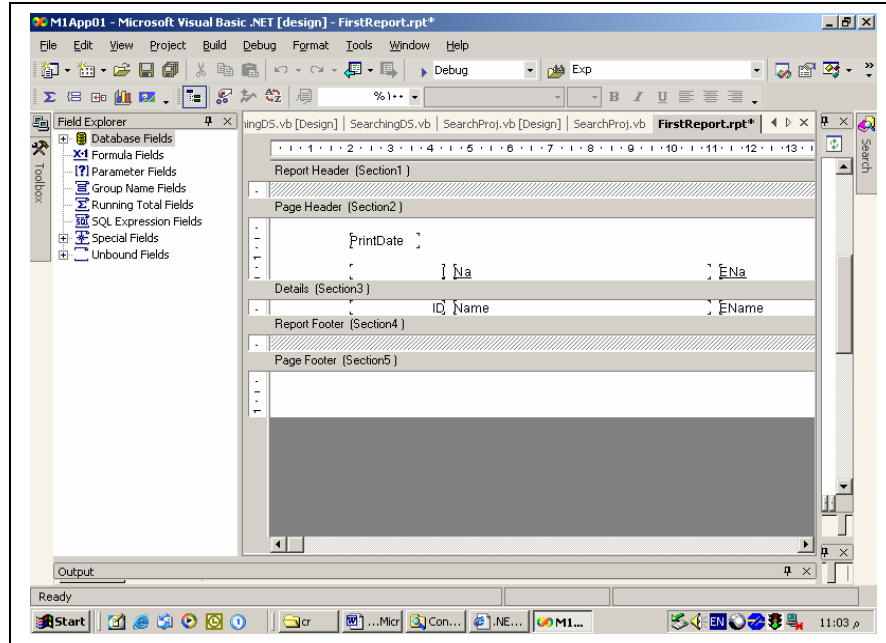


: -

:



Finish



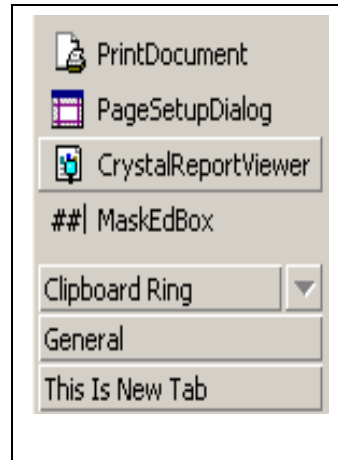
: ReportHeader

: Page Header

يـخـ.

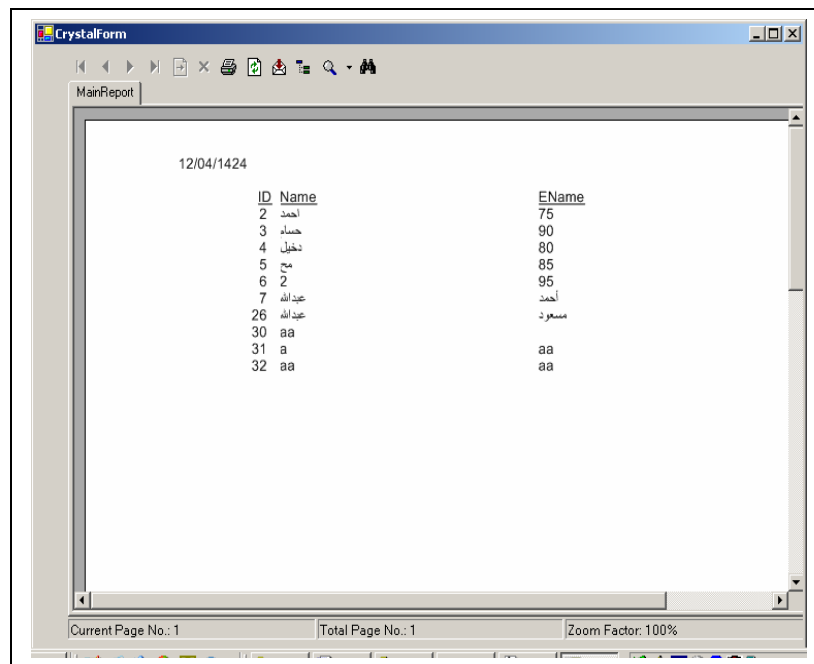
١٢-الخطوة الأخيرة هي عرض التقرير. لعرض التقرير قم بالخطوات التالية :

CrystalReportViewer



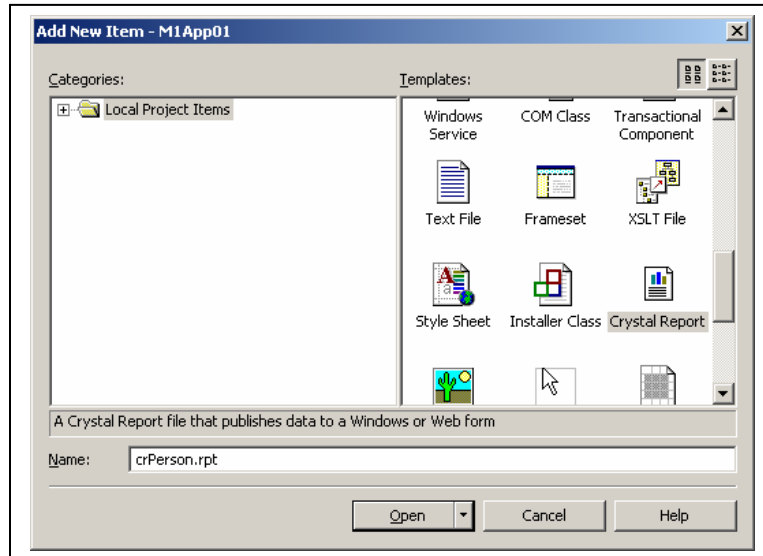
ReportSource

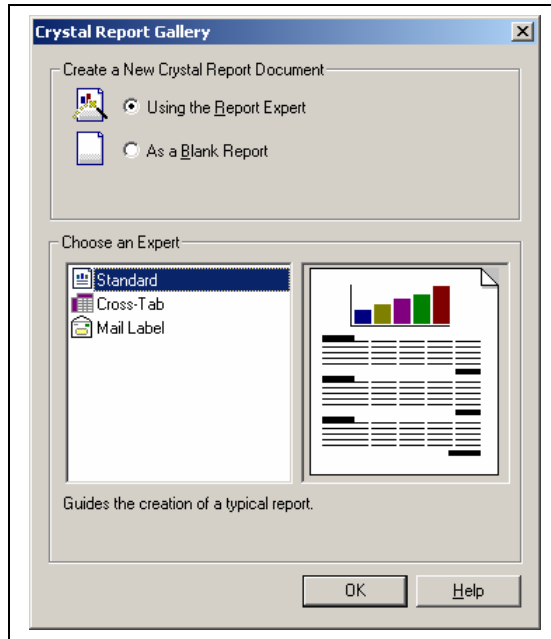
FirstReport.rpt





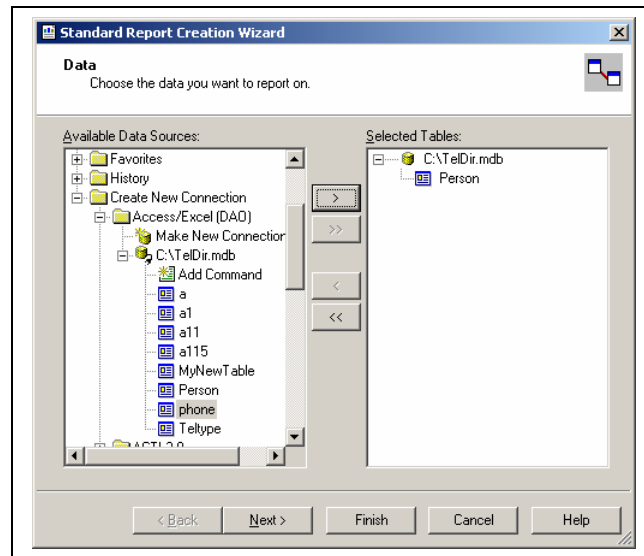
crPerson

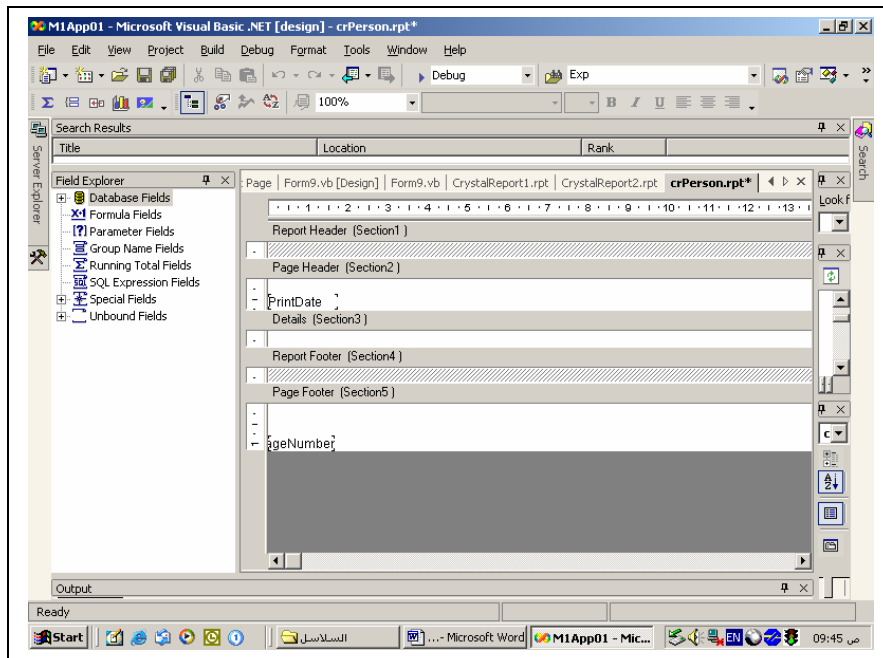




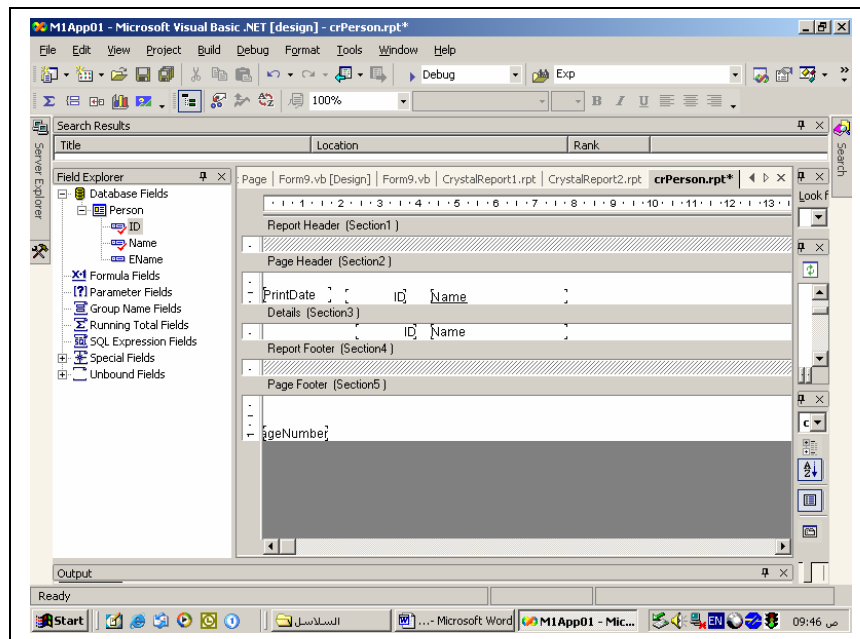
Access/Excel Create New Connection

Finish





.Name ID



ID Name



-  
CrystalReportViewer  
-ReportSource-  
.crPerson  
-

```

Dim con As New OleDb.OleDbConnection()
Dim sql As String
sql = "select id,name from person where name Like '" &
txtEmpName.Text & "%'"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
>Data Source=C:\TelDir.mdb;Persist Security Info=False"
Dim myds As New DataSet()
myds.Clear()
Try
Dim Adapter As New OleDb.OleDbDataAdapter(sql, con)
Adapter.Fill(myDs, "Person")
If myds.Tables("Person").Rows.Count >= 1 Then
MessageBox.Show(myds.Tables("Person").Rows.Count)
Dim rpt As New crPerson()
rpt.SetDataSource(myds)
CrystalReportViewer1.Zoom(20%)
CrystalReportViewer1.ReportSource = rpt
CrystalReportViewer1.RefreshReport()
End If
Catch err As System.Exception
MsgBox(err.Message)
End Try

```

