الألوان

هناك ثلاثة طرق لتغيير الألوان (بالاضافة الى التغيير في وقت التصميم والتغيير باستخدام أداة ستشرح لاحقا).

1 -اعطاء اللون: حيث يعطى اللون مسبوقا بحرفين هما **VB** مختصر من Visual Basic

مثلا:

```
Label1.BackColor=Vbred
Label1.ForeColor=VbBlue
```

Color Constants

Constant	Value	Description
vbBlack	&H0	Black
vbRed	&HFF	Red
vbGreen	&HFF00	Green
vbYellow	&HFFFF	Yellow
vbBlue	&HFF0000	Blue
vbMagenta	&HFF00FF	Magenta
vbCyan	&HFFFF00	Cyan
vbWhite	&HFFFFFF	White

System Colors

Constant	Value	Description
vbScrollBars	&H80000000	Scroll bar color
vbDesktop	&H80000001	Desktop color
vbActiveTitleBar	&H80000002	Color of the title bar for the active window
vbInactiveTitleBar	&H80000003	Color of the title bar for the inactive window
vbMenuBar	&H80000004	Menu background color
vbWindowBackground	&H80000005	Window background color
vbWindowFrame	&H80000006	Window frame color
vbMenuText	&H80000007	Color of text on menus
vbWindowText	&H80000008	Color of text in windows
vbTitleBarText	&H80000009	Color of text in caption, size box, and scroll arrow
vbActiveBorder	&H8000000A	Border color of active window
vbInactiveBorder	&H8000000B	Border color of inactive window

vbApplicationWorkspace	&H8000000C	Background color of multiple-document interface (MDI) applications
vbHighlight	&H8000000D	Background color of items selected in a control
vbHighlightText	&H8000000E	Text color of items selected in a control
vbButtonFace	&H8000000F	Color of shading on the face of command buttons
vbButtonShadow	&H80000010	Color of shading on the edge of command buttons
vbGrayText	&H80000011	Grayed (disabled) text
vbButtonText	&H80000012	Text color on push buttons
vbInactiveCaptionText	&H80000013	Color of text in an inactive caption
vb3DHighlight	&H80000014	Highlight color for 3D display elements
vb3DDKShadow	&H80000015	Darkest shadow color for 3D display elements
vb3DLight	&H80000016	Second lightest of the 3D colors after vb3Dhighlight
vb3DFace	&H8000000F	Color of text face
vb3Dshadow	&H80000010	Color of text shadow
vbInfoText	&H80000017	Color of text in ToolTips
vbInfoBackground	&H80000018	Background color of ToolTips

2 – دالة **QBColor**

```
Label1.BackColor=QBColor(5)
Label1.ForeColor=QBColor(Rnd*15) ' لاظهار اللون بصورة عشوائية
```

أي يعطى بين قوسين رقم محصور بين (0-15)

Bits of Color				
3	2	1	0	Value (Color)
Light	Red	Green	Blue	
0	0	0	0	0 Black أسود
0	0	0	1	1 Blue أزرق
0	0	1	0	2 Green أخضر
0	0	1	1	3 Cyan شذري
0	1	0	0	4 Red أحمر
0	1	0	1	5 Magenta ماروني

0	1	1	0	6 Yellow أصفر
0	1	1	1	7 White أبيض
1	0	0	0	8 Gray رصاصي (أسود فاتح)
1	0	0	1	9 Light Blue أزرق فاتح
1	0	1	0	10 Light Green أخضر فاتح
1	0	1	1	11 Light Cyan شذري فاتح
1	1	0	0	12 Light Red أحمر فاتح
1	1	0	1	13 Light Magenta ماروني فاتح
1	1	1	0	14 Light Yellow أصفر فاتح
1	1	1	1	15 Bright White أبيض ناصع

أي الألوان الثمانية الأخيرة هي نفس الألوان الثمانية الأولى لكن فاتحة.

3 - دالة **RGB**:

R: Red, G: Green, B: Blue أي ان اللون الناتج من هذه الدالة هو خليط من هذه الألوان الثلاثة، حيث ان كل لون منها يمثل بـــرقم محصور بين (0-255). حيث ان (صفر) يعني عدم وجود اللون، والرقم بين (1-255) يعني وجود 255 درجة لونية مختلفة من ذلك اللون.

RGB(red, green, blue)

مثلا:

```
Label1.BackColor=RGB(50,80,255)
Label1.ForeColor=RGB(Rnd*255,Rnd*255,Rnd*255) ' لاظهار اللون بصورة عشوائية
```

مثلا:

Color	Red Value	Green Value	Blue Value
Black	0	0	0
Blue	0	0	255
Green	0	255	0
Cyan	0	255	255
Red	255	0	0
Magenta	255	0	255
Yellow	255	255	0
White	255	255	255

صندوق النص TextBox Control

وتستخدم هذه الأداة للادخال حيث ممكن ان نقوم بإدخال رقم فيها أو سلسلة حرفية وممكن ان نقوم بإدخال نص كامل فيها متكون من أكثر من سطر.

الخصائص TextBox 's Properties

Name, Alignment, Appearance, Backcolor, BorderStyle, Top, Left, Height, Width, Tag, MousePointer, MouseIcon, Enabled, Visible, RightToLeft, Font, FontName, FontBold, FontItalic, FontStrikethru, FontUnderline, TabIndex, Index, ToolTipText, DragMode and DragIcon .

الخصائص أعلاه مشابهة لخصائص شريط الطباعة Label، عدا ان خصائص الخط تؤثر على الخاصية Text التي تقابل الخاصية Caption لشريط الطباعة والغير موجودة هنا.

Text: أهم خاصية هنا وتمثل النص أو الرقم المعطى أو المسترجع. (الادخال دائما هو سلسلة حرفية واذا اردنا تحويله الى رقم فباستخدام الدالة Val)

Locked: صندوق النص مقفل لا يسمح بالادخال عندما تكون هذه الخاصية True.

MaxLength: أقصى عدد للاحرف المسموح بادخالها في صندوق النص.

MultiLine: السماح بالكتابة على أكثر من سطر اذا كانت هذه الخاصية True، ولا يسمح بذلك أي ان الكتابة ستكون على سطر واحد اذا كانت هذه الخاصية False.

ScrollBars: اشرطة التمرير: صفر: لا يوجد، : 1 يوجد شريط أفقي فقط، : 2 يوجد شريط عمودي فقط، 3 : يوجد شريط أفقي وشريط عمودي.

الأفضل في حالة MultiLine=True أي الكتابة على أكثر من سطر ان نضع ScrollBars=3.

Constant	Setting	Description
vbSBNone	0	(Default) None
vbHorizontal	1	Horizontal
vbVertical	2	Vertical
vbBoth	3	Both

SelStart: موقع الحرف ضمن النص والذي يمثل بداية اختيار نص. (أول حرف بالنص يكون موقعه صفر)

object.SelStart [= index]

SelLength: عدد الأحرف المختارة. object.SelLength [= number]

SelText: النص المختار. object.SelText [= value]

HideSelection: اخفاء الاضاءة العالية High Lighted عن النص المختار في حالة الخروج خارج صندوق النص ، True: اخفاء، False: اظهار.

فعند الحاجة الى عمل اختيار لجزء من النص فيجب ان نجعل (منذ البداية) أي في وقت التصميم هذه الخاصية = False.

DataField & DataSource: تستخدم مع قاعدة البيانات.

الأحداث Events :

Click, DblClick, MouseDown, MouseUp, MouseMove, DragDrop, KeyDown, KeyUp, KeyPress, GetFocus and LostFocus .

وهي مشابهة لأحداث النموذج

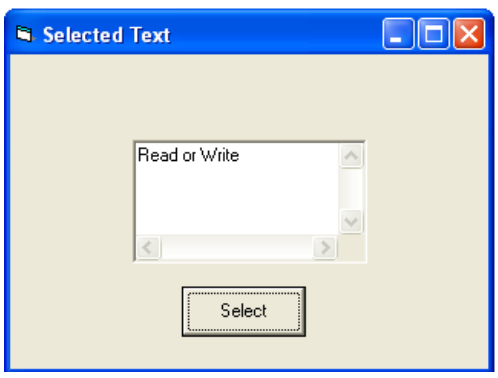
Change: يعمل هذا الحدث عند تغير محتوى صندوق النص بواسطة قاعدة البيانات. أو عندما يقوم المستخدم بادخال النص أي عندما تتغير خاصية Text.

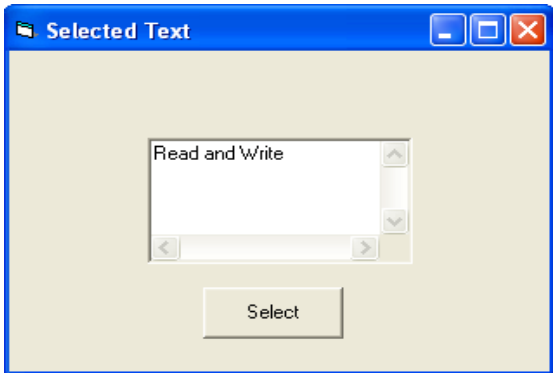
الاجراءات Methods :

Move, Refresh, SetFocus and Zorder Methods.

* يتم فحص خصائص صندوق النص أعلاه.

مثال حول تحديد جزء من النص وتبديله بنص آخر بواسطة ايعازات مكتوبة في البرنامج.





بعد تغيير كلمة and بـــ or قبل التغيير

Project properties

Object	Property	Setting
Form1	Name Caption	Form1 Selected Text
CommandButton1	Name Caption	Command1 Select
Text Box1	Name Text HideSelection MultiLine ScrollBars	Text1 فراغ False True 3

Poject code

```
Private Sub Command1_Click()
 Text1.SelStart = 5
 Text1.SelLength = 3
 Text1.SelText = "or"
End Sub
```

عند التنفيذ نكتب في صندوق النص read and write وننقر على زر الأوامر Select نلاحظ تبديل الكلمة and بالكلمة or حيث سنحدد موقع بداية الاختيار وهو الموقع الخامس وهو بداية الكلمة and ونحدد طول الاختيار وهو ثلاثة فيصبح النص المختار and فنبدله بالكلمة or.

* ممكن ان نختار نحن كلمة and بواسطة الماوس ونكتب بدله or مباشرة أو ان نضع في حدث النقر على زر الأوامر فقط آخر خطوة وهي تبديل النص المختار بـــ or (لاننا حددنا النص بواسطة الماوس).

* ممكن ان نمسح كلمة and اي نحذفها فنقول Text1.SelText="".

* ذكرنا ان خصائص الخط تؤثر على خاصية Text أي على النص كله ولا تؤثر على خاصية SelText النص المختار كما الحالة في برنامج Word (مثل هذه الحالة توجد أداة اضافية اسمها Rich Edit Text Box وهي أقرب للـــ Word)

Clipboard object الحافظة

الحافظة هي منطقة خزنية في الذاكرة تحفظ النص الذي استنسخناه أو حذفناه ومن ثم نجلب منها هذا النص للصقه. (ممكن ان نخزن فيها رسم لكن ما يهمنا هنا هو النص).

الاجراءات **Methods**

1- **Clear**: تنظيف الحافظة. Clipboard.clear

2- **Settext**: خزن النص في الحافظة object.SetText data, format

حيث data: هو النص، أما Format فممكن ان تكون:

Format	Value	Description
vbCFLink	&HBF00	DDE conversation information
vbCFRTF	&HBF01	RichText Format
vbCFText	1	(Default) Text

مثال:

```
Clipboard.settext text1.seltext
```

Gettext: جلب النص من الحافظة object.GetText (format)

Format ممكن ان تكون:

Format	Value	Description
vbCFLink	&HBF00	DDE conversation information
vbCFText	1	(Default) Text
vbCFRTF	&HBF01	Rich Text Format (.rtf file)

مثال:

```
text1.seltext =Clipboard.gettext()
```

الاستنساخ:-

1- تنظيف الحافظة

2- خزن النص في الحافظة

القص:-

1- تنظيف الحافظة

2- خزن النص في الحافظة

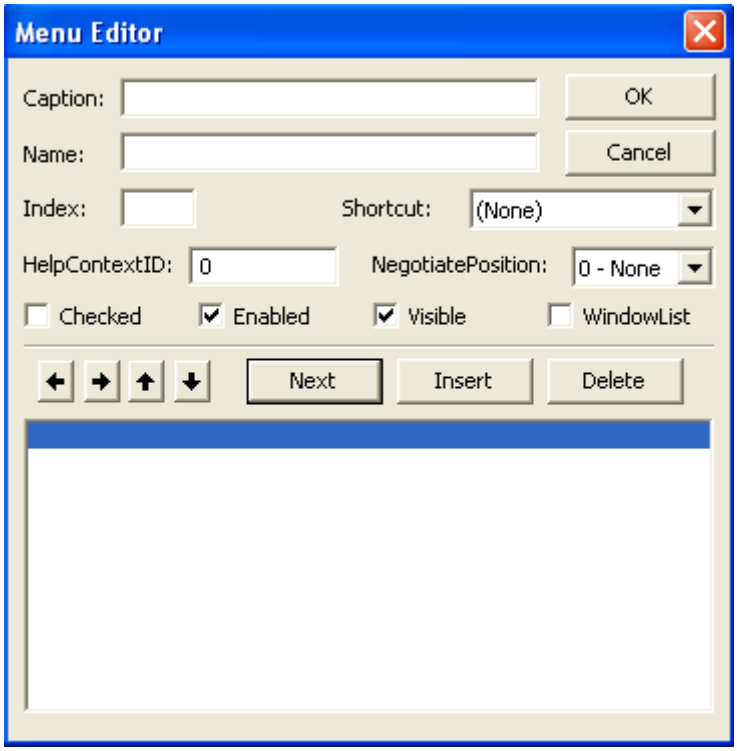
3- مسح النص

اللصق:-

جلب النص من الحافظة.

شريط القوائم MenuBar :-

لعمل شريط قوائم خاص بنا فمن القائمة Tools نختار Menu Editor فتظهر لنا الشاشة التالية:-



Caption: الكلمة التي تظهر في القائمة. (الحرف الذي يظهر في القائمة تحته _ نضع قبله &)

Name: اسم الاختيار يكون نفس الكلمة في Caption عادة وبدون فراغات (بدون &) ونسبقه عادة بـــ mnu.

* عندما نريد حد فاصل بين مجموعة من الاختيارات وأخرى فنضع في Caption نضع - (Minus) وأي اسم في Name.

ShortCut: احيانا تظهر أمام الاختيار (حرف+ctrl) أو (حرف+Shift) للتنفيذ السريع للاختيار بدون الدخول في قائمة الاختيارات.

Checked: تظهر قبل الكلمة علامة صح أو لا تظهر (Toggle).

Enabled: يعمل الاختيار أو لا (يستجيب لحدث النقر)

Visible: الاختيار مرئي أو لا

Next: عندما نبني القائمة لإدخال الاختيار التالي نضغط على Next

Insert: عند الوقوف على أي اختيار ونضغط Insert فيتم السماح بادخال اختيار جديد قبل هذا الاختيار.

Delete: لحذف اختيار محدد.

: لنجعل الاختيار يتفرع الى قائمة فرعية.

: لنصعد مستوى بالتفرع.

: لنغير موقع هذا الاختيار مع الاختيار السابق.

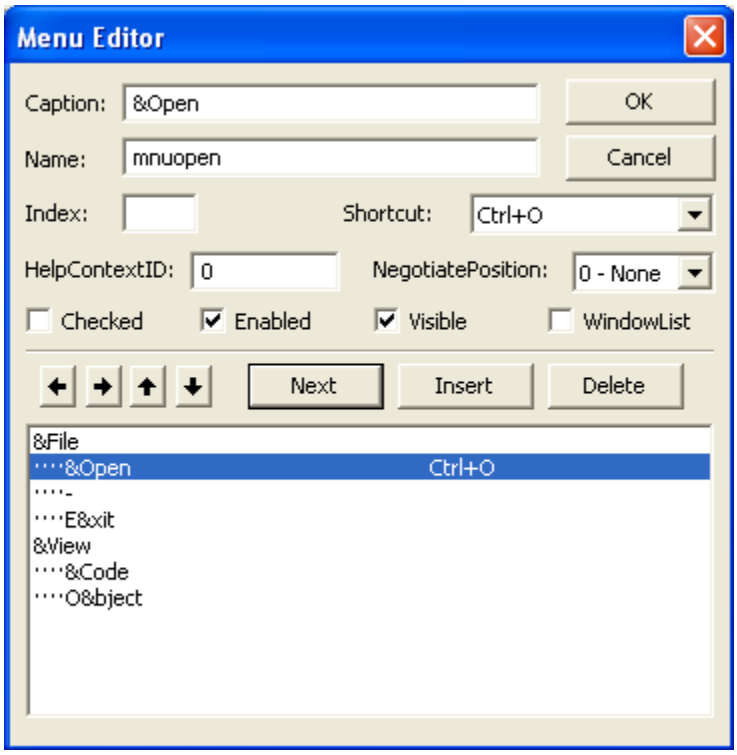
: لنغير موقع هذا الاختيار مع الاختيار اللاحق.

مثال لعمل شريط القوائم التالي:

File
Open Ctrl+O
Exit

View
Code
Object

* الاختيارات File و View ستمثل عناوين لقوائم اختيارت وستظهر في شريط القوائم



- من القائمة Tools نختار Menu Editor
- في Caption نضع &File وفي Name نضع mnufile
- نضعط على Next لننقوم بإدخال الاختيار التالي ونضغط على ➔ لأن لدينا قائمة فرعية فستظهر ....
ثم في Caption نضع &Open وفي Name نضع mnuopen (نختار Ctrl+O كـــ Shortcut.
- نضعط على Next لندخل الاختيار التالي ولا نضغط على ➔ لأننا اساسا متفرعين أي موجودة ....
ثم في Caption نضع - (Minus) وفي Name نضع أي اسم.
- نضعط على Next لندخل الاختيار التالي ولا نضغط على ➔ لأننا اساسا متفرعين أي موجودة ....
ثم في Caption نضع E&xit وفي Name نضع mnuexit
- نضعط على Next فتظهر .... لأننا اساساً متفرعين ولكننا لا نريد هذا التفرع هنا لان View هي قائمة رئيسية فنضغط على ← لنلغي هذا التفرع فتختفي ....
ثم في Caption نضع &View وفي Name نضع mnuview
- نضعط على Next فلا تظهر .... ولكن هنا نريدها لان Code قائمة فرعية فنضغط ➔ فتظهر ....
ثم في Caption نضع &Code وفي Name نضع mnucode
- نضعط على Next لندخل الاختيار التالي ولا نضغط على ➔ لأننا اساسا متفرعين أي موجودة ....

ثم في Caption نضع O&bject وفي Name نضع mnuobject

- نختار OK ونخرج

ويكتب جدول قوائم نموذج1 بالشكل التالي:-

Caption	Name
&File	Mnufile
....&Open	Mnuopen ' Shortcut=ctrl+o
....-	Mnuun
....E&xit	Mnuexit
&View	Mnuview
....&Code	Mnucode
....O&bject	Mnuobject

* لنبرمج الاختيار Exit في القائمة فنختار عند التحول الى كتابة البرناج mnuexit من موقع اختيار الأدوات فيظهر البرنامج الفرعي الخاص بالحدث Click لهذا الاختيار).

* بالنسبة للاختيارات فنتعامل معها معاملة الأدوات فيوجد فقط الحدث Click لها والاختيارات المذكورة في Menu Editor تعتبر خصائص له.

```
Private Sub mnuexit_click()
 Beep
 End
End Sub
```

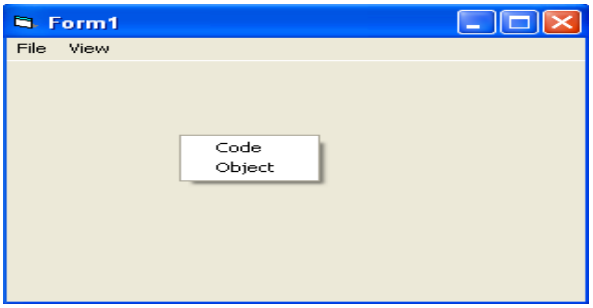
PopupMenu

ممكن ان نعمل القائمة View على سبيل المثال كـــ PopupMenu. ونبرمج النقر على النموذج (النقر مباشرة) فبعد التنفيذ عند النقر على النموذج تظهر الاختيارت التابعة لهذه القائمة (بدون ان يظهر عنوان القائمة View).

```
Private Sub Form_Click()
  Form1.PopupMenu mnuview
End Sub
```

أو ان نظهر هذه القائمة عند النقر بالزر الأيمن للماوس على النموذج:

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then  Form1.PopupMenu mnuview
End Sub
```



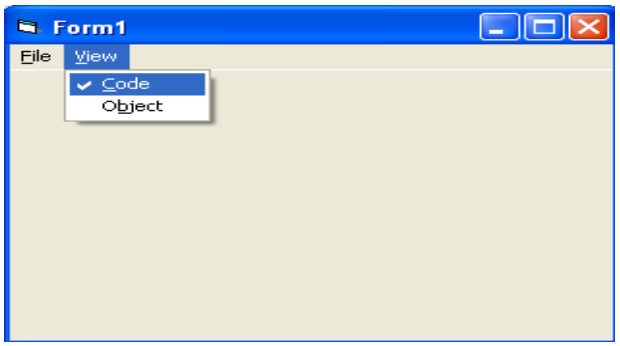
ملاحظة:-

ممكن ان نتعامل مع خصائص قائمة الاختيارات.

مثلاً: لايقاف عمل هذا الاختيار وممكن تفعيله فيما بعد ' Mnuopen.Enabled=False

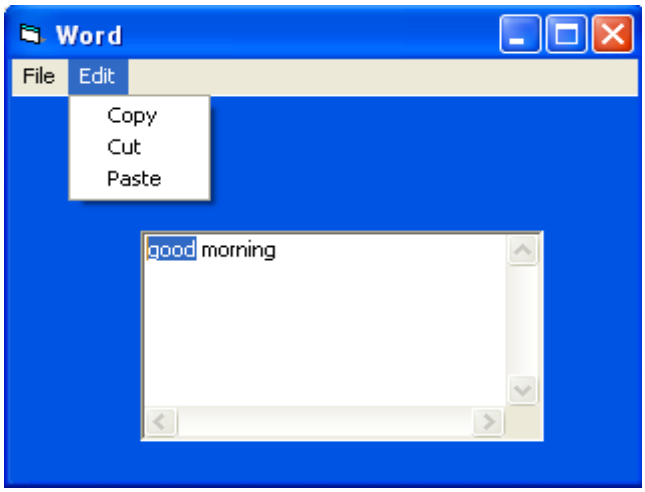
* ممكن ان نعمل Toggle لخاصية Checked لأي اختيار، مثلا:

```
Private sub mnucode_Click()
' ليتم وضع علامة صح مرة ومرة أخرى ترفع هذه العلامة
 Mnucode.Checked=Not mnucode.Checked
End Sub
```



س) برنامج يضم صندوق نصوص وشريط قوائم يحتوي على ما يأتي:-

File Edit
Exit Copy
Cut
Paste



Project properties

Object	Property	Setting
Form1	Name Caption BackColor	Form1 Word أي لون تختاره
TextBox1	Name Text MultiLine ScrollBars HideSelection	Text1 فراغ True 3 False

جدول قوائم نموذج1

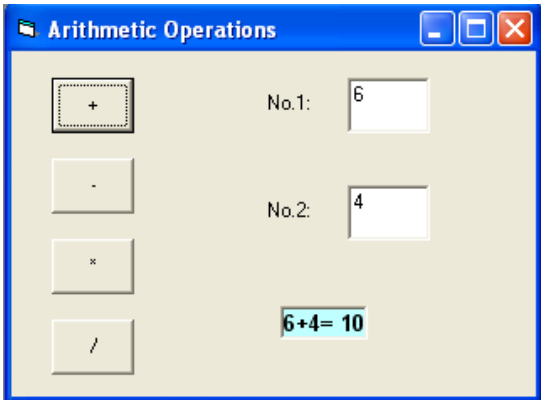
Caption	Name
&File ….E&xit &Edit ….&Copy ….Cu&t ….&Paste	Mnufile Mnuexit Mnuedit Mnucopy Mnucut Mnupaste

Project code

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
 If  Button = 2 Then Form1.PopupMenu mnuedit
End Sub
```

```
Private Sub mnucopy_Click()
 Clipboard.Clear
 Clipboard.SetText Text1.SelText
End Sub

Private Sub mnucut_Click()
 Clipboard.Clear
 Clipboard.SetText Text1.SelText
 Text1.SelText = ""
End Sub

Private Sub mnuexit_Click()
 Beep
 End
End Sub

Private Sub mnupaste_Click()
 Text1.SelText = Clipboard.GetText()
End Sub
```

س) عمل برنامج يقوم بإدخال رقمين ثم يقوم باختيار عملية الجمع أو الطرح أو الضرب أو القسمة ومن ثم طباعة الناتج (باستخدام أزرار الأوامر)



Project properties

Object	Property	Setting
Form1	Name Caption BackColor	Form1 Arithmetic Operations أي لون تختاره
Label1	Name Caption BackStyle	Label1 No.1: 0 'transparent
Label2	Name Caption BackStyle	Label2 No.2: 0 'transparent
Label3	Name Caption BackColor BackStyle BorderStyle Autosize Font	Label3 Result أي لون 1 'Opaque 1 'Fixed Single True تغيير اسم الخط وحجمه
TextBox1	Name Text	Text1 فراغ
TextBox2	Name Text	Text2 فراغ
CommandButton1	Name Caption	Command1 +
CommandButton2	Name Caption	Command2 -
CommandButton3	Name Caption	Command3 *
CommandButton4	Name Caption	Command4 /

Project code

```
Private Sub Command1_Click()
d = Val(Text1.Text) + Val(Text2.Text)
Label3.Caption = Text1.Text + "+" + Text2.Text + "=" + Str(d)
End Sub

Private Sub Command2_Click()
d = Val(Text1.Text) - Val(Text2.Text)
Label3.Caption = Text1.Text + "-" + Text2.Text + "=" + Str(d)
End Sub

Private Sub Command3_Click()
d = Val(Text1.Text) * Val(Text2.Text)
Label3.Caption = Text1.Text + "*" + Text2.Text + "=" + Str(d)
End Sub

Private Sub Command4_Click()
If  Val(Text2.Text) = 0 Then
 Label3.Caption = "error: division by 0"
Else
 d = Val(Text1.Text) / Val(Text2.Text)
 Label3.Caption = Text1.Text + "/" + Text2.Text + "=" + Str(d)
End If
End Sub
```

س) عمل برنامج يحتوي على عبارة "welcome" ويضم شريط قوائم يتكون مما يأتي:

Color	Font
Red	Bold
Green	Italic
Blue	Underline

عند اختيار أي لون نضع علامة صح عليه وبقية الألوان نرفع العلامة منهم ونلون العبارة باللون الذي اخترناه، وعند اختيار Bold نضع علامة صح عليها ونعمل خط العبارة أغمق وإذا اخترناه مرة أخرى نجعل الخط فاتح ونرفع علامة الصح وهكذا للبقية.

Project properties

Object	Property	Setting
Form1	Name BackColor	Form1 أي لون تختاره
Label1	Name Caption BackColor BackStyle BorderStyle Font	Label1 Welcome أي لون 1 'Opaque 1 ' Fixed Single اختيار اسم خط وحجم مناسب

جدول قوائم نموذج1

Caption	Name
&Color	Mnucolor
….&Red	Mnured
….&Green	Mnugreen
….&Blue	Mnublue
&Font	Mnufont
….&Bold	Mnubold
….&Italic	Mnuitalic
….&Underline	Mnuunderline

Project code

```
Private Sub mnured_Click()
 Label1.BackColor = vbRed
 mnured.Checked = True
 mnugreen.Checked = False
 mnuBlue.Checked = False
End Sub

Private Sub mnugreen_Click()
 Label1.BackColor = vbGreen
 mnured.Checked = False
 mnugreen.Checked = True
 mnuBlue.Checked = False
End Sub

Private Sub mnuBlue_Click()
 Label1.BackColor = vbBlue
 mnured.Checked = False
 mnugreen.Checked = False
 mnuBlue.Checked = True
End Sub

Private Sub mnuBold_Click()
 mnuBold.Checked = Not mnuBold.Checked
 Label1.FontBold = mnuBold.Checked
End Sub

Private Sub mnuitalic_Click()
 mnuitalic.Checked = Not mnuitalic.Checked
 Label1.FontItalic = mnuitalic.Checked
End Sub
```

```
Private Sub mnuunderline_Click()
 mnuunderline.Checked = Not mnuunderline.Checked
 Label1.FontUnderline = mnuunderline.Checked
End Sub
```

زر الأوامر CommandButton Control

كل زر يخصص لاداء عملية معينة فممكن ان نجعل زر للون وزر للخط الغامق وزر للخروج ونبرمج حدث النقر الخاص بها أفضل من ان نعمل كل ما نحتاجه في حدث النقر على النموذج.

خصائص زر الأوامر

Name, Appearance, Backcolor, Caption, Top, Left, Height, Width, Tag, MousePointer, MouseIcon, Enabled, Visible, RightToLeft, Font, FontName, FontBold, FontItalic, FontStrikethru, FontUnderline, TabIndex, Index, ToolTipText, DragMode and DragIcon.

وهي مشابهة لخصائص شريط الطباعة Label.

الخصائص المميزة لزر الأوامر

Default : عندما تكون True أي ان هذا الزر هو الزر التلقائي default command button في النموذج.

Cancel: تكون True لزر واحد في النموذج اي انه يعمل على Esc

وهذا يعني ان حدث النقر (click) لزر الأوامر يعمل أما بالنقر على الماوس أو (بالحركة بالـــ Tab أو بالتركيز عليه بـــ SetFocus ثم الضغط على Enter Key) وممكن بالـــ Esc في حالة كون خاصية Cancel=True لهذا الزر.

BackColor: لون الخلفية.

Picture: الصورة الظاهرة على زر الأوامر.

DownPicture: الصورة التي تظهر على زر الأوامر عند النزول بالماوس عليه أي عند النقر عليه.

DisabledPicture: الصورة التي تظهر على الزر عندما تكون خاصية Enabled=False أي ان الزر لا يستجيب للحدث.

Style: عندما تكون= صفر فان الزر شكله عادي Standard ، بينما عندما تكون هذه الخاصية=1 فان هذا الزر يصبح بشكل رسومي Graphical. فتظهر عليه تأثير الخصائص: **Picture ،BackColor ،DownPicture ،DisabledPicture.**

أحداث زر الأوامر المهمة Events

Click, MouseDown, MouseUp, MouseMove, DragDrop, KeyDown, KeyUp, KeyPress, GotFocus and LostFocus

وهي مشابهة لأحداث النموذج.

الحدث **GotFocus**: ويحدث عند التركيز على هذا الزر.

الحدث **LostFocus** ويحدث عندما يفقد الزر التركيز.

اجراءات زر الأوامر المهمة **Methods**

Move, Refresh, SetFocus and Zorder Methods.

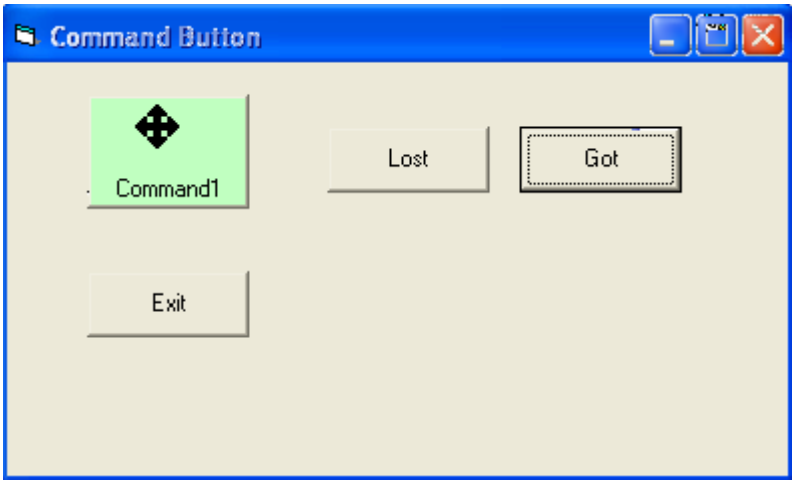
SetFocus : يعني اجعل التركيز على زر معين فيصبح هو الزر الفعال، وبالتالي ممكن النقر عليه بالماوس أو الضغط على Enter Key

مثال:

سندرس في هذا المثال: 1) خصائص زر الأوامر باستخدام command1

2) عمل زر خروج باستخدام command2

3 استخدام command4، command3 لدراسة الاجـــــــراء SetFocus، والأحـــــداث LostFocus ،GotFocus



Project properties

Object	Property	Setting
Form1	Name Caption	Form1 Command Button
CommandButton1	Name Backcolor Picture Down picture Disabled Picture Enabled Style	Command1 اخضر صورة صورة تظهر عند النزول بالماوس صورة تظهر عندما Enabled=False في حالة وضع Disabled Picture فقط False 1
CommandButton2	Name Caption	Command2 Exit
CommandButton3	Name	Command3
CommandButton4	Name	Command4

Poject code

```
Private Sub Command2_Click()
 Beep
 End
End Sub

Private Sub Command3_Click()
 Command4.SetFocus
End Sub

Private Sub Command3_LostFocus()
 Command3.Caption = "Lost"
End Sub

Private Sub Command4_Click()
 Command4.Caption = "click"
End Sub

Private Sub Command4_GotFocus()
 Command4.Caption = "Got"
End Sub
```

عند النقر على زر الأوامر الثالث يتم وضع التركيز على زر الأوامر الرابع بواسطة الاجراء SetFocus الخاص بزر الأوامر الرابع أي Command4.SetFocus حينها سيفقد زر الأوامر الثالث التركيز فيعمل له

الحدث LostFocus وسيأخذ زر الأوامر الرابع التركيز فيعمل له الحدث GotFocus حينها ممكن النقر على زر الأوامر الرابع بالماوس او بالضغط على Enter Key

MsgBox Function

دالة صندوق الرسالة. وصيغتها العامة

MsgBox(prompt[, buttons] [, title] [, helpfile, context])

Prompt: الرسالة، buttons: الأزرار، title : العنوان، helpfile, context: المساعدة

الأزرار

الزر التلقائي الذي يظهر في حالة عدم اعطاء ازرار ضمن الرسالة هو: OK

الأزرار (0-5) كل حسب استخدامه، واقصى عدد للأزرار هو 3. وممكن اضافة ايقونة للرسالة وهي (16، 32، 48، 44) حسب المعنى المقصود من الرسالة. القيم التالية للأيقونات تعني من هو الزر الـ default اي يكون التركيز عليه من بين الازرار الظاهرة. هناك زر رابع للمساعدة وهو vbMsgBoxHelpButton

The buttons argument settings are:

Constant	Value	Description
vbOKOnly	0	Display OK button only.
vbOKCancel	1	Display OK and Cancel buttons.
vbAbortRetryIgnore	2	Display Abort, Retry, and Ignore buttons.
vbYesNoCancel	3	Display Yes, No, and Cancel buttons.
vbYesNo	4	Display Yes and No buttons.
vbRetryCancel	5	Display Retry and Cancel buttons.
vbCritical	16	Display Critical Message icon.
vbQuestion	32	Display Warning Query icon.
vbExclamation	48	Display Warning Message icon.
vbInformation	64	Display Information Message icon.
vbDefaultButton1	0	First button is default.
vbDefaultButton2	256	Second button is default.
vbDefaultButton3	512	Third button is default.
vbDefaultButton4	768	Fourth button is default.
vbApplicationModal	0	Application modal; the user must respond to the message box before continuing work in the current application.
vbSystemModal	4096	System modal; all applications are suspended until the user responds to the message box.
vbMsgBoxHelpButton	16384	Adds Help button to the message box

VbMsgBoxSetForeground	65536	Specifies the message box window as the foreground window
vbMsgBoxRight	524288	Text is right aligned
vbMsgBoxRtlReading	1048576	Specifies text should appear as right-to-left reading on Hebrew and Arabic systems

ملاحظة:- ممكن ان تسترجع دالة صندوق الرسالة جواب المستخدم باختيار احد الازرار الظاهرة لديه.

حيث يمكن لاحقا فحص هذا الجواب والتصرف على ضوئه.

Return Values

Constant	Value	Description
vbOK	1	OK
vbCancel	2	Cancel
vbAbort	3	Abort
vbRetry	4	Retry
vbIgnore	5	Ignore
vbYes	6	Yes
vbNo	7	No

InputBox Function

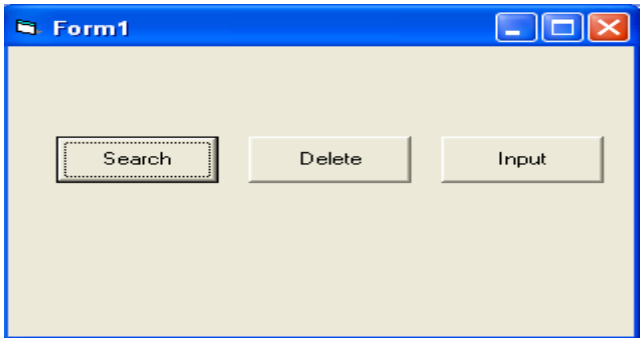
دالة صندوق الادخال وتعيد سلسلة حرفية string. وصيغتها العامة:-

InputBox(prompt[, title] [, default] [, xpos] [, ypos] [, helpfile, context])

Prompt: الرسالة، title : العنوان، default، اعطاء قيمة أولية، xpos, ypos: الموقع على الشاشة، helpfile,context: المساعدة

هنا لا نعطي أزرار لانه يعطي زرين هما OK, Cancel

مثال: – حول استحدام دوال صندوق الرسالة وصندوق الادخال

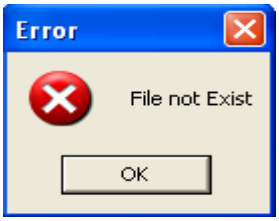


Project properties

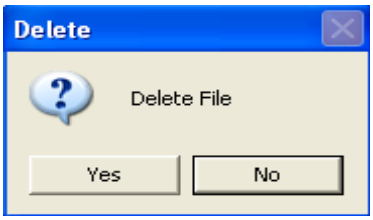
Object	Property	Setting
Form1	Name	Form1
CommandButton1	Name Caption	Command1 Search
CommandButton2	Name Caption	Command2 Delete
CommandButton3	Name Caption	Command3 Input

Poject code

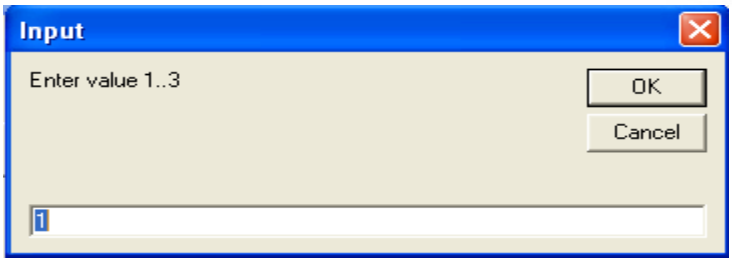
```
Private Sub Command1_Click()
 MsgBox "File not Exist", vbCritical + vbOKOnly, "Error"  ' اعطاء رسال الملف غير موجود
End Sub
```




```
Private Sub Command2_Click()
' هنا صندوق الرسالة يسترجع جواب المستخدم بنعم أو لا
 a = MsgBox("Delete File", vbYesNo + vbQuestion + vbDefaultButton2, "Delete")
 If a = vbYes Then Print "File Deleted"
End Sub
```




```
Private Sub Command3_Click()
' ادخال رقم محصور بين 1..3
 vs = InputBox("Enter value 1..3", "Input", "1")
 v = Val(vs)
 If (v >= 1) And (v <= 3) Then Print v              'we can use label. (label1.caption=v)
End Sub
```



ملاحظة:-

الدالة **Val** عكس الدالة (str) حيث ان:-

Val: تحول السلسلة الحرفية (string) الى رقم

Str: تحول الرقم الى سلسلة الحرفية (string)

س) اكتب الشفرة الخاصة بحدث تحميل النموذج والمتضمنة ما يأتي:-

1- جعل النص الظاهر على شريط الطباعة الأول يظهر وتحته خط

2- تحريك زر الأوامر الثالث إلى الموقع x1=1000، y1=2000

3- مسح النص من شريط الطباعة الثاني

```
Private sub Form_Load()
  Label1.FontUnderline=True
Command3.Move 1000,2000
    Label2.Caption=""
End Sub
```

س) عند النقر على زر الأوامر الأول يظهر صندوق إدخال يضم رسالة لإدخال كلمة تخزن مباشرة في شريط الطباعة الأول وتظهر بصورة غامقة وتحتها خط.

```
Private Sub Command1_Click()
 Label1.Caption = InputBox("Enter a word", "Word", "")
 Label1.FontBold = True
 Label1.FontUnderline = True
End Sub
```

ملاحظة:- إذا أردنا أن تكون الرسالة على أكثر من سطر فنكتب بين سطر وآخر +vbCrLf+ كما في المثال التالي:-

```
Label1.Caption = InputBox("Please" + vbCrLf + "Enter a word", "Word", "")
```

حيث vbCrLf تعني :-

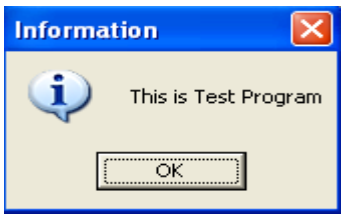
Cr للنزول سطر، Lf: للوقوف على بداية السطر.

س) عند النقر على أي شريط طباعة من ضمن مصفوفة أشرطة طباعة يتغير لون خلفيته إلى أزرق ويتم إيقاف عمله.

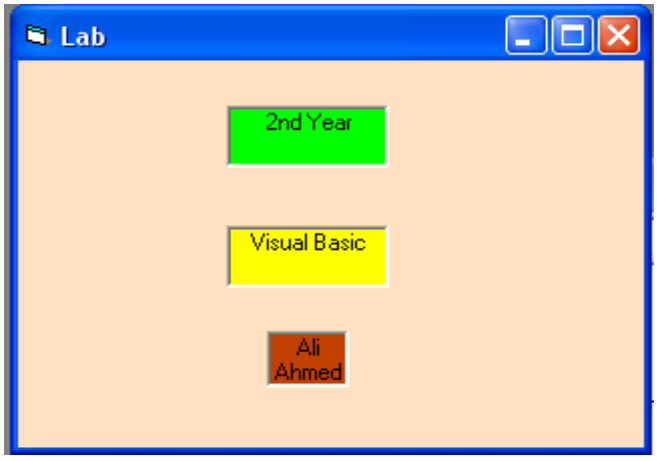
```
Private Sub Label1_Click(Index As Integer)
 Label1(Index).BackColor = vbBlue
 Label1(Index).Enabled = False
End Sub
```

س) عند الضغط على الزر F1 تظهر رسالة تتضمن المعلومة التالية: "This is test program"

```
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
 If KeyCode = vbKeyF1 Then MsgBox "This is Test Program", vbOKOnly + vbInformation, "Information"
End Sub
```



س) قم بإعداد نموذج بسيط يعرض مرحلتك الدراسية واسم المختبر وأسماء أفراد مجموعة العمل في المختبر بصورة مرتبة، وباختيار اسم المختبر يصبح الخط غامق، وباختيار أسماء المجموعة يصبح الخط مرة غامق ومرة فاتح، وباختيار المرحلة الدراسية يتم وضع خط تحته مرة ومرة أخرى يتم رفع الخط. (الاختيار يكون بالضغط على الماوس ضغطة واحدة). وعند الضغط على الماوس بالزر الأيمن فوق اسم المختبر يتم مسحه، عند الضغط على الحرف "h" تختفي أسماء المجموعة وعند الضغط على الحرف "v" تظهر أسماء المجموعة.



Project properties

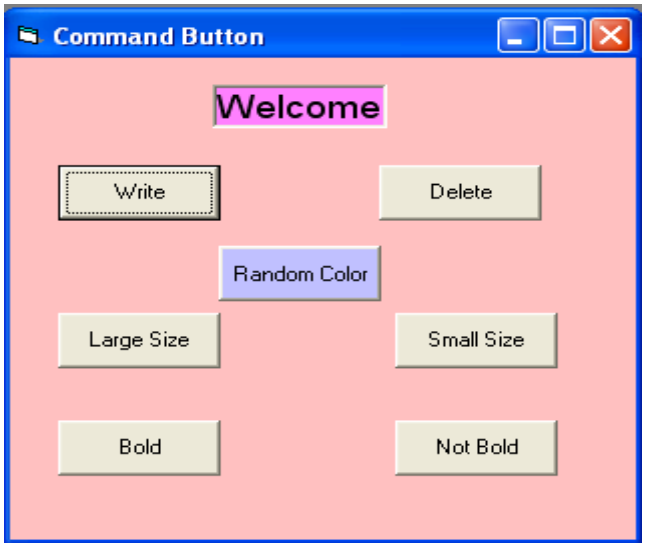
Object	Property	Setting
Form1	Name Caption BackColor	Form1 True أي لون تختاره
Label1	Name Caption BackColor BackStyle BorderStyle Alignment	Label1 2nd Year أي لون 1 1 'Fixed Single 2 ' center
Label2	Name Caption BackColor BackStyle BorderStyle Alignment	Label2 Visual Basic أي لون 1 1 'Fixed Single 2
Label3	Name BackColor BackStyle BorderStyle Alignment AutoSize WordWrap	Label3 أي لون 1 1 'Fixed Single 2 True True

Project code

```
Private Sub Form_KeyPress(KeyAscii As Integer)
 a = Chr(KeyAscii)
 If (a = "h") Or (a = "H") Then
  Label3.Visible = False
 ElseIf (a = "v") Or (a = "V") Then
  Label3.Visible = True
 End If
End Sub

Private Sub Form_Load()
 Label3.Caption = "Ali" + Chr(10) + "Ahmed"
End Sub

Private Sub Label1_Click()
 Label1.FontUnderline = Not Label1.FontUnderline
End Sub
```

```
Private Sub Label2_Click()
 Label2.FontBold = True
End Sub

Private Sub Label2_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As
Single)
 If Button = 2 Then Label2.Caption = ""
End Sub

Private Sub Label3_Click()
 Label3.FontBold = Not Label3.FontBold
End Sub
```

س) عمل شاشة تتكون مما يأتي:- شريط طباعة Label ، زر لكتابة "Welcome " على شريط الطباعة، زر آخر لمسح كلمة "Welcome" من شريط الطباعة، وأزرار أخرى لتغيير اللون بصورة عشوائية ولتكبير حجم الخط ولتصغيره ولجعله أغمق أو افتح



Project properties

Object	Property	Setting
Form1	Name Caption BackColor	Form1 Command Button أي لون تختاره
Label1	Name Caption BackColor BackStyle BorderStyle Alignment Font AutoSize	Label1 فراغ أي لون 1 1 'Fixed Single 2 تحديد اسم للخط Arial وحجم للخط 16 True
CommandButton1	Name Caption	Command1 Write
CommandButton2	Name Caption	Command2 Delete
CommandButton3	Name Caption Style BackColor	Command3 Random Color 1 'graphic أي لون
CommandButton4	Name Caption	Command4 Large Size
CommandButton5	Name Caption	Command5 Small Size
CommandButton6	Name Caption	Command6 Bold
CommandButton7	Name Caption	Command7 Not Bold

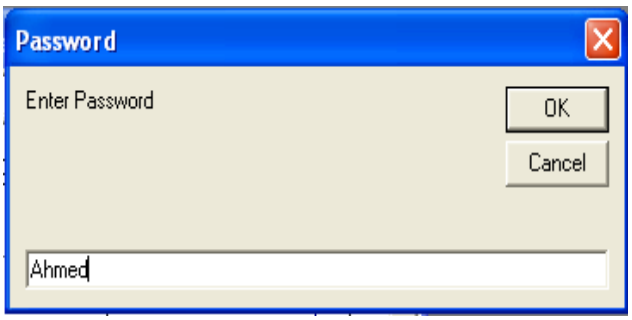
Project code

```
Private Sub Command1_Click()
 Label1.Caption = "Welcome"
End Sub

Private Sub Command2_Click()
 Label1.Caption = ""
End Sub

Private Sub Command3_Click()
 Label1.BackColor = RGB(Rnd * 255, Rnd * 255, Rnd * 255)
End Sub

Private Sub Command4_Click()
```

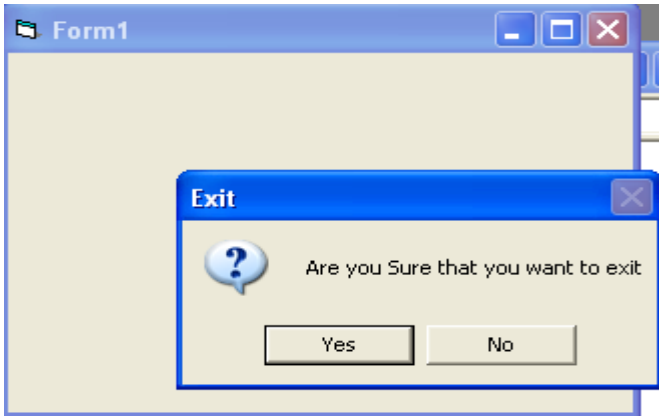
```
 If Label1.FontSize < 72 Then
   Label1.FontSize = Label1.FontSize + 2
 End If
End Sub

Private Sub Command5_Click()
 If Label1.FontSize > 8 Then
   Label1.FontSize = Label1.FontSize - 2
 End If
End Sub

Private Sub Command6_Click()
 Label1.FontBold = True
End Sub

Private Sub Command7_Click()
 Label1.FontBold = False
End Sub
```

س) قم بعمل برنامج بحيث عند بدء تحميل النموذج يعمل كلمة سر فإذا لم يكن الاسم المطلوب فينتهي. وكذلك التأكد من الرغبة في الخروج عند النقر على زر الاغلاق في شريط العنوان وإجراء اللازم عند رفض الخروج.

```
Private Sub Form_Load()
 r = InputBox("Enter Password", "Password", "")
 If  r <> "Ahmed" Then End
End Sub
```



```
Private Sub Form_Unload(Cancel As Integer)
 a = MsgBox("Are you Sure that you want to exit", vbYesNo + vbQuestion, "Exit")
 If  a = vbNo Then Cancel = True
End Sub
```



أحداث النموذج **Events**:

Load: حدث تحميل النموذج في الذاكرة

```
Private Sub Form_Load()

End Sub
```

Unload: حدث حذف النموذج وادواته من الذاكرة ويحدث عند النقر على الزر Close الموجود في شريط العنوان للنموذج

```
Private Sub Form_Unload(Cancel As Integer)

End Sub
```

* Cancel معناها اذا كانت الرغبة في عدم الخروج فنضع في شفرة هذا الحدث Cancel=True أي الغاء عملية الخروج وممكن ان نعمل ذلك بعد سؤال المستخدم عن رغبته في الخروج.

Activate: عندما يصبح النموذج هو النموذج الفعال بالنقر عليه أو عمل الاجراءٍ Show لاظهاره أو SetFocus للتركيز عليه فيحدث هذا الحدث.

Deactivate: عندما لا يصبح هذا النموذج هو النموذج الفعال عندما يفقد التركيز Focus باظهار نموذج آخر أو بالنقر أو التركيز على غير نموذج.

GotFocus: عندما يأخذ النموذج التركيز بواسطة الاجراء SetFocus (سيتم شرحه لاحقا مع زر الأوامر CommandButton)

LostFocus: عندما يفقد النموذج التركيز.

Resize: عند تغيير حجم النموذج

Paint: عندما يعاد رسم النموذج بعد التصغير أو التكبير أو تغيير الموقع ويعتبر System event

DragDrop: افلات السحب (سيتم شرحه لاحقا مع شريط الطباعة Label)

KeyDown، **KeyUp**: الضغط على مفتاح في لوحة المفاتيح KeyBoard، رفع اليد عن المفتاح وتعتبر User event تعطي الشفرة لكل الأزرار نستفاد منها مثلا في حالة الضغط على F1..F12

```
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
  If KeyCode = vbKeyF1 Then Print "F1"          ' عند الضغط على F1
End Sub
```

KeyCode هو شفرة المفتاح، Shift سيتم شرحها مع الماوس.

KeyPress: يعطي AsciiCode للمفتاح الذي ضغطناه (لا يرجع F1..F12)

```
Private Sub Form_KeyPress(KeyAscii As Integer)
  v = Chr(KeyAscii)
 If  v = "c" Then Form1.Cls
End Sub
```

* الدالة Chr تستخدم لتحويل الرقم AsciiCode الى حرف، عكسها الدالة Asc التي تحول الحرف الى رقم AsciiCode

Click: حدث النقر المفرد بالماوس على النموذج

DblClick: حدث النقر المزدوج بالماوس على النموذج

MouseMove: عند الحركة بالماوس ويعتبر User event

```
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)


End Sub
```

MouseDown، **MouseUp**: عند الضغط على الماوس، عند رفع اليد من الماوس.

```
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 1 Then
    Print "Left"
ElseIf Button = 2 Then
  Print "Right"
End If
End Sub
```

جملة **if** الشرطية

```
If condition Then [statements] [Else elsestatements]
```

أو

```
If condition Then
[statements]
[ElseIf condition-n Then
[elseifstatements] ...
[Else
[elsestatements]]
End If
```

Bits of Button			
2	1	0	Value
Middle	Right	Left	
0	0	1	1 (left)
0	1	0	2 (right)
0	1	1	3 (left and right)
1	0	0	4 (middle)

Bits of Shift		
2	1	0
Alt	Ctrl	Shift

جمل *Statements*

Beep: لاعطاء صوت قصير (تون) عبر مجهار الحاسبة.

End : لانهاء البرنامج باكمله، اي ايقاف تنفيذ البرنامج. وممكن وضعها في أي مكان من البرنامج وستقوم اوتوماتيكيا بغلق جميع الملفات المفتوحة.

Exit Sub: للخروج من البرنامج الفرعي عند أي موضع منه.

Randomize: لتهيأة مولد الأرقام العشوائية حيث توضع مرة واحدة في البرنامج في داخل الحدث Form_Load أو في الـــ File Module وتستعمل لضمان توليد قيم عشوائية عند كل بدء تشغيل للبرنامج.

Rnd: وهي دالة **Function** وليست جملة لانها تعيد رقم عشوائي أكبر او يساوي صفر وأقل من واحد

Rnd[(number)]

Label1

شريط الطباعة Label Control

تستخدم هذه الأداة لطباعة النتائج أو العبارات بصورة أكثر ترتيب من الايعاز Print

خصائص شريط الطباعة Label's Properties

لشريط الطباعة خصائص مماثلة لخصائص النموذج. مثل:-

Name, **Backcolor**, **Top, Left, Height, Width, Tag**, **MousePointer, MouseIcon, Enabled, Visible, RightToLeft, Font, FontName, FontBold, FontItalic, FontStrikethru, and FontUnderline**

مثال:

```
Label1.FontBold=True (set FontBold to True)
Label1.FontBold=False (reset FontBold to False)
Label1.FontBold= Not Label1.FontBold (toggle FontBold)
```

Caption: وهي أهم خاصية لشريط الطباعة حيث تمثل النص الظاهر عليه وهذا ما يهمنا. كما ان خصائص الخط تؤثر على هذه الخاصية حيث لا يوجد نص غيره في شريط الطباعة.

أمثلة:-

	في وقت التصميم	في وقت التنفيذ
1- كلمة	no	Label1.caption="no"
2- رقم	50	Label1.caption=50
		(Label1.caption=Label1.caption+20 ممكن بعد ذلك)
3- رقم وكلمة	no=30	Label1.caption="no="+str(x)
		where x=30, str to convert integer no. to string
4- مسح الكلمة	فراغ	Label1.caption="" 'null caption

5- الطباعة على أكثر من سطر

لا يجوز Label1.Caption = "Book" + Chr(10) + "Chair" + Chr(10) + "CopyBook"

أو تكتب Label1.Caption = "Book" + vbCrLf + "Chair" + vbCrLf + "CopyBook"

حيث vbCrLf تقابل Chr(10). وبالحقيقة فهي تقابل Chr(10)+Chr(13) لكن مع شريط الطباعة يكفي Chr(10) للنزول إلى السطر التالي. **حيث** Chr(13): للنزول إلى السطر التالي، Chr(10): للوقوف من بداية السطر.

'multiple lines (where autosize property=True and WordWrap=True)

قد يظهر أكثر من سطر على شريط الطباعة. عندما تكون الكتابة كثيرة والشريط صغير الحجم ولكن من غير تحديد لموقع الكلمات ضمن السطر الواحد.

Alignment: تحديد موقع العبارة ضمن شريط الطباعة صفر: يسار، 1: يمين، 2: يسار

Alignment	Setting	Description
vbLeftJustify	0	(Default) Text is left-aligned.
vbRightJustify	1	Text is right-aligned.
vbCenter	2	Text is centered.

AutoSize: تغير حجم الشريط اوتوماتيكيا.

True: يصبح الشريط بحجم الكلام الذي يحتوي عليه (عندما لا نعرف بالضبط حجم العبارة التي ستظهر).

False: يكون حجم الشريط ثابت فلا تظهر باقي الأحرف التي تتجاوز عرض الشريط.

BorderStyle: شكل حدود الشريط. صفر: ليس له حدود، 1: له حدود ثابتة.

BackStyle: شكل خلفية الشريط. صفر : شفاف فيعكس لون الخلفية ورائه (عندما ترافق صندوق النص Text Box)، 1: غير شفاف فيظهر لون خلفيته المعطى في خاصية BackColor لشد الانتباه نحو العبارة الظاهرة في الشريط.

BackStyle	Description
0	Transparent — background color and any graphics are visible behind the control.
1	(Default) Opaque — the control's BackColor property setting fills the control and obscures any color or graphics behind it.

Appearance: الظهور. صفر : لا تظهر عليه تأثيرات ثلاثية الأبعاد، 1: تظهر عليه تأثيرات ثلاثية الأبعاد.

DragIcon: صورة السحب والتي تظهر عند تحريك المؤشر اثناء عملية السحب من المصدر وحتى الافلات على المستقر

DragMode: طريقة السحب ممكن ان تكون (صفر) وتتم عن طريق استخدام الاجراء Drag المستخدم ضمن البرنامج اي يتم بدء السحب بواسطة ايعازات في البرنامج. أو تكون (1) اوتوماتيكي حيث يستطيع المستخدم بدء عملية السحب متى شاء ذلك.

Constant	Setting	Description
vbManual	0	(Default) Manual — requires using the Drag method to initiate a drag-and-drop operation on the source control.
vbAutomatic	1	Automatic — clicking the source control automatically initiates a drag-and-drop operation. OLE container controls are automatically dragged only when they don't have the focus.

ToolTipText: كلمة تظهر تحت الشريط عند التأشير عليه.

WordWrap: لضبط الكلمة عرضيا وطوليا عندما يكون AutoSize= True .

object.WordWrap [= boolean]

The settings for boolean are:

Setting	Description
True	The text wraps; the Label control expands or contracts vertically to fit the text and the size of the font. The horizontal size doesn't change.
False	(Default) The text doesn't wrap; the Label expands or contracts horizontally to fit the length of the text and vertically to fit the size of the font and the number of lines.

TabIndex: للحركة على أكثر من أداة بالـــ Tab ويمثل رقم هذه الأداة عند الحركة بالـــ Tab.

Index: موقع هذا الشريط ضمن مصفوفة من شريط الطباعة. كما سنلاحظ ذلك في المثال لاحقاً.

الخصائص **DataField & DataSource** تتعلق بقاعدة البيانات.

أحداث شريط الطباعة Events :

Click, DblClick, MouseDown, MouseUp, MouseMove, DragDrop events.

Change: يتم هذا الحدث عند تغيير محتوى شريط الطباعة بواسطة الربط مع قاعدة البيانات أو عند تغيير الخاصية Caption.

اجراءات شريط الطباعة: Methods

Move, Refresh, and Zorder Methods.

Zorder: موقعه الى الامام أو وراء الأداة التي تقع خلفه

Object.Zorder position

Position : عندما يكون الموقع صفر أو عندما لا يعطى الموقع يتقدم الى الأمام. واذا كان 1 فيكون الى الوراء.

Label1.Zorder ' شريط الطباعة الأول الى الأمام

Label1.Zorder 0 ' شريط الطباعة الأول الى الأمام

Label1.Zorder 1 ' شريط الطباعة الأول الى الوراء

ملاحظة:– تستخدم ' أي (single quotation mark) لعمل ملاحظة Comment في البرنامج وتقابل بذلك Rem في لغة بيسك.

مثال:

في هذا المثال

– سنطبع Ali Muna في شريط الطباعة الأول (عند تحميل النموذج لأن لدينا أكثر من سطر) ونجعل الخط مرة غامق ومرة فاتح بتعاقب النقر على هذا الشريط.

– أما شريط الطباعة الثاني فسنعمل منه مصفوفة (نجلب شريط الطباعة الثاني ثم نعمل له استنساخ Copy ثم نعمل لصق Paste لتكوين العنصر التالي في هذه المصفوفة فيسأل عند أول لصق هل تريد عمل مصفوفة فنختار Yes فيصبح Label2(0) بدل Label2 ويتكون Label2(1) وهكذا نعمل لصق للشريط التالي فيصبح لدينا Label2(2)، (نلاحظ ان المصفوفة تبدأ من الموقع 0)). (لاحظ الخاصية index والتي تمثل موقع هذا الشريط ضمن المصفوقة. كما نلاحظ هنا اختلاف الشفرة الخاصة بحدث النقر، حيث ان هناك برنامج فرعي واحد يمثل حدث النقر لكل عناصر المصفوفة لهذا الشريط وما يميز عنصر عن آخر هو الـــ index موقع هذا الشريط في المصفوفة، وبالامكان فحصه لمعرفة من الذي حدث عليه النقر من هذه الأشرطة الواقعة ضمن مصفوفة واحدة.

على سبيل المثال عند النقر على أي منها نكتب الموقع index ضمن المصفوفة. وممكن ان نفحص قيمة الـــ index بـــ if اذا كان التعامل أو الاجراء يختلف من واحد لآخر.

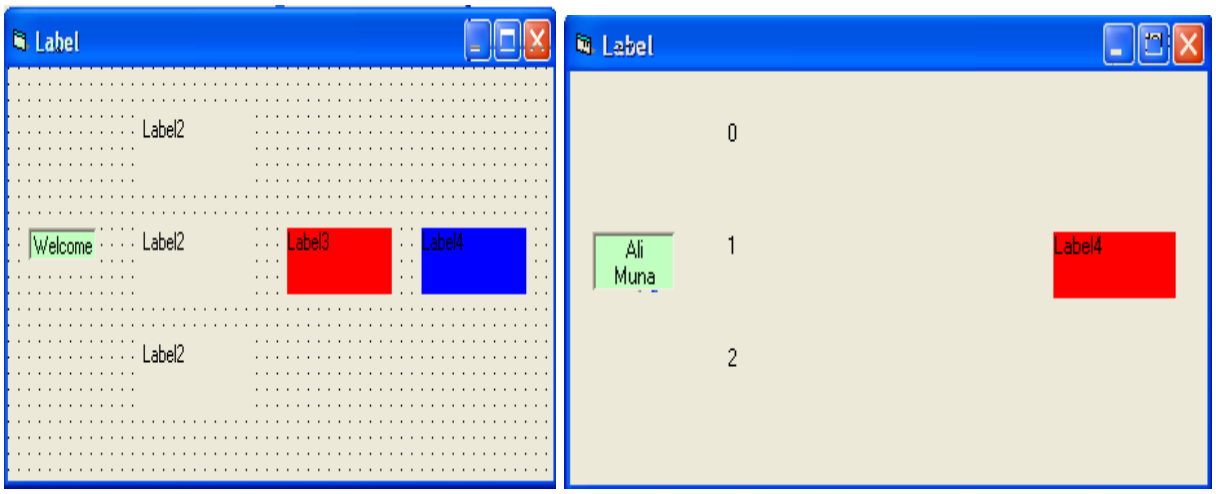
```
Private Sub Label2_Click(Index As Integer)
 Label2(Index).Caption = Index
End Sub
```

–أما شريط الطباعة الثالث والرابع فسنجرب عليها حدث افلات السحب DragDrop أي افلات السحب وسيكون شريط الطباعة الثالث هو الـــ Source المصدر للسحب فيجب ان نجعل

DragMode=1 لهذا الشريط أي Automatic ونعطي له صورة في DragIcon . وسنفلت السحب على شريط الطباعة الرابع حيث سيجري عليه الحدث DragDrop. ونبرمج هذا الحدث حيث سنجعل لون خلفية شريط الطباعة الرابع مساوي للون خلفية شريط الطباعة الثالث (المصدر) فيصبح لون خلفية شريط الطباعة أحمر ثم سنخفي شريط الطباعة الثالث.

```
Private Sub Label4_DragDrop(Source As Control, X As Single, Y As Single)
 Label4.BackColor = Source.BackColor
 Source.Visible = False
End Sub
```

- ثم سندرس خاصية Zorder لشريط الطباعة الثالث.



قبل التنفيذ بعد التنفيذ

Project properties

Object	Property	Setting
Form1	Name Caption	Form1 Label
Label1	Name Caption Alignment AutoSize BackColor BackStyle Appearance ToolTipText WordWrap	Label1 Welcome 2 الكتابة في الوسط True يصبح الشريط بحجم الكلمة اخضر 1 BackColor حتى يأخذ لون خلفية من 1 ثلاثي الأبعاد Test كلمة تظهر تحته عندما نؤشر عليه بالماوس True لضبط الكلمة
Label2(0)	Name Index	Label2 0
Label2(1)	Name Index	Label2 1
Label2(2)	Name Index	Label2 2
Label3	Name BackColor BackStyle DragMode DragIcon	Label3 احمر 1 1 'Automatic صورة
Label4	Name BackColor BackStyle	Label4 ازرق 1

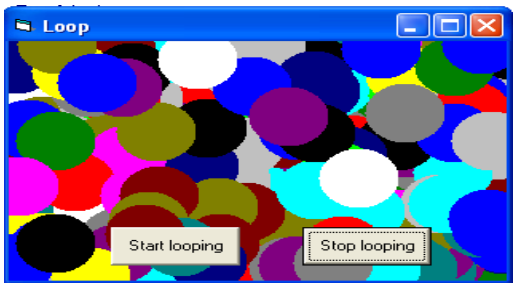
Poject code

```
Private Sub Form_Load()
 Label1.Caption = "Ali" + Chr(10) + "Muna"
End Sub

Private Sub Label1_Click()
 Label1.FontBold = Not Label1.FontBold
End Sub

Private Sub Label2_Click(Index As Integer)
 Label2(Index).Caption = Index
End Sub

Private Sub Label3_Click()
 Label3.ZOrder 0
End Sub
```

```
Private Sub Label4_DragDrop(Source As Control, X As Single, Y As Single)
 Label4.BackColor = Source.BackColor
 Source.Visible = False
End Sub
```

تطبيقات حول الرسم

س) لرسم دوائر بصورة عشوائية على الشاشة



Project properties

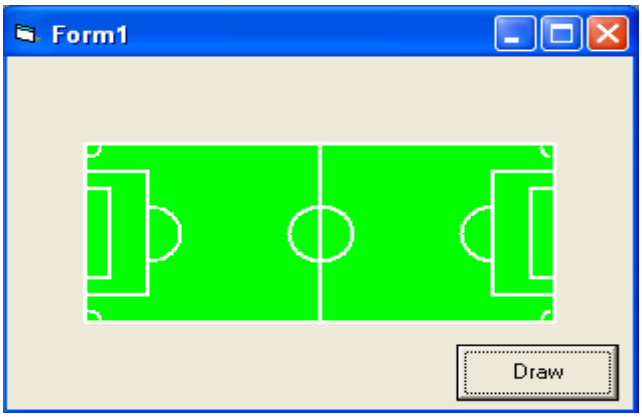
Object	Property	Setting
Form1	Name BackColor Caption	Form1 فضي Loop
CommandButton1	Name Caption	Command1 Start Loop
CommandButton2	Name Caption	Command2 Stop Loop

Poject code

```
Option Explicit
Dim g
Private Sub Command1_Click()
Dim c
Dim x, y
 g = 1
 Form1.CurrentX = Form1.ScaleWidth / 2
 Form1.CurrentY = Form1.ScaleHeight / 2
 Form1.Cls
 Form1.DrawWidth = 50
 c = 1
 Do While c < 1000
  x = Rnd * Form1.ScaleWidth
  y = Rnd * Form1.ScaleHeight
  PSet (x, y), QBColor(Rnd * 15)  'هنا تم رسم الدوائر على شكل نقاط عريضة
  c = c + 1
  If c = 999 Then
   c = 1
   Form1.Cls
  End If
  If g = 0 Then Exit Do
  DoEvents
  Loop
End Sub
```

```
Private Sub Command2_Click()
 g = 0
End Sub
```

س) ارسم ملعب كرة القدم بطريقة الاجراءات على النموذج



Project properties

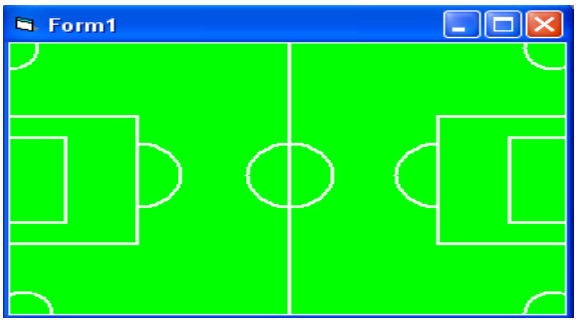
Object	Property	Setting
Form1	Name AutoRedraw	Form1 True
CommandButton1	Name Caption	Command1 Draw

Poject code

```
Private Sub Command1_Click()
 Form1.Line (-150, -100)-(150, 100), , B
 Form1.Circle (0, 0), 20
 Form1.Line (0, -100)-(0, 100)
  f = 1
  For i = 0 To 1
  Form1.Line (f * -150, -70)-(f * -110, 70), , B
  Form1.Line (f * -150, -50)-(f * -135, 50), , B
  Form1.Circle (f * -110, 0), 20, , ((270 + 180 * i) Mod 360) * 3.14 / 180, ((90 + 180
* i) Mod 360) * 3.14 / 180
  Form1.Circle (f * -147, -97), 6, , (270 - 90 * i) * 3.14 / 180, (360 - 90 * i) * 3.14 /
180
  Form1.Circle (f * -147, 97), 6, , (0 + 90 * i) * 3.14 / 180, (90 + 90 * i) * 3.14 / 180
  f = -f
 Next i
End Sub
Private Sub Form_Load()
 Form1.Scale (-200, -200)-(200, 200)
 Form1.DrawWidth = 2
```

```
 Form1.FillColor = vbGreen
 Form1.FillStyle = 0 'solid
 Form1.ForeColor = vbWhite
End Sub
```

س) ارسم الملعب بطريقة الأدوات



Project properties

Object	Property	Setting
Form1	Name	Form1
Shape1	Name Shape Left Top Width Height BorderWidth BorderColor FillStyle FillColor	Shape1 0 -rectangle 0 0 4695 3105 2 ابيض 0 -solid اخضر
Shape2(0)	Name Index Shape Left Top Width Height BorderWidth BorderColor FillStyle	Shape2 0 3 –circle 2000 1080 735 855 2 ابيض 1-Transparent
Shape2(1)	Name Index Shape Left Top Width Height	Shape2 1 3-circle 720 1080 735 855

	BorderWidth BorderColor FillStyle	2 ابيض 1-Transparent
Shape2(2)	Name Index Shape Left Top Width Height BorderWidth BorderColor FillStyle	Shape2 2 3-circle 3240 1080 735 855 2 ابيض 1-Transparent
Shape2(3)	Name Index Shape Left Top Width Height BorderWidth BorderColor FillStyle	Shape2 3 3-circle -240 -360 495 855 2 ابيض 1-Transparent
Shape2(4)	Name Index Shape Left Top Width Height BorderWidth BorderColor FillStyle	Shape2 4 3-circle 4320 -360 495 855 2 ابيض 1-Transparent
Shape2(5)	Name Index Shape Left Top Width Height BorderWidth BorderColor FillStyle	Shape2 5 3-circle -120 2640 495 855 2 ابيض 1-Transparent
Shape2(6)	Name Index Shape Left	Shape2 6 3-circle 4320

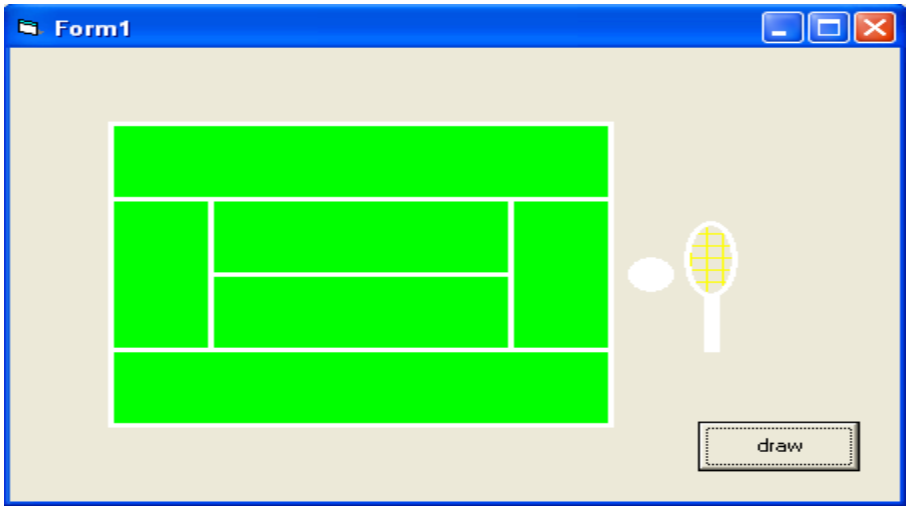
	Top Width Height BorderWidth BorderColor FillStyle	2640 495 855 2 ابيض 1-Transparent
Line1	Name X1 X2 Y1 Y2 BorderWidth BorderColor	Line1 2350 2350 0 3100 2 ابيض
Shape3(0)	Name Index Shape Left Top Width Height BorderWidth BorderColor FillStyle FilColor	Shape3 0 0-rectangle 0 480 1095 1455 2 ابيض 0-solid اخضر
Shape3(1)	Name Index Shape Left Top Width Height BorderWidth BorderColor FillStyle FilColor	Shape3 1 0-rectangle 3600 480 1095 1455 2 ابيض 0-solid اخضر
Shape3(2)	Name Index Shape Left Top Width Height BorderWidth BorderColor FillStyle FilColor	Shape3 2 0-rectangle 0 1080 495 975 2 ابيض 0-solid اخضر
Shape3(3)	Name	Shape3

	Index	3
	Shape	0-rectangle
	Left	4200
	Top	1080
	Width	495
	Height	975
	BorderWidth	2
	BorderColor	ابيض
	FillStyle	0-solid
	FilColor	اخضر

* تم استخدام المصفوفة هنا لكي تأخذ الاشكال المتقابلة نفس المواصفات من حيث الشكل واللون والعرض والارتفاع والجهة العليا.

* لعدم وجود منحني arc بطريقة الأدوات فسنقوم برسم دائرة شفافة أولاً ثم يرسم فوقها مستطيل أخضر لاخفاء الجزء الداخلي من الدائرة.

س) ارسم ملعب كرة التنس بطريقة الاجراءات على النموذج



Project properties

Object	Property	Setting
Form1	Name AutoRedraw ScaleMode	Form1 True 3 'pixel
CommandButton1	Name Caption	Command1 Draw

Poject code

```
Private Sub Command1_Click()
 Form1.FillColor = vbGreen
 Form1.FillStyle = vbSolid
 Form1.DrawWidth = 3
 Form1.Line (50, 50)-(300, 250), vbWhite, B
 Form1.Line (50, 100)-(300, 100), vbWhite
 Form1.Line (50, 200)-(300, 200), vbWhite
 Form1.Line (100, 100)-(100, 200), vbWhite
 Form1.Line (250, 100)-(250, 200), vbWhite
 Form1.Line (100, 150)-(250, 150), vbWhite

 Form1.FillColor = vbWhite
 Form1.FillStyle = vbSolid
 Form1.Circle (320, 150), 10, vbWhite

 Form1.Line (348, 164)-(353, 200), vbWhite, B
 Form1.FillColor = vbYellow
 Form1.FillStyle = vbCross
 Form1.Circle (350, 140), 24, vbWhite, , , 2
End Sub
```

النماذج المتعددة

جملة **Load** لتحميل النموذج في الذاكرة.

Load object

جملة **UnLoad** لحذف النموذج من الذاكرة.

Unload object

الاجراء **Show** لاظهار النموذج Form أو النموذج المتعدد MDIForm

object.Show style, ownerform

object.Show أو يكتب مباشرة '

القيمة Style اما ان تكون صفر (النموذج modeless) أو واحد(النموذج modal)

في حالة عدم تحميل النموذج واستخدام هذا الاجراء فان النموذج سيتحمل اوتوماتيكيا في الذاكرة.

يقابل Object.Visible=True ويختلف عنه في ان خاصية Visible لا تحمل النموذج اوتوماتيكيا الى الذاكرة.

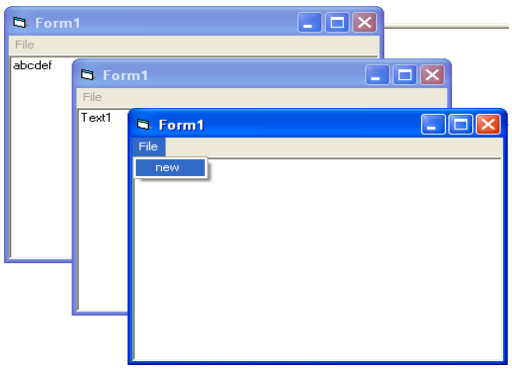
الاجراء **Hide** لاخفاء النموذج Form أو النموذج المتعدد MDIForm ولا يعمل حذف اوتوماتيكي من الذاكرة.

object.Hide

يقابل Object.Visible=False

1- النماذج المتشابهة

المثال التالي يقوم بخلق نماذج متشابهة.



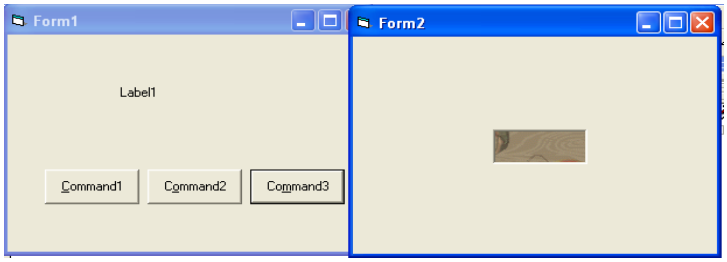
كون مشروعا جديدا وقم بإعداد القائمة New ثم اكتب الشفرة التالية:

```
Private Sub Form_Resize()
Me.Text1.Height = Me.ScaleHeight                ' Me أي نموذج يكون هو الفعال حاليا
Me.Text1.Width = Me.ScaleWidth
End Sub

Private Sub mnunew_Click()
 Dim newf As New Form1
 newf.Show              'test for newf.show 1
End Sub
```

2- النماذج المختلفة

المثال التالي يقوم بخلق نماذج مختلفة:-

كون مشروع جديد وقم باعداد النموذج1 ثم من القائمة Project اختر الأمر Add Form لاضافة نموذج2 وقم باعداده.




```
'In Form1
Private Sub Command1_Click()
' Form2.show
'Or
 Load Form2
 Form2.Show      'form2.Visible =True
End Sub
```

عند الرغبة في بدء العمل من نموذج2 وليس من نموذج 1 ففي هذه الحالة يجب اختيار القائمة project ومنها نختار project properties ثم من General وتحت عنوان :startup object نختار Form2

ملاحظات

1- بالامكان اضافة برنامج فرعي ضمن أي نموذج وكما يلي: من القائمة tools اختر الامر add procedure ثم اكتب اسم البرنامج الفرعي المطلوب

```
Private Sub Form_Click()
update
End Sub
```

```
Public Sub update()
 Form1.Caption = "hello"
 Form1.BackColor = vbRed
End Sub
```

2- لتعريف اي متغير في داخل أي حدث أو برنامج فرعي فبالامكان استخدام المتغير مباشرة أو تعريفه عن طريق الايعاز dim في بداية الحدث أو البرنامج الفرعي، قيمة هذا المتغير ستنتهي بانتهاء هذا البرنامج الفرعي أو الحدث. في حالة تعريف المتغير كـ static وليس dim فان قيمة هذا المتغير ستبقى لهذا البرنامج الفرعي فقط ولن تلغى حتى ولو خرجنا من هذا البرنامج الفرعي. ستكون قيمة المتغير الابتدائية عند بدء التعريف هي صفر.

```
 Private Sub Form_Click()
  Dim I as integer
  Static k
  K=k+1
  I=I+1
  Label1.caption=I
 Label2.caption=k
End Sub
```

3- عند الرغبة في عمل متغير عام على كل النموذج (يحتفظ بقيمته على طول البرنامج) فبالامكان تعريفه في بداية النموذج وبالصورة التالية:

```
Option Explicit
Dim e As Integer

Private Sub Form_Click()
 e = e + 1
 Label3.Caption = e
End Sub
Private Sub Label3_Click()
 e = e + 1
 Label3.Caption = e
End Sub
```

حيث ستزداد قيمة e سواء بالضغط على النموذج أو على Label3

4- ضمن نموذج معين مثلا نموذج2 ممكن تغيير الصفات لأدوات موجودة في نموذج آخر مثلا نموذج 1 وكما يأتي:

```
Form1.command1.caption="welcome"
```

أي يجب تحديد اسم النموذج أولاً ثم اسم الأداة ثم اسم الخاصية.

5- بالنسبة لتبادل قيم متغيرة بين النماذج فممكن تبادلها عن طريق الخاصية Tag والموجودة لكل أداة وبالاسلوب المكتوب في الملاحظة 4:

```
 Form1.command1.Tag=1
```

أو عن طريق تعريف متغيرات عامة على كل النماذج، حيث يتم تعريفها في Module، وكما يأتي: من القائمة Project اختر الأمر Add Module ثم اختر New Module، عندها ستظهر صفحة فارغة مخصصة للـ Module ممكن تعريف متغيرات عامة فيها أو اضافة البرنامج الفرعي الرئيسي فيها أو برامج فرعية عادية وسيخزن الـ Module كملف بامتداد bas. ، في حالة كتابة برنامج فرعي رئيسي فيه فنتبع نفس اسلوب اضافة البرنامج الفرعي في الملاحظة الأولى ولكن بكتابة اسم البرنامج الفرعي Main وممكن بدء تشغيل المشروع منه وذلك بعد اختيار القائمة project ومنها نختار project properties ثم من General وتحت عنوان :startup object نختـــار Sub Main

- ضمن الـ Module ضع على سبيل المثال الشفرة التالية:

```
Option Explicit
Global a
Public Sub main()
 Form1.Caption = "hi"
 Form1.Show
End Sub
```

- ضمن النموذج1، ضع على سبيل المثال الشفرة التالية:

```
Private Sub Form_click()
 a=a+1
 Label1.Caption = a
End Sub
```

- ضمن النموذج 2، ضع على سبيل المثال الشفرة التالية:

```
Private Sub Form_click()
 a=a+1
 Form1.Label1.Caption = a
 End Sub
```

هنا سيتنفذ أولاً البرنامج الفرعي الرئيسي وسيظهر النموذج1 وعنوانه hi وعند كل نقرة على النموذج 1 ستزداد قيمة a بواحد والذي تم تعريفه في Module وكذلك عند تحميل النموذج2 وعند كل نقرة عليه ستزداد قيمة a بواحد.

3– البرنامج متعدد الوثائق Multiple Document Interface MDI:

MDIForm Object

MDI (multiple-document interface) وهو النموذج الأب يكون بمثابة خلفية للنماذج الابناء التابعة له حيث تكون خاصية MDIChild= True لهذه النماذج الابناء. والنماذج الابناء ممكن ان تكون متشابهة أو مختلفة.

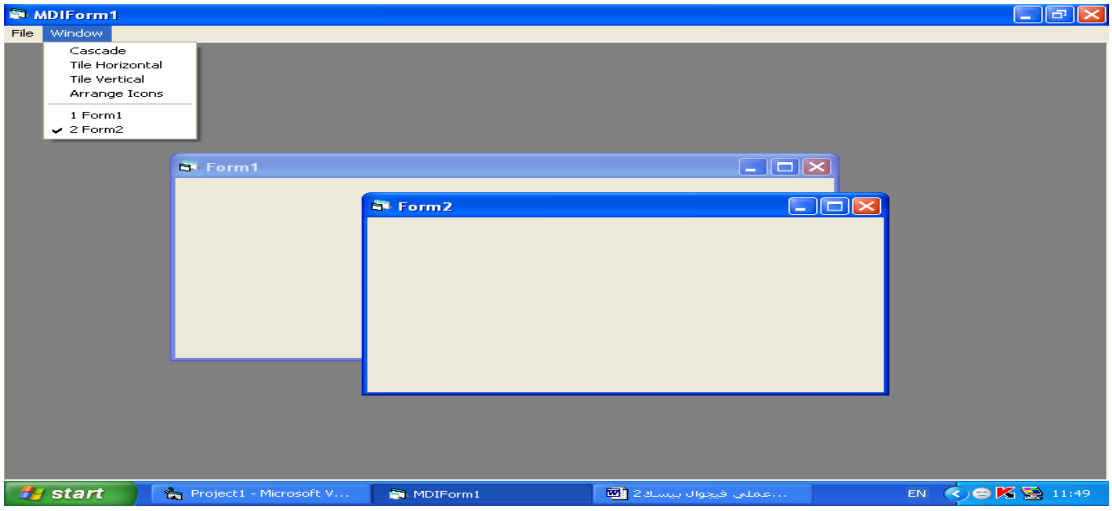
النموذج الأب ممكن ان يحوي فقط Menu and PictureBox controls أي قائمة اختيارات وصندوق صورة

الاجراء **Arrange** يستخدم لترتيب النماذج الابناء ويأخذ القيم (0..3) تتالي أو تجانب عرضي أو عمودي أو ترتيب النماذج المصغرة كايقونات.

Example:

```
MDIForm1.Arrange 1
```

Example:



خطوات العمل

1 - قم بانشاء مشروع جديد

2 - احذف نموذج1 وذلك باختيار الأمر Remove Form من القائمة Project

3 - من القائمة Project قم باختيار الأمر Add MDI Form

4 - من القائمة Project اختر الأمر Project Properties ثم من General وتحت عنوان Startup Object قم باختيار MDI Form ليكون بدء التنفيذ منه

5 - من القائمة Project اختر الأمر Add Form حينها سيتولد نموذج 1 (Form1) ثم اجعل خاصية MDIChild=True لهذا النموذج

6 - كرر الخطوة 5 لتكوين نموذج2

7 - يمكن التنقل بين النماذج عن طريق اختيار النموذج المطلوب ضمن Project Explorer

8 - من Project Explorer اختر النموذج الأب MDI Form ثم من القائمة Tools اختر الأمر Menu Editor لانشاء قائمة الاختيارات التالية، لا تنسى جعل الاختيار WindowList=True للقائمة Window

جدول قوائم النموذج الأب MDI Form

Caption	**Name**	
&File	Mnufile	
....Show Form1	Mnushow1	
....Show Form2	Mnushow2	
....-	Mnuu	
....Show All	Mnusall	
....Clear All	Mnucall	
&Window	Mnuwin	' Windowlist=True
....&Cascade	Mnucas	
....Tile &Horizontal	Mnuh	
....Tile &Vertical	Mnuv	
....&Arrange Icons	Mnui	

```
Private Sub MDIForm_Load()
 mnusall_Click
End Sub

Private Sub mnucall_Click()
 Unload Form1
 Unload Form2
End Sub

Private Sub mnucas_Click()
 MDIForm1.Arrange 0  'vbCascade
End Sub

Private Sub mnuh_Click()
 MDIForm1.Arrange 1  'vbTileHorizontal
End Sub

Private Sub mnui_Click()
 MDIForm1.Arrange 3  'vbArrangeIcons
End Sub

Private Sub mnusall_Click()
 Form1.Show
 Form2.Show
End Sub

Private Sub mnushow1_Click()
 Form1.Show
End Sub
```

```
Private Sub mnushow2_Click()
 Form2.Show
End Sub

Private Sub mnuv_Click()
 MDIForm1.Arrange 2  'vbTileVertical
End Sub
```

الاتصالات: للتعامل مع Serial Port تستخدم الأداة Mscomm Control الموجودة في الملف Microsoft comm control 5.0. هنا سيتم ذكر أهم الايعازات وبالإمكان تطوير ذلك إلى برنامج أكثر ترتيب.

قيم الـ **baudrate**: (110، 300، 600، 1200، 2400، 9600،..)

حالة التحقق Parity:

القيمة	الوصف
E	Even
M	Mask
N	None
O	Odd
S	Space

```
Private Sub Form_Load ()
        ' Buffer to hold input string
        Dim Instring As String
        ' Use COM1.
        MSComm1.CommPort = 1            ' تحديد رقم الـ port بين 1..4
        ' 9600 baud, no parity, 8 data, and 1 stop bit.
        MSComm1.Settings = "9600,N,8,1"        ' تهيأة الـ Port
        ' Tell the control to read entire buffer when Input
        ' is used.
        MSComm1.InputLen = 0    ' تهيأة الـ Port لإدخال القيم
        ' Open the port.
        MSComm1.PortOpen = True         ' فتح الـ port للتعامل معه
        ' Send the attention command to the modem.
        MSComm1.Output = "AT" + Chr$(13)
```

```
    ' Wait for data to come back to the serial port.
     Do
          DoEvents                    ' انتظار استلام بايتين أو أكثر
    Loop Until MSComm1.InBufferCount >= 2
    ' Read the "OK" response data in the serial port.
    Instring = MSComm1.Input          'القراءة من الـ port (استلام المعلومات)
    ' Close the serial port.
    MSComm1.PortOpen = False          ' إغلاق الـ Port
End Sub
```

فيجوال بيسك

فيجوال بيسك وهي من اللغات البرمجية العالية المستوى التي تعمل under window وتستخدم واجهات رسومية وأشكال أدت الى تسهيل العمل على المبرمج، حيث ان هناك اشياء (أدوات) جاهزة بالامكان استخدامها بصورة مباشرة بعد تغيير صفاتها لتصبح ملائمة للبرنامج المطلوب، كما انه بالامكان التعامل مع التطبيقات الأخرى وبسهولة. وهذه اللغة مبنية على أساس البرمجة الشيئيـــة Object Oriented Programming O.O.P والتي تقول ان البرنامج يتكون من مجموعة من الأشياء أو الأدوات، حيث تظهر أهميتها في خلق هذه الأشياء أو تحميلها في الذاكرة أوحذفها عند الحاجة مما يحقق سرعة في التنفيذ واستفادة أكبر من الذاكرة وعدم تحميلها ببرامج قد لا نحتاج لها طول عمل البرنامج.

مثلاً : عند الرغبة في عرض صورة معينة على الشاشة في لغة بيسك على سبيل المثال فكان على المبرمج قراءة ملف تلك الصورة ومن ثم رسمها كنقطة أثر نقطة حسب محتوى الصورة وحسب اسلوب خزنها في الملف، أما في فيجوال بيسك فان على المبرمج تحميل ملف الصورة لأي أداة تضم خاصية عرض صور وستعرض هذه الصورة مباشرة على الشاشة بدون حاجة إلى قراءة محتوى الملف أو معرفة طريقة الخزن فيه ومن ثم الرسم.

مثال آخر: عند الحاجة لعمل قائمة اختيارات فكان على المبرمج اعداد برنامج كامل لعرض القائمة والانتقال بين الاختيارات، أما في الفيجوال بيسك فهناك قائمة اختيارات جاهزة ممكن الانتقال بين اختياراتها بسهولة بدون برمجة عملية الانتقال بين خياراتها كما إنها تعرض بصورة لطيفة وكل ما على المبرمج فعله هو ان يضع فيها الاختيارات التي تخص برنامجه ومن ثم برمجة اختياراته.

المشروع project

في لغة بيسك التي تعمل under dos كنا نكتب البرنامج ثم ننفذه فتظهر لنا شاشة التنفيذ أما في لغة فيجوال بيسك فان البرنامج وشاشة التنفيذ موجودة لدينا قبل التنفيذ وما علينا هو تهيأة الشاشة بأدوات جميلة ثم نقوم بالتنفيذ حيث تظهر لنا شاشة التنفيذ وادواتها وكل ما علينا فعله هو الادخال والاختيار وملاحظة النتائج. شاشة التنفيذ تدعى **نموذج** Form وتعتبر أداة كباقي الأدوات عليهـــا

(أشياء objects) فمثلا هناك أداة للادخال (Text Box) صندوق نص، وأداة للطباعة (Label) شريط طباعة، وغيرها من الأدوات.

كل هذه الأدوات تشترك بأن لها:-

1- **صفات أو خصائص** (properties) كالارتفاع Height والعرض width والبعض من هذه الخصائص متشابهة لكل الأدوات والبعض الآخر مختلف بما يميز أداة عن أخرى حسب طريقة عملها أو الاستفادة منها.

2- **اجراءات** (methods) مثل Move للتحريك.

3- **أحداث** (events) قد تكون **user event** التي يقوم بها المستخدم كحدث النقر click أو قد تكون **system event** أي احداث النظام مثل الحدث Timer الخاص بالأداة (الميقاتي Timer) والذي يحدث من قبل النظام عند انقضاء الوقت المحدد من قبل المبرمج.

فالمشروع يتكون من برنامج واحد أو اكثر (كل له شاشة التنفيذ (النموذج Form)) الخاصة به (Form1, Form2,...) وليس من الضروري ان تكون محملة جميعها في الذاكرة من البداية.

فالمشروع هو العمل الذي يقوم بادارة كل الملفات المختلفة الخاصة بالتطبيق ويتكون من الملفات التالية:

1 -ملف المشروع ويحتوي على جميع الملفات المرتبطة بمشروع معين بالاضافة الى المعلومات التي تتعلق بالمشروع وله الامتداد vbp.

2 -ملف خاص لكل نموذج وله الامتداد frm. (حيث يتكون المشروع من نموذج واحد أو أكثر)

3 - ملف خاص لكل خطوات البرمجة MODULE FILE ويأخذ امتداد bas. (حيث ممكن ان يضم متغيرات عامة على كل النماذج وقد يتم الانطلاق بالعمل من هذا الملف وليس من النموذج)

الجدول التالي يوضح أنواع الملفات الأكثر استخداما عند إنشاء المشاريع داخل برنامج فيجوال بيسك:-

File Type نوع الملف	Extension الامتداد
Form	FRM
Module	BAS
Automatically generated file for every graphic in your project	FRX
ActiveX control	OCX
Class module	CLS
Visual Basic project	VBP

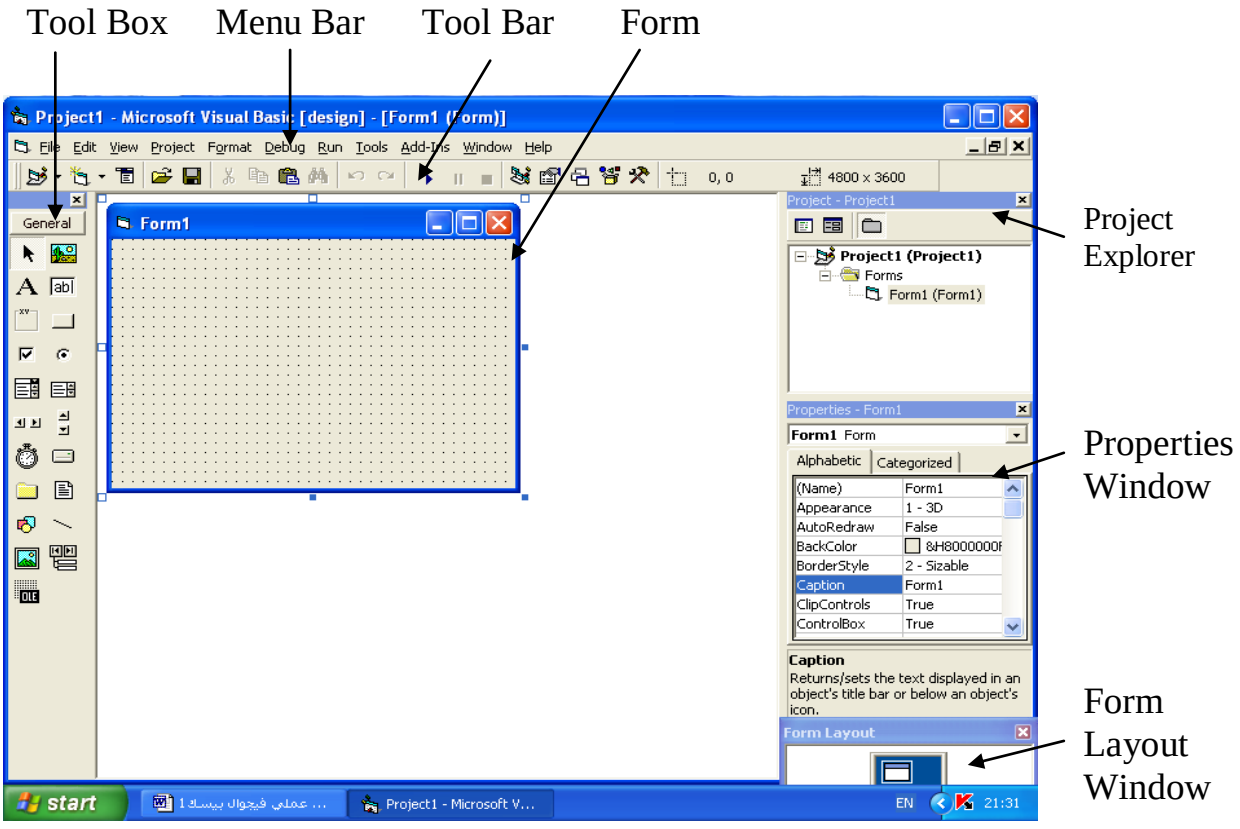
التشغيل

عند الضغط على الايقونة الخاصة بتنفيذ برنامج فيجوال بيسك ستظهر لنا شاشة بالامكان ان نختار عن طريقها عمل مشروع جديد NEW أو فتح مشروع موجود أصلاً تم عمله مسبقاً EXISTING أو بالامكان اختيار المشروع من قائمة آخر مجموعة من المشاريع تم فتحها مؤخراً RECENT .

عند اختيارنا لمشروع جديد فستظهر لدينا عدة اختيارات أخرى وعلينا اختيار الأمـــــر STANDARD EXE بالضغط عليه مرتين بالماوس أو الضغط عليه مرة واحدة والضغط على الأمر OPEN ليتم فتح مشروع جديد. عندها ستظهر لنا الواجهة الأساسية للبرنامج **(نافذة بيئة التطوير المتكاملة)**

نافذة بيئة التطوير المتكاملة (Integrated Development Environment: IDE) :

وهي الواجهة الرئيسية لبرنامج فيجوال بيسك. لاحظ الشكل التالي:-



(Integrated Development Environment: IDE) نافذة بيئة التطوير المتكاملة

وتتكون مما يأتي:-

1 شريط القوائم menu bar ويضم مجموعة من القوائم التي تتعلق بالتعامل مع الملفات الموجودة وعمليات العرض والتحرير والتنفيذ ... الخ

2 شريط الأدوات tool bar ويضم مجموعة من الأدوات الخاصة بالاختيار السريع للخزن والتنقيح... الخ

3 نافذة النموذج Form وتمثل منطقة العمل الأساسية التي يتم فيها تصميم المشروع والتطبيق وعليها تضاف الأدوات الأخرى.

4 صندوق أدوات التحكم Toolbox controls وهي العناصر الأساسية لتكوين أي مشروع حيث يمكن اختيار الأداة المطلوبة بالضغط بالماوس عليها مرتين (النقر المزدوج على الماوس باستخدام الزر الأيسر) عندها ستظهر تلك الأداة في وسط النموذج Form وبإمكان المبرمج تغيير موقعها حسب الرغبة، كما يمكن جلب الأداة بطرية السحب والافلات بالنقر المفرد على الأداة بالز الأيسر للماوس وباستمرار النقر تسحب الأداة (أو لا تسحب) يتم النقر على الموضع المطلوب ثم مع استمرار النقر يتم التحرك يمينا أو يسارا لتحديد عرض الأداة والحركة أعلى أو أسفل لتحديد ارتفاع الأداة ومن ثم رفع اليد عن الماوس (تنفع هذه الطريقة لجلب الأدوات داخل النموذج Form أو داخل صندوق الصورة Box Picture أو داخل الإطار Frame كما سنلاحظ ذلك لاحقا لتصبح الأدوات التي تم جلبها بهذه الطريقة تابعة للنموذج أو لأداة الصورة أو للإطار). ومن أمثلة هذه الأدوات أزرار الأوامر وأزرار الاختيار... الخ كما سيتم شرحها لاحقاً، وبالامكان جلب أدوات أحرى باختيار الايعاز components من القائمة project واختيار الملف الذي يضم أداة أو أكثر. ويمكن عمل ذلك بالضغط بالزر الأيمن للماوس على المنطقة الخالية ضمن صندوق الأدوات وستظهر قائمة نختار منها الأمر component. عند اختفاء صندوق الأدوات فبالامكان اظهاره باختيار الأمر Tool Box من القائمة View

5 نافذة مكتشف المشروع project explorer حيث تظهر مكونات المشروع من نمـــاذج و module file، عند اختيار النموذج المطلوب فممكن حينها ان نختار view code لاظهار البرنامج الخاص بهذا النموذج أو ان نختار view object ليظهر لنا النموذج وأدواته.

6 نافذة تصميم النموذج form layout window تبين موقع ظهور نافذة النموذج على الشاشة بعد تنفيذ التطبيق.

7 نافذة الخصائص properties window بالامكان اختيار النموذج أو أي أداة تم اضافتها على النموذج وتغيير الخصائص التابعة له، أي تغيير نعمله هنا يعتبر تغيير في وقت تصميم البرنامج **at design time** ويظهر تأثيره مباشرة أثناء التصميم، أما تغيير خصائص أي أداة عند كتابة البرنامج الفرعي الخاص بأي حدث فسيظهر التأثير في وقت التنفيذ عند حدوث ذلك الحدث **at run time** (سيتم ملاحظة ذلك لاحقاً)

من أهم الخصائص الموجودة:

Name: اعطاء اسم لأداة التحكم يتم من خلاله التعامل مع الأداة

Backcolor: اختيار لون الخلفية لبعض الأدوات التي تسمح بذلك

Caption: تعطي عنواناً ظاهرياً للأداة على نافذة النموذج

Enabled: امكانية استخدام الأداة على الشاشة

Font: يتم من خلاله تحديد صفات الخط Format

Height : تحديد طول الأداة

Left : تحديد الجهة اليسرى من الأداة

Text : تقوم بادخال وتعديل النص داخل الأداة

Tag: وهي قيمة ممكن ان تكون كـ flag

Top تحديد الجهة العليا للأداة

Width : تحديد عرض أداة التحكم

Visible: تبين امكانية اظهار الاداة أو اختفائها من على سطح النموذج

مراحل كتابة البرنامج وتنفيذه

1 مرحلة البرمجة المرئية (visual programming) والمتضمنة انشاء أدوات التحكم المطلوبة للبرنامج وتغيير خصائص هذه الأدوات من نافذة الخصائص.

2 مرحلة كتابة الشفرة (code programming) وتتم بواسطة محرر النصوص الخاص ببرنامج visual basic

3 لكي نقوم بكتابة البرنامج فنختار View Code حينها سيظهر لنا محرر النصوص الخاص ببرنامج visual basic (العودة الى الأدوات المرئية تكون باختيار View Object) ثم سنقوم باختيار الأداة المطلوبة والحدث المطلوب (حيث تم تقسيم البرنامج الى برامج فرعية خاصة بأحداث كل أداة) حينها سيظهر لنا السطر الاول والسطر الأخير من الحدث.

```
Private Sub object_event([parameters])

End Sub
```

السطر الأول من الشفرة (code) الخاصة بالحدث يبدأ بعبارة Private Sub ثم يأتي بعدها اسم الأداة ثم _ ومن ثم اسم الحدث، أما السطر الأخير من الشفرة (code) فهو عبارة End Sub ولا يمكن أن يخلو أي برنامج فرعي خاص بحدث أي أداة من هذين الجملتين.

كلمة Sub وتعني subroutine أي برنامج فرعي، أما Private فتعني ان هذا البرنامج الفرعي خاص بتلك الأداة فقط. أما Parameters فهي واحدة أو أكثر قد تكون موجودة أو غير موجودة ( ) وتعني تمريرات النظام الخاصة بالحدث.

أما السطر الأخير End Sub فتعني نهاية برنامج فرعي.

بين هذين السطرين نكتب الايعازات الخاصة بتغيير الخصائص أو نكتب الاجراءات المطلوبة.

لتغيير خاصية معينة نكتب (الصيغة العامة):- Object.Property=value

Object: أو (Toolname) اسم الأداة أو الشيء (الكائن)، Property: الخاصية المطلوبة، value: القيمة التي قد تكون رقم أو قد تكون سلسلة حرفية نضعها بين "" أو قد تكون True or False أو قد نساويها بخاصية أو نفي الخاصية الخاصة بنفس الأداة أو أداة أخرى وسنلاحظ ذلك لاحقاً.

أما **استخدام الاجراء** فيكون بالصيغة العامة:- Object.Method [values]

Method: اسم الاجراء، وبعده لا نكتب = ، عندما يحتاج الاجراء الى قيم فبعد اسم الاجراء نضع فراغ واحد ثم نكتب القيم مباشرة وبينها فارزة ,

ملاحظة:- عندما نضع النقطة . بعد اسم الأداة مباشرة تظهر لنا قائمة فنكتب أول حرف من الخاصية أو الاجراء فتظهر لنا كل الخصائص والاجراءات التي تبدأ بهذا الحرف وما علينا فعله هو ان نختار ما نطلبه عن طريق الأسهم أو الماوس ثم نضغط Space bar حينها ستكتب الخاصية المطلوبة أو الاجراء مباشرة بعد اسم الأداة.

التنفيذ:-

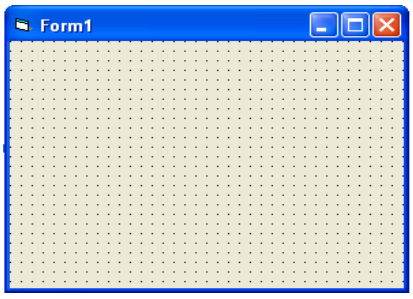
يتم تنفيذ البرنامج بالضغط على المفتاح F5 من لوحة المفاتيح أو اختيار الايعاز run من شريط القوائم ثم اختيار start أو بالضغط على المثلث الأزرق في شريط الأدوات.

الخزن:-

يفضل انشاء مجلد خاص لكل مشروع ويخزن فيه المشروع وملفاته من نماذج وغيرها. باستخدام SaveAs أو Save لأول مرة لخزن هذا المشروع. ثم الخزن بـــ Save لمرات الخزن اللاحقة عند التعديل. ولتكوين برنامج تنفيذي لهذا المشروع فمن القائمة File يتم اختيار Make Exe.

النموذج Form

وهو سطح توضع عليه باقي الأدوات كما ويمكن الرسم عليه.



خصائص النموذج

Name: الاسم بالامكان تغييره فقط في نافذة الخصائص أي في وقت التصميم. ويسمى أول نموذج تلقائيا بـــ Form1 والنموذج الثاني Form2 وهكذا. (تقرير تغيير الاسم يكون منذ البداية لأي أداة عند جلبها مباشرة).

Caption: النص الظاهر على النموذج قرب الايقونة في شريط العنوان Title Bar يبدأ بنفس اسم النموذج التلقائي وممكن ان نغيره بأي كلمة. object.Caption [= value]

Backcolor: لون خلفية النموذج وبالامكان تغييره بأي لون. object.BackColor [= color]

مثال:-

- نغير الـــ Caption الى Hello ولون الخلفية الى احمر في نافذة الخصائص نلاحظ ان التغيير يحدث قبل التنفيذ ويظهر بعد التنفيذ ايضا. (at design time)
- لنعمل ذلك في البرنامج بواسطة الايعازات فنحول الى محرر نصوص البرنامج باختيـــــــــار View Code ثم نختار مثلا للأداة Form نختار الحدث Load ويعني حدث تحميل النموذج في الذاكرة ويحدث مباشرة بعد التنفيذ (at run time) ولا يحتاج الى تدخل المستخدم. ونعمل ذلك بالشكل التالي:-

```
Private Sub Form_Load()
 Form1.Caption = "Hi"
 Form1.BackColor = vbGreen
End Sub
```

نلاحظ ان اسم الأداة في أول سطر تكون Form وليست Form1 او الاسم الذي نغيره لان في داخل النموذج الواحد يوجد نموذج واحد. (ولكن لبقية الادوات تأخذ الاسم المذكور في خاصية Name). ولكن عند تغيير الخصائص او استخدام الاجراءات فنكتب دائما الاسم المذكور في خاصية Name.

مثلا: عند تغيير اسم النموذج الى test في نافذة الخصائص

```
Private Sub Form_Load()
    test.Caption = "Hi"
End Sub
```

نلاحظ ان اللون مسبوق بـــ Vb وتعني Visual Basic وهذه أحدى طرق التلوين التي ستدرس لاحقا.

عندما نضغط على الزر F5 نرى هذه التغييرات مباشرة بدلا من تغييرات وقت التصميم.

- بينما اذا اخترنا user event للنموذج مثل الحدث click وهو حدث النقر المفرد على النموذج:-

```
Private Sub Form_Click()
 Form1.Caption = "Welcome"
 Form1.BackColor = vbBlue
End Sub
```

هنا عند التنفيذ نشاهد التغييرات التي عملناها في حدث التحميل Load ولا نشاهد التغييرات التي عملناها في حدث النقر click فنشاهد هذه التغييرات بعد النقر المفرد على النموذج لان هذا الحدث هو user event.

BorderStyle: شكل الحدود مثلا صفر: بلا شريط عنوان أو حدود وهذا غير مرغوب به، 1: حدود ثابتة فلا يسمح بتغيير الحجم للنموذج ReSize، 2: الافضل حيث يسمح بتغيير الحجم ،.. الخ. تحدد هذه القيمة في وقت التصميم فقط وممكن ان نقرأها في وقت التنفيذ read-only at run time.

BorderStyle	Setting	Description
vbBSNone	0	None (no border or border-related elements).
vbFixedSingle	1	Fixed Single. Can include Control-menu box, title bar, Maximize button, and Minimize button. Resizable only using Maximize and Minimize buttons.
vbSizable	2	(Default) Sizable. Resizable using any of the optional border elements listed for setting 1.
vbFixedDouble	3	Fixed Dialog. Can include Control-menu box and title bar; can't include Maximize or Minimize buttons. Not resizable.
vbFixedToolWindow	4	Fixed ToolWindow. Displays a non-sizable window with a Close button and title bar text in a reduced font size. The form does not appear in the Windows 95 task bar.
vbSizableToolWindow	5	Sizable ToolWindow. Displays a sizable window with a Close button and title bar text in a reduced font size. The form does not appear in the Windows 95 task bar.

Enabled: True يعمل يستجيب للحدث مثلاً يتقبل النقر ، False لا يعمل لا يتقبل الحدث (وهذا لا ينفع عند وجود نموذج واحد)

Visible: True مرئي، False غير مرئي (وهذا لا ينفع عند وجود نموذج واحد)

MaxButton: True يوجد زر تكبير أي يسمح بتكبير النموذج، False لا يوجد زر تكبير (التغيير في وقت التصميم فقط)

MinButton: True يوجد زر تصغير أي يسمح بتصغير النموذج ، False لا يوجد زر تصغير (التغيير في وقت التصميم فقط)

Moveable: True قابل للحركة وتغيير الموقع، False غير قابل للحركة(التغيير في وقت التصميم فقط)

RightToLeft: True الكتابة من اليمين الى اليسار، False الكتابة من اليسار الى اليمين (التغيير في وقت التصميم فقط)

StartUpPosition: الموقع الذي يبدأ فيه النموذج بالنسبة لشاشة الحاسبة (التغيير في وقت التصميم فقط)

StartUpPosition	Value	Description
vbStartUpManual	0	No initial setting specified.
vbStartUpOwner belongs.	1	Center on the item to which the UserForm
vbStartUpScreen	2	Center on the whole screen.
vbStartUpWindowsDefault screen.	3	Position in upper-left corner of

ShowInTaskbar: True يظهر عند التصغير في شريط المهام ، False لا يظهر في شريط المهام (التغيير في وقت التصميم فقط)

WindowState: حالة النموذج: صفر حجم عادي، 1 مصغر كايقونة في شريط المهام، 2: مكبر بحجم الشاشة (التغيير في وقت التصميم فقط)

Tag: يستخدم كعلم Flag يضم رقم أو سلسلة حرفية String أو يستخدم لتبادل قيمة بين أكثر من نموذج.

Left : تحديد الجهة اليسرى للنموذج نسبة لشاشة الحاسبة (ممكن اعتبارها X1)

Top : تحديد الجهة العليا للنموذج نسبة لشاشة الحاسبة (ممكن اعتبارها Y1)

Height : تحديد ارتفاع النموذج (وممكن تمثيله بـــ y2-y1)، حيث y2 تمثل الجهة السفلى للنموذج

Width : تحديد عرض النموذج (وممكن تمثيله بـــ X2-X1)، حيث X2 تمثل الجهة اليمنى للنموذج

ممكن ان نحدث أي منها في داخل الشفرة الخاصة بأي حدث مثلا:- Form1.Left=1000

- في حالة الحاجة لتغيير أكثر من واحدة منها في وقت التنفيذ فممكن ان نعوض عن ذلك باستخدام الاجراء Move مثلا:- Form1.Move left,Top,Width,Height

فممكن ان نحرك النموذج الى الموقع X1=1000، Y1=2000 نسبة لشاشة الحاسبة فنقول:

Form1.Move 1000,2000

القيمة التي لا نغيرها ممكن ان نضع أمامها فراغ أو نفس القيمة مثلا Form1.Width مقابل Width اذا لا نريد تغييره، أما اذا اردنا زيادته بـــ 50 مثلا فنقابل الـــ Width بـــ Form1.Width+50

Icon: اعطاء صورة للآيقونة التي تظهر في شريط عنوان النموذج.

Picture: اعطاء صورة للنموذج.

MousePointer: تحديد شكل ايقونة الماوس القيمة بين (0-15)

MousePointer	Value	Description
vbDefault	0	(Default) Shape determined by the object.
VbArrow	1	Arrow.
VbCrosshair	2	Cross (crosshair pointer).
VbIbeam	3	I beam.
VbIconPointer	4	Icon (small square within a square).
VbSizePointer	5	Size (four-pointed arrow pointing north, south, east, and west).
VbSizeNESW	6	Size NE SW (double arrow pointing northeast and southwest).
VbSizeNS	7	Size N S (double arrow pointing north and south).
VbSizeNWSE	8	Size NW SE (double arrow pointing northwest and southeast).
VbSizeWE	9	Size W E (double arrow pointing west and east).
VbUpArrow	10	Up Arrow.
VbHourglass	11	Hourglass (wait).
VbNoDrop	12	No Drop.
VbArrowHourglass	13	Arrow and hourglass.
vbArrowQuestion	14	Arrow and question mark.
vbSizeAll	15	Size all.
vbCustom	99	Custom icon specified by the MouseIcon property.

MouseIcon: اعطاء صورة من عندنا لايقونة الماوس ولكن يجب ان تكون MousePointer=99 أي Custom. وممكن اعطاء اي صورة مثلا من المسار: "c:\windows\cursors\"

```
object.MouseIcon = LoadPicture(pathname)
object.MouseIcon [= picture]
```

في حالة تغيير الصورة في وقت التنفيذ فبالحدث المطلوب نعمل ما يأتي:

```
Form1.Picture=LoadPicture("c:\windows\cursors\اسم صورة")
```

حيث استخدمنا دالة تحميل الصورة LoadPicture.

وممكن ان نستحدم نفس الدالة لمسح الصورة: Form1.Picture=LoadPicture()

وممكن ان نساوي الصورة بصورة اخرى لغير أداة وهكذا Form1.Picture=Image1.Picture

FontName: اسم خط object.FontName [= font]

مثلا: object.FontName ="Arial"

الخصائص FontBold: الخط غامق أم لا، **FontItalic**: الخط مائل أم لا، **FontStrikethru**: يوجد خط وسط الكلمة أم لا، **FontUnderline**: يوجد خط اسفل الكلمة أم لا كلها تأخذ قيم ثنائية Boolean أما True أو False.

مثلا:

Form1.FontBold=True غامق

Form1.FontBold=False فاتح

لعمل Toggle مرة غامق ومرة فاتح على حدث النقر مثلا ' Form1.FontBold=Not Form1.FontBold

Font: هذه الخاصية تحوي باقي خصائص الخط بداخلها مثلا: Form1.Font.Bold=True

FontSize: حجم الخط مثلا Form1.FontSize=16

ForeColor: لون خط الكتابة أو لون (النقطة، الخط، اطار المستطيل والمربع والدائرة) بالنسبة للرسم

مثلا Form1.ForeColor=VbRed

ملاحظة مهمة: خصائص الخط بالنسبة للنموذج تؤثر على الايعاز Print وليس Caption ولكن في بقية الأدوات تؤثر على Caption وفي صندوق النص Text Box تؤثر على Text

مثال: عند النقر على النموذج اطبع الكلمة Welcome بالخط Arial وحجم 14 وتحتها خط وبلون احمر

```
Private Sub Form_Click()
  Form1.FontName = "Arial"
  Form1.FontSize = 14
  Form1.FontUnderline = True
```

```
  Form1.ForeColor = vbRed
  Print "Welcome"
End Sub
```

ملاحظة: هناك مجموعة من الخصائص الخاصة بالرسم المباشر على النموذج تبدأ بكلمـــــــــــــة Draw or Scaleسنتطرق لها لاحقاً .

الاجراءات Methods

Move: لتغيير موقع أو حجم النموذج object.Move left, top, width, height

Refresh: لانعاش النموذج

Cls: لتنظيف النموذج Form1.Cls

س) اكتب الشفرة الخاصة بحدث النقر على النموذج والمتضمنة ما يأتي:-

```
Private sub Form_Click( )
```

1- تغيير لون خلفية النموذج إلى أزرق

```
  Form1.BackColor=VbBlue
```

2- تغيير النص الظاهر على النموذج إلى "Help"

```
  Form1.Caption="Help"
```

3- مسح الصورة المحملة على النموذج

```
  Form1.Picture=LoadPicture()
```

4- تنظيف النموذج

```
 Form1.Cls
 End Sub
```

القائمة ListBox Control :

وهي قائمة لعرض مصفوفة لسلسلة حرفية أو رقمية List of items وبإمكان المستخدم اختيار واحدة أو أكثر منها.

الخصائص Properties :

List: وهي مصفوفة لسلسلة حرفية تضم الكلمات التي يتم عرضها في هذه الأداة. وممكن أن يتم تحميلها في وقت التصميم أو في وقت التنفيذ مثلا في حدث تحميل النموذج، وتبدأ من الموقع صفر.

object.List(index) [= string]

فالعنصر الأول للقائمة الأولى يمثل بـــ List1.List(0). والعنصر الثاني للقائمة الأولى يمثل بـــ List1.List(1).

ItemData: وهي مصفوفة رقمية تقابل المصفوفة list حيث هناك رقم يمثل شفرة يقابل كل كلمة. وعند وضع أي كلمة في القائمة list يوضع اوتوماتيكيا مقابله الرقم صفر الذي يمكن تحديثه سواء في وقت التصميم أو في وقت التنفيذ.

مثلا: عند إضافة عنصر جديد إلى القائمة list باستخدام الإجراء AddItem فإن موقع العنصر الذي تم إضافته يمثل بالخاصية NewIndex الذي يمكن الاستفادة منه لتحديث القيمة الموجودة في المصفوفة ItemData وعلى النحو التالي:-

```
List1.AddItem "Judy Phelps"
List1.ItemData(List1.NewIndex) = 42310
```

NewIndex: موقع العنصر الجديد الذي تم إضافته بواسطة الإجراء AddItem في وقت التنفيذ أينما كان.

ListCount: عدد العناصر الموجودة في القائمة.

Sorted: عندما تكون True فإن العناصر الموجودة في القائمة سيتم ترتيبها حسب الأحرف الأبجدية، أما إذا كانت False فستبقى العناصر على حالها وحسب إدخالها أي لا يهم الترتيب حسب الأحرف الأبجدية.

Column: عدد الأعمدة التي تظهر بها عناصر القائمة، فعندما تكون هذه الخاصية صفر فستعرض على شكل عمود واحد، أما إذا كانت أكثر من صفر مثلا 2 فستعرض العناصر على شكل عمودين حيث يملىء العمود الأول بالعناصر ومن ثم العمود الثاني وسيضاف شريط تمرير أفقي لرؤية باقي العناصر حيث بالحركة عليه سيظهر العمود الثاني الذي كان ظاهرا كعمود أول ويظهر على يمينه عناصر جديدة تالية في العمود الثاني.

object.Columns [= number]

number	Description
0	(Default) Items are arranged in a single column and the ListBox scrolls vertically.
1 to n	Items are arranged in snaking columns, filling the first column, then the second column, and so on. The ListBox scrolls horizontally and displays the specified number of columns.

MultiSelect: تعد هذه الخاصية في وقت التصميم وهي أما أن تكون صفر فبامكان المستخدم اختيار عنصر واحد من القائمة بالنقر عليه، أو أن تكون 1 فبإمكان المستخدم اختيار أكثر من عنصر بصورة عشوائية أي غير متسلسلة بالنقر على العنصر بواسطة الماوس أو بالحركة عليه بالأسهم ومن ثم الضغط على المفتاح SPACEBAR وبنفس الاسلوب ممكن التراجع عن الاختيار أي بالنقر عليه بالماوس أو بالضغط على المفتاح SPACEBAR، عندما تكون 2 فإن بالإمكان اختيار أكثر من عنصر بصورة متسلسلة فيكون الاختيار بالضغط على المفتاح SHIFT وبالنقر على الماوس واستمرار الضغط على المفتاح SHIFT مع الحركة بواسطة الماوس أو الأسهم (UP ARROW, DOWN ARROW, LEFT ARROW, and RIGHT ARROW) حتى نصل إلى آخر عنصر حينها يتم رفع اليد عن المفاتيح أو الماوس، ويمكن استخدام المفتاح CTRL بدل SHIFT والذي يسمح بالاختيار أو التراجع عن الاختيار.

MultiSelect	Description
0	(Default) Multiple selection isn't allowed.
1	Simple multiple selection. A mouse click or pressing the SPACEBAR selects or deselects an item in the list. (Arrow keys move the focus.)
2	Extended multiple selection. Pressing SHIFT and clicking the mouse or pressing SHIFT and one of the arrow keys (UP ARROW, DOWN ARROW, LEFT ARROW, and RIGHT ARROW) extends the selection from the previously selected item to the current item. Pressing CTRL and clicking the mouse selects or deselects an item in the list.

ListIndex: موقع الكلمة أو العنصر الذي تم اختياره عندما تكون MultiSelect=0 أما في حالة كون MultiSelect أكبر من صفر فإنها تمثل موقع آخر كلمة أو عنصر تم اختياره.

Text: الكلمة أو العنصر الذي تم اختياره عندما تكون MultiSelect=0 أما في حالة كون MultiSelect أكبر من صفر فإنها تمثل آخر كلمة أو عنصر تم اختياره.

فمثلا للقائمة الأولى تمثل هذه الكلمة المختارة بالاسلوب:- List1.Text

أو تمثل بالاسلوب:- List1.List(List1.ListIndex)

أي الكلمة الموجودة حسب الموقع الذي تم اختياره في المصفوفة List.

* عندما لا يتم اختيار أي عنصر فإن ListIndex سوف يساوي -1، وستكون الخاصية Text=""

SelCount: عدد العناصر الذي تم اختيارها.

Selected: تمثل مصفوفة حالة الاختيار (صح أو خطأ) مقابلة للمصفوفة List فالعنصر الذي يتم اختياره في المصفوفة List سيخزن مقابله True في المصفوفة Selected وفي حالة عدم اختيار العنصر فسيخزن مقابله False في هذه المصفوقة.

ملاحظة:- ممكن تحديد عنصر في وقت التنفيذ، مثلا تحديد أو اختيار العنصر الثاني للقائمة وكما يأتي:-

```
List1.ListIndex = 1
```

وممكن عمل نفس الشيء بالاسلوب التالي:-

```
List1.Selected(1) = True
```

ممكن الغاء اختيار العنصر وبالاسلوب التالي:-

```
List1.Selected(1) = False
```

Style: خاصية ممكن تحديدها في وقت التصميم فقط وعندما تكون الخاصية MultiSelect=صفر فقط، فإذا جعلنا خاصية Style صفرا فتظهر القائمة بالشكل العادي أما إذا كانت هذه الخاصية مساوية لواحد فتظهر عناصر القائمة وأمامها صندوق تحقق checkbox لاختيار العنصر عن طريقه.

Style	Value	Description
vbListBoxStandard	0	(Default) Standard. The ListBox control displays as it did in previous versions of Visual Basic; That is, as a list of text items.
vbListBoxCheckbox	1	CheckBox. The ListBox control displays with a checkbox next to each text item. Multiple items in the ListBox can be selected by selecting the checkbox beside them.

الأحداث Events

Click, DblClick, MouseDown, MouseUp, MouseMove, DragDrop, KeyDown, KeyUp, KeyPress, GetFocus and LostFocus events

وهي مشابهة لأحداث النموذج.

Scrool: يتم هذا الحدث عند تحريك صندوق التمرير الموجود في شريط التمرير العمودي أو الأفقي لهذه الأداة.

ItemCheck: يتم هذا الحدث عندما يتم اختيار رفع الاختيار (التراجع عنه) لعنصر موجود في القائمة عندما تكون خاصية Style مساوية لواحد.

```
Private Sub List1_ItemCheck(Item As Integer)
End Sub
```

الإجراءات Methods :

Move, Refresh, SetFocus and Zorder Methods

Additem : وهو الإجراء الذي يستخدم لإضافة عنصر جديد إلى القائمة. وصيغته العامة:-

```
object.AddItem item, index
```

حيث يتم إعطاء الكلمة وموقع الإضافة في القائمة.

الإضافة في نهاية القائمة:-

مثلا لإضافة السلسلة الحرفية "Judy Phelps" إلى نهاية القائمة:-

```
List1.AddItem "Judy Phelps"
```

ممكن أن تكون الكلمة مخزونة في متغير مثل x فيكتب الإيعاز بالشكل التالي:-

```
X="Judy Phelps"
List1.AddItem x
```

أو قد تكون مخزونة في صندوق إدخال:-

```
List1.AddItem Text1.Text
```

أو قد تكون مخزونة في سلسلة حرفية لأي أداة أخرى.

الإضافة في أي موقع للقائمة:-

وممكن إضافة نفس السلسلة الحرفية ولكن حسب موقع معين مثلاً الموقع الثالث في القائمة حينها سيتم إضافة هذه الكلمة في هذا الموقع وستزحف الكلمات من هذا الموقع حتى الأخير موقع واحد إلى الأسفل:-

```
List1.AddItem "Judy Phelps",2
```

وممكن أن يكون الموقع مخزون في متغير، أو أن يكون مخزون في صندوق نص:-

```
List1.AddItem Text1.Text, Val(Text2.Text)
```

RemoveItem: وهو الإجراء المستخدم لحذف كلمة من القائمة وبالصيغة العامة التالية:-

```
object.RemoveItem index
```

حيث نعرف موقع الكلمة المراد حذفها.

ممكن أن نعرف الموقع مباشرة:- `List1.RemoveItem 2`

أو قد يكون مخزون في متغير:- `List1.RemoveItem x`

أو قد يكون مخزون في صندوق نص:- `List1.RemoveItem Val(Text1.Text)`

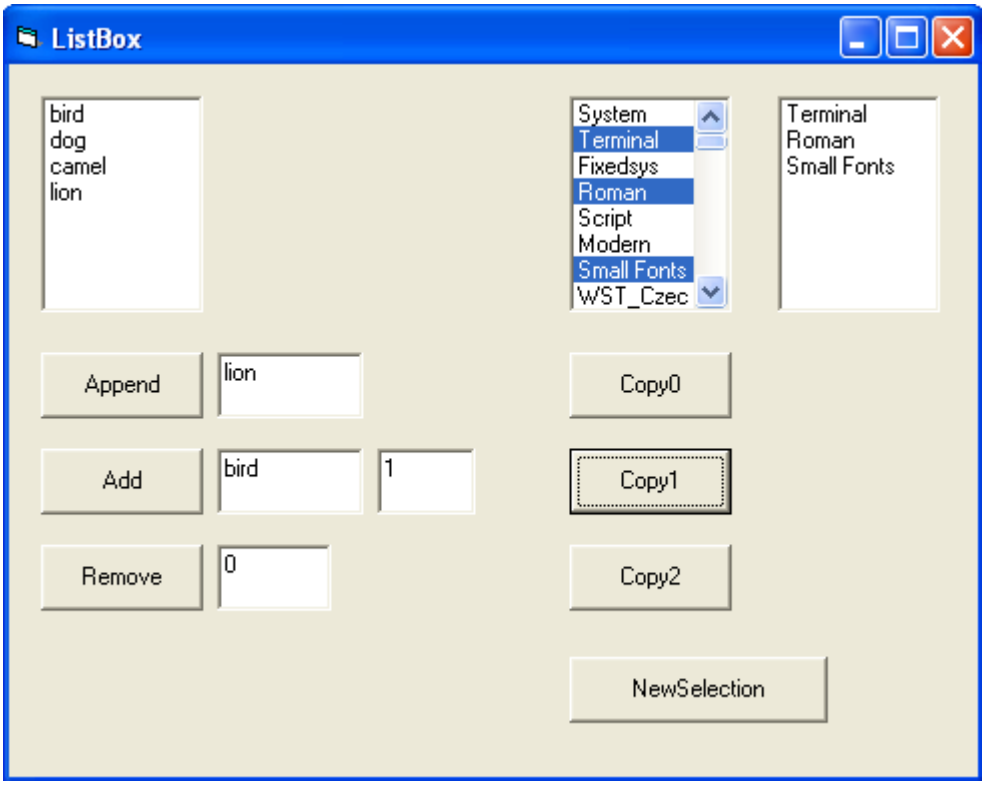
أو قد يكون موقع الكلمة التي تم اختيارها من القائمة:- `List1.RemoveItem List1.ListIndex`

أو بأي اسلوب آخر.

Clear: ويستخدم هذا الإجراء لتنظيف القائمة أي مسح كل الكلمات المخزونة فيها.

```
List1.Clear
```

مثال توضيحي لخصائص وإجراءات القائمة، حيث إن القائمة الأولى سيتم فيها فحص إجراءات الإضافة والحذف. أما القائمة الثانية سيتم فيها دراسة خصائص الاختيار وسيتم على ضوئها استنساخ الكلمات المختارة إلى القائمة الثالثة، حيث تم تخصيص زر أوامر لعملية الاستنساخ لكل حالة من حالات الخاصية MultiSelect التي تحدد في وقت التصميم ومن ثم يتم اختيار الكلمات من القائمة الثانية ثم اختيار الزر المناسب لعملية الاستنساخ:-



Project properties

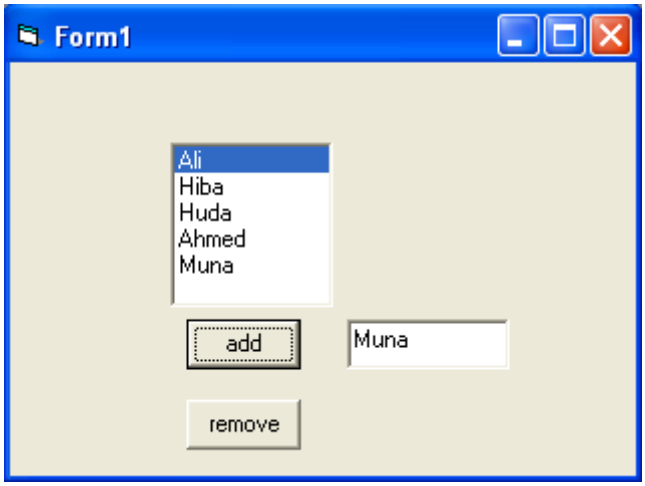
Object	Property	Setting
Form1	Name Caption	Form1 ListBox
ListBox1	Name List Sorted ItemData	List1 Cat ' ممن أن توضع في حدث تحميل النموذج Dog Camel False ' True دراسة 546 0 234
ListBox2	Name Column MultiSelect Style	List2 0 ' 2 دراسة 0 ' 2 ،1 فحص 0 ' 0 = MultiSelect فحص عندما يكون
ListBox3	Name	List3
CommandButton1	Name Caption	Command1 Append
TextBox1	Name Text	Text1 فراغ
CommandButton2	Name Caption	Command2 Add
TextBox2	Name Text	Text2 فراغ
TextBox3	Name Text	Text3 فراغ
CommandButton3	Name Caption	Command3 Remove
TextBox4	Name Text	Text4 فراغ
CommandButton4	Name Caption	Command4 Copy0
CommandButton5	Name Caption	Command5 Copy1
CommandButton6	Name Caption	Command6 Copy2
CommandButton7	Name Caption	Command7 New Selection

Poject code

```
Private Sub Command1_Click()
 'List1.AddItem "lion"   'إضافة عنصر إلى نهاية القائمة
 If Text1.Text <> "" Then
  List1.AddItem Text1.Text
 ' List1.ItemData(List1.NewIndex) = 564
 End If
End Sub
Private Sub Command2_Click()
' List1.AddItem "bird", 1          'إضافة عنصر إلى موقع معين في القائمة
 If (Text2.Text <> "") And (Val(Text3.Text) >= 0) And (Val(Text3.Text) <= List1.ListCount) Then
   List1.AddItem Text2.Text, Val(Text3.Text)
   'List1.ItemData(List1.NewIndex) = 738
 End If
End Sub

Private Sub Command3_Click()
 'list1.RemoveItem 2        'حذف عنصر من القائمة
If (Val(Text4.Text) >= 0) And (Val(Text4.Text) < List1.ListCount) Then
 List1.RemoveItem Val(Text4.Text)
End If
End Sub

Private Sub Command4_Click()
 List3.Clear          ' صفر=MultiSelect استنساخ عنصر واحد من قائمة إلىقائمة أخرى عندما
 List3.AddItem List2.Text
 ' list3.AddItem list2.List (list2.ListIndex )
 Command4.FontName = List2.Text ' ممكن الاستفادة من الاختيار لأعمال أخرى
End Sub

Private Sub Command5_Click()
 List3.Clear          " 1=MultiSelect استنساخ عناصر متفرقة من قائمة إلىقائمة أخرى عندما
 For i = 0 To List2.ListCount - 1
  If List2.Selected(i) Then
   List3.AddItem List2.List(i)
  End If
  Next i
End Sub
```

```
Private Sub Command6_Click()
  List3.Clear          " 2=MultiSelect استنساخ عناصر متسلسلة من قائمة إلى قائمة أخرى عندما
 For i = 0 To List2.ListCount - 1
  If List2.Selected(i) Then
   List3.AddItem List2.List(i)
  End If
  Next i
End Sub


Private Sub Command7_Click()
 For i = 0 To List2.ListCount – 1   'لرفع الاختيار عن العناصر التي كانت مختارة سابقا والتهيأة لاختيار جديد
  If List2.Selected(i) Then List2.Selected(i) = False
 Next i
End Sub

Private Sub Form_Load()
 Dim a                 ' ممكن إضافة الأسماء هنا في وقت التنفيذ
 a = Array("cat", "dog", "camel")
 List1.Clear
 For i = 0 To 2
  List1.AddItem a(i)
 Next i

 For i = 0 To Screen.FontCount – 1  'إضافة أسماء الخطوط إلى القائمة الثانية
  List2.AddItem Screen.Fonts(i)
 Next i
End Sub
```

س) إضافة أسماء طلبة إلى قائمة ListBox باستخدام صندوق النص وزر أوامر للإضافة، واستخدام زر أوامر آخر لحذف الاسم الذي تم اختياره



Project properties

Object	Property	Setting
Form1	Name	Form1
CommandButton1	Name Caption	Command11 Add
CommandButton2	Name Caption	Command12 Remove
TextBox1	Name Text	Text1 فراغ
ListBox1	Name List	List1 Ali Muna Huda

Project code

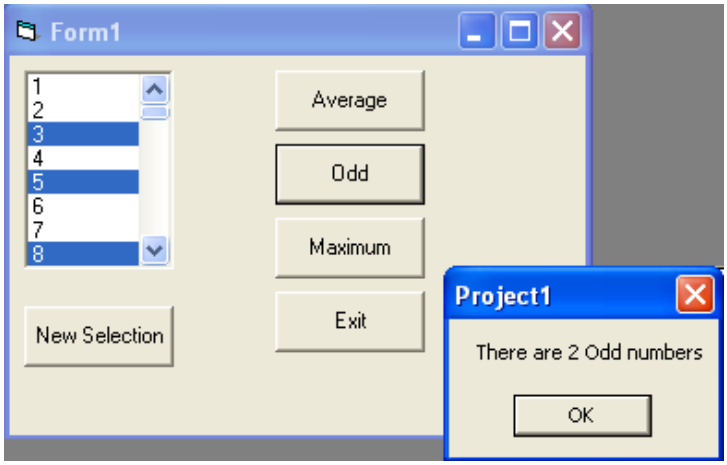
```
Private Sub Form_Load()
Dim x   ' ممكن خزن الاسماء هنا في القائمة
x = Array("Ali", "Muna", "Huda")
List1.Clear
For i = 0 To 2
 List1.AddItem x(i)
Next i
 List1.ListIndex = 0 ' ممكن وضعها
End Sub

Private Sub Command1_Click()
If Text1.Text <> "" Then
 List1.AddItem Text1.Text
 If List1.ListCount = 1 Then List1.ListIndex = 0    ' في حالة وضعها في حدث تحميل النموذج
End If
End Sub

Private Sub Command2_Click()
                أو ممكن إعطاء رقم العنصر المراد حذفه وفحص الرقم إذا كان أقل من List1.ListCount فيتم الحذف
'If List1.ListIndex <> -1 Then List1.RemoveItem List1.ListIndex
 If List1.ListIndex = -1 Then
  MsgBox "No item selected or empty list", vbCritical, "Error"
 Else
  List1.RemoveItem List1.ListIndex
  If List1.ListCount > 0 Then List1.ListIndex = 0   ' في حالة وضعها في حدث تحميل النموذج
 End If
End Sub
```

س) صمم نافذة يمكن من خلالها اختيار مجموعة من الأرقام الصحيحة (1-99) من خلال قائمة ListBox بحيث تعطي النافذة إمكانية حساب الآتي باستخدام أزرار أوامر Command Buttons:-

1- معدل الأرقام

2- أكبر رقم

3- عدد الأرقام الفردية



Project properties

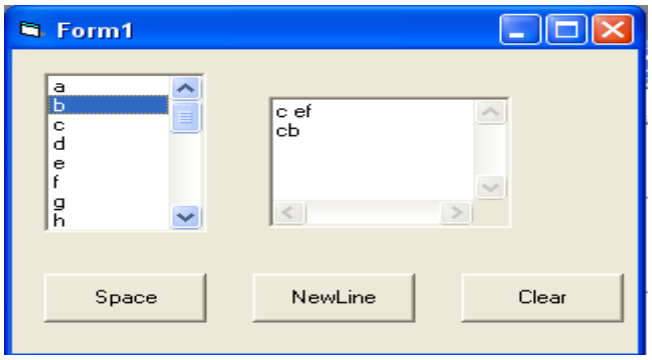
Object	Property	Setting
Form1	Name	Form1
CommandButton1	Name Caption	cmdave Average
CommandButton2	Name Caption	cmdodd Odd
CommandButton3	Name Caption	cmdmax Maximum
CommandButton4	Name Caption	cmdend End
CommandButton5	Name Caption	cmdnew New Selection
ListBox1	Name MultiSelect	List1 1 'simple

Project code

```
Private Sub Form_Load()
 For i = 1 To 99
  List1.AddItem i
 Next i
```

```
End Sub

Private Sub cmdave_Click()
 Dim sum As Single, n As Integer
 For i = 0 To List1.ListCount - 1
  If List1.Selected(i) Then
   sum = sum + List1.List(i)
   n = n + 1
  End If
 Next i
 MsgBox "The average is " & sum / n
End Sub

Private Sub cmdend_Click()
 End
End Sub

Private Sub cmdmax_Click()
 Dim m As Integer
 m = List1.Text
 For i = 0 To List1.ListCount - 1
  If List1.Selected(i) Then
   If List1.List(i) > m Then m = List1.List(i)
  End If
 Next i
 MsgBox "The maximum number is " & m

End Sub

Private Sub cmdnew_Click()
 For i = 0 To List1.ListCount - 1
  List1.Selected(i) = False
 Next i
End Sub

Private Sub cmdodd_Click()
Dim n As Integer
 For i = 0 To List1.ListCount - 1
  If List1.Selected(i) Then
   If List1.List(i) Mod 2 <> 0 Then n = n + 1
  End If
 Next i
 MsgBox "There are " & n & " Odd numbers"

End Sub
```

س) صمم نافذة مشروع يمكن من خلالها كتابة أي نص في صندوق نص TextBox دون استخدام لوحة المفاتيح Keyboard.



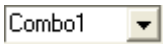
Project properties

Object	Property	Setting
Form1	Name	Form1
CommandButton1	Name Caption	Command1 Space
CommandButton2	Name Caption	Command2 New Line
CommandButton3	Name Caption	Command3 Clear
TextBox1	Name Text Multiline ScrollBars	Text1 فراغ True 3
ListBox1	Name	List1

Project code

```
Private Sub Command1_Click()
 Text1.Text = Text1.Text + " "
End Sub

Private Sub Command2_Click()
 Text1.Text = Text1.Text + vbCrLf
End Sub

Private Sub Command3_Click()
 Text1.Text = ""
End Sub

Private Sub Form_Load()
 For i = Asc("a") To Asc("z")
  List1.AddItem Chr(i)
 Next i
End Sub
```

```
Private Sub List1_Click()
 Text1.Text = Text1.Text + List1.Text
End Sub
```

صندوق السرد ComboBox Control : 

إن هذه الأداة تجمع في عملها بين أداة صندوق النص TextBox من حيث عرض وتعديل البيانات وأداة القائمة ListBox من حيث عرض البيانات في قائمة يمكن الاختيار منها.

الخصائص Important Properties :

List , ListCount, ListIndex, Sorted, ItemData, NewIndex.

وهي نفس خصائص القائمة ListBox. ولا توجد هنا خاصية MultiSelect

Style:- النمط حيث يتم من خلال هذه الخاصية تحديد شكل صندوق السرد والحوار، وتأخذ القيم التالية:

أ -السحب Drop-down Comb -0 : حيث يسمح هذا الشكل للمستخدم بإجراء عملية الاختيار Select والادخال Enter ويظهر هذا النوع بشكل TextBox وعلى يمينه سهم نحو الأسفل يقوم بسرد عناصر القائمة عند النقر عليه.

ب -النمط البسيط Simple Combo -1: حيث يسمح هذا الشكل للمستخدم بالاختيار أو إدخال البيانات، في هذا النوع يختفي سهم السرد لأن عناصر القائمة تكون ظاهرة ويمكن الاختيار أو التغيير فيها. عند استخدام هذا النوع من النمط يجب تكبير حجم أداة السرد والحوار عند التصميم لإظهار العناصر جميعها.

ج قائمة السحب Dropdown List -2:حيث يسمح هذا الشكل للمستخدم بالاختيار فقط من خلال القائمة. فهو يشبه في اسلوب عمله أداة صندوق القائمة ListBox.

Style	Value	Description
vbComboDropDown	0	(Default) Dropdown Combo. Includes a drop-down list and a text box. The user can select from the list or type in the text box.
vbComboSimple	1	Simple Combo. Includes a text box and a list, which doesn't drop down. The user can select from the list or type in the text box. The size of a Simple combo box includes both the edit and list portions. By default, a Simple combo box is sized so that none of the list is displayed. Increase the Height property to display more of the list.
vbComboDrop-DownList	2	Dropdown List. This style allows selection only from the drop-down list.

0- Drop-down Comb

1- Simple Combo

2- Dropdown List

Text: النص الذي يتم اختياره (حالة Style=2). أو إدخاله أو اختياره في صندوق النص (حالة Style=صفر أو 1) والذي يظهر في صندوق النص.

ممكن إضافة كلمة يتم إدخالها في صندوق نص هذه الإداة إلى قائمة الأداة لتصبح من ضمن عناصره، وذلك بإدخال الكلمة في الصندوق ومن ثم اختيار زر أوامر يقوم بعملية الإضافة التالية:-

Combo1.Additem Combo1.Text

في حالة اختيار كلمة موجودة أصلا في قائمة صندوق السرد فإن هذه الكلمة ستمثل Combo1.Text أو Combo1.List(Combo1.ListIndex)

Locked, SelStart, Sellength, SelText properties. **الخصائص**

نفس خصائص صندوق النص.

الأحداث Events:

Click, DblClick, MouseDown, MouseUp, MouseMove, DragDrop, KeyDown, KeyUp, KeyPress, GetFocus, LostFocus and Scroll events are similar to ListBox's events. **ListBox**.نفس أحداث القائمة

ان الأحداث الثلاثة التالية يتوقف استخدامها على نوع النمط المستخدم:-

Change: يعمل مع النمط 0- Drop-down Combo والنمط 1- Simple Combo

DropDown: يعمل مع النمط 0- Drop-down Combo والنمط 2- Dropdown List

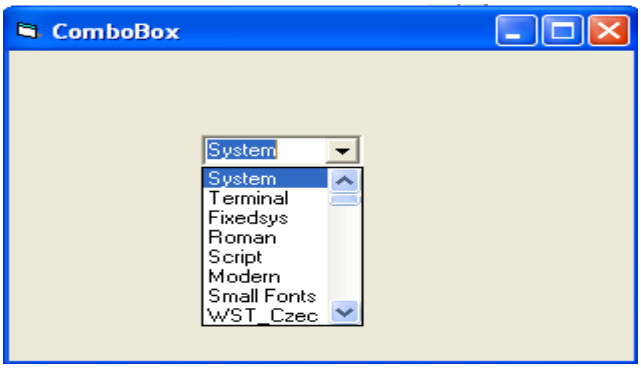
Click: يعمل مع جميع الأنماط

الإجراءات Methods:

Move, Refresh, SetFocus, Additem, RemoveItem, Clear and Zorder Methods

نفس إجراءات القائمة .

مثال عن إضافة أسماء الخطوط إلى صندوق السرد ComboBox



Project properties

Object	Property	Setting
Form1	Name Caption	Form1 ComboBox
ComboBox1	Name Style	Combo1 2 ' 0,1 فحص

Poject code

```
Private Sub Form_Load()
 For i = 0 To Screen.FontCount - 1
  Combo1.AddItem Screen.Fonts(i)
 Next i
 Combo1.ListIndex = 0
End Sub
```

معلومة مفيدة

قد نحتاج أحياناً لعرض الصور بصورة عشوائية. عندها بإلامكان الإستفادة من القائمة ListBox لعمل ذلك ولمنع التكرار كما في المثال التالي.

مثال :- لاختيار رقم عشوائي محصور بين (0..9)، هذه الأرقام قد تمثل مواقع أسماء مخزونة في مصفوفة أو قد تمثل مواقع لصور مخزونة في مصفوفة من الصور أو في أداة تسمى قائمة صور ImageList.

Project properties

Object	Property	Setting
Form1	Name Caption	Form1 Random
CommandButton1	Name Caption	Command1 New Random No.
Label1	Name Caption BorderStyle BackColor BackStyle Autosize	Label1 يترك فارغ 1 أي لون 1 True
ListBox1	Name Visible	List1 False

Poject code

```
Option Explicit
Dim r
Private Sub Command1_Click()
Dim d
If r >= 0 Then
 d = Rnd * r
 Label1.Caption = "No.=" + Str(List1.List(d))
  ' List1.List(d) هذا الرقم قد يمثل موقع لصورة ضمن مصفوفة صور أو كلمات أو أرقام .
  ' ممكن أن تخزن كل المواقع في قائمة واستغلالها لاحقاً
 List1.RemoveItem d
 r = r - 1
Else
 Label1.Caption = "End of Game"
 Command1.Enabled = False
End If
End Sub
Private Sub Form_Load()
Dim i
 Randomize
 r = 9
 For i = 0 To r
  List1.AddItem i
 Next i
End Sub
```

Timer Control الميقاتي

وهي اداة غير مرئية تعمل مثل عمل الساعة، حيث بعد انتهاء الفترة المحددة لها في خاصية Interval يعمل الحدث Timer وهو حدث نظام System event. يستخدم لعمل تأخير delay أو يستفاد منه كعداد للوقت أو لعمل صورة متحركة مثلا animation.

Important Properties الخصائص المهمة

Interval: الفترة بالملي ثانية

object.Interval [= milliseconds]

The settings for milliseconds are:

Setting	Description
0	(Default) Disables a Timer control.
1 to 65,535	Sets an interval (in milliseconds) that takes effect when a Timer control's Enabled property is set to True. For example, a value of 10,000 milliseconds equals 10 seconds. The maximum, 65,535 milliseconds, is equivalent to just over 1 minute

Enabled: السماح بعمل الميقاتي

object.Enabled [= boolean]

The settings for boolean are:

Setting	Description
True	(Default) Allows object to respond to events.
False	Prevents object from responding to events.

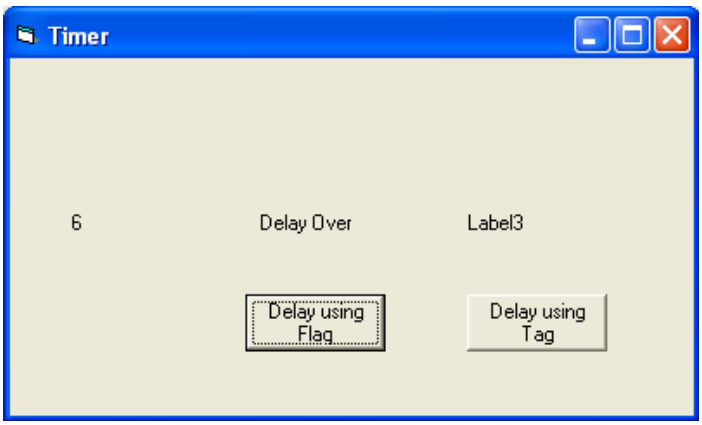
حيث بالامكان التحكم في بدء عمل الميقاتي. فاذا كان Interval=0 فانه لا يعمل وكذلك اذا كانت Enabled=False فانه لا يعمل.

Events الأحداث

Timer: يعمل هذا الحدث النظامي System event عند انقضاء الوقت المحدد له في خاصية Interval عندما تكون خاصية Enabled=True.

مثال:

- استخدام الميقاتي الأول كعداد للثواني
- استخدام الميقاتي الثاني لعمل تأخير واستخدام متغير عام كعلم Flag
- استخدام الميقاتي الثالث لعمل تأخير واستخدام الخاصية Tag كعلم Flag



Project properties

Object	Property	Setting
Form1	Name Caption	Form1 Timer
Timer1	Name Enabled Interval	Timer1 True ليعمل مباشرة' 1000 الفترة بالملي ثانية وتقابل ثانية '
Timer2	Name Enabled Interval	Timer2 False ليعمل عند الحاجة' 2000 لعمل تأخير ثانيتين '
Timer3	Name Enabled Interval	Timer3 False 2000
CommandButton1	Name Caption	Command1 Delay using Flag
CommandButton2	Name Caption	Command2 Delay using tag
Label1	Name Caption AutoSize	Label1 Label1 True
Label2	Name Caption AutoSize	Label2 Label2 True
Label3	Name Caption AutoSize	Label3 Label3 True

Poject code

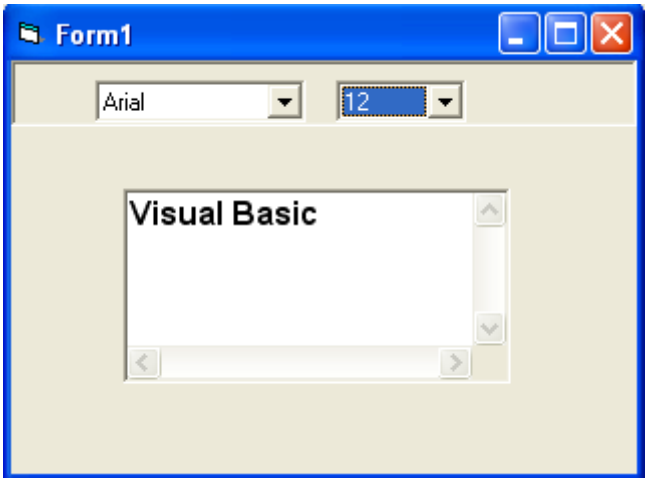
```
Option Explicit                    تعريف متغير عام في بداية البرنامج وقبل كل الأحداث'
Dim f

Private Sub Command1_Click()
 f = 0                             اخفاض العلم'
 Timer2.Enabled = True             بدء عمل هذا الميقاتي '
```

استمر في الـــ Loop لانتظار انقضاء التأخير مع تنفيذ الأحداث الأخرى ان حصلت والخروج عندما يصبح العلم =1 أي مرفوعا من قبل الحدث Timer الخاص بهذا الميقاتي دلالة انقضاء فترة الانتظار

```
 Do While DoEvents()
  If f = 1 Then Exit Do
 Loop
 Label2.Caption = "Delay Over"
 Timer2.Enabled = False            ايقاف عمل هذا الميقاتي لعدم الحاجة له بعد انقضاء التأخير'
End Sub

Private Sub Command2_Click()
 Timer3.Tag = 0
 Timer3.Enabled = True
 Do While DoEvents()
  If Timer3.Tag = 1 Then Exit Do
 Loop
 Label3.Caption = "Delay Over"
 Timer3.Enabled = False
End Sub

Private Sub Timer1_Timer()
 Static i          للحفاظ على قيمة المتغير عند الخروج ومن ثم الدخول الى هذا البرنامج الفرعي ويبدأ بصفر '
  Label1.Caption = Str(i)          طباعة العداد '
  i = i + 1                        زيادة العداد '
End Sub

Private Sub Timer2_Timer()
 f = 1                             رفع المتغير العام كعلم دلالة على انقضاء التأخير '
End Sub
```

```
Private Sub Timer3_Timer()
 Timer3.Tag = 1       ' رفع خاصية Tag كعلم دلالة على انقضاء التأخير
End Sub
```

س) قم بعمل شريط أدوات يتضمن صندوقي سرد combobox 2، يضم الأول أسماء الخطوط ويضم الثاني حجم الخط، وعند تغيير أي منها يظهر تأثير ذلك على حجم النص المكتوب في صندوق النصوص.



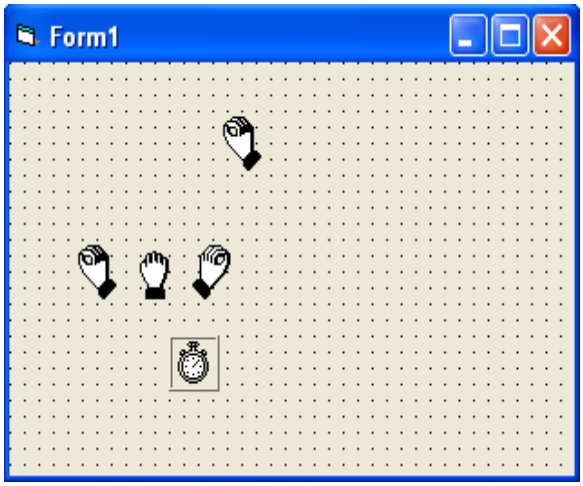
Project properties

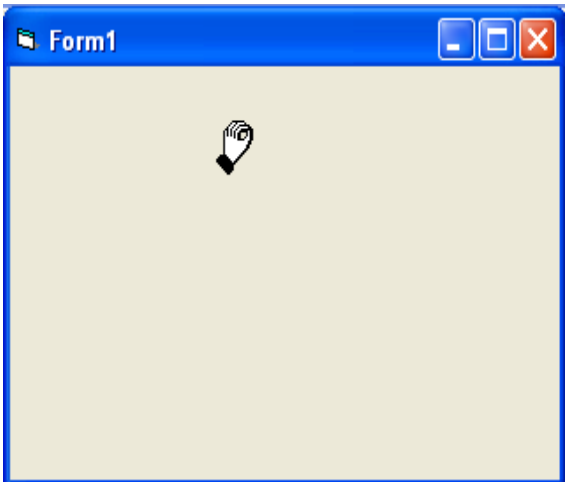
Object	Property	Setting
Form1	Name	Form1
PictureBox1	Name Align	Picture1 1 'top
TextBox1	Name Text MultiLine ScrollBars	Text1 فراغ True 3 'Both
ComboBox1	Name Style	Combo1 2
ComboBox2	Name Style	Combo2 2

Project code

```
Private Sub Combo1_Click()
 Text1.FontName = Combo1.Text
End Sub


Private Sub Combo2_Click()
 Text1.FontSize = Combo2.Text
End Sub

Private Sub Form_Load()
For i = 0 To Screen.FontCount - 1
   Combo1.AddItem Screen.Fonts(i)
 Next i
  Combo1.ListIndex = 0
  Text1.FontName = Combo1.Text


 For i = 8 To 48 Step 2
   Combo2.AddItem i
 Next i
  Combo2.ListIndex = 0
  Text1.FontSize = Combo2.Text

End Sub
```

س) اعمل صورة متحركة باستخدام الميقاتي Timer





قبل التنفيذ بعد التنفيذ

Project properties

Object	Property	Setting
Form1	Name	Form1
Timer1	Name Enabled Interval	Timer1 True 2000
Image1(0)	Name Index visible Picture	Image1 0 False صورة
Image1(1)	Name Index visible Picture	Image1 1 False صورة
Image1(2)	Name Index visible Picture	Image1 2 False صورة
Image2	Name Visible Stretch Picture	Image2 True True Image1(0) نفس صورة

Poject code

```
Private Sub Timer1_Timer()
Static i
 Image2.Picture = Image1(i).Picture
 i = (i + 1) Mod 3
End Sub
```

ممكن حل السؤال بطريقة أخرى إذا كانت الصورة مليئة بالألوان أي لها خلفية ملونة.

Project properties

Object	Property	Setting
Form1	Name	Form1
Timer1	Name Enabled Interval	Timer1 True 2000
Image1(0)	Name Index Visible Stretch Picture	Image1 0 True True صورة
Image1(1)	Name Index Visible Stretch Picture	Image1 1 True True صورة
Image1(2)	Name Index Visible Stretch Picture	Image1 2 True True صورة

ملاحظة:- هنا تكون الصور مرئية ومرتبة واحدة فوق الأخرى ولجميعها نفس الموقع والأبعاد (Left, Top, Width, Height) على أن تكون Image1(0) للأمام أولا.

Poject code

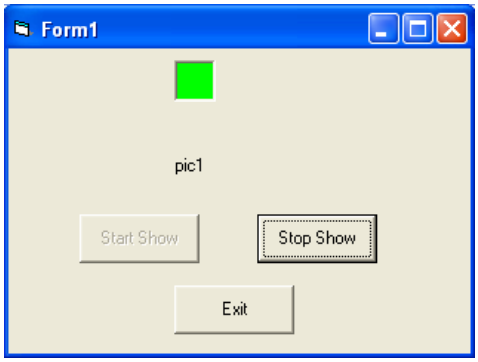
```
Private Sub Timer1_Timer()
Static i
 Image1(i).ZOrder 0
 i = (i + 1) Mod 3
End Sub
```

مثال:- صمم نافذة المشروع التالي والتي من خلالها

1 -عند الضغط على الأمر Start Show يتم إظهار ثلاث صور Pic1.bmp..Pic3.bmp موجودة داخل المجلد c:\pictures بصورة تلقائية وبفارق زمني مقداره ثانية بين صورة والتي تليها.

2 -عند الضغط على الأمر Stop Show فان عرض الصور يتوقف عند الصورة الحالية.

3 -عند الضغط على الأمر Exit ينتهي البرنامج.



Project properties

Object	Property	Setting
Form1	Name	Form1
Timer1	Name Enabled Interval	Timer1 False 1000
PictureBox1	Name AutoSize	Picture1 True
Label1	Name Caption AutoSize	Label1 فراغ True
CommandButton1	Name Caption	Command1 Start Show
CommandButton2	Name Caption	Command2 Stop Show
CommandButton3	Name Caption	Command3 Exit

Poject code

```
Option Explicit
Dim a
Private Sub Command1_Click()
 Timer1.Enabled = True
 Command1.Enabled = False
 Command2.Enabled = True
End Sub

Private Sub Command2_Click()
 Timer1.Enabled = False
 Command1.Enabled = True
 Command2.Enabled = False
End Sub

Private Sub Command3_Click()
 End
End Sub

Private Sub Form_Load()
 a = 1
End Sub

Private Sub Timer1_Timer()
  Label1.Caption = "pic" & a
  Picture1.Picture = LoadPicture("c:\pictures\pic" & Format(a) & ".bmp")
  a = a + 1
  If a = 4 Then
   Timer1.Enabled = False
   Command1.Enabled = True
   Command2.Enabled = False
   a = 1
  End If
End Sub
```

الملفات

هناك حاجة ماسة لاستخدام الملفات لحفظ النتائج والمعلومات من الضياع ولاسترجاعها فيما بعد. لذلك يجب ان نتعلم كيفية جلب اسم ملف ومن ثم قرائته، أو اعطاء اسم ملف لحفظ المعلومات أو لتعديلها فيه. كما قد نحتاج الى عرض الصور عن طريق معرفة اسم الملف المخزونة فيه.

هناك طريقتان لجلب اسم ملف:-

1 - باستخدام ثلاث أدوات:

DriveListBox Control, DirListBox Control, and FileListBox Control

2- باستخدام الأداة CommonDialog Control وهي الأفضل

الطريقة الاولى

قائمة محرك الأقراص DriveListBox Control

ويستخدم لجلب اسم محرك الأقراص disk drive اثناء التنفيذ فقط.

الخصائص المهمة Important Properties

Drive: استرجاع أو تحديد اسم محرك الأقراص اثناء التنفيذ object.Drive [= drive]

ممكن ان يكون محرك الأقراص واحد من الأنواع التالية

·Floppy disks — "a:" or "b:", and so on
·Fixed media — "c: [volume id]"
·Network connections — "x: \\server\share"

ملاحظة: ممكن ملاحظة تغيير محرك الأقراص عن طريق الحدث Change الخاص بهذه الأداة

قائمة المجلدات DirListBox Control

تستخدم لعرض المجلدات اثناء وقت التنفيذ. حيث تعرض على شكل شجرة مجلدات.

الخصائص المهمة Important Properties

Path: يستخدم لجلب أو تحديد المسار. object.Path [= pathname]

ملاحظة: ممكن ملاحظة تغيير المسار عن طريق الحدث Change الخاص بهذه الأداة

قائمة الملفات FileListBox Control

وهي قائمة تعرض اسماء الملفات التابعة للمسار المحدد في الخاصية Path.

الخصائص المهمة Important Properties

Path: يستخدم لجلب أو تحديد المسار object.Path [= pathname]

FileName: اسم الملف الذي تم اختياره أو تحديده. object.FileName [= pathname]

Pattern: ممكن ان نحدد امتداد الملفات التي نريد عرض اسماؤها. object.Pattern [= value]

مثلا نضع في الحدث Form_Load:

```
File1.Pattern="*.txt"
File1.Pattern="*.com;*.bat"                    أو
```

الخصائص **Archive, Hidden, Normal, System** : ممكن ان نحدد انواع الملفات التي نريد عرض اسماؤها كملفات النظام والملفات العادية والملفات المخفية والملفات الارشيفية.

```
object.Archive [= boolean]
object.Hidden [= boolean]
object.Normal [= boolean]
object.System [= boolean]
```

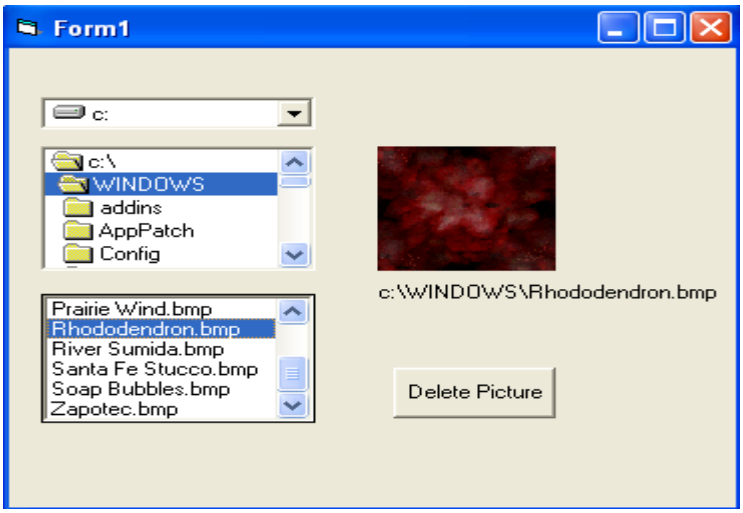
مثلا نضع في الحدث Form_Load:

```
File1.Hidden=True          ' لعرض اسماء الملفات المخفية
```

ملاحظة: ممكن ملاحظة اختيار اسم الملف عن طريق الحدث Click الخاص بهذه الأداة

ملاحظة: بالامكان الربط بين الأدوات الثلاثة المذكورة سلفا عن طريق كتابة برنامج يشمل هذه الأدوات والسيطرة على احداثها.

مثال: قراءة اسم ملف صورة باستخدام ثلاث أدوات ثم عرض تلك الصورة، وحذفها عند الرغبة.



Project properties

Object	Property	Setting
Form1	Name	Form1
Label1	Name Caption AutoSize BackStyle	Label1 فراغ True يصبح الشريط بحجم الكلمة 0 شفاف يعكس لون الخلفية ورائه
DriveListBox1	Name	Drive1
DirListBox1	Name	Dir1
FileListBox1	Name	File1
Image1	Name Stretch	Image1 True ' لتأخذ الصورة حجم الاطار الذي يحويها
CommandButton1	Name Caption	Command1 Delete Picture

Poject code

```
Private Sub Command1_Click()
 Image1.Picture = LoadPicture()
 Label1.Caption = ""
End Sub

Private Sub Dir1_Change()
 File1.Path = Dir1.Path          'set file path
End Sub

Private Sub Drive1_Change()
 Dir1.Path = Drive1.Drive        'set directory path
End Sub
```

```
Private Sub File1_Click()
' بالامكان عرض اسم الملف و قرائته أو عرض الصورة حسب المطلوب
'  للحصول على اسم الملف كاملا مع مساره ومن ثم طباعة الاسم كاملا وعرض الصورة المحتواة فيه
  Path = File1.Path
  If Right$(Path, 1) <> "\" Then Path = Path + "\"          'لفحص الحرف الأول على يمين المسار
  Label1.Caption = Path + File1.filename
  Image1.Picture = LoadPicture(Label1.Caption)
 'A=Path  + File1.filename  أو تكتب مباشرة
 'Image1.Picture = LoadPicture(A)
End Sub

Private Sub Form_Load()
 File1.Pattern = "*.bmp"
End Sub
```

الطريقة الثانية:

ممكن ان نجلب اسم ملف باستخدام أداة واحدة تدعى CommonDialog الموجودة في الملف Microsoft common dialog 5.0 حيث نصل لهذا الملف أما باختيار Components من القائمة Project أو باستخدام Ctrl+T أو بالنقر بالزر الأيمن للماوس على المنطقة الفارغة في صندوق الأدوات ToolBox ومن ثم اختيار Components. في كل هذه الحالات تظهر مجموعة من الملفات كل منها يحتوي على أداة اضافية أو اكثر لا تقل أهمية عن الأدوات التي تم شرحها فنجلبها عند الحاجة.

CommonDialog Control

هذه الأداة لها اجراءات كثيرة مثل فتح الملف ShowOpen لجلب اسم ملف للقراءة، ShowSave جلب اسم ملف للخزن، ShowColor لجلب اللون، ShowFont لجلب الخط، ShowPrinter لعمل الاختيارات الخاصة بالطباعة، ShowHelp.

Method	Dialog Displayed
ShowOpen	Show Open Dialog Box
ShowSave	Show Save As Dialog Box
ShowColor	Show Color Dialog Box
ShowFont	Show Font Dialog Box
ShowPrinter	Show Print or Print Options Dialog Box
ShowHelp	Invokes the Windows Help Engine

الخصائص المهمة Important Properties

CancelError: خاصية مهمة لتسمح بفحص الخطأ عندما تكون هذه الخاصية True.

object.CancelError [= boolean]
The settings for boolean are:

Setting	Description
True	An error is generated.
False	(Default) No error is generated.

DialogTitle: عنوان للشاشة التي ستظهر.

Filter: تحديد امتداد الملفات التي سيتم عرضها

object.Filter [= description1 |filter1 |description2 |filter2...]
CommonDialog1.Filter="Text (*.txt)|*.txt|Pictures (*.bmp;*.ico)|*.bmp;*.ico"

FilterIndex: لتحديد امتداد الملفات من بين مجموعة من الامتدادات معطاة في الخاصية Filter. أي منها يعرض أولا.

object.FilterIndex [= number]
CommonDialog1.FilterIndex=0

MaxFileSize: اقصى حجم للملف بين 1 و 32كيلو بايت، التلقائي هو 256 بايت.

FileName: اسم الملف كامل مع مساره.

FileTitle: اسم الملف بدون مسار.

Color: لتحديد أو استرجاع اللون المختار يستخدم مع الإجراء ShowColor.

FromPage, ToPage: لتحديد بدء ونهاية صفحات الطباعة يستخدم مع الإجراء ShowPrinter.

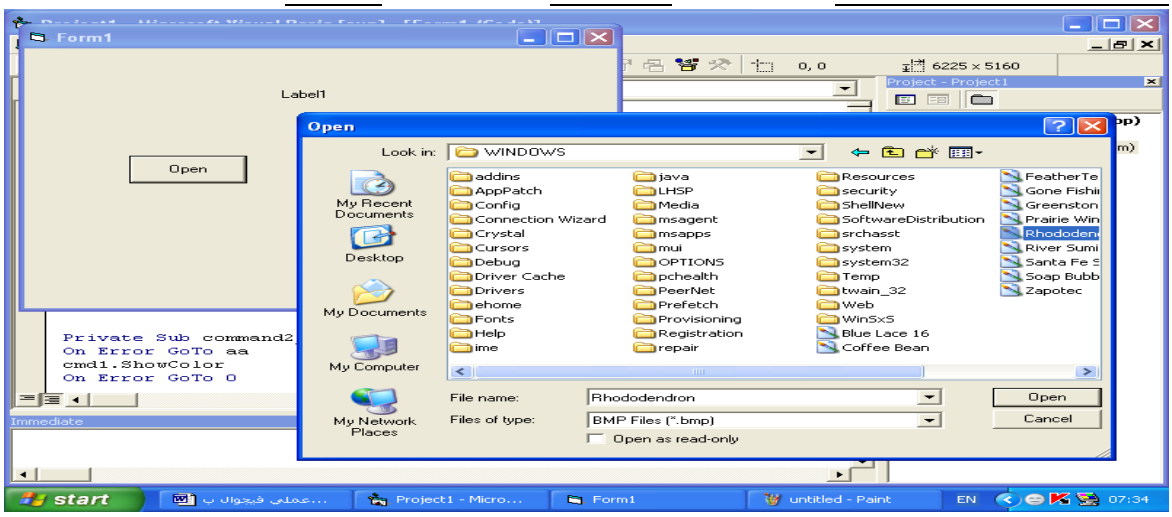
object.FromPage [= number]
object.ToPage [= number]

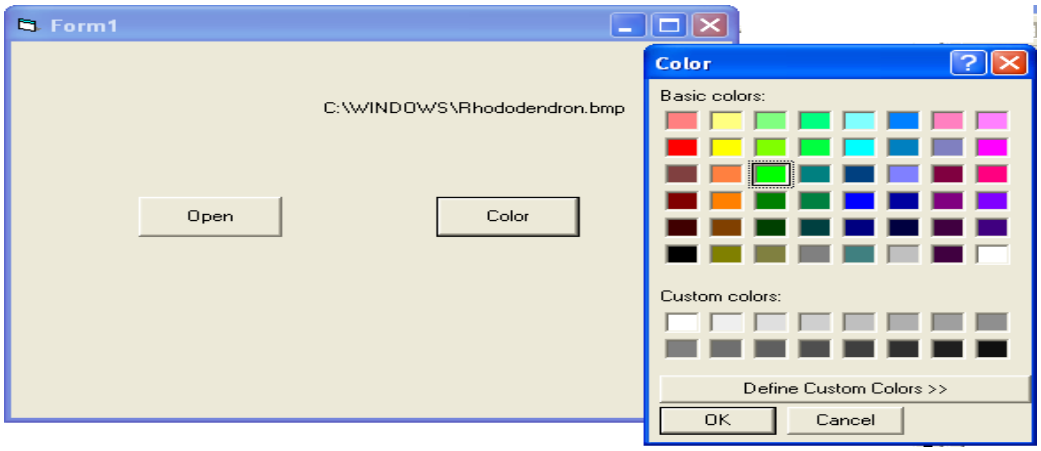
Copy: لتحديد عدد نسخ الطباعة يستخدم مع الإجراء ShowPrinter.

Max, Min: لتحديد اكبر واصغر حجم خط يستخدم مع الإجراءات ShowPrinter ، ShowFont

object.Min [= points]
object.Max [= points]
object.Min [= number]
object.Max [= number]

مثال عن استخدام الاجراء ShowOpen والاجراء ShowColor للاداة **CommonDialog**:





Project properties

Object	Property	Setting
Form1	Name	Form1
Label1	Name Caption AutoSize BackStyle	Label1 فراغ True يصبح الشريط بحجم الكلمة 0 شفاف يعكس لون الخلفية ورائه
CommonDialog1	Name CancelError	cmd1 True
CommandButton1	Name Caption	Command1 Open
CommandButton2	Name Caption	Command2 Color
Image1	Name Stretch	Image1 True ' لتأخذ الصورة حجم الاطار الذي يحويها

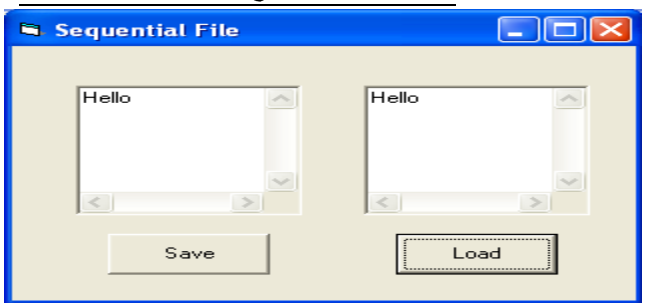
Poject code

```
Private Sub command1_click()
On Error GoTo errorf2                  ' في حالة وجود خطأ اذهب الى نهاية هذا البرنامج الفرعي
cmd1.filename = ""
cmd1.Filter = "BMP Files (*.bmp) |*.bmp"        'لتحديد ملف صورة
cmd1.FilterIndex = 0    ' Filter لتحديد نوع الامتداد (التسلسل) من بين مجموعة امتدادات معطاة في الخاصية
cmd1.ShowOpen      ' عرض الشاشة الخاصة بجلب اسم ملف للقراءة
On Error GoTo 0                ' disable the error
' dir في حالة ان المستخدم كتب اسم الملف فيجب التأكد من وجوده في هذا المسار أم لا وتستخدم لذلك الدالة
If Dir(cmd1.filename) = "" Then
' في حالة كون الملف غير موجود اصلا تظهر رسالة خطأ والخروج من هذا البرنامج الفرعي
 MsgBox "File not Exist", vbCritical + vbMsgBoxRight, "Error"
  Exit Sub
End If
' في حالة كون الملف موجود فيعرض اسمه ومن ثم يمكن قرائته أو عرض الصورة المخزونة فيه
 Label1.Caption = cmd1.filename
Image1.Picture=LoadPicture(cmd1.filename)
errorf2:
End Sub


Private Sub command2_click()
On Error GoTo aa
cmd1.ShowColor                 'لعرض شاشة الألوان
On Error GoTo 0
Form1.BackColor = cmd1.Color    ' تغيير لون خلفية النموذج باللون الذي تم اختياره
aa:
End Sub
```

انواع الملفات:-

1 - الملف التسلسلي Sequential File لحفظ واسترجاع النصوص
2 - الملف الثنائي Binary File
3 - الملف العشوائي Random File

مثال عن الملف التسلسلي Sequential File، **خزن واسترجاع النص المخزون فيه:**



Project properties

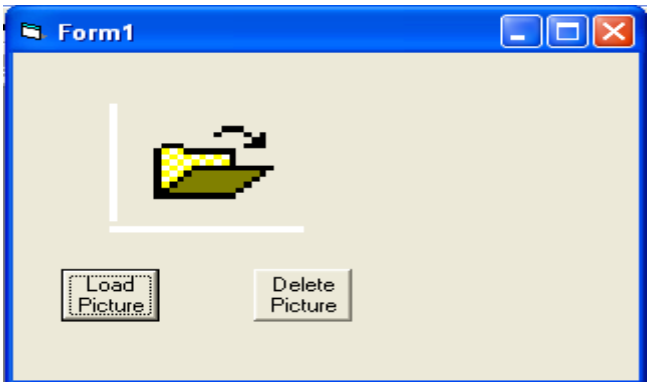
Object	Property	Setting
Form1	Name Caption	Form1 Sequential File
Text Box1	Name Text MultiLine ScrollBars	Text1 فراغ True 3 'Both
Text Box2	Name Text MultiLine ScrollBars	Text2 فراغ True 3 'Both
CommandButton1	Name Caption	Command1 Save
CommandButton2	Name Caption	Command2 Load

Poject code

```
Private Sub Command1_Click()
 filenum = FreeFile                    ' اعطاء رقم للملف
 Open "example.txt" For Output As filenum ' فتح الملف للخزن "example.txt" واعطائه الرقم أعلاه
 Print #filenum, Text1.Text            ' خزن النص في الملف
 Close filenum                         'غلق الملف
End Sub
Private Sub Command2_Click()
 filenum = FreeFile
 Open "example.txt" For Input As filenum ' فتح الملف للقراءة "example.txt" واعطائه الرقم أعلاه
 Text2.Text = Input$(LOF(filenum) - 2, filenum)  ' قراءة عدد معين من الأحرف، هنا على طول الملف
```

```
 Close filenum
End Sub
```

س) قراءة اسم ملف صورة باستخدام CommonDialog ثم عرض تلك الصورة، وحذفها عند الرغبة.



Project properties

Object	Property	Setting
Form1	Name	Form1
CommonDialog1	Name CancelError	Cmd1 True
Image1	Name Stretch Picture	Image1 True فراغ
CommandButton1	Name Caption	Command1 Load Picture
CommandButton2	Name Caption	Command2 Delete Picture

Poject code

```
Private Sub Command1_Click()
On Error GoTo errof2
cmd1.filename = ""
cmd1.Filter = "Bmp Files (*.bmp)|*.bmp"
cmd1.FilterIndex = 0
cmd1.ShowOpen
On Error GoTo 0

 If Dir(cmd1.filename) = "" Then
  MsgBox "File not exist", vbCritical, "Error"
  Exit Sub
 End If
 Image1.Picture = LoadPicture(cmd1.filename)
errof2:
End Sub
```

```
Private Sub Command2_Click()
 Image1.Picture = LoadPicture()
End Sub
```

س) اخزن سلسلة من الحروف في ملف تسلسلي يتم تحديد اسمه باستخدام CommonDialog . ثم قراءة اسمه باستخدام CommonDialog وعرض محتواه.

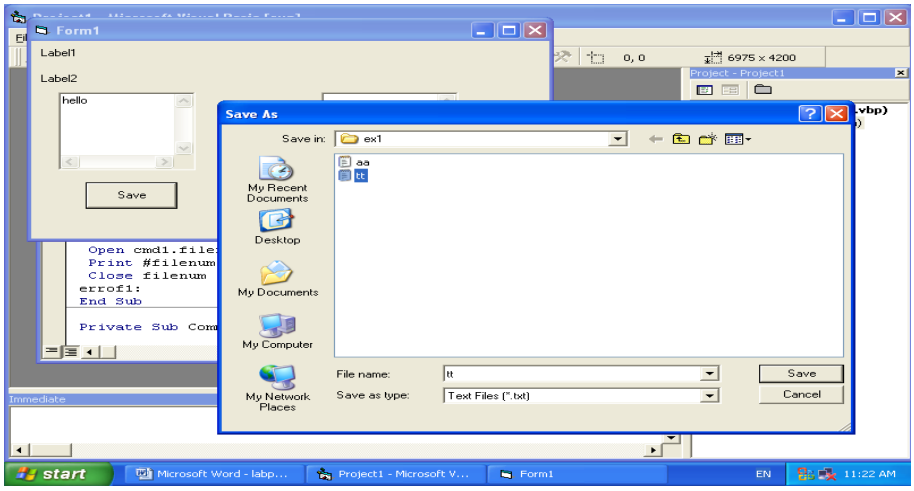
Project properties

Object	Property	Setting
Form1	Name	Form1
CommonDialog1	Name CancelError	Cmd1 True
TextBox1	Name Text MultiLine ScrollBars	Text1 فراغ True 3 ' Both
TextBox2	Name Text MultiLine ScrollBars	Text2 فراغ True 3 ' Both
CommandButton1	Name Caption	Command1 Save
CommandButton2	Name Caption	Command2 Load
Label1	Name Autosize	Label1 True
Label2	Name Autosize	Label2 True

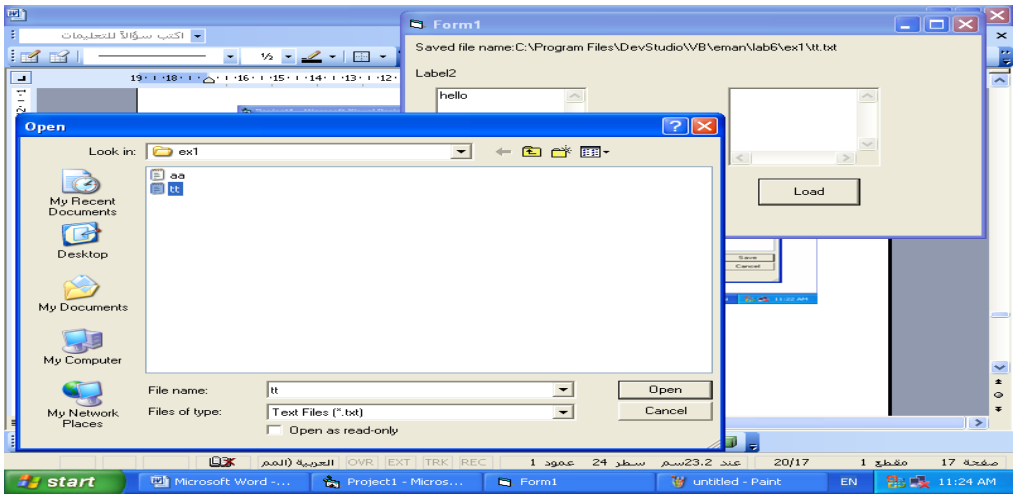
Poject code

```
Private Sub Command1_Click()
On Error GoTo errof1
cmd1.filename = ""
cmd1.Filter = "Text Files (*.txt)|*.txt"
cmd1.FilterIndex = 0
cmd1.ShowSave
On Error GoTo 0
 If Dir(cmd1.filename) <> "" Then
  y = MsgBox("Overwrite File", vbQuestion + vbYesNo, "Overwrite") 'قا  سابد  وجوده  حالة  ي  ف
  If y = vbNo Then Exit Sub
  End If

 'save file
Label1.Caption = "Saved file name:" + cmd1.filename
 filenum = FreeFile
 Open cmd1.filename For Output As filenum
 Print #filenum, Text1.Text
 Close filenum
```

```
errof1:
End Sub
```




```
Private Sub Command2_Click()
On Error GoTo errof2
cmd1.filename = ""
cmd1.Filter = "Text Files (*.txt)|*.txt"
cmd1.FilterIndex = 0
cmd1.ShowOpen
On Error GoTo 0

 If Dir(cmd1.filename) = "" Then
  MsgBox "File not exist", vbCritical, "error" ' في حالة عدم وجوده
  Exit Sub
 End If
' Load file
 Label2.Caption = "Loaded file name:" + cmd1.filename
 filenum = FreeFile
 Open cmd1.filename For Input As filenum
 Text2.Text = Input$(LOF(filenum) - 2, filenum)
 Close filenum
 'Text2.Text = Text2.Text + "good bye"
errof2:
End Sub
```


Form1
Saved file name:C:\Program Files\DevStudio\VB\eman\lab6\ex1\tt.txt
Label2
hello
Load
Open
Look in:
ex1
aa
tt
My Recent Documents
Desktop
My Documents
My Computer
My Network Places
File name:
tt
Files of type:
Text Files (*.txt)
Open as read-only
Open
Cancel
start
Microsoft Word -...
Project1 - Micros...
Form1
untitled - Paint
EN
11:24 AM

الرسم

هناك طريقتان للرسم:-

1 استخدام أدوات الرسم (خط Line، شكل Shape)

2 استخدام خصائص واجراءات الرسم الخاصة بالنموذج Form أو صندوق الصورة Picture Box

طريقة الرسم بالأدوات

الأداة خط Line

بالامكان رسم خط افقي أو عمودي أو مائل.

خصائص الخط:-

X1,Y1: الاحداثي السيني والصادي لنقطة البداية للخط

```
object.X1 [= value]
object.Y1 [= value]
```

X2,Y2: الاحداثي السيني والصادي لنقطة النهاية للخط

```
object.X2 [= value]
object.Y2 [= value]
```

BorderColor: لون الخط

```
object.BorderColor [= color]
```

BorderWidth: عرض الخط (سمكه)

```
object.BorderWidth [= number]
```

BorderStyle: شكل الخط (شفاف لا يظهر، مستمر، متقطع بصورة نقطية أو خطوط أو نقط وخطوط) ويأخذ الارقام (0..6). عندما يكون عرض الخط >1 فلا يظهر شكل الخط (1..4) وانما يظهر بدلها الخط المستمر Solid.

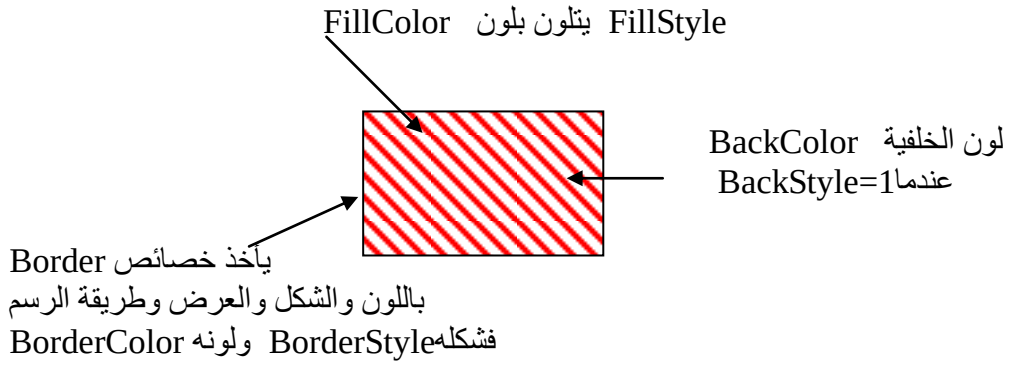
```
object.BorderStyle = [value]
```

Value	Setting	Description
vbTransparent	0	Transparent
vbBSSolid	1	(Default) Solid. The border is centered on the edge of the shape.
vbBSDash	2	Dash
vbBSDot	3	Dot
vbBSDashDot	4	Dash-dot
vbBSDashDotDot	5	Dash-dot-dot
vbBSInsideSolid	6	Inside solid. The outer edge of the border is the outer edge of the shape.

DrawMode: حالات رسم الخط وهناك عدة حالات فقد يرسم الخط مباشرة باللون المعطى عندما تكون قيمة الحالة 13-VbCopyPen أو قد تتم عملية معينة بين لون الخط ولون الخلفية ورائه فيظهر الخط بلون آخر مثل VbXorPen -7 الخ من الحالات (1..16). object.DrawMode [= number]

number	Setting	Description
vbBlackness	1	Blackness.
vbNotMergePen	2	Not Merge Pen — Inverse of setting 15 (Merge Pen).
vbMaskNotPen	3	Mask Not Pen — Combination of the colors common to the background color and the inverse of the pen.
vbNotCopyPen	4	Not Copy Pen — Inverse of setting 13 (Copy Pen).
vbMaskPenNot	5	Mask Pen Not — Combination of the colors common to both the pen and the inverse of the display.
vbInvert	6	Invert — Inverse of the display color.
vbXorPen	7	Xor Pen — Combination of the colors in the pen and in the display color, but not in both.
vbNotMaskPen	8	Not Mask Pen — Inverse of setting 9 (Mask Pen).
vbMaskPen	9	Mask Pen — Combination of the colors common to both the pen and the display.
vbNotXorPen	10	Not Xor Pen — Inverse of setting 7 (Xor Pen).
vbNop	11	Nop — No operation — output remains unchanged. In effect, this setting turns drawing off.
vbMergeNotPen	12	Merge Not Pen — Combination of the display color and the inverse of the pen color.
vbCopyPen	13	Copy Pen (Default) — Color specified by the ForeColor property.
vbMergePenNot	14	Merge Pen Not — Combination of the pen color and the inverse of the display color.
vbMergePen	15	Merge Pen — Combination of the pen color and the display color.
vbWhiteness	16	Whiteness.

الأداة شكل Shape



خصائص الشكل:-

Shape: مستطيل، مربع، بيضوي، دائرة، مستطيل زواياه مدورة، مربع زواياه مستديرة (0..5)

object.Shape [= value]

Value	Setting	Description
vbShapeRectangle	0	(Default) Rectangle
vbShapeSquare	1	Square
vbShapeOval	2	Oval
vbShapeCircle	3	Circle
VbShapeRoundedRectangle	4	Rounded Rectangle
VbShapeRoundedSquare	5	Rounded Square

* نفس خصائص (**DrawMode, Bordecolor, BorderWidth, BorderStyle**) للخط

* نفس خصائص (**Height, Width, Top, Left, BackStyle, Backcolor**) لشريط الطباعة Label

1= BackStyle يظهر لون الخلفية BackColor عندما يكون شكل المليء

object.BackStyle [= number]

Number	Description
0	Transparent — background color and any graphics are visible behind the control.
1	(Default) Opaque — the control's BackColor property setting fills the control and obscures any color or graphics behind it.

FillStyle: شكل المليء: شفاف فلا يظهر وانما يظهر لون الخلفية، أو صلب Solid فيظهر لون المليء FillColor ولا يظهر لون الخلفية BackColor، أو على شكل خطوط (طولية أو عرضية أو مائلة أو شبكة) متقطعة تتلون بلون المليء وترسم فوق لون الخلفية. object.FillStyle [= number]

Number	Setting	Description
vbFSSolid	0	Solid
vbFSTransparent	1	(Default) Transparent
vbHorizontalLine	2	Horizontal Line
vbVerticalLine	3	Vertical Line
vbUpwardDiagonal	4	Upward Diagonal
vbDownwardDiagonal	5	Downward Diagonal
vbCross	6	Cross
vbDiagonalCross	7	Diagonal Cross

FillColor: لون المليء object.FillColor [= value]

مثال: عند النقر على زر الأوامر الأول يتم ما يأتي:- Private Sub Command1_Click()

1 - يتغير شكل الأداة شكل إلى مربع
Shape1.Shape=1
2 يتم ملىء الأداة شكل بخطوط عمودية حمراء
3 يكون إطار الأدة شكل نقط وخطوط وبلون أزرق

```
Shape1.FillStyle=3
Shape1.FillColor=VBRed
Shape1.BorderStyle=4
Shape1.BorderColor=VBBlue
End Sub
```

طريقة الرسم باسخدام خصائص واجراءات الرسم الخاصة بالنموذج Form أو صندوق الصورة Picture Box

الخصائص

DrawMode، FillStyle، FillColor نفس خصائص الأداة شكل Shape

DrawWidth: بدل BorderWidth وهي عرض النقطة أو الخط أو سمك الاطار

object.DrawWidth [= size]
Size: A numeric expression from 1 through 32,767.

DrawStyle : بدل BorderStyle وهو شكل هذا الخط أو الاطار.object.DrawStyle [= number]

ForeColor: لون النقطة أو الخط أو الاطار

AutoReDraw: اعادة الرسم اوتوماتيكيا فاذا كانت False وصغرنا النموذج مثلا minimize ثم كبرناه أو وضع نموذج أخر فوق هذا النموذج فيختفي الرسم الذي رسمناه بالايعازات. فيجب ان نضعه True.

object.AutoRedraw [= boolean]

CurrentX, CurrentY: الموقع الحالي للرسم بالاحداثيات (س، ص) وممكن ان نطبع عندها مباشرة بالايعاز print أو ان نكمل الرسم نسبة لها.

object.CurrentX [= x]
object.CurrentY [= y]

ScaleMode: نمط مقياس الرسم (0-7)

ScaleLeft, ScaleTop, ScaleWidth, ScaleHeight : قيم الاحداثي السيني والصادي لنقطة البداية للرسم على النموذج أو صندوق الصورة (الجهة العليا اليسرى) وقيم العرض والارتفاع حسب مقياس الرسم.

-ScaleMode

خاصية نمط مقياس الرسم تأخذ القيم بين (0-7). القيم من (1-7) تعني تقسيم النموذج أو صندوق الصورة الى تويب (هناك 1440 تويب لكل انج و 567 تويب لكل سم) أو انج أو سنتيمتر أو pixel... الخ. بينما القيمة صفر تعني ان مقياس الرسم يحدد من قبل المبرمج باستخدام الاجراء Scale.

object.ScaleMode [= value]

Value	Setting	Description
vbUser	0	Indicates that one or more of the ScaleHeight, ScaleWidth, ScaleLeft, and ScaleTop properties are set to custom values.
VbTwips	1	(Default) Twip (1440 twips per logical inch; 567 twips per logical centimeter).
VbPoints	2	Point (72 points per logical inch).
VbPixels	3	Pixel (smallest unit of monitor or printer resolution).
vbCharacters	4	Character (horizontal = 120 twips per unit; vertical = 240 twips per unit).
VbInches	5	Inch.
VbMillimeters	6	Millimeter.
VbCentimeters	7	Centimeter.

عندما يكون مقياس الرسم بين (1-7) حينها سيكون ScaleTop=0 ,ScaleLeft=0 اما ScaleWidth, ScaleHeight فيأخذان القيم حسب عرض وارتفاع النموذج أو صندوق الصورة Picture Box (في حالة الرسم فيه) وبمقياس الرسم الجديد. وتزداد قيمة الاحداثي السيني X من اليسار الى اليمين بينما تزداد قيمة الاحداثي الصادي Y من الأعلى الى الأسفل.

ملاحظة:- لا نضع اجراءات الرسم داخل الحدث Form_Load

ملاحظة:- لرسم النقطة (50,80) فممكن ان نرسمها مباشرة نسبة الى (Scaleleft,ScaleTop):

```
Form1.Pset(50,80)
```

أو ممكن ان نرسمها نسبة الى نقطة الأصل أي نقطة تقاطع المحور الأفقي والمحور العمودي في وسط الشاشة (cx,cy) فتكون:

```
Form1.Pset(cx+50,cy-80)
```

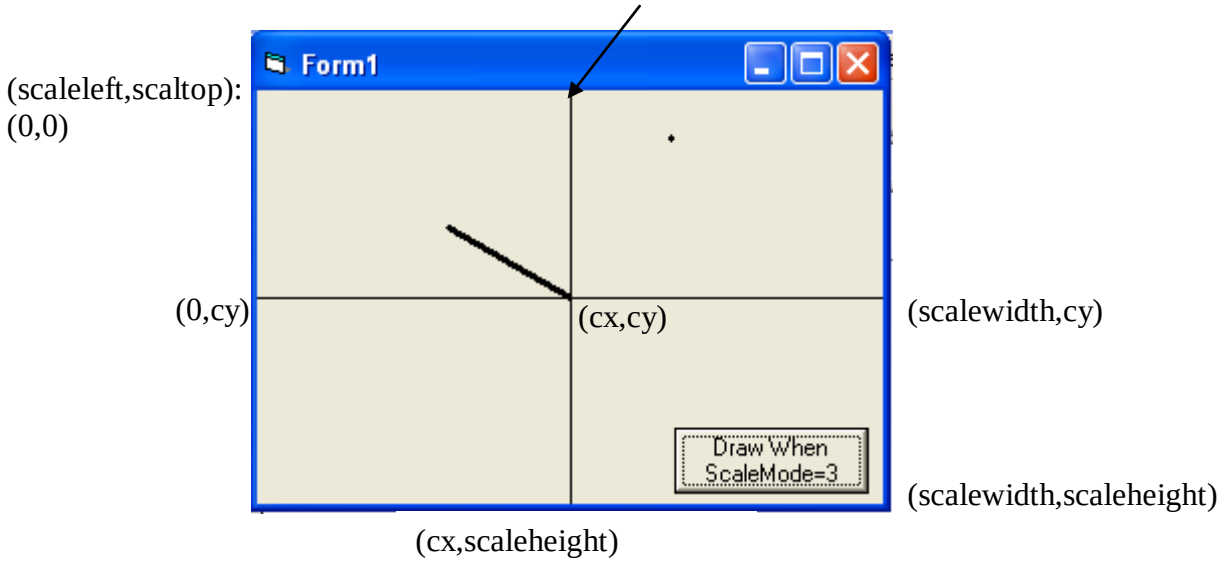
كما في المثال التالي:

مثال:

عندما ScaleMode=3 للنموذج أي ان مقياس الرسم هو Pixel أو أي رقم بين (1-7)

نجعل AutoRedraw=True للنموذج

(cx,0)



Project properties

Object	Property	Setting
Form1	Name AutoRedraw ScaleMode	Form1 True 3
CommandButton1	Name Caption	Command1 Draw When ScaleMode=3

Poject code

```
Option Explicit
Dim cx, cy
Private Sub Command1_Click()
Dim q As Double
cx = Form1.ScaleWidth / 2
cy = Form1.ScaleHeight / 2
 Form1.Line (0, cy)-(Form1.ScaleWidth, cy) ' المحور الافقي
 Form1.Line (cx, 0)-(cx, ScaleHeight) ' المحور العمودي
 Form1.DrawWidth = 3
 Form1.PSet (cx + 50, cy - 80)    '        نسبة الى نقطة الأصل   (50,80) لرسم النقطة
' to draw line angle=150 degree, range=70
q = 150 * 3.14 / 180         'convert angle from degree to radian
 Form1.Line (cx, cy)-(cx + 70 * Cos(q), cy - 70 * Sin(q))
End Sub
```

ملاحظة:- الانتباه عندما يكون رسم أي نقطة (x,y) نسبة الى نقطة الأصل (cx,cy) نقول:

cx+x و cy-y

عندما نضع ScaleMode=0 للنموذج أو صندوق الصورة الذي نرسم فيه

أي تحديد مقياس الرسم من قبل المبرمج حينها يجب ان نعرف المقياس باستخدام الاجرء Scale

Scale Method

Defines the coordinate sytem for a form, picture box, or printer.

Syntax:

Object.scale (x1,y1)- (x2,y2)

(x1,y1): الاحداثي السيني والصادي لنقطة البداية (الجهة العليا اليسرى) للرسم على النموذج أو صندوق الصورة

(x2,y2): الاحداثي السيني والصادي لنقطة النهاية (الجهة السفلى اليمنى) للرسم على النموذج أو صندوق الصورة

حتى اذا لا نضع ScaleMode=0 فمجرد أن نذكر حدود الـــ Scale اي ان نستخدم هذا الاجراء فسيعتبر ScaleMode=0 وتعطى قيم ScaleWidth, ScaleHeight, ScaleLeft, ScaleTop حسب تعريفنا للـــ Scale. وستزداد قيمة الاحداثي السيني X من اليسار الى اليمين بينما تزداد قيمة الاحداثي الصادي Y من الأعلى الى الأسفل.

- عند رسم أي نقطة فيجب ان تكون ضمن حدود هذا الـــ Scale
- ممكن ان نعطي أي Scale مثلاً : Form1.Scale (0,0)-(50,50)

أو Form1.Scale (-100,-100)-(50,50)

أو أي أرقام أخرى حسب الرغبة وليس شرط ان نرسم المحاور (ترسم حسب الحاجة)

ملاحظة:- لرسم النقطة (50,80) والتي تقع ضمن حدود المقياس الذي حددناه بالاجراء Scale فممكن ان نرسمها مباشرة نسبة الى (Scaleleft,ScaleTop) أيا كانت قيمها حسب تحديدنا بالاجراء Scale :

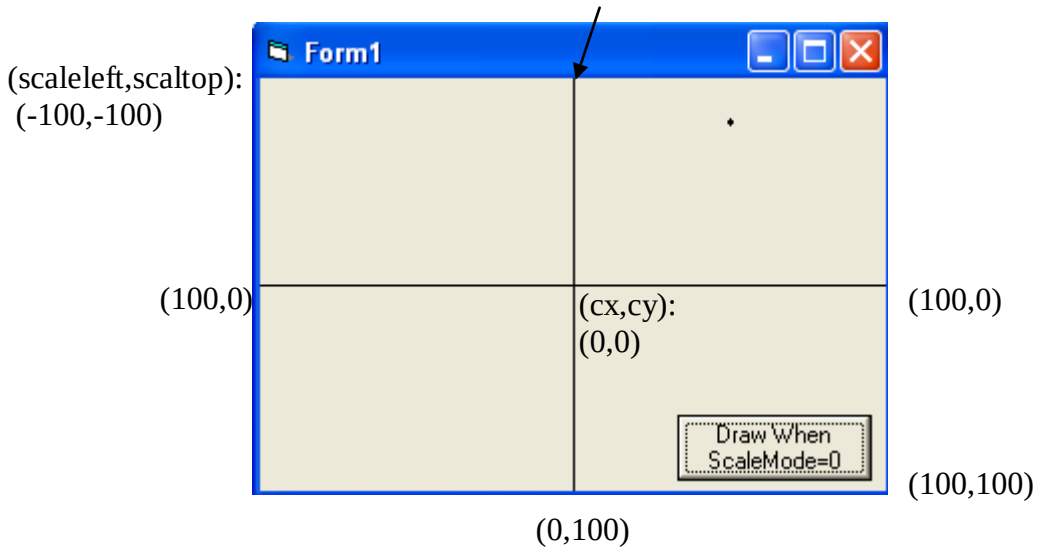
Form1.Pset(50,80)

أو ان نرسم المحاور الافقية والعمودية ونرسم النقطة نسبة الى نقطة تقاطع المحاور ففي حالة اصبح (cx,cy)=(0,0) فنقول: Form1.Pset(50,-80) كما في المثال التالي.

أما اذا لم يكن (cx,cy) مساويا لـــ (0,0) فنقول: Form1.Pset(cx+50,cy-80)

مثال:

(0,-100)



Project properties

Object	Property	Setting
Form1	Name AutoRedraw ScaleMode	Form1 True 0
CommandButton1	Name Caption	Command1 Draw When ScaleMode=0

Poject code

```
Private Sub Command1_Click()
 Form1.Line (-100, 0)-(100, 0)
 Form1.Line (0, -100)-(0, 100)
 Form1.DrawWidth = 3
 Form1.PSet (50, -80) ' (x,-y) فنقول مباشرة (cx,cy):(0,0) لان (cx+x,cy-y) هنا لم نذكر
End Sub
Private Sub Form_Load()
 Form1.Scale (-100, -100)-(100, 100)
End Sub
```

اجراءات الرسم الخاصة بالنموذج وصندوق الصورة

1- **Point**: لاسترجاع لون نقطة مثلا:

```
Dim color
Color=Form1.Point(50,80)
```

2- **Pset**: رسم نقطة

الصيغة العامة:

Object.Pset [Step] (x,y), [Color]

القيمة Step اختيارية وتعني الموقع نسبة الى موقع الرسم الحالي (currentx,currenty)، اللون اختياري فاذا لا نذكره تتلون النقطة حسب لون ForeColor

مثلاً: Form1.Pset (60,100), VbRed

اللون ممكن ان يعطى بهذه الصورة او عن طريق QBColor أو الدالة RGB

3 **Line**: لرسم خط أو مربع أو مستطيل

الصيغة العامة:

object.Line [Step] (x1, y1) [Step] (x2, y2), [color], [B][F]

القيمة Step اختيارية وتعني الموقع نسبة الى موقع الرسم الحالي (currentx,currenty).

(x1, y1) نقطة البداية، (x2, y2) نقطة النهاية.

اللون اختياري فاذا لا نذكره يتلون الخط أو الاطار حسب لون ForeColor (اللون التلقائي هو الأسود) اذا نضع B أو BF فيعني ذلك رسم صندوق (مربع أو مستطيل)، حيث BF تعني ملىء المربع أو المستطيل بلون حسب لون الاطار. واذا نكتب B فقط فيعني ان لون الملىء سيكون حسب FillColor مع الانتباه الى قيمة FillStyle وأحيانا يقال عنها صندوق فارغ في حالة كون FillStyle=1 أي شفاف

أمثلة:

Form1.Line (50,80)-(100,100) ' رسم خط حسبForeColor

Form1.Line (50,80)-(100,100), VbRed ' رسم خط أحمر

لرسم صندوق بلون اطار حسب ForeColor وكذلك الملىء بلون حسب ForeColor :-

Form1.Line (50,80)-(100,100),,BF

لرسم صندوق بلون اطار أحمر والملىء بنفس اللون الأحمر:-

Form1.Line (50,80)-(100,100),VbRed,BF

لرسم صندوق بلون اطار حسب ForeColor والملىء بلون حسب Fillcolor:-

Form1.Line (50,80)-(100,100),,B

لرسم صندوق بلون اطار أحمر والملىء بلون حسب Fillcolor:-

Form1.Line (50,80)-(100,100),VbRed,B

Form1.Line -(100,100) ' يرسم الخط ابتداءا من موقع الرسم الحالي الى هذه النقطة

4 – **Circle**: لرسم دائرة أو قوس دائرة أو مقطع دائرة أو شكل بيضوي

الصيغة العامة:

object.Circle [Step] (x, y), radius, [color, start, end, aspect]

القيمة Step اختيارية وتعني الموقع نسبة الى موقع الرسم الحالي (currentx,currenty).

(x, y) مركز الدائرة، radius نصف قطر الدائرة.

color لون اطار الدائرة وفي حالة عدم ذكر اللون ضمن هذا الايعاز فيتلون الإطار حسب ForeColor.

وتملىء الدائرة بلون حسب FillColor مع مراعاة FillStyle.

start, end زاوية البداية وزاوية النهاية بوحدة القياس نصف قطرية عند الحاجة لرسم منحني Arc أي قوس دائرة واذا تعطى قيم سالبة فيتم رسم مقطع دائرة Sector.

aspect وهي نسبة الارتفاع الى العرض ففي حالة الدائرة تساوي 1 فلاحاجة لذكرها أما في حالة الشكل البيضوي فتعطى رقم لا يساوي واحد.

أمثلة:-

لرسم دائرة بلون اطار حسب ForeColor و المليء بلون حسب FillColor :-

```
Form1.Circle (60,70),10
```

لرسم شكل بيضوي بلون اطار أحمر والمليء بلون حسب Fillcolor:-

```
Form1.Circle (60,70),10,VbRed,,,2
```

لرسم قوس دائرة بلون اطار أحمر :-

```
Form1.Circle (60,70),10,VbRed,90*3.14/180,270*3.14/180
```

مثال: عند النقر على زر الأوامر الأول يتم ما يأتي:-

1 -رسم دائرة مركزها (90,30) ونصف قطرها 10 مملؤة بخطوط حمراء افقية واطارها أخضر

2- رسم النقطة (170,200) بلون عشوائي وبعرض 4

```
Private Sub Command1_Click()
    Form1.FillStyle=2
    Form1.FillColor=VBRed
Form1.Circle(90,30),10,VBGreen
    Form1.DrawWidth=4
Form1.Pset(170,200),RGB(Rnd*255,Rnd*255,Rnd*255)
End Sub
```