وزارة التعليم العالي والبحث العلمي

ملزمـــــة البرمجة المرئيـــــــــــــة
Visual Basic

أعـــداد
م. احمـــد تكليف الحساني
2010

PDF created with pdfFactory trial version www.pdffactory.com

المشروع الاول

-:----------

أساسيات Visual Basic

(أسم المشروع : آلة حاسبة بسيطة)

سوف نتعلم من خلال تنفيذ هذا المشروع كمايلي:

1. **التعامل مع عناصر التحكم الأساسية.**

   - **النموذج Form.**
   - **صندوق النص Text Box.**
   - **التسمية Label.**
   - **زر الأمر Command Button.**

2. **كيفية إسناد قيمة الى خاصية عنصر تحكم.**
3. **الدالة Val.**
4. **العمليات الرياضية الاساسيه في Visual Basic.**
5. **أحد أشكال الأوامر الشرطية IF.**

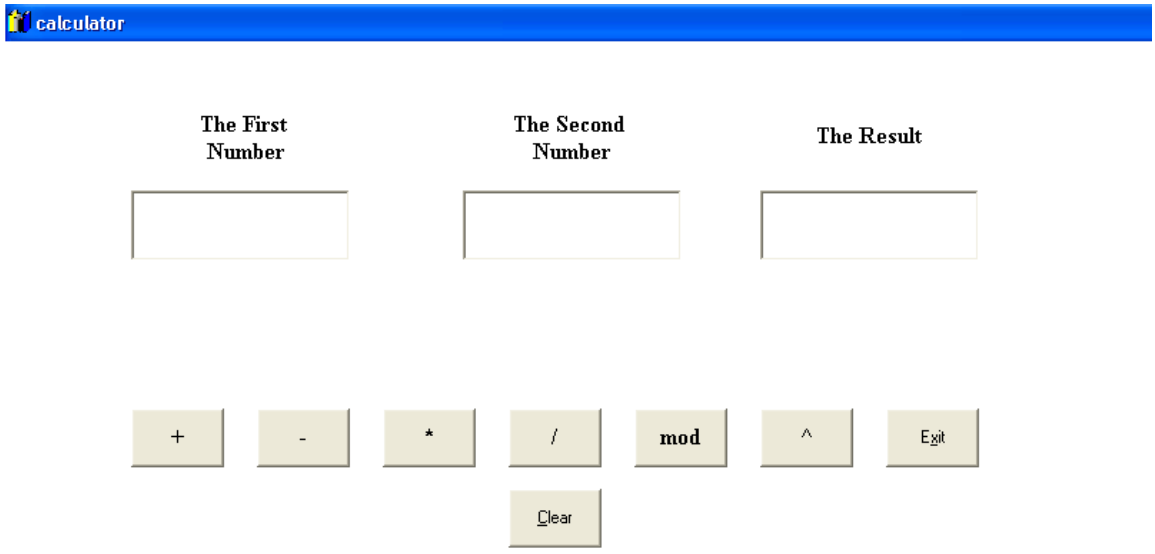
البرنامج (Source Code) الخاص بهذا المشروع موضح كمايلي:-

```
Private Sub Command1_Click()
Text3.Text = Val(Text1.Text) + Val(Text2.Text)
End Sub

Private Sub Command2_Click()
End
End Sub

Private Sub Command3_Click()
Text3.Text = Val(Text1.Text) - Val(Text2.Text)
End Sub

Private Sub Command4_Click()
Text3.Text = Val(Text1.Text) * Val(Text2.Text)
End Sub

Private Sub Command5_Click()
Text3.Text = Val(Text1.Text) / Val(Text2.Text)
End Sub

Private Sub Command6_Click()
Text3.Text = Val(Text1.Text) Mod Val(Text2.Text)
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

```
Private Sub Command7_Click()
Text3.Text = Val(Text1.Text) ^ Val(Text2.Text)
End Sub

Private Sub Command8_Click()
Text3.Text = ""
Text2.Text = ""
Text1.Text = ""
End Sub
```

ناتج تنفيذ البرنامج أعلاه موضح بالشكل التالي:



1. **Properties Of Form.**
2. **Properties Of Labels.**
3. **Properties Of Bottuns.**

PDF created with pdfFactory trial version www.pdffactory.com

1. Properties Of Form1.

PROPERTIES OF FORM1	
Alphabetic	**Categorized**
(Name)	Form1
Caption	Calculator
Font	Times New Roman Font Style Bold Font Size 12
ForeColor	Black اسود

2. Properties Of Labels.

PROPERTIES OF LABEL1	
Alphabetic	**Categorized**
(Name)	Label1
Alignment	2-Center
Caption	The First Number
Font	Times New Roman Font Style Bold Font Size 12
ForeColor	Black أسود

PROPERTIES OF LABEL2	
Alphabetic	**Categorized**
(Name)	Label2
Alignment	2-Center
Caption	The ◌Second Number
Font	Times New Roman Font Style Bold Font Size 12
ForeColor	Black أسود

PDF created with pdfFactory trial version www.pdffactory.com

PROPERTIES OF LABEL3	
Alphabetic	**Categorized**
(Name)	Label3
Alignment	2-Center
Caption	The Result
Font	Times New Roman Font Style Bold Font Size 12
ForeColor	Black أسود

3. Properties Of TextBox.

PROPERTIES OF TEXT1	
Alphabetic	**Categorized**
(Name)	Text1
Alignment	2-Center
Font	Times New Roman Font Style Bold Font Size 16
ForeColor	Blue ازرق
Text	

PROPERTIES OF TEXT2	
Alphabetic	**Categorized**
(Name)	Text2
Alignment	2-Center
Font	Times New Roman Font Style Bold Font Size 16
ForeColor	Blue ازرق
Text	

PROPERTIES OF TEXT3	
Alphabetic	**Categorized**
(Name)	Text3
Alignment	2-Center
Font	Times New Roman Font Style Bold Font Size 16
ForeColor	Blue ازرق

PDF created with pdfFactory trial version www.pdffactory.com

Text	

4. Properties Of Command Bottuns.

PROPERTIES OF COMMAND1	
Alphabetic	**Categorized**
(Name)	Command1
Caption	+
Font	Times New Roman Font Style Bold Font Size 16

PROPERTIES OF COMMAND3	
Alphabetic	**Categorized**
(Name)	Command3
Caption	-
Font	Times New Roman Font Style Bold Font Size 16

PROPERTIES OF COMMAND4	
Alphabetic	**Categorized**
(Name)	Command4
Caption	*
Font	Times New Roman Font Style Bold Font Size 16

PROPERTIES OF COMMAND5	
Alphabetic	**Categorized**
(Name)	Command5
Caption	/
Font	Times New Roman Font Style Bold Font Size 16

PROPERTIES OF COMMAND6	
Alphabetic	**Categorized**
(Name)	Command6
Caption	Mod
Font	Times New Roman

PDF created with pdfFactory trial version www.pdffactory.com

	Font Style Bold Font Size 16
Properties of Command7	
Alphabetic	**Categorized**
(Name)	Command7
Caption	^
Font	Times New Roman Font Style Bold Font Size 16

PROPERTIES OF COMMAND2	
Alphabetic	**Categorized**
(Name)	Command2
Caption	E&xit
Font	Times New Roman Font Style Bold Font Size 16

PROPERTIES OF COMMAND8	
Alphabetic	**Categorized**
(Name)	Command8
Caption	&Clear
Font	Times New Roman Font Style Bold Font Size 16

PDF created with pdfFactory trial version www.pdffactory.com

المشروع الثاني

-:----------

الة حاسبه بسيطه متكاملة

سوف نتعلم من خلال هذا المشروع مايلي:

- التعامل مع عناصر التحكم الاساسيه

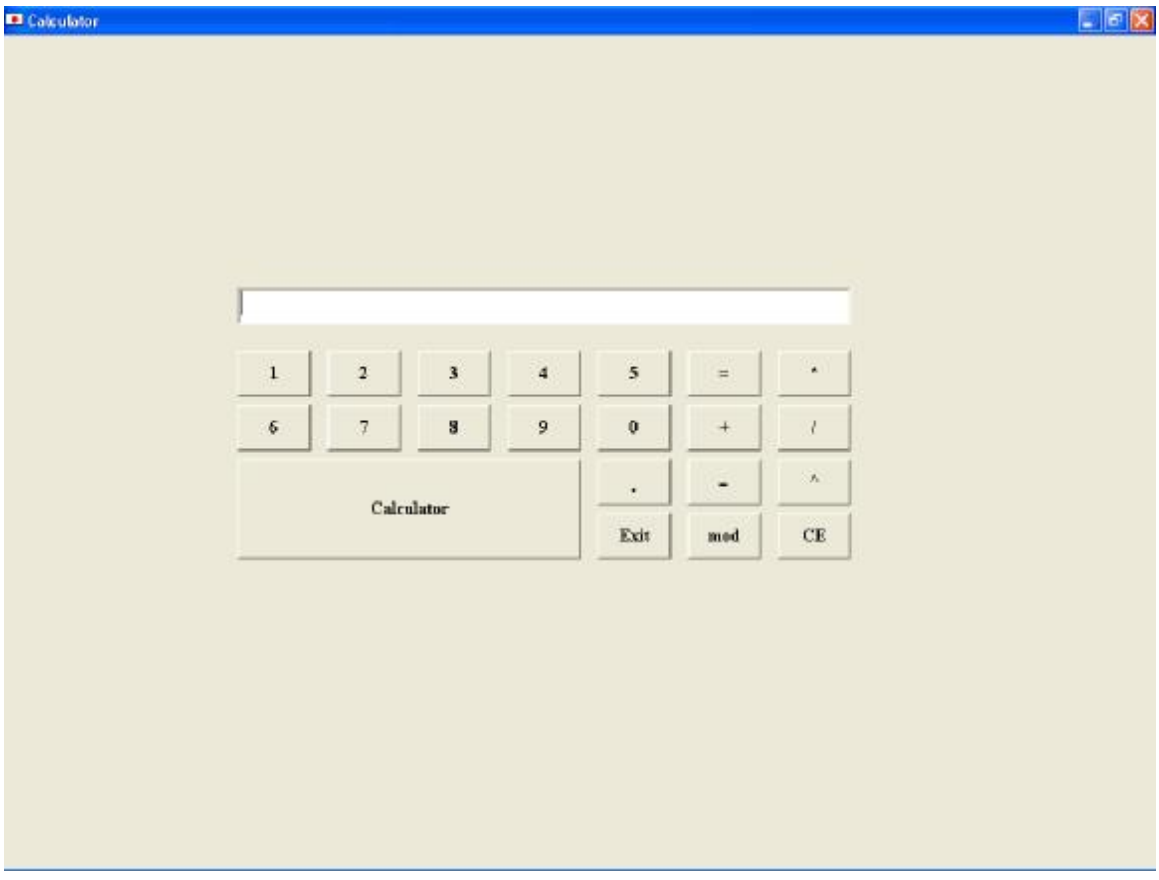
- النموذج Form
- صندوق النص Text Box
- التسمية Label
- زر الامر Command Button

- اسناد قيمه ثابته الى خاصيه عنصر التحكم Text Box
- الدالة Val
- العمليات الاساسيه الرياضيه في Visual Basic
- الشرط If .. Then .. Else

الهدف من المشروع

سوف نقوم ببناء الة حاسبة بسيطه متكاملة تسمح للمستخدم تنفيذ العمليات الرياضيه الاساسيه على عددين (يستخدم Text Box) واحد لتنفيذ عمليات الحاسبه هذه وليس ثلاثه مثل المثال السابق.

عند تنفيذ البرنامج سوف نحصل على الشاشه التاليه:



PDF created with pdfFactory trial version www.pdffactory.com

البرنامج الذي يقوم بتنفيذ الحاسبه اعلاه مدون كالتالي:

```
Dim n1 As Single
Dim n2 As Single
Dim op As String
Private Sub Command1_Click()
Text1.Text = (Text1.Text) + "1"
End Sub

Private Sub Command10_Click()
Text1.Text = (Text1.Text) + "8"
End Sub

Private Sub Command11_Click()
Text1.Text = (Text1.Text) + "9"
End Sub

Private Sub Command12_Click()
Text1.Text = (Text1.Text) + "0"
End Sub

Private Sub Command13_Click()
n1 = Val(Text1.Text)
op = "+"
Text1.Text = ""
End Sub

Private Sub Command14_Click()
n1 = Val(Text1.Text)
op = "/"
Text1.Text = ""
End Sub

Private Sub Command15_Click()
n1 = Val(Text1.Text)
op = "mod"
Text1.Text = ""
End Sub

Private Sub Command16_Click()
Text1.Text = ""
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

```
Private Sub Command17_Click()
End
End Sub

Private Sub Command18_Click()
n1 = Val(Text1.Text)
op = "^"
Text1.Text = ""
End Sub

Private Sub Command19_Click()
Text1.Text = (Text1.Text) + "."
End Sub

Private Sub Command2_Click()
Text1.Text = (Text1.Text) + "2"
End Sub

Private Sub Command20_Click()
Text1.Text = "Mathematical Calculator"
End Sub

Private Sub Command21_Click()
n1 = Val(Text1.Text)
op = "-"
Text1.Text = ""
End Sub

Private Sub Command3_Click()
Text1.Text = (Text1.Text) + "3"
End Sub

Private Sub Command4_Click()
Text1.Text = (Text1.Text) + "4"
End Sub

Private Sub Command5_Click()
Text1.Text = (Text1.Text) + "5"
End Sub

Private Sub Command6_Click()
n2 = Val(Text1.Text)
If op = "+" Then
Text1.Text = n1 + n2
```


PDF created with pdfFactory trial version www.pdffactory.com

```
ElseIf op = "*" Then
Text1.Text = n1 * n2
ElseIf op = "-" Then
Text1.Text = n1 - n2
ElseIf op = "/" Then
Text1.Text = n1 / n2
ElseIf op = "^" Then
Text1.Text = n1 ^ n2
ElseIf op = "mod" Then
Text1.Text = n1 Mod n2
Else
End If
End Sub

Private Sub Command7_Click()
n1 = Val(Text1.Text)
op = "*"
Text1.Text = ""
End Sub

Private Sub Command8_Click()
Text1.Text = (Text1.Text) + "6"
End Sub

Private Sub Command9_Click()
Text1.Text = (Text1.Text) + "7"
End Sub
```

نستخدم الخصائص التاليه:

1. الشكل Form

PROPERTIES OF FORM1	
Alphabetic	**Categorized**
(Name)	Form1
Caption	Calculator
Font	Times New Roman Font Style Bold Font Size 12
Back Color	High Light
ForeColor	Black اسود

PDF created with pdfFactory trial version www.pdffactory.com

2. صندوق النص Text Box

PROPERTIES OF TEXT BOX - 1	
Alphabetic	**Categorized**
(Name)	Text1
Alignment	0 – Left Justify
Back Color	High Light
Font	Times New Roman Font Style Bold Font Size 14
Text	

3. زر الامر Command Button

PROPERTIES OF COMMAND1	
Alphabetic	**Categorized**
(Name)	Command
Caption	1, 2, 3, 4, 5, 6, 7, 8, 9, 0
Font	Times New Roman Font Style Bold Font Size 14

PROPERTIES OF COMMAND1	
Alphabetic	**Categorized**
(Name)	Command
Caption	Exit, CE
Font	Times New Roman Font Style Bold Font Size 14

PROPERTIES OF COMMAND1	
Alphabetic	**Categorized**
(Name)	Command
Caption	*, /, +, -, ^, Mod, =
Font	Times New Roman Font Style Bold Font Size 14

PDF created with pdfFactory trial version www.pdffactory.com

PROPERTIES OF COMMAND1	
Alphabetic	**Categorized**
(Name)	Command
Caption	Calculator
Font	Times New Roman Font Style Bold Font Size 14

PDF created with pdfFactory trial version www.pdffactory.com

المشروع الثالث

-:----------

(تحويل العملات)

سوف نتعلم من خلال هذا المشروع مايلي:-

1. التعامل مع عناصر التحكم التاليه:-

- النموذج Form
- صندوق النص Text Box
- التسميه Label
- زر الاوامر Command Button
- صندوق القائمه المنسدله Combo Box
- الصوره Image

2. الشكل العام للاوامر الشرطيه If .. then ..else
3. الشكل العام للامر الشرطي Select .. Case

الهدف من المشروع

-:----------

بناء واجهة لتحويل العملات او المبلغ المكافيء بالدولار لقيمة مبلغ بعملة بلد معين,

عند تنفيذ هذا المشروع تظهر الواجه التاليه:

PDF created with pdfFactory trial version www.pdffactory.com

نستخدم الخصائص التاليه الخاصه بكل جزء من الاجزاء المستخدمه في بناء المشروع:

4. الشكل Form

PROPERTIES OF FORM1	
Alphabetic	**Categorized**
(Name)	FrmEx
Caption	Money Changer
Font	Times New Roman Font Style Bold Font Size 12
Back Color	High Light
ForeColor	Black اسود

5. صندوق النص Text Box

PROPERTIES OF TEXT BOX - 1	
Alphabetic	**Categorized**
(Name)	TxtIn
Alignment	2 - Center
Back Color	High Light
Font	Times New Roman Font Style Bold Font Size 14
Text	

PROPERTIES OF TEXT BOX - 2	
Alphabetic	**Categorized**
(Name)	TxtOut
Alignment	2 - Center
Back Color	High Light
Font	Times New Roman Font Style Bold Font Size 14
Text	

PDF created with pdfFactory trial version www.pdffactory.com

6. صندوق القوائم Combo Box

PROPERTIES OF COMBO BOX	
Alphabetic	**Categorized**
(Name)	CboIn
Back Color	Window Background
Font	Times New Roman Font Style Bold Font Size 14
list	France Germany Italy

7. مفتاح الاوامر Command Button

PROPERTIES OF COMMAND1	
Alphabetic	**Categorized**
(Name)	Command1
Caption	How Many Dollars
Font	Times New Roman Font Style Bold Font Size 12

PROPERTIES OF COMMAND2	
Alphabetic	**Categorized**
(Name)	Command2
Caption	How Many Dollars
Font	Times New Roman Font Style Bold Font Size 12

PROPERTIES OF COMMAND3	
Alphabetic	**Categorized**
(Name)	Command3
Caption	How Many Dollars
Font	Times New Roman Font Style Bold

PDF created with pdfFactory trial version www.pdffactory.com

	Font Size 12

8. التسميه Label

PROPERTIES OF LABEL	
Alphabetic	**Categorized**
(Name)	LblIn
Alignment	2-Center
Caption	-
Font	Times New Roman Font Style Bold Font Size 12
ForeColor	Black أسود

PROPERTIES OF LABEL2	
Alphabetic	**Categorized**
(Name)	Label2
Alignment	2-Center
Caption	The Amount
Font	Times New Roman Font Style Bold Font Size 14
ForeColor	Black أسود

PROPERTIES OF LABEL3	
Alphabetic	**Categorized**
(Name)	Label3
Alignment	2-Center
Caption	The Country
Font	Times New Roman Font Style Bold Font Size 14
ForeColor	Black أسود

PROPERTIES OF LABEL4	
Alphabetic	**Categorized**
(Name)	Label4
Alignment	2-Center
Caption	The Result
Font	Times New Roman Font Style Bold

PDF created with pdfFactory trial version www.pdffactory.com

	Font Size 12
ForeColor	Black أسود
Properties of label1	
Alphabetic	**Categorized**
(Name)	Label1
Alignment	2-Center
Caption	$
Font	Times New Roman Font Style Bold Font Size 14
ForeColor	Black أسود

9. صندوق الصور Picture Box

PROPERTIES OF PICTURE BOX	
Alphabetic	**Categorized**
(Name)	ImgIn
AutoSize	True
Visible	False
Picture	Micro Soft Visual Studio, Common, Graphics, Icons, Flags, Choose Country Flag, CTRUSA

PROPERTIES OF PICTURE BOX	
Alphabetic	**Categorized**
(Name)	ImgF
AutoSize	True
Visible	False
Picture	Micro Soft Visual Studio, Common, Graphics, Icons, Flags, Choose Country Flag, FLGFRAN

PDF created with pdfFactory trial version www.pdffactory.com

PROPERTIES OF PICTURE BOX	
Alphabetic	**Categorized**
(Name)	ImgG
AutoSize	True
Visible	False
Picture	Micro Soft Visual Studio, Common, Graphics, Icons, Flags, Choose Country Flag, FLGGERM

PROPERTIES OF PICTURE BOX	
Alphabetic	**Categorized**
(Name)	ImgI
AutoSize	True
Visible	False
Picture	Micro Soft Visual Studio, Common, Graphics, Icons, Flags, Choose Country Flag, FLGITALY

Sourse Code البرنامج

```
Private Sub Command1_Click()
   If CboIn.Text = "France" Then
      TxtOut.Text = Val(TxtIn.Text) * 0.2
      LblIn.Caption = "FR"
      ImgIn.Picture = ImgF.Picture
   ElseIf CboIn.Text = "Germany" Then
      TxtOut.Text = Val(TxtIn.Text) * 0.6
      LblIn.Caption = "DM"
      ImgIn.Picture = ImgG.Picture
   ElseIf CboIn.Text = "Italy" Then
      TxtOut.Text = Val(TxtIn.Text) * 0.1
      LblIn.Caption = "IL"
      ImgIn.Picture = ImgI.Picture
   Else
      MsgBox "Please Choose The Country From The List", vbCritical, "Error"
   End If
```


PDF created with pdfFactory trial version www.pdffactory.com

```
End Sub

Private Sub Command2_Click()
If CboIn.ListIndex = 0 Then
        TxtOut.Text = Val(TxtIn.Text) * 0.2
        LblIn.Caption = "FR"
        ImgIn.Picture = ImgF.Picture
    ElseIf CboIn.ListIndex = 1 Then
        TxtOut.Text = Val(TxtIn.Text) * 0.6
        LblIn.Caption = "DM"
        ImgIn.Picture = ImgG.Picture
    ElseIf CboIn.ListIndex = 2 Then
        TxtOut.Text = Val(TxtIn.Text) * 0.1
        LblIn.Caption = "IL"
        ImgIn.Picture = ImgI.Picture
    Else
        MsgBox "Please Choose The Country From The List", vbCritical, "Error"
    End If
End Sub

Private Sub Command3_Click()
Select Case CboIn.ListIndex
Case 0
        TxtOut.Text = Val(TxtIn.Text) * 0.2
        LblIn.Caption = "FR"
        ImgIn.Picture = ImgF.Picture
Case 1:
        TxtOut.Text = Val(TxtIn.Text) * 0.6
        LblIn.Caption = "DM"
        ImgIn.Picture = ImgG.Picture
Case 2:
        TxtOut.Text = Val(TxtIn.Text) * 0.1
        LblIn.Caption = "IL"
        ImgIn.Picture = ImgI.Picture
Case Else
        MsgBox "Please Choose The Country From The List", vbCritical, "Error"
End Select
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الرابع

-:----------

(أدارة الفنادق)

سوف نتعلم من خلال هذا المشروع

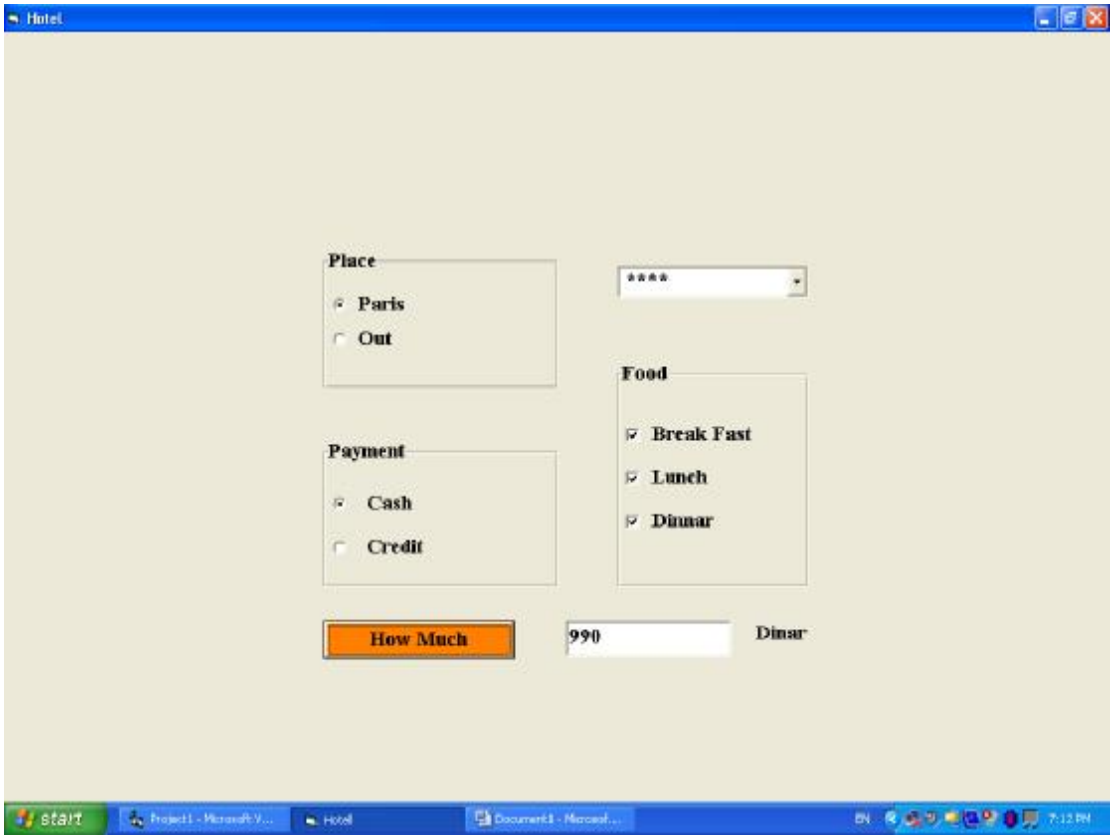
- التعامل مع عناصر التحكم:

  - الاطار Frame
  - زر الخيار Option Button
  - صندوق التحقق Check Box

- تعليمات الشرط

الهدف من المشروع:- تكوين برنامج بلغة Visual Basic في أدارة الفندق من حيث كلفة المبيت المعتمده على تصنف الفندق (من حيث عدد النجوم), مكان الفندق هل هو في مركز المدينه او في الضواحي, الوجبات التي يريد تناولها (فطور, غداء, عشاء) طرقيقة الدفع (نقدا او شيك).

عند تنفيذ هذا المشروع سوف تظهر الواجهه التاليه الخاصه بتحديد مواصفات الاقامه في فندق في فرنسا مثلا:



PDF created with pdfFactory trial version www.pdffactory.com

سوف نقوم في هذا الرنامج بحساب كلفة الاقامة وكما يلي:

1. سعر الليله في فندق (*) 100
   سعر الليله في فندق (**) 200
   سعر الليله في فندق (***) 300
   سعر الليله في فندق (****) 400
   سعر الليله في فندق (*****) 500

2. أذا كان الفندق في باريس يضاف 50% للتسعيره السابقه.

3. اسعار الطعام:-

   سعر الفطور 50
   سعر الغداء 100
   سعر العشاء 200

4. اذا كان الدفع نقدا يخصم 10% من السعر الكلي.

البرنامج الذي يقوم بتنفيذ هذه الواجهه مدون ادناه

```
Dim X As Single
Private Sub Command1_Click()
Select Case CboStar.ListIndex
Case 0:
X = 100
Case 1:
X = 200
Case 2:
X = 300
Case 3:
X = 400
Case 4:
X = 500
Case Else
MsgBox "Choose how many stars", vbCritical, "Error"
Exit Sub
End Select
If OptParis.Value = True Then X = X * 1.5
If ChkBreak.Value = 1 Then X = X + 50
If ChkLunch.Value = 1 Then X = X + 100
If ChkDinner.Value = 1 Then X = X + 200
If OptCash.Value = True Then X = X * 0.9
TxtOut.Text = X
```


PDF created with pdfFactory trial version www.pdffactory.com

End Sub

مراحل التنفيذ

----------:-

نفتح مشروعا قياسيا جديدا .

- نضبط خصائص النموذج وكمايلي:
- نضبط خصائص الشكل Form كمايلي:

PROPERTIES OF FORM1	
Alphabetic	**Categorized**
(Name)	Form1
Caption	Hotel
Font	Times New Roman Font Style Bold Font Size 14
ForeColor	Black اسود

- ننشي صندوق قائمة منسدلة ونضبط الخصائص كالاتي:

PROPERTIES OF COMBOBOX	
Alphabetic	**Categorized**
(Name)	CboStar
List	* ** *** **** *****
Text	Choose the stars

- ننشأ ثلاثة عناصر من نوع الاطار Frame ونضبط الخاصية Caption لكل منهما: , Place Payment, Food.

يستخدم الاطار لضم مجموعه من العناصر المترابطة فيما بينها عمليا, يجب انشاء الاطار اولا ثم انشاء العناصر بداخله (اي لاننشيء مثلا ازرار الخيار في مشروعنا على النموذج ثم نسحبة داخل الاطار), للتأكد من تابعية العناصر للاطار يجب عند سحب الاطار ان تسحب العناصر معه.

PDF created with pdfFactory trial version www.pdffactory.com

يستخدم الاطار بشكل اساسي لضم مجموعه من ازرار الخيار Option Button ويمكن لزر واحد فقط في المجموعه ان يكون محددا, وعند النقر على زر في المجموعه نفسها يتحدد هذا الزر ويفقد التحديد الزر الذي علية التحديد.

- ننشيء زر خيار Option Button داخل الاطار Food ونضبط خصائصة كمايلي:

PROPERTIES OF OPTION BUTTON	
Alphabetic	**Categorized**
(Name)	OptParis
Caption	Paris
Value	True

PROPERTIES OF OPTION BUTTON	
Alphabetic	**Categorized**
(Name)	OptProv
Caption	Out
Value	False

تصميم الخاصيه Value لعنصر التحكم Option Button تأخذ القيمة True اذا كان زر الخيار محددا , اما اذا كان زر الخيار غير محددا فتاخذ القيمه False.

- ننشيء زر خيار Option Button داخل الاطار Payment ونضبط خصائصة كمايلي:

PROPERTIES OF OPTION BUTTON	
Alphabetic	**Categorized**
(Name)	OptCash
Caption	Cash
Value	True

PROPERTIES OF OPTION BUTTON	
Alphabetic	**Categorized**
(Name)	OptCheq
Caption	Credit
Value	False

- ننشيء ثلاثة صناديق تحقق Check Box ضمن الاطار Food ونضبط الخصائص كمايلي:

PDF created with pdfFactory trial version www.pdffactory.com

PROPERTIES OF CHEK BOX	
Alphabetic	**Categorized**
(Name)	ChkBreak
Caption	Break Fast
Value	1- Checked

PROPERTIES OF CHEK BOX	
Alphabetic	**Alphabetic**
(Name)	ChkLunch
Caption	Lunch
Value	0 - Unchecked

PROPERTIES OF CHEK BOX	
Alphabetic	**Alphabetic**
(Name)	ChkDinner
Caption	Dinner
Value	0 - Unchecked

تصميم الخاصية Value لعنصر التحكم Check Box تاخذ القيم الثلاثة التالية

0-UNCHECKED	غير محدد
1-Checked	محدد
2-Grayed	رمادي

- ننشيء صندوق نص لوضع الناتج فيه واسمه (TxtOut) وزر امر لايجاد المبلغ الكلي اسمه (How Much) لاجراء الحساب عند النقر علية .

PDF created with pdfFactory trial version www.pdffactory.com

المشروع الخامس

-:----------

اسم المشروع :

(أسم المستخدم User Name وكلمة السر Password)

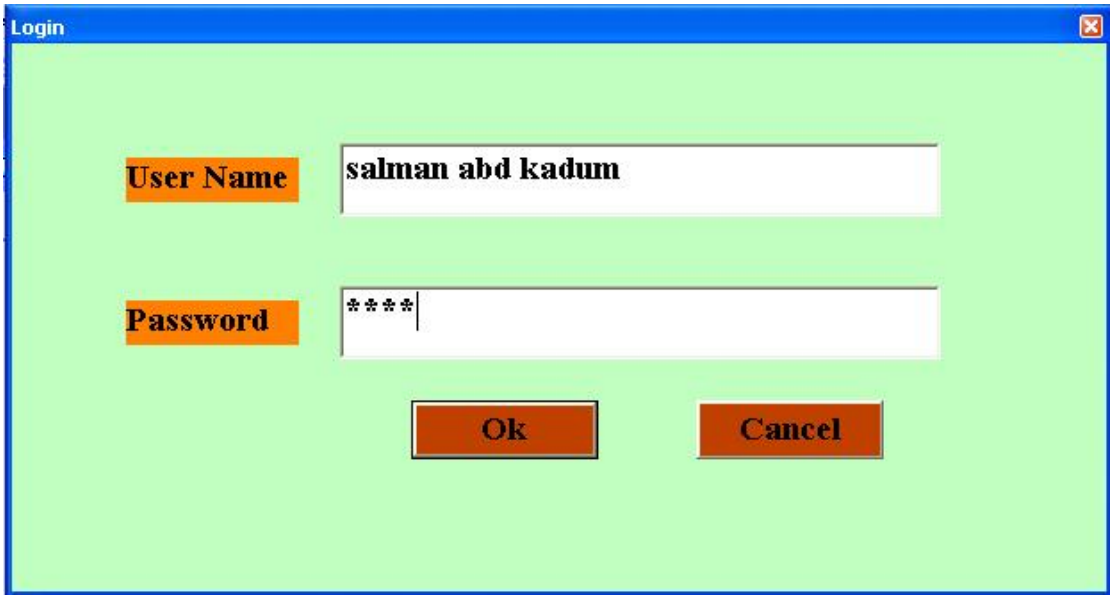
سوف نتعلم من خلال هذا المشروع مايلي:

- v التعامل مع المتغيرات Variables
  - المتغيرات المحلية للاجراء procedure
  - المتغيرات الساكنة Static للاجراء
  - المتغيرات العامة للنموذج
  - المتغيرات العامة للمشروع
- v التعامل مع اكثر من نموذج في المشروع
- v التعامل مع الوحدات النمطية Module

الهدف من المشروع

-:----------

عند تنفيذ المشروع سوف تظهر الواجهة التالية التي تطلب اسم المستخدم وكلمة السر



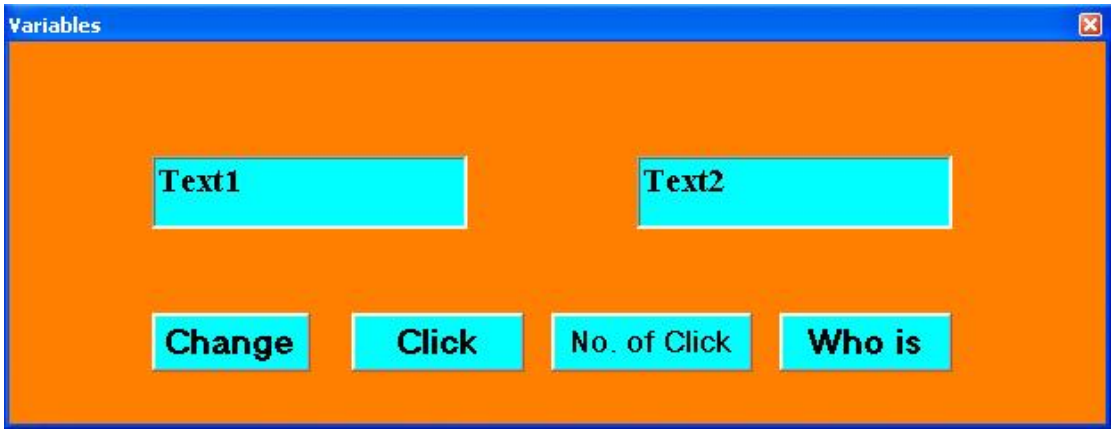
PDF created with pdfFactory trial version www.pdffactory.com

عندما يدخل المستخدم اسمة في مربع النص ثم كلمة السر في مربع النص الثاني

في مثالنا هذا, أذا أخطأ المستخدم في ادخال كلمة السر سوف تظهر العبارة التنبيهية التالية: toto

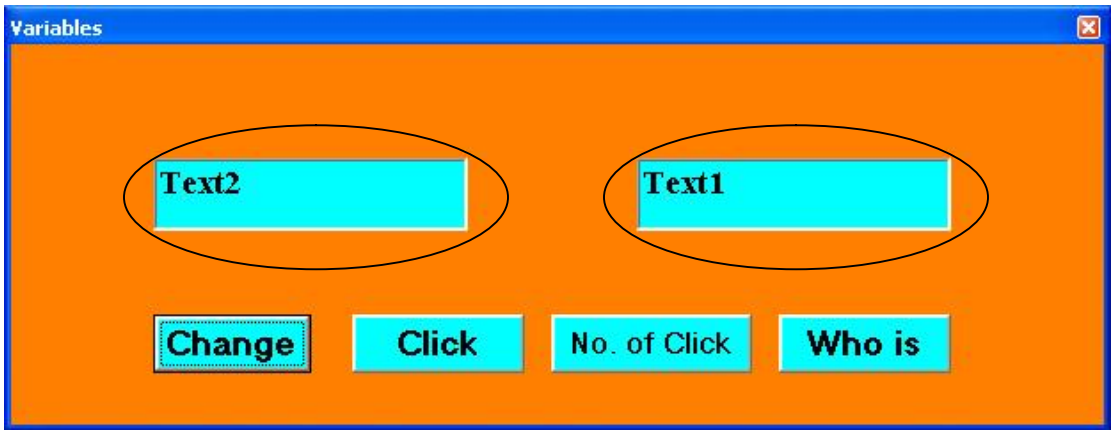
- يسمح للمستخدم ان يخطأ مرتين في كلمة السر, أذا أخطأ في المرة الثالثة سوف يظهر صندوق الرسالة التالي ويغلق البرنامج.

- في حالة كتابة كلمة السر الصحيحة , تظهر الواجهة التالية (Form2)



PDF created with pdfFactory trial version www.pdffactory.com

- عند النقر على مفتاح Change سوف يؤدي الى التبديل بين محتويات الصندوق الاول مع محتويات الصندوق الثاني وكمايلي:



- يمكن للمستخدم ان ينقر عدد من المرات على المفتاح Click , بعدها اذا نقر على المفتاح No. of Click سوف يظهر له مربع رسالة يحتوي على عدد النقرات التي قام بها المستخدم على المفتاح Click وكمايلي :.

- عند النقر على مفتاح Who is تظهر رسالة تحتوي على اسم المستخدم الحالي وكمايلي:

PDF created with pdfFactory trial version www.pdffactory.com

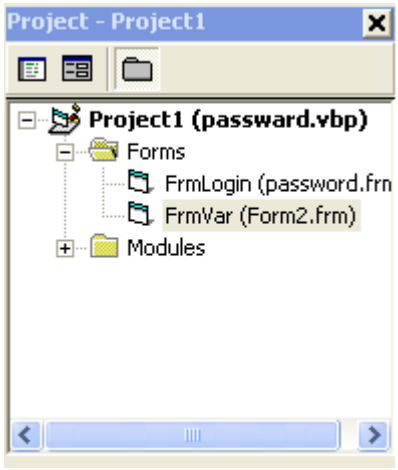
مراحل التنفيـــــذ

-:----------

1. **نفتح مشروعا قياسيا جديدا.**
2. **نظيف الى المشروع نموذجا ثانيا Form2, ووحدة نمطية جديدة Module.**

- **لاضافة نموذج اخر الى المشروع , نختار القائمة Project/Add Form.**
- **لاضافة وحدة نمطية , نختار القائمة Project / Add Module.**

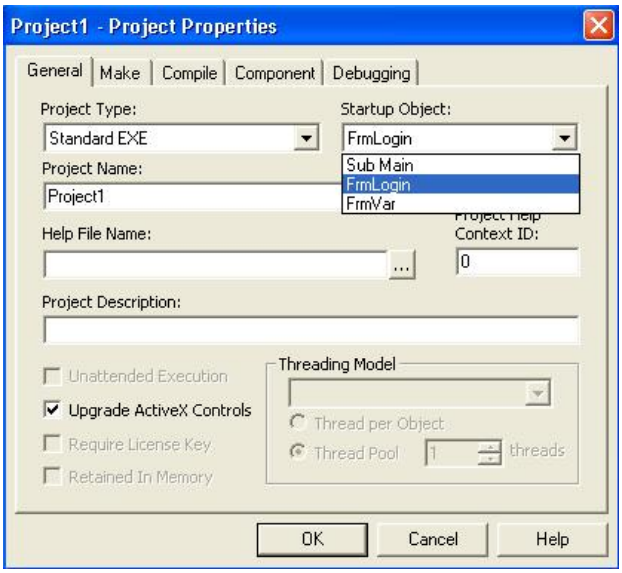
ملاحظـة -1:- نستخدم مستكشف المشروع Project Explorer للتنقل بين مكونات المشروع المختلفة وكمايلي:



ملاحظـه- 2:- نقوم خلال هذا المشروع بحفظ اربعة ملفات , ملف للنموذج الاول وملف وملف اخر للمشروع. moduleللنموذج الثاني وملف للوحدة النمطية

ملاحظـة -3:- لجعل تنفيذ المشروع يبدأ من نموذج معين , نختار القائمة Project / Project Properties ثم نختار النموذج المطلوب من القائمة Startup Object

PDF created with pdfFactory trial version www.pdffactory.com



3. نضبط خصائص النوذجين كمايلي: -

FORM1 PROPERTIES	
Name	FrmLogin
BorderStyle	3-FixedDialoag
Caption	Login
MinButton	False

FORM2 PROPERTIES	
Name	FrmVar
BorderStyle	2-Sizable
Caption	Variables
MinButton	True

PDF created with pdfFactory trial version www.pdffactory.com

4. تصميم الخاصية Border Style : - تستخدم لتحديد شكل اطار النموذج , حيث ان القيمة الافتراضية لهذه الخاصيه هي (2-Sizable) اي انه يمكن للمستخدم تغيير ابعاد النموذج. اما اذا وضعنا القيمة (3-FixedDialog) فيصبح النموذج غير قابل للتمدد.
5. تصميم الخاصية MinButton تستخدم لالغاء او لابقاء مفتاح تصغير النموذج.
6. ننشأ عنصرين من النوع (TextBox) صندوق نص على النموذج الاول ونضبط خصائصة كمايلي:-

TEXTBOX PROPERTIES	
Name	TxtUserName
PasswordChar	
Text	

TEXTBOX PROPERTIES	
Name	TxtPassword
PasswordChar	*
Text	

7. تصميم الخاصية PasswordChar: تستخدم لجعل حرف ما يظهر عوضا عن الاحرف المدخله(عادة نضع الحرف *).
8. ننشأ مفتاحين على النموذج الاول ونضبط الخصائص كمايلي:

COMMANDBUTTON	
Name	CmdOk
Cancel	False
Caption	Ok
Default	True

PDF created with pdfFactory trial version www.pdffactory.com

CommandButton	
Name	CmdCancel
Cancel	True
Caption	Cancel
Default	False

9. تصميم الخاصية Cancel: تستخدم لتحديد ان مفتاحا معينا على النموذج هو مفتاح الالغاء Cancelسوف يتم تنفيذ هذا المفتاح بالنقر عليه او بالضغط على مفتاح ESC.
10. تصميم الخاصية Default : تستخدم لتحديد ان مفتاحا معينا على النموذج هو المفتاح الافتراضي , سوف يتم تنفيذ هذا المفتاح بالنقر عليه او بالضغط على المفتاح Enter.
11. ننشأ عنصرين من النوع Label على النموذج الاول ونضبط الخاصيه Caption لوضع الاسم وكلمة السر.
12. ننشأ صندوقي نص على النموذج الثاني وندع الاسماء الافتراضيه لهما Text1, Text2 .
13. ننشأ اربعة مفاتيح على النموذج الثاني ونضبط الخاصية caption لهما وكمايلي: Change, Click, No. of Click, Who is.
14. نبرمج المفتاح Change على النموذج الثاني: للتبديل بين محتوى صندوقي النص , علينا ان نستخدم مكان وسيط لوضع محتوى الصندوق الاول فيه, ثم وضع محتوى الصندوق الثاني في الصندوق الاول, ثم نضع محتوى المكان الوسيط في الصندوق الثاني, يعتبر هذا الوسيط كوسيلة تخزين مؤقت وهو ماندعوه بالمتغيير Variable.

المتغيير Variable

هو عنوان منطقة محجوزه في الذاكرة , يمكن الكتابة فيها والقراءه منها , يتم التصريح او التعبير عن المتغيير المحلي في داخل الاجراء Procedure باستخدام الامر التالي:

```
Dim VariableName As Type
```


PDF created with pdfFactory trial version www.pdffactory.com

حيث ان VariableName هو اسم المتغيير و Type هو نوع المتغيير, عند تنفيذ البرنامج سوف يتم حجز المكان الموافق للتغيير في الذاكره, وسوف يتم تحرير هذا المكان المحجوز عند انتهاء تنفيذ البرنامج, ومن اهم انواع المتغييرات هي:

INTEGER	عدد صحيح
Long	عدد صحيح طويل
Single	عدد عشري
Double	عدد عشري ذو دقة مضاعفه
String	سلسلة نصية

في مثالنا هذا سوف نعرف متغيرا من نوع سلسلة نصيه String لتخزين محتوى صندوق النص الاول فيه.

```
Dim s As String
```

حيث ان S هو الاسم المعطى للمتغير , ويصبح الاجراء الخاص بالمفتاح Change كالاتي:

```
Private Sub Command1_Click()
Dim s As String
s = Text1.Text
Text1.Text = Text2.Text
Text2.Text = s
End Sub
```

15.برمجة المفتاح Click على النموذج الثاني:-

المطلوب في التمرين عد كم مرة ينقر فيها المستخدم على هذا المفتاح, نحتاج اذا الى متغيير لنضيف اليه واحد في كل مرة يتم النقر فيها على المفتاح Click , لايمكن في هذه

PDF created with pdfFactory trial version www.pdffactory.com

الحالة استخدام الامر Dim في البرنامج لتعريف المتحول المطلوب , أذ أن قيمة المتغيير ستضيع بعد الخروج من البرنامج الخاص بالاجراء , كما انه لايمكن الوصول الى هذا المتغيير في مفتاح No. of Click .

نحتاج في هذه الحالة الى متغير عام على مستوى النموذج اي انه سوف يخلق عند فتح النموذج ويموت عند اغلاق النموذج , كما انه تتشارك جميع البرامج الفرعيه للنموذج.

لتعريف متغيير عام للنموذج يجب وضع الامر Dim للمتغيير في قسم التصريحات للنموذج. يكون هذا القسم فر رأس صفحة البرمجة للنموذج. في مثالنا هذا التغيير العام هو n في قسم التصريحات العامة للنموذج الثاني:

```
Dim n As Integer
```

ونكتب الاجراء التالي على المفتاح Click:

```
Private Sub Command2_Click()
n = n + 1
End Sub
```

16.برمجة المفتاح No. of Click :- يهدف البرنامج الفرعي لهذا المفتاح الى اضهار قيمة المتغيير العام الذي يعد عدد النقرات No. of Click على المفتاح Click, نستخدم الامر Msgbox لاظهار قيمة هذا المتغيير وكمايلي:

```
Private Sub Command3_Click()
MsgBox n
End Sub
```

17.برمجة المفتاح Ok على النموذج الاول:

نريد ان نبرمج على هذا المفتاح كتابة اجراء يختبر كلمة السر المدخلة صحيحه , كما سنسمح للمستخدم المحاولة لثلاث مرات على الاكثر لادخال كلمة السر الصحيحه, اذا نحتاج الى عد هذه المحاولات (اي عدد مرات النقر على مفتاح Ok), يمكن ان نستخدم لذلك متغيرا عاما على مستوى النموذج الاول كما فعلنا في مفتاح Click على النموذج

PDF created with pdfFactory trial version www.pdffactory.com

الثاني , الا ان هذا المتغير لسنا بحاجته في هذه المره الا ضمن اجراء المفتاح Ok , لتعريف متغيير محلي يحافظ على قيمته من استدعاء لاستدعاء اخر.

نستخدم المتغيير المعرف باتخدام الامر Static في الاجراء Procedure بحيث يحافظ على قيمته من استدعاء لاخر. مثال :

```
Static i As Integer
```

18.لاظهار النموذج Form نستخدم الامر Show ولاخفائه نستخدم الامر Hide.

مثلا لاظهار النموذج FrmVar نستخدم الامر التالي:

```
FrmVar.Show
```

يمكن استخدام Me للدلاله على النموذج الحالي. مثال ذلك:

```
Me.Hide
```

تؤدي الى اخفاء النموذج الذي ينفذ فية الامر.

19.الامر SetFocus :- يستخدم لوضع التحديد على عنصر تحكم . مثال ذلك:

```
txtPassword.SetFocus
```

تضع التحديد (مؤشر الماوس على صندوق النص txtPassword

20.الامر SendKeys يستخدم لاسال مجموعه من الاحرفل للنافذه النشطه او الفعاله حاليا كما لو كنا كتبناه من لوحة المفاتيح, مثال

```
SendKeys "{Home}+{End}"
```

تشبة النقر على المفتاح Home ثم النقر على المفتاحين معا Shift و End , سيؤدي ذلك في مثالنا هذا الى تحديد النص المكتوب في صندوق النص.

البرنامج او الاجراء الذي يكتب عند النقر على مفتاح Ok يكون كالاتي:

```
Private Sub CmdOk_Click()
Static i As Integer
i = i + 1
UserName = TxtUserName.Text
If TxtPassword = "toto" Then
```


PDF created with pdfFactory trial version www.pdffactory.com

```
Me.Hide
FrmVar.Show
Else
If i = 3 Then
MsgBox "Finished your try", vbCritical, "Entrance"
End
Else
MsgBox "Try again, it's Wrong", vbExclamation, "Error"
TxtPassword.SetFocus
SendKeys "{Home}+{End}"
End If
End If
End Sub
```

21.برمجة المفتاح Cancel على النموذج الاول : نكتب الاجراء التالي:

```
Private Sub CmdCancel_Click()
End
End Sub
```

22.برمجة المفتاح من هو Who is:

نريد عند النقر على مفتاح Who is في النموذج الثاني اضهار اسم المستخدم المدخل في مربع النص txtUserName , في النموذج الاول. نحتاج في هذه الحالة الى متغير عام على مستوى المشروع, اي انه متغير يشارك كافة اجراءات النماذج في المشروع كي نخزن فيه اسم المستخدم من خلال اجراء Ok في النموذج الاول , ثم نظهر محتوى هذا المتغير من خلال اجراء Who is في النموذج الثاني .

لتعريف متغير عام على مستوى المشروع كلة نعرفه في وحدة نمطيه Module باستخدام الامر Public وكمايلي :-

PDF created with pdfFactory trial version www.pdffactory.com

Public Username As String

سوف تتشارك كافة اجراءات النماذج في المشروع بالمتغير UserName.

سابقا كتبنا اجراء المفتاح Ok على النموذج الاول الامر التالي:

Username = txtUserName.Text

لتخزين اسم المستخدم عند الدخول.وبالتالي يكفي ان نكتب على مفتاح Who is الاجراء التالي:

Private Sub Command4_Click()

MsgBox UserName

End Sub

والذي سيؤدي تنفيذه الى اظهار صندوق رسالة يحتوي على محتوى المتغير txtUserName.

البرنامج كاملا مدون ادناه:-

الوحده النمطية Module

Public UserName As String

النموذج الاول

Private Sub CmdCancel_Click()

End

End Sub

Private Sub CmdOk_Click()

Static i As Integer

i = i + 1

UserName = TxtUserName.Text

If TxtPassword = "toto" Then

Me.Hide

PDF created with pdfFactory trial version www.pdffactory.com

```
FrmVar.Show
Else
If i = 3 Then
MsgBox "Finished your try", vbCritical, "Error"
End
Else
MsgBox "Try again, it's Wrong", vbExclamation, "Error"
TxtPassword.SetFocus
SendKeys "{Home}+{End}"
End If
End If
End Sub
```

النموذج الثاني

```
Dim n As Integer

Private Sub Command1_Click()
Dim s As String
s = Text1.Text
Text1.Text = Text2.Text
Text2.Text = s
End Sub


Private Sub Command2_Click()
n = n + 1
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

```
Private Sub Command3_Click()
MsgBox n
End Sub

Private Sub Command4_Click()
MsgBox UserName
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع السادس

-:----------

اوامر التكرار

سوف نتعلم من خلال هذا المشروع مايلي:-

- **تعليمات التكرار الاساسية.**
- **For .. Statement**
- **Do .. While Statement**
- **Do .. Until Statement**
- **الطباعة على النموذج (Print)**
- **الطباعه على ا لطابعة (Printer)**

الهدف من المشروع

-:----------

عند تنفيذ هذا المشروع سوف تظهر الواجهة التالية:

PDF created with pdfFactory trial version www.pdffactory.com

يؤدي النقر على احد المفاتيح الاربعة WithOut, For, Do While, Do Until الى كتابة الارقام من 0 الى 15 بخط ذي حجم متزايد وكل رقم بلون معين.

كذلك يؤدي النقر على مفتاح جدول الضرب 5 (Multi Table 5) الى كتابة جدول الضرب الى حد 10 * 5 والذي يساوي 50.

اما مفتاح جدول الضرب Multi table يكتب على النموذج جداول الضرب العشره وكل جدول بلون يختلف عن الاخر.

عند النقر على مفتاح الطابعة Printer سيتم طباعة جدول الضرب على الطابعة الموصولة مع الحاسبة الالكترونية.

بينما يؤدي النقر على مفتاح Clear الى مسح النموذج , وكذلك مفتاح Exit يدل على الخروج من التنفيذ الكلي.

مراحل التنفيذ

----------:-

- **نفتح مشروعا قياسيا جديدا.**
- **ننشيء تسعة مفاتيح ونضبط خاصية العنوان Caption لها كمايلي:-**

Without, For, Do While, Do Until, Multi table 5, Multi Table, Printer, Clear, Exit.

- **الامر Cls : - يقوم بمسح النموذج.**
- **الامر Print:- يستخدم للكتابة على النموذج.**
- **الامر Me:- للدلالة على النموذج الحالي.**
- **الخاصية FontSize تحدد حجم الخط للكتابة.**
- **الخاصة ForeColor تحدد لون خط الكتابة.**
- **الدالة QBColor(Color_Number) تعطي عدد صحيحا من Long يمثل اللون الذي رقمة *Number_color*, ترقم الارقام من 0 الى 15 كمامبين في الجدول التالي:**

PDF created with pdfFactory trial version www.pdffactory.com

0	BLACK	8	GRAY
1	Blue	9	Light Blue
2	Green	10	Light Green
3	Cyan	11	Light Cyan
4	Red	12	Light Red
5	Magenta	13	Light Magenta
6	Yellow	14	Light Yellow
7	White	15	Bright White

- يكتب الاجراء التالي على المفتاح بدون تكرار WithOut لطباعة الارقام من 0 الى 15 بحيث يتزايد حجم الخط من 10 فصاعدا ويكون كل سطر بلون مختلف.

```
Private Sub Command1_Click()
Me.Cls
Me.FontSize = 10 + 0
Me.ForeColor = QBColor(0)
Me.Print 0

Me.FontSize = 10 + 1
Me.ForeColor = QBColor(1)
Me.Print 1

Me.FontSize = 10 + 2
Me.ForeColor = QBColor(2)
Me.Print 2

Me.FontSize = 10 + 3
Me.ForeColor = QBColor(3)
Me.Print 3
Me.FontSize = 10 + 4
```


PDF created with pdfFactory trial version www.pdffactory.com

```
Me.ForeColor = QBColor(4)
Me.Print 4
Me.FontSize = 10 + 5
Me.ForeColor = QBColor(5)
Me.Print 5

Me.FontSize = 10 + 6
Me.ForeColor = QBColor(6)
Me.Print 6

Me.FontSize = 10 + 7
Me.ForeColor = QBColor(7)
Me.Print 7

Me.FontSize = 10 + 8
Me.ForeColor = QBColor(8)
Me.Print 8

Me.FontSize = 10 + 9
Me.ForeColor = QBColor(9)
Me.Print 9

Me.FontSize = 10 + 10
Me.ForeColor = QBColor(10)
Me.Print 10

Me.FontSize = 10 + 11
Me.ForeColor = QBColor(11)
Me.Print 11
Me.FontSize = 10 + 12
```


PDF created with pdfFactory trial version www.pdffactory.com

```
Me.ForeColor = QBColor(12)
Me.Print 12

Me.FontSize = 10 + 13
Me.ForeColor = QBColor(13)
Me.Print 13

Me.FontSize = 10 + 14
Me.ForeColor = QBColor(14)
Me.Print 14

Me.FontSize = 10 + 15
Me.ForeColor = QBColor(15)
Me.Print 15

End Sub
```

لاحظ اننا قمنا بتكرار الاوامر التالية :

```
Me.FontSize = 10 + i
Me.ForeColor = QBColor(i)
Me.Print i
```

من اجل ان نجعل *i* تبدأ من الصفر وتتزايد كل مرة بقيمة مقدارها واحد الى ان تصل الى القيمة 15

توجد في لغة البرمجة Visual Basic عدد من اوامر التكرار , سوف نعالج ثلاثة من جمل التكرار في هذا المثال بهدف استعراض وفهم هذه الحلقات التكرارية.

- **برمجة المفتاح For**

تاخذ الحلقة التكرارية For الشكل التالي:

PDF created with pdfFactory trial version www.pdffactory.com

For *counter* = *start* to *end* [Step]

[*statements*]

…

[*statements*]

Next [*counter*]

الاقواس المربعة لاتستخدم في البرمجة بل تستخدم في الشكل العام للامر للدلالة على ان جزءا من الامر يكون خياري اي يمكن كتابتة او تركة.

حيث ان:-

- **Counter هو متغير يدعى بالعداد.**
- **Start القيمة الابتدائية للعداد.**
- **End القيمة النهائية للعداد.**
- **Step هو مقدار الزيادة او النقصان لقيمة العداد**

ملاحظة

-:----------

يمكن ان يكون Step موجبا او سالبا. أذا كان موجبا فأن Start يجب ان يكون اصغر اويساوي End والا فان تعليمات الحلقة لن تنفذ. امام أذا كان سالبا فان start يجب ان يكون اكبر او يساوي كي تنفذ تعليمات الحلقة التكرارية, أذا لم يحدد مقدار الزيادة او النقصان عن طريق الايعاز step فان كل لغات البرمجة تعتبر الزياده بمقدار وحدة واحدة فقط, البرنامج التالي يقوم بتنفيذ اجراء الحلقة التكرارية For.

```
Private Sub Command2_Click()
Dim i As Integer
   Me.Cls
For i = 0 To 15
   Me.FontSize = 10 + i
   Me.ForeColor = QBColor(i)
   Me.Print i
Next i
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- برمجة المفتاح Do While

تأخذ جملة التكرار Do .. While الشكل العام التالي:

```
Do While Condition
[Statement]
…
[Statement]
Loop
```

عندما يصل Visual Basic لتنفيذ هذه الحلقة التكرارية فانه سيبدأ اولا باختبار الشرط *Condition* فاذا وجده غير محقق *False* فانه سيقفز فورا لما بعد سطر Loop اي لن ينفذ اوامر الحلقة التكرارية *Statements* اما اذا وجده *True* فأنه سينفذ التعليمات *Statements* ثم يعود الى سطر Do While ليختبر الشرط ثانية ويعاود التنفيذ اذا كان الشرط *True* او يخرج من الحلقة التكرارية اذا كان الشرط *false* وهكذا, وبالتالي فان تعليمات الحلقة التكرارية ستنفذ عدد من المرات طالما بقي الشرط True ولن تنفذ ابدا اذا كان الشرط منذ البداية False .

الاجراء التالي الخاص بهذه الحلقة التكرارية موضح كمايلي:-

```
Private Sub Command3_Click()
Dim i As Integer
   Me.Cls
Do While i <= 15
   Me.FontSize = 10 + i
   Me.ForeColor = QBColor(i)
   Me.Print i
   i = i + 1
Loop
End Sub
```

- برمجة المفتاح Do Until

تأخذ الحلقة التكرارية Do .. until الشكل العام التالي:

PDF created with pdfFactory trial version www.pdffactory.com

Do until *condition*

[*statements*]

…

[*statements*]

Loop

يقوم Visual Basic بتكرار هذه الحلقة حتى يصبح الشرط محققا.

الاجراء التالي يقوم بتنفيذ هذه الحلقة التكرارية.

```
Private Sub Command4_Click()
Dim i As Integer
   Me.Cls
   i = 0
Do Until i = 15
   Me.FontSize = 10 + i
   Me.ForeColor = QBColor(i)
   Me.Print i
   i = i + 1
Loop
End Sub
```

- **برمجة المفتاح 5 Multi table :- يطبع هذا المفتاح جدول الضرب لحد 5 على النموذج , حيث نستخدم الحلقة لتكرارية For لذلك. الاجراء التالي يحقق الجدول.**

```
Private Sub Command5_Click()
Me.Cls
Dim i As Integer
Me.FontSize = 12
Me.ForeColor = QBColor(0)
For i = 1 To 10
      Me.Print i & " * 5 = " & i * 5
Next i
End Sub
```

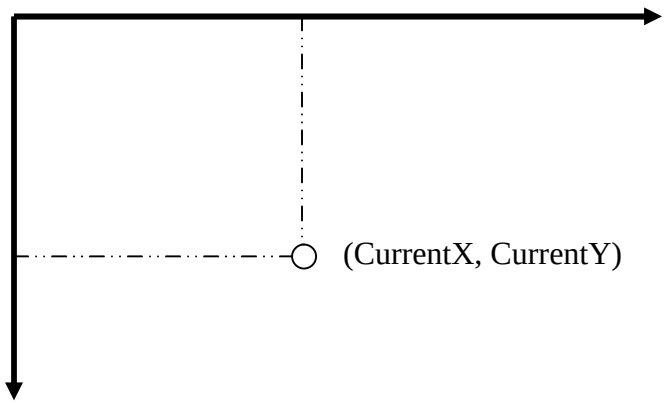

PDF created with pdfFactory trial version www.pdffactory.com

- **برمجة المفتاح Multi Table :-** يطبع هذا المفتاح جدول الضرب على النموذج , بحيث يكون كل جدول بلون معين. الاجراء التالي يقوم بتنفيذ هذه العملية.

```
Private Sub Command6_Click()
Dim i As Integer, j As Integer
Me.Cls
Me.FontSize = 10
For i = 1 To 10
   Me.ForeColor = QBColor(i)
   For j = 1 To 10
      Me.CurrentX = 800 * i
      Me.CurrentY = 500 * j
      Me.Print j & " * "; i & " = " & i * j
   Next j
Next i
End Sub
```

- **تصميم محاور احداثيات النموذج**

تصمم محاور احداثيات النموذج كمايلي:



PDF created with pdfFactory trial version www.pdffactory.com

للكتابة في مكان معين على النموذج يجب تحديد نقطة محور X عن طريق النقطة CurrentX حيث تتراوح قيم هذا المحور بين الصفر وعرض النموذج width ونقطة محور Y عن طريق النقطة CurrentY التي تتراوح بين الصفر وارتفاع النموذج Height.

يستخدم مصطلح Twip كوحدة قياس اساسية في VB حيث ان كل 576 تويب تعادل واحد سنتيمتر , في مثالنا هذا نعتبر ان المسافة الافقية الفاصلة بين جدول الضرب واخر تساوي 800 تويب وان المسافة العمودية بين سطر واخر تساوي 500 تويب.

نحتاج في الاجراء الذي يقوم بتنفيذ جدول الضرب الى حلقتين تكراريتين : حلقة تكرارية i من الواحد الى العشرة لجدول الضرب العشرة , ومن اجل كل جدول ضرب i نحتاج الى j من الواحد الى العشرة لطباعة اسطر جدول ضرب i العشرة. يكون الاجراء بالشكل التالي:

```
Private Sub Command6_Click()
Dim i As Integer, j As Integer
Me.Cls
Me.FontSize = 10
For i = 1 To 10
   Me.ForeColor = QBColor(i)
   For j = 1 To 10
      Me.CurrentX = 800 * i
      Me.CurrentY = 500 * j
      Me.Print j & " * "; i & " = " & i * j
   Next j
Next i
End Sub
```

- برمجة مفتاح الطبع:-

يطبع هذا المفتاح جدول الضرب على الطابعة , بحيث يكون كل جدول بلون, الاجراء الخاص بتنفيذ هذا الامر موضح كالاتي:-

لاحظ اوجة التشابهه مع اجراء جدول الضرب.

PDF created with pdfFactory trial version www.pdffactory.com

```
Private Sub Command7_Click()
Dim i As Integer, j As Integer
Printer.FontSize = 10
For i = 1 To 10
   Printer.ForeColor = QBColor(i)
   For j = 1 To 10
      Printer.CurrentX = 800 * i
      Printer.CurrentY = 500 * j
      Printer.Print j & " * "; i & " = " & i * j
   Next j
Next i
End Sub
```

يستخدم ايعاز الطبع Printer للقيام بجميع عمليات الرسم والطباعة على الطابعة, يقبل النموذج جميع الطرق المعرفة على الكائن Printer , لذا فاننا نقوم عمليا بكتابة اجراءات الرسم والطباعة على النموذج اولا, وبعد الوصول الى الشكل النهائي المطلوب نستبدل اسم النموذج في الاجراءات ب Printer وكما موضح في البرنامج الخاص بالطبع اعلاه.

من اهم خصائص الكائن Printer موضحة في الجدول التالي:

CurrentX	الاحداثيات الافقية
CurrentY	الاحاثيات العمودية
FontName	اسم خط الكتابة
FontSize	حجم خط الكتابة
FontCount	عدد الخطوط المتاحة
Fonts (n)	اسم الخط ذي الرقم n حيث ترقم اسماء الخطوط من 0 الى FontCount-1
Orientation	اتجاه صفحة الطباعة : 1 راسي Portrait و 2 Landscape

PDF created with pdfFactory trial version www.pdffactory.com

ومن اهم طرق الكائن Printer هي:

Circle (X, Y), R, C	رسم دائرة
Line (X1, Y1)-(X2,Y2)	رسم خط
Pset (X,Y),C	رسم نقطة
Print	الطباعة عند الاحداثيات المحددة
printPicture	نقل صورة للطابعة
EndDoc	اخراج الصفحة من الطابعة بعد الطباعة عليها

- **مفتاح المسح Clear**

يستخدم الاجراء التالي لمسح النموذج

```
Private Sub Command9_Click()
Me.Cls
End Sub
```

- **مفتاح الخروج Exit**

```
Private Sub Command8_Click()
End
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع السابع

-:----------

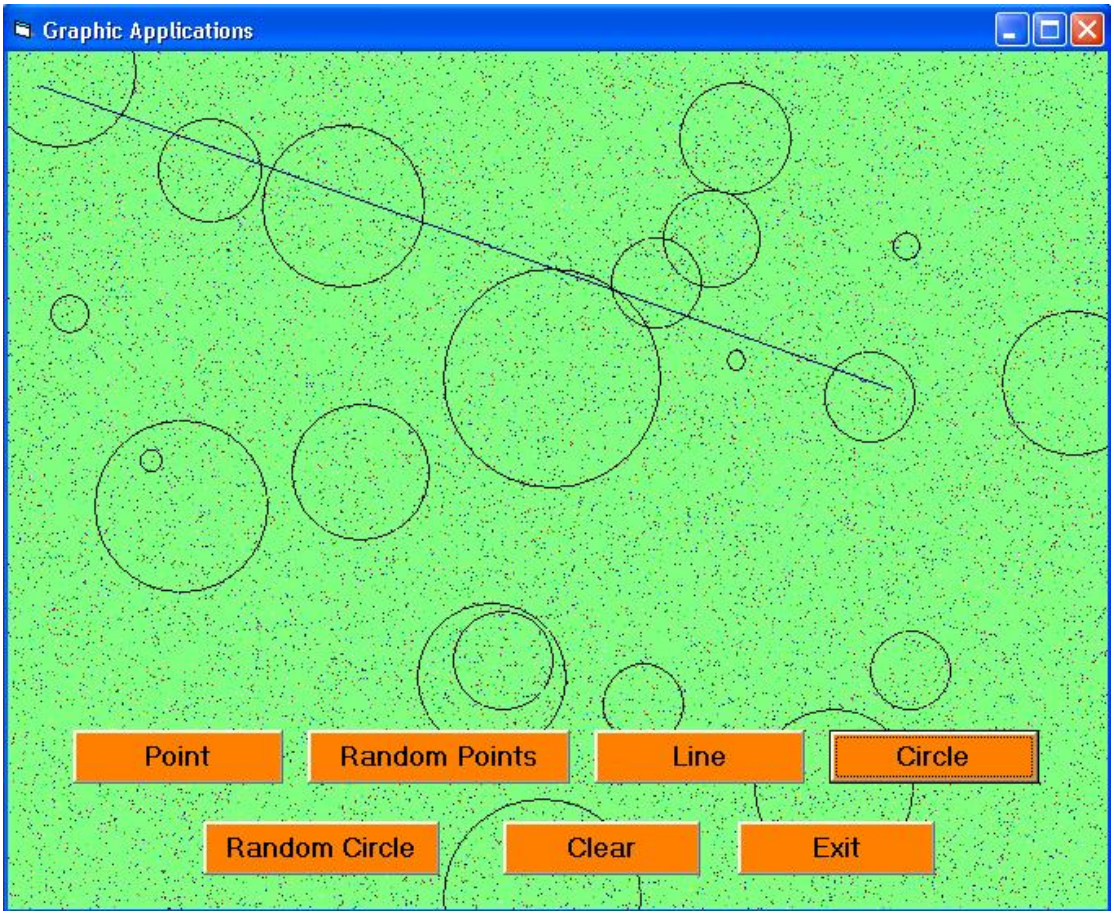
اسم المشروع : (رسم النقطة , النقاط العشوائية, الخط, الدائره, الدوائر العشوائية)

سوف نتعلم من خلال هذا المشروع مايلي:

- رسم النقطة.
- رسم النقاط العشوائية.
- رسم الخط.
- رسم الدائره.
- رسم الدوائر العشوائية.

الهدف من المشروع:

عند تنفيذ هذا المشروع تظهر الواجهة التالية:



PDF created with pdfFactory trial version www.pdffactory.com

مراحل التنفيذ

-:----------

- **ننشي نموذجا قياسيا جديدا, ونضبط خصائصة كمايلي:**

FORM1 PROPERTIES	
Name	**Graphics_1**
BorderStyle	**2-Sizable**
Caption	**Graphic Applications**
BackColor	**White Green اخضر فاتح**
Height	**8400**
Width	**10240**
Window State	**0 – Normal**

- **ننشي سبعة مفاتيح اوامر Command Buttons ونضبط الخصائص لها كمايلي:**

COMMAND BUTTONS 1	
Name	**Point**
BackColor	**Orange**
Caption	**Point**

COMMAND BUTTONS 2	
Name	**CmdRandPoint**
BackColor	**Orange**
Caption	**Random Point**

COMMAND BUTTONS 3	
Name	**CmdLine**
BackColor	**Orange**
Caption	**Line**

PDF created with pdfFactory trial version www.pdffactory.com

command Buttons 4	
Name	**CmdCircle**
BackColor	**Orange**
Caption	**Circle**

COMMAND BUTTONS 5	
Name	**Command1**
BackColor	**Orange**
Caption	**Random Circle**

COMMAND BUTTONS 6	
Name	**CmdClear**
BackColor	**Orange**
Caption	**Clear**

COMMAND BUTTONS 7	
Name	**CmdExit**
BackColor	**Orange**
Caption	**Exit**

- **البرنامج الخاص بالرسومات موضح كمايلي:**

```
Private Sub CmdCircle_Click()
Graphics_1.AutoRedraw = True
Circle (5000, 3000), 1000
End Sub

Private Sub CmdClear_Click()
Graphics_1.Cls
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

```
Private Sub CmdExit_Click()
End
End Sub
Private Sub CmdLine_Click()
Graphics_1.AutoRedraw = True
Line (300, 310)-(8140, 3100), QBColor(1)
End Sub

Private Sub CmdPoint_Click()
Graphics_1.AutoRedraw = True
PSet (1500, 600), QBColor(4)
End Sub

Private Sub CmdRandPoint_Click()
Dim r, x, y, c As Integer
For r = 1 To 20000
   x = Rnd * Graphics_1.ScaleWidth
   y = Rnd * Graphics_1.ScaleHeight
   c = Rnd * 15
   PSet (x, y), QBColor(c)
Next r
End Sub

Private Sub Command1_Click()
Dim r, x, y, c As Integer
For r = 1 To 20
   x = Rnd * Graphics_1.ScaleWidth
   y = Rnd * Graphics_1.ScaleHeight
   c = Rnd * 1000
   Circle (x, y), c
Next r
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الثامن

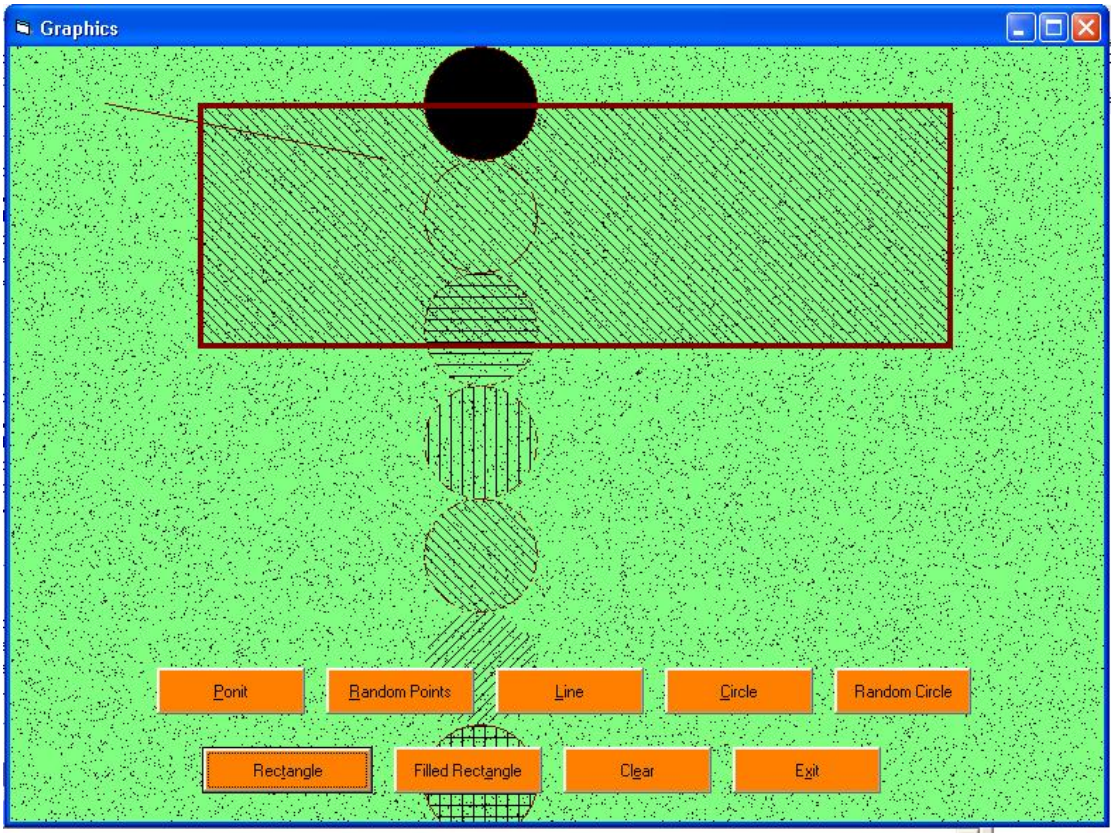
-:----------

(رسم الاشكال الهندسية مع امكانية تغير سمك الخط, لون الخط, مليء الشكل)

الهدف من المشروع

-:----------

عند تنفيذ المشروع تظهر الواجهه التاليه:



اوامر الرسوم المباشرة

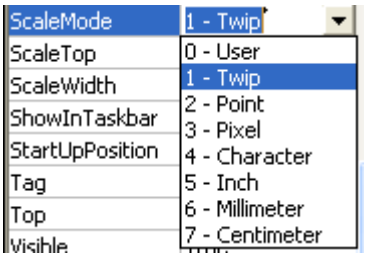
-:----------

تحتوي Visual Basic على مجموعه كبيره من الاوامر التي تمكننا من رسم الاشكال المختلفة فوق النموذج, وقبل ان نبدأ في معرفة هذه الاوامر علينا اولا معرفة احداثيات النموذج, عند بداية تشغيل Visual Basic هناك وحدة افتراضية خاصة لقياس احداثيات النموذج تسمى TWIP وكل انج واحد يعادل 1440 TWIP, من الممكن ان تسبب لك هذه الوحدة بعض الازعاجات لعدم ضبط حدود الرسم لذلك يمكن تغييرها الى وحدات قياس اخرى كما مدون في الجدول التالي:

PDF created with pdfFactory trial version www.pdffactory.com

الوحـــــــده	الشرح
TWIP	ويحتوي كل انج على 1440 TWIP
POINTS	ويحتوي كل انج على 72 POINTS
PIXELS	تعتبر هذه الوحده من اكثر الوحدات شيوعا واستخداما منذ بداية التعامل مع لغات البرمجة المختلفة, ويتحدد عدد النقاط PIXELS التي يحتويها النموذج على دقة الشاشة Resolution
CHARACTER	حرف
INCHES	انج
MILIMETERS	مليمتر
CENTIMETERS	سنتميمتر

نتكمن من ضبط وحدات القياس المطلوبة من خلال استخدام الخاصية Scale Mode الموجوده ضمن خصائص النموذج الموضحه في الشكل التالي:



- الامر PSet

يقوم هذا الامر او الايعاز برسم نقطة على النموذج بالاحداثيات التي تحددها انت.

الصيغة العامه لها:

Pset (x, y), Color

حيث ان x هي الاحداثي الافقي لمكان النقطة على النموذج.

وان y هي الاحداثي العمودي لمكان النقطة على النموذج.

Color هو لون النقطه ويمكن تغيير لون النقطة عن طريق الدالة QBColor(*i*)

حيث ان *i* يمثل لون النقطه من 0 الى 15 لون كما موضح في الجدول التالي:

PDF created with pdfFactory trial version www.pdffactory.com

الرقم	اللون
0	الاسود
1	الازرق
2	الاخضر
3	البني
4	الاحمر
5	البرتقالي
6	الاصفر
7	الابيض
8	الرمادي
9	الازرق الفاتح
10	الاخضر الفاتح
11	السماوي
12	الاحمر الفاتح
13	الروز
14	الاصفر الفاتح
15	الابيض الناصع

- الدالة Rnd لتوليد النقاط العشوائية اي Random.
- الامر Line

يقوم هذا الامر او الايعاز برسم خط على النموذج بالاحداثيات التي تحدد لرسمه

الصيغة العامة له :

Line (x1, y1) - (x2, y2), QBColor(i)

- لرسم مربع او مستطيل نستخدم ايعاز Line مع اضافة الحرف B للدلالة على انه Box وحسب الاحداثيات المعطاة له, الصيغه العامه له:

Line (x1, y1) - (x2, y2), QBColor(i), B

واذا اردنا الصندوق مملوء نستخدم الايعاز BF حيث ان F للدلالة على انه Filled اي مملوء

- ايعاز رسم الدائرة: نستخدم الايعاز Circle لرسم الدائره, الصيغه العامة له هي:

Circle (x, y), Radius, QBColor(i)

حيث ان Radius هي نصف قطر الدائرة.

PDF created with pdfFactory trial version www.pdffactory.com

- **الخاصيه FillStyle**

تستخدم هذه الخاصيه لتحديد الشكل الداخلي للشكل المرسوم سواء كان دائرة او مستطيل, القيم الموضحه في الجدول التالي تبين القيم المستخدمة لتحديد الشكل الداخلي للرسم:

الرقم	الاسم
0	Solid معتمه
1	Transparant شفافه
2	Horizontal Lines خطوط افقية على طول الشكل
3	Vertical Lines خطوط عمودية على طول الشكل
4	UpWard Diagonal Lines خطوط مائلة للاعلى
5	DownWard Diagonal Lines خطوط مائلة للاسفل
6	خطوط متقاطعه Cross Hatch
7	خطوط متقاطعه مائلة Diagonal Cross Hatch

- **الخاصية DrawStyle**

تستخدم هذه الخاصية لتحديد شكل الخط المرسوم , القيم الموضحه في الجدول التالي تبين شكل الخط المطلوب:

الرقم	الاسم
0	خط متصل Solid
1	خط متقطع Dash
2	خط يتكون من مجموعة نقط Dot
3	خط يتكون من خط يليه نقطه Dash Dot
4	خط يتكون من خط يليه نقطتين Dash Dot Dot
5	خط شفاف غير مرئي Transparant
6	خط ممتليء من الداخل Inside Solid

- **الخاصية DrawWidth**

تحدد هذه الخاصيه سمك الخط للرسم

- **كل الخواص الموضحه في الاعلى تم استخدامها في البرنامج, يجب علينا ان نمعن في كيفية استخدامها داخل البرنامج وبالتالي نكون ملزمين عن فهم طريقة استخدامها .**

PDF created with pdfFactory trial version www.pdffactory.com

البرنامج التالي يقوم بتنفيذ شكل النموذج الموجود في البداية:

```
Private Sub CmdCircle_Click()
Dim r, x, y, c, z
z = 0
For r = 0 To 6
   FrmGraphics.FillStyle = r
   FrmGraphics.DrawStyle = r
   x = 5000
   y = 600 + z
   Circle (x, y), 600, QBColor(4)
   z = z + 1200
Next r
End Sub

Private Sub CmdClear_Click()
FrmGraphics.Cls
End Sub

Private Sub CmdExit_Click()
End
End Sub

Private Sub CmdLine_Click()
Line (1000, 600)-(4000, 1200), QBColor(4)
End Sub

Private Sub CmdPoint_Click()
Rem FrmGraphics.DrawWidth = 4
PSet (1000, 600), QBColor(4)
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

```
 Private Sub CmdRndCircle_Click()
Dim r, x, y, c As Integer
For r = 1 To 20
   x = Rnd * FrmGraphics.ScaleWidth
      y = Rnd * FrmGraphics.ScaleHeight
      c = Rnd * 1000
   Circle (x, y), c
Next r
End Sub

Private Sub CmdRndPoints_Click()
Dim r, x, y, c
For r = 1 To 20000
   x = Rnd * FrmGraphics.ScaleWidth
   y = Rnd * FrmGraphics.ScaleHeight
   PSet (x, y), QBColor(c)
Next r
End Sub

Private Sub Command1_Click()
FrmGraphics.DrawWidth = 4
FrmGraphics.FillStyle = 4
Line (2000, 600)-(10000, 3200), QBColor(4), B
End Sub

Private Sub Command2_Click()
FrmGraphics.DrawWidth = 4

Line (2000, 600)-(10000, 3200), QBColor(4), BF
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع التاسع

-:----------

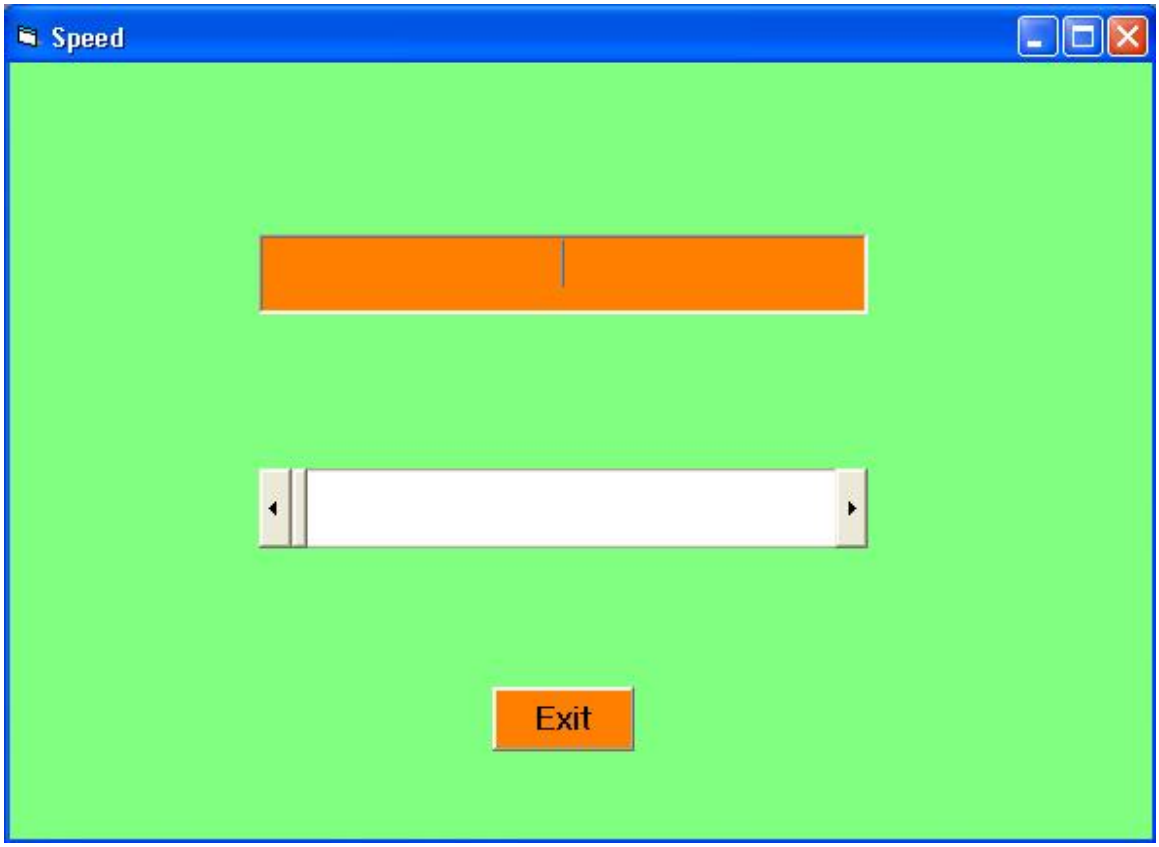
اسم المشروع : محاكاة السرعة Speed

سوف نتعلم من خلال هذا المشروع استخدام الكائن Horizontal Scroll Bar اي شريط الطي الافقي

الهدف من المشروع

-:----------

عند تنفيذ هذا المشروع سوف تظهر الواجهة التالية:



PDF created with pdfFactory trial version www.pdffactory.com

خطوات تنفيذ المشروع

-:----------

- **ننشيء مشروعا قياسيا جديدا ونضبط خصائصه كمايلي:**

FORM1 PROPERTIES	
Name	Form1
BorderStyle	2-Sizable
Caption	Speed
BackColor	White Green اخضر فاتح
Height	6480
Width	8925
Window State	0 – Normal

- **ننشي عنصر واحد من نوع Text Box ونضبط خصائصه كمايلي:**

TEXT BOX1	
Name	TxtSpeed
Alignment	2 – Center
BackColor	Orange
Text	

- **ننشيء عنصر من نوع HScrollBar اي شريط الطي الافقي ونضبط اسمه HSpeedScroll**
- **ننشيء عنصر من نوع Command Button ونضبط عنوانة ب Exit**
- **عند النقر على Text Box نكتب البرنامج التالي:**

```
Private Sub TxtSpeed_Change()
TxtSpeed.Text = Str(HSpeedScroll.Value) + "  km/hour"
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- **عند النقر على مفتاح شريط الطي الافقي نكتب الاجراء التالي:**

```
Private Sub HSpeedScroll_Change()
TxtSpeed_Change
End Sub
```

- **عند النقر على مفتاح Exit نكتب الاجراء التالي:**

```
Private Sub Command1_Click()
End
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع العاشر

-:----------

(حساب الفائدة المركبة في البنك)

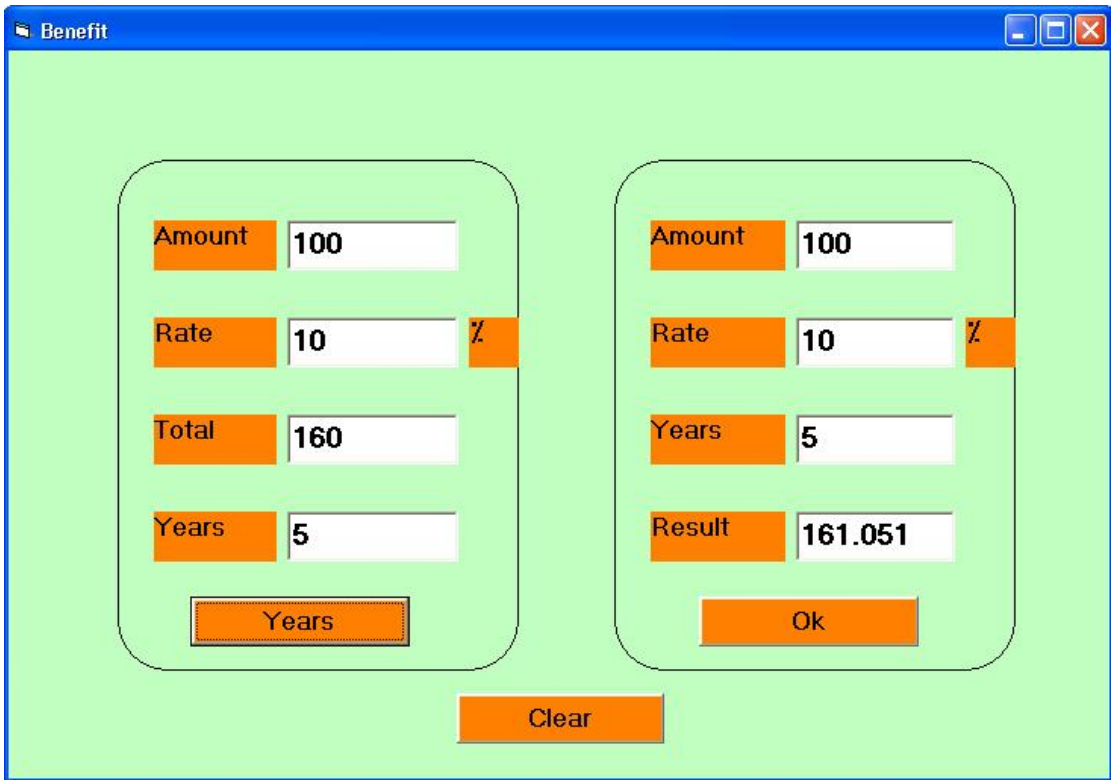
سنتعلم من خلال هذا المشروع:

- **الحلقات التكرارية Do .. While , For والفرق بينهما:**
- **حساب الفائدة المركبة**

الهدف من المشروع

-:----------

عند تنفيذ هذا المشروع تظهر الواجهة التالية:



تسمح هذه الواجهة باجراء بعض حسابات الفائدة المركبة:

- **في القسم الايمن من الواجهة اعلاه يقوم المستخدم بادخال المبلغ والفائدة السنوية وعدد السنوات, وعند النقر على المفتاح Ok سيتم حساب المبلغ الناتج بعد مرور السنوات المحددة.**

PDF created with pdfFactory trial version www.pdffactory.com

- في القسم الايسر يقوم المستخدم بادخال المبلغ والفائدة لسنوية والمبلغ المطلوب, وعند النقر على المفتاح Years سيتم حساب عدد السنوات اللازم كي يصبح المبلغ اكبر او يساوي المبلغ المطلوب.

مراحل التنفيـــــــــــذ

----------:-

- نفتح مشروعا قياسيا جديدا.
- ننشيء عنصرين من النوع Shape على شكل مستطيل ذي حواف مستديرة, نضع عناصر كل مسألة في مستطيل.

يستخدم عنصر التحكم Shape لرسم شكل هندسي على النموذج, وتحدد الخاصية Shape للشكل الهندسي وكمايلي:

0 - RECTANGLE	مستطيل
1 – Square	مربع
2 – Oval	قطع ناقص
3 – Circle	دائرة
4 – Rounded Rectangle	مستطيل ذو حواف مستديرة
5 – Rounded Square	مربع ذو حواف مستديرة

- نضبط خصائص النموذج كمايلي:

FORM1 PROPERTIES	
Name	Form1
BorderStyle	2-Sizable
Caption	Benefit
BackColor	White Green اخضر فاتح
Height	7710
Width	11025
Window State	0 – Normal

PDF created with pdfFactory trial version www.pdffactory.com

- ننشيء شكلين من النوع Shape ونضبط الخصائص كمايلي:

SHAPE	
Name	Shape1
Shape	4 – Rounded Rectangle

SHAPE	
Name	Shape2
shape	4 – Rounded Rectangle

- ننشيء اربعة صناديق نص لوضع المبلغ (TxtAmount) والفائدة (TxtRate) وعدد السنين (TxtYears) والمبلغ الناتج (TxtOut) في المستطيل الايمن. نضبط الخصائص كمايلي:

TEXT BOX	
Name	TxtAmount
Text	

TEXT BOX	
Name	TxtRate
Text	

TEXT BOX	
Name	TxtYears
Text	

TEXT BOX	
Name	TxtOut
Text	

PDF created with pdfFactory trial version www.pdffactory.com

- ننشيء مفتاح الاوامر Ok في المستطيل الايمن لاجراء عملية الحساب.
- ننشيء اربعة صناديق نص لوضع المبلغ (TxtAmount1) والفائدة (TxtRate1) وعدد السنين (TxtYears1) والمبلغ الناتج (TxtOut1) في المستطيل الايسر.

TEXT BOX	
Name	TxtAmount1
Text	

TEXT BOX	
Name	TxtRate1
Text	

TEXT BOX	
Name	TxtYears1
Text	

TEXT BOX	
Name	TxtOut1
Text	

- ننشيء مفتاح الاوامر Command Button ونضبط العنوان باسم Years في المستطيل الايسر لاجراء عملية الحساب.
- ننشي التسميات Labels الظاهره في الواجهة ونضبط عناوينها Caption.
- برمجة المفتاح Ok:

لحساب المبلغ الناتج بعد عدة سنوات, يجب ان يكون من اجل كل سنة اضافة الفائدة الى المبلغ الناتج الذي يكون مبلغ اول سنة قادمة (فائدة مركبة او تراكمية) وهكذا لبقية السنين.

اي انة مثلا اذا كان المبلغ هو 100 والفائدة هي 10 وعدد السنوات 5 سوف يتم الحساب كمايلي:

PDF created with pdfFactory trial version www.pdffactory.com

السنة	مبلغ اول سنة	الفائدة	مبلغ اخر سنة
1	100	10=100\10*100	110=10+100
2	110	11=100\10*110	121=11+110
3	121	12,1=100\10*121	133,1=12,1+121
4	133,1	13,31=100\10*133,1	146,41=13,31+133,1
5	146,41	14,641=100\10*146,41	161,051=14,641+146,41

اي انه يجب تكرار العملية التالية بعدد السنوات:

مبلغ نهاية السنة = مبلغ بداية السنة + مبلغ بداية السنة X الفائدة \ 100.

من الانسب في هذه الحالة استخدام تعليمة التكرار For , اذا اردنا ان نعرف عدد مرات التكرار (اي عدد السنوات).

نكتب الاجراء التالي على مفتاح الاوامر Ok وكمايلي:

```
Private Sub Command1_Click()
Dim a As Single
Dim r As Single
Dim y As Integer
Dim b As Single
Dim i As Integer
a = Val(TxtAmount.Text)
r = Val(TxtRate.Text)
y = Val(TxtYears.Text)
b = a
For i = 1 To y
   b = b + b * r / 100
Next i
TxtOut.Text = b
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- **برمجة المفتاح Years :**

في هذه الحالة لدينا المبلغ والفائده , ونريد ان نحسب عدد السنوات اللازمة كي يصبح هذا المبلغ اكبر من مبلغ معطى

اي اننا يجب ان نكرر عملية اضافة الفائدة الى المبلغ الى ان يصل الى اكبر من المبلغ المطلوب, لاحظ اننا لانعرف مسبقا عدد مرات التكرار , وبالتالي لايمكن في هذه الحالة استخدام الحلقة التكرارية For. الحلقة التكراريةالمناسبة في هذه الحالة هي الحلقة التكرارية Do .. While اذا سنكرر عملية اضافة الفائدة الى المبلغ طالما كان اصغر من المبلغ المطلوب, وعندما يصبح المبلغ اكبر او يساوي المبلغ المطلوب سوف تتوقف الحلقة التكرارية وسوف يكون عدد مرات التكرار هو بقدر عدد السنوات.

نكتب الاجراء التالي بعد الضغط على مفتاح الاوامر Years:

```
Private Sub Command2_Click()

Dim a As Single
Dim r As Single
Dim y As Integer
Dim b As Single

a = Val(TxtAmount1.Text)
r = Val(TxtRate1.Text)
b = Val(TxtOut1.Text)
y = 0
Do While a < b
   a = a + a * r / 100
   y = y + 1
Loop
TxtYears1.Text = y
End Sub
```

- **ننشيء مفتاح الاوامر Command Button رقم 3 ونضبط العنوان له ب Clear**

نبرمج هذا المفتاح كمايلي لغرض المسح في حالة اعادة تنفيذ البرنامج مرة اخرى لغرض اجراء عملية حسابات اخرى:

PDF created with pdfFactory trial version www.pdffactory.com

```
Private Sub Command3_Click()
TxtAmount = ""
TxtRate = ""
TxtYears = ""
TxtOut = ""
TxtAmount1 = ""
TxtRate1 = ""
TxtYears1 = ""
TxtOut1 = ""
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الحادي عشر

-:----------

(بعض التطبيقات الرياضيه)

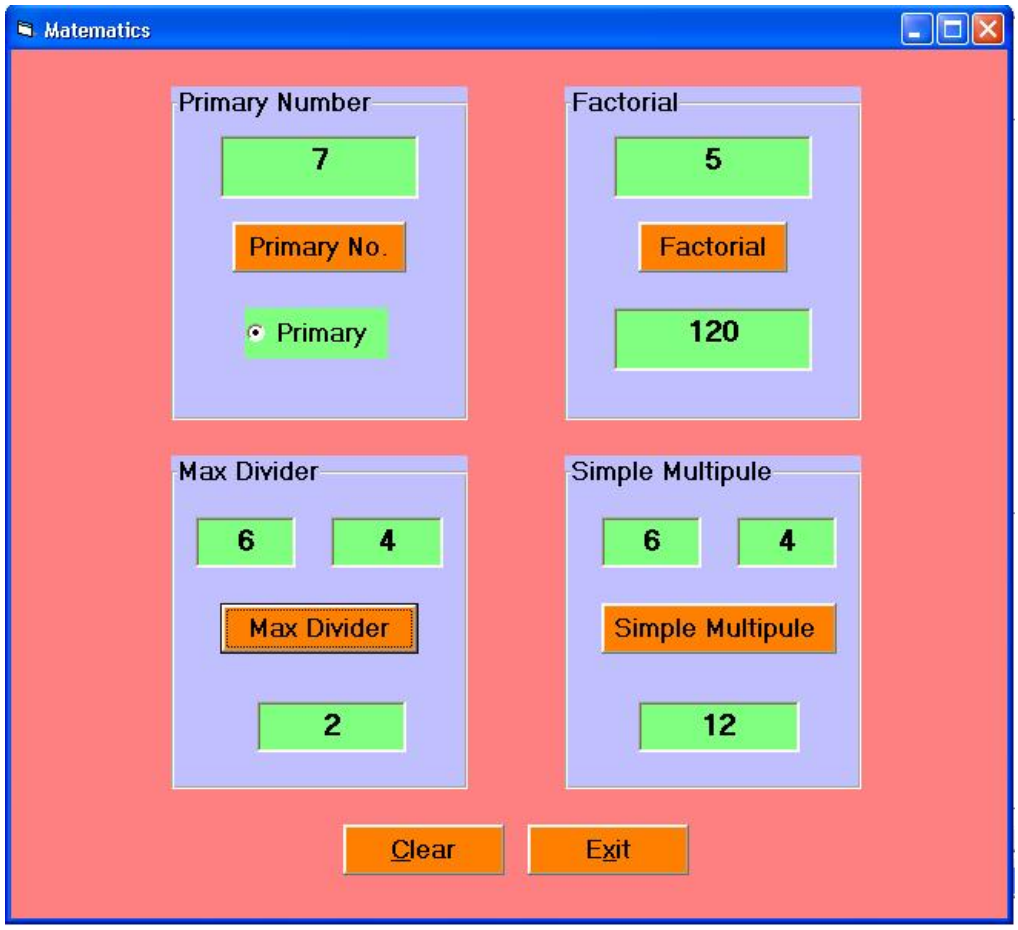
سوف نتعلم من خلال هذا المشروع مايلي:-

- اومر التكرار
- برمجة عدة مسائل رياضية
- حساب المفكوك Factorial
- أختبار هل ان العدد اولي ام لا
- المضاعف المشترك البسيط لعددين
- القاسم المشترك الاعظم لعددين

الهدف من المشروع

-:----------

عند تنفيذ المشروع تظهر الواجهه التالية:



PDF created with pdfFactory trial version www.pdffactory.com

نتمكن من خلال هذه البرنامج حساب مايلي:

1. المفكوك Factorial
2. اختبار هل ان العدد اولي ام لا Primary or not
3. حساب المضاعف المشترك البسيط لعددين Simple Multiple
4. حساب القاسم المشترك الاعظم لعددين Max Divider

مراحل التنفيذ

-:----------

- نفتح مشروعا قياسيا جديدا.
- ننشيء عناصر التحكم الظاهره في النموذج أعلاه.
- برمجة المفكوك Factorial

المفكوك هو حاصل ضرب الاعداد من 1 الى n : 1 * 2 * 3 * ... * n

لحساب المفكوك لعدد n نحتاج الى حلقة تكرارية For من 1 الى n لتراكم ضرب الاعداد من 1 الى n .

نكتب الاجراء التالي الخاص بحساب المفكوك:

```
Private Sub Command1_Click()
Dim n As Integer
Dim nf As Double
Dim i As Integer
n = Val(Text1.Text)
nf = 1
For i = 1 To n
   nf = nf * i
Next i
   Text2.Text = nf
End Sub
```

- برمجة المفتاح Primary No

العدد الاولي هو العدد الذي لايقبل القسمة على اي عدد (ماعدا على الواحد او على العدد نفسه).

لاختبار ان العدد n هو عدد اولي نقسمة على كل الاعداد من 2 الى n – 1

PDF created with pdfFactory trial version www.pdffactory.com

- اذا كان هناك باقي القسمة على جميع الاعداد من 2 الى n - 1 عند ذلك يكون العدد اولي.
- اما اذا قبل n القسمه على احد الاعداد من 2 الى n – 1 يكون العدد n غير اولي.

نكتب الاجراء التالي على المفتاح Primary No.

```
Private Sub Command2_Click()
Dim n As Integer
n = Val(Text3.Text)
Dim i As Integer
Option1.Value = True
For i = 2 To n - 1
   If n Mod i = 0 Then
      Option1.Value = False
Exit For
   End If
Next i
End Sub
```

- برمجة المفتاح Simple Multiple

المضاعف المشترك البسيط لعددين هو اصغر عدد يقبل القسمه على كلا العددين وبالتالي فهو محصور بين اكبر عدد وحاصل ضرب العددين (حاصل ضرب العددين حتما يقبل القسمه عليهما)

مثلا المضاعف المشترك البسيط لعددين 4 و 6 هو 12

الخوارزمية الخاصه بهذا الاجراء موضحه بالشكل التالي:

- ليكن العددان n, m
- نوجد اكبر عدد بينهما وليكن max
- نجري الحلقة التكرارية For i à max to n*m
- في كل تكرار نختبر فيما اذا كان متغير الحلقة i يقبل القسمه على n, m
- اذا قبل i القسمه على n و m يكون المضاعف المشترك الاصغر وعندها ننهي الحلقة التكرارية Exit For
- نكتب الاجراء التالي على مفتاح Simple Multiple

PDF created with pdfFactory trial version www.pdffactory.com

```
Private Sub Command3_Click()
Dim n As Integer
Dim m As Integer
Dim i As Integer
Dim max As Integer
n = Val(Text4.Text)
m = Val(Text5.Text)
If n > m Then max = n Else max = m
   For i = max To n * m
      If (i Mod n = 0) And (i Mod m = 0) Then
   Exit For
End If
   Next i
      Text6.Text = i
End Sub
```

- برمجة مفتاح القاسم المشترك الاعظم Max Divider

القاسم المشترك الاعظم لعددين هو اكبر عدد يقسم كلا العددين بدون باقي وبالتالي فهو محصور بين اصغر عدد بينهما وبين العدد 1 مثلا القاسم المشترك الاعظم بين 4 و 6 هو 2.

لحل هذه المسألة نكتب الخوارزمية التاليه:

- ليكن العددان n و m
- نوجد اصغر عدد بينهما وليكن min
- ننفذ الحلقة التكرارية For من العدد min الى العدد 1

```
For i à min to 1 step – 1
```

- في كل تكرار نختبر فيما اذا كان المتغيير i يقسم كلا العددين n و m
- اذا قسم العدد i العددين n و m يكون القاسم المشترك الاعظم وعندها ننهي الحلقة التكرارية
- نكتب الاجراء التالي على مفتاح Max Divider وكمايلي:

```
Private Sub Command4_Click()
Dim n As Integer
```


PDF created with pdfFactory trial version www.pdffactory.com

```
Dim m As Integer
Dim i As Integer
Dim min As Integer
n = Val(Text7.Text)
m = Val(Text8.Text)
If n < m Then min = n Else min = m
   For i = min To 1 Step -1
      If (n Mod i = 0) And (m Mod i = 0) Then
   Exit For
End If
   Next i
      Text9.Text = i
End Sub
```

- **برمجة المفتاح Clear**

لتنفيذ برمجة المفتاح Clear نكتب الاجراء التالي:

```
Private Sub Command5_Click()
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
Text6.Text = ""
Text7.Text = ""
Text8.Text = ""
Text9.Text = ""
End Sub
```

- **برمجة مفتاح الخروج Exit**

للخروج من البرنامج نكتب الاجراء التالي:

```
Private Sub Command6_Click()
End
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الثاني عشر

-:----------

(دوال الحوار مع المستخدم)

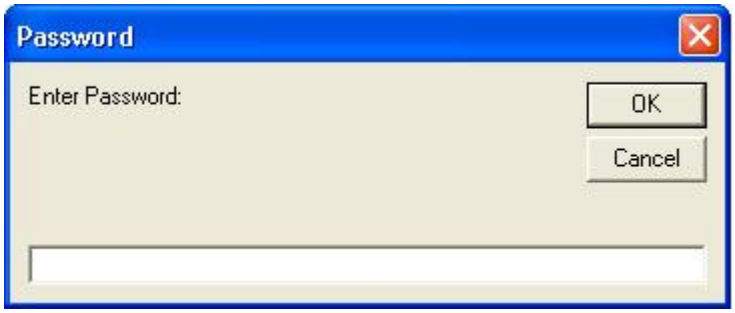
سوف نتعلم من خلال هذا المشروع مايلي:

- **دوال الحوار مع المستخدم.**
- **صندوق الادخال InputBox**
- **صندوق الرسالة MsgBox**
- **الحلقات التكرارية.**

الهدف من المشروع

-:----------

عند تنفيذ هذا المشروع سوف يظهر صندوق الحوار التالي الذي يطلب من المستخدم ادخال كلمة السر.



أذا ادخل المستخدم كلمة السر الصحيحه toto في مثالنا هذا ثم نقر على المفتاح Ok او المفتاح Enter سوف يظهر نموذج المشروع التالي:

PDF created with pdfFactory trial version www.pdffactory.com

أما أذا أخطأ المستخدم في أدخال كلمة السر الصحيحه فسوف يظهر له صندوق الرسالة التالية

- أذا نقر المستخدم على المفتاح Cancel سوف يتم أغلاق البرنامج, اما أذا نقر المستخدم على مفتاح Retry فسوف يظهر صندوق أدخال كلمة السر ليجرب أدخال كلمة السر مرة أخرى.
- يسمح البرنامج للمستخدم ان يخطيء مرتين في كلمة السر, أما اذا أخطأ في المرة الثالثة فسيتم أغلاق البرنامج.
- يحوي نموذج هذا المشروع التعليمي مفتاحا وحيدا لاغلاق البرنامج Exit عند النقر على هذا المفتاح يظهر صندوق الحوار التالي:

- أذا نقر المستخدم على المفتاح No لايحدث أي شيء, أما أذا نقر على مفتاح Yes سيتم أغلاق البرنامج.

مراحل التنفيـــــذ

----------:-

- برمجة الدالة InputBox : تستخدم هذه الدالة لاظهار صندوق الحوار الذي يطلب قيمة من المستخدم أي تنتظر الدالة من المستخدم ان يكتب نصا في صندوق النص او ينقر على احد المفتاحين Ok او Cancel حيث تعيد الدالة سلسلة حرفية String هي عباره عن محتوى صندوق النص, تأخذ الدالة الشكل العام التالي:

InputBox(Prompt[,title][,Default][,xpos][,ypos][,helpfile,context])

PDF created with pdfFactory trial version www.pdffactory.com

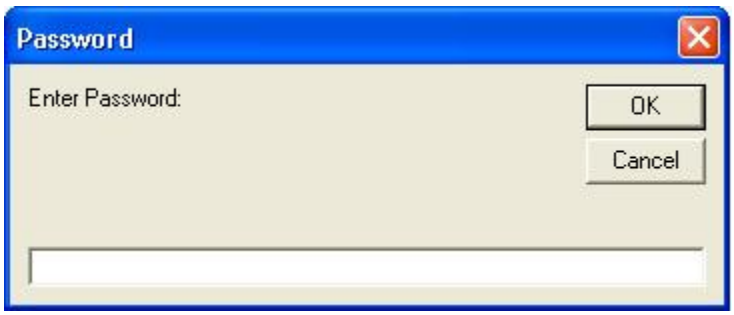
حيث ان

- Prompt تشير الى الرسالة الظاهره في الصندوق.
- Title عنوان الصندوق.
- Default القيمة الافتراضية التي تظهر في صندوق نص نافذة الحوار.
- Xpos, Ypos أحداثيات الصندوق.
- Helpfile, context تستخدم مع ملفات المساعده.

الايعاز التالي يمثل كيفية أستخدام الدالة InputBox ضمن البرنامج لاظهار صندوق الحوار أعلاه:

```
P = InputBox("Enter Password: ", "Password")
```

عند النقر على مفتاح Ok او الضغط على مفتاح Enter فأن القيمة المرجعة للدالة InputBox تكون النص المكتوب في صندوق النص, أما أذا نقر المستخدم على مفتاح Cancel فأن القيمة المرجعه هي لاشيء مهما كان النص الموجود ضمن صندوق النص, وكما موضح في الشكل التالي:



- برمجة الدالة MsgBox تستخدم لاظهار رسالة حوار ضمن صندوق خاص بها , تنتظر هذه الدالة من المستخدم ان ينقر على احد المفاتيح التي يحتويها الصندوق, تعيد هذه الدالة قيمة رقمية Integer تمثل المفتاح الذي تم النقر علية.

تأخذ هذه الدالة الشكل العام التالي:

```
MsgBox(Prompt[,buttons][,title][,helpfile,context])
```

حيث ان :

- Prompt تمثل الرسالة الظاهره في الصندوق.
- Title عنوان الصندوق.
- Helpfile, context تستخدم مع ملفات المساعده.

PDF created with pdfFactory trial version www.pdffactory.com

- **Buttons تعبير رقمي: يمثل مجموع ثوابت رقمية تحدد ايقونة ومفاتيح صندوق الحوار والمفتاح الافتراضي في الصندوق.**

مثلا : لاظهار ايقونة الاستفهام مع المفاتيح (Yes, No, Cancel) بحيث يكون المفتاح الثاني هو المفتاح الافتراضي, يكون هذا التعبير كمايلي:

vbQuestion + vbYesNoCancel + vbDefaultButton2

وكما موضح في الشكل التالي:

```
R = MsgBox("Sure to Exit", vbQuestion + vbYesNoCancel + vbDefaultButton2, "Exit")
```

- **تحدد الثوابت التالية الايقونة الظاهرة في صندوق الحوار عند استخدامها في البرمجة وكمايلي:**

VBQUESTION	**أشارة الاستفهام**
vbExclamation	**أشارة التعجب**
vbCritical	**اشارة الخطأ**
vbInformation	**أشارة المعلومات**

- **تحدد الثوابت التالية مفاتيح صندوق الرسائل MsgBox**

VBOKONLY	**OK**
vbOKCancel	**OK, Cancel**
vbAbortRetryIgnore	**Abort, Retry, Ignore**
vbYesNoCancel	**Yes, No, Cancel**
vbYesNo	**Yes, No**
vbRetryCancel	**Retry, Cancel**

PDF created with pdfFactory trial version www.pdffactory.com

- تحدد الثوابت التالية المفتاح الافتراضي

VBDEFAULTBUTTON1	المفتاح الاول هو المفتاح الافتراضي
VbDefaultButton2	المفتاح الثاني هو المفتاح الافتراضي
VbDefaultButton3	المفتاح الثالث هو المفتاح الافتراضي

- نضغط على النموذج فيظهر حدث تحميل النموذج Form_Load , سوف نكتب أجراء طلب كلمة السر للدخول الى المشروع في هذا الحدث.
- برمجة الحدث Form_Load يحصل عند تحميل النموذج في الذاكره, نحتاج في هذا الاجراء الى حلقة تكرارية لتكرار السؤال عن كلمة السر ثلاثة مرات على الاكثر.
- في حالة معرفة المستخدم لكلمة السر يتم انهاء الحلقة التكرارية.
- أذا ضغط المستخدم على Cancel في نافذة طلب كلمة السر او في نافذة أعادة المحاولة يتم أغلاق البرنامج.
- أذا وصل عدد المحاولات الى ثلاث دون معرفة كلمة السر يغلق البرنامج.
- نكتب الاجراء التالي على حدث تحميل النموذج:

```
Private Sub Form_Load()
Dim P As String
Dim R As Integer 'Retry or Cancel
Dim i As Integer
Do While True
i = i + 1
P = InputBox("Enter Password: ", "Password")
If P = "toto" Then Exit Do
If P = "" Then End
If i = 3 Then End
R = MsgBox("Wrong Password", vbQuestion + vbRetryCancel + vbDefaultButton1, "Error")
'R = MsgBox("Wrong Password", vbExclamation + vbRetryCancel + vbDefaultButton1, "Error")
'R = MsgBox("Wrong Password", vbCritical + vbRetryCancel + vbDefaultButton1, "Error")
'R = MsgBox("Wrong Password", vbInformation + vbRetryCancel + vbDefaultButton1, "Error")
```


PDF created with pdfFactory trial version www.pdffactory.com

```
If R = vbCancel Then End
Loop

End Sub
```

- انشاء مفتاح Exit للخروج من البرنامج

نريد قبل أغلاق البرنامج أظهار صندوق الحوار للتأكد من أن المستخدم يريد أغلاق البرنامج, يحوي هذا الصندوق على مفتاح Yes ومفتاح No وهو المفتاح الافتراضي. لن يتم أغلاق البرنامج الا اذا نقر المستخدم على مفتاح Yes.

- نكتب الاجراء التالي على مفتاح Exit

```
Private Sub Command1_Click()
Dim R As Integer
R = MsgBox("Sure to Exit", vbQuestion + vbYesNo + vbDefaultButton2, "Exit")
'R = MsgBox("Sure to Exit", vbQuestion + vbYesNoCancel + vbDefaultButton2, "Exit")
'R = MsgBox("Sure to Exit", vbQuestion + vbOKCancel + vbDefaultButton2, "Exit")
If R = vbYes Then End
'If R = vbYes Then End
'If R = vbOk Then End
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الثالث عشر

-:----------

مصفوفات عناصر التحكم

سوف نتعلم من خلال هذا المشروع مايلي:

- انشاء مصفوفة من عناصر التحكم.
- القايم بالعمليات التالية على مجموعة من الارقام.
- أيجاد المعدل (الوسط الحسابي).
- أيجاد القيمة العظمى.
- أيجاد القيمة الصغرى.
- الترتيب التصاعدي (الفرز) لمجوعة من الارقام.

الهدف من المشروع

-:----------

عند تنفيذ هذا المشروع تظهر الواجة التالية:

PDF created with pdfFactory trial version www.pdffactory.com

- يمكن للمستخدم ان يدخل ارقاما في صناديق النص, ثم يطلب ايجاد الوسط الحسابي (المعدل Average) والقيمة العظمى (Max) والقيمة الصغرى (MIN) لهذه الارقام. عند النقر على المفتاح تلوين (Color) سوف يتلون العدد الزوجي بلون احمر والعدد الفردي بلون ازرق, عند النقر على مفتاح ترتيب (Ordered) سيتم ترتيب الارقام تصاعديا.

مراحل التنفيذ

-:----------

- نفتح مشروعا قياسيا جديدا.
- ننشيء مصفوفة من (10) مر بعات نص (Text Box):

M(0), M(1), M(2), …, M(9)

- تصميم المصفوفة: وهي عبارة عن مجموعة من عناصر التحكم من نفس النوع, لها نفس الاسم, حيث يكون لكل منهما فهرس (Index) متسلسل يبدأ من الصفر. وبالتالي يمكن لنا استخدام الحلقة التكرارية (For) للتعامل مع جميع عناصر المصفوفة.

لانشاء هذه المصفوفة, نقوم بأنشاء اول عنصر ونسمية ونضبط خصائصة ثم نقوم بعملية نسخة (Copy) ثم نقوم بعملية لصقه (Paste) . مثلا لانشاء مصفوفة من صناديق النص نقوم بمايلي:

- ننشيء صندوق نص , نسمية (M) ونضبط خصائصة (مع ملاحظة ان الخاصية Index له فارغه
- نحدد صندوق النص (M) ومن ثم ننسخه بأختيار القائمة (Edit/Copy).
- نقوم بعملية لصقه (Paste) من خلال القائمة (Edit/Paste), سوف يظهر صندوق حواري يحتوي على مفتاح "No" or "Yes" والرسالة التالية:

You already have a control named 'M'

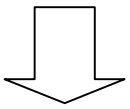
Do you want to create a control array?

- اذا أخترنا "No" فسوف ينشيء صندوق نص لاعلاقة له ب (M).
- أما اذا اخترنا "Yes" تصبح الخاصية (Index) لصندوق النص الاول مساويا الى صفرا.
- ننشيء صندوق نص جديد اسمة (M) عن طريق الاستنساخ واللصق (Copy/Paste) والخاصيه (Index) له مساويا الى العدد (1), وهكذا نكرر العملية حتى صندوق النص M(9) اي بعدد عناصر المصفوفه.

PDF created with pdfFactory trial version www.pdffactory.com

M(0) TEXT BOX	
Name	M
Index	0
Text	

M(0) TEXT BOX	
Name	M
Index	1
Text	



M(0) TEXT BOX	
Name	M
Index	9
Text	

- ننشيء مفتاح لايجاد المعدل (Average) وصندوق نص (TxtAvg).
- ننشيء مفتاح لايجاد العدد الاكبر (Max), وصندوق نص (TxtMax).
- ننشيء مفتاح لايجاد العدد الاصغر (Min) وصندوق نص (TtMin).
- ننشيء مفتاح للتلوين (Color).
- ننشيء مفتاح لاجراء عملية الترتيب (Ordered) .

- **برمجة المفتاح (Average)**

لحساب الوسط الحسابي (المعدل Average) لمجموعة من الاعداد نقسم مجموع تلك الارقام على عددها. عند ذلك نحتاج الى متغير (Avg) لتراكم او جمع الارقام المكتوبة في عناصر المصفوفة, وللوصول الى عناصر تلك امصفوفة نحتاج الى حلقة تكرارية For من صفر الى 9 , لان عناصر المصفوفة العشرة مفهرسة من صفر الى 9 .

PDF created with pdfFactory trial version www.pdffactory.com

- **برمجة الخاصية Count**

تعيد هذه الخاصية عدد عناصر المصفوفة.

حيث ان عدد عناصر المصفوفة 10 بالاعتماد على M.count = 10, وبالتالي تكون حلقة For من صفر الى M.Count-1, أذ أن فهرسة عناصر المصفوفة تبدأ من الصفر.

البرنامج الخاص بهذا الاجراء موضح بالشكل التالي:

```
Private Sub Command1_Click()
Dim i As Integer
Dim Avg As Single
Avg = 0
For i = 0 To M.Count - 1
   Avg = Avg + Val(M(i).Text)
Next i
   TxtAvg.Text = Avg / M.Count
End Sub
```

- **برمجة المفتاح Max**

لايجاد القيمة العظمى لمجموعة من الارقام نستخدم الطريقة التالية:

- نفرض ان اول عدد هو القيمة العظمى ونضعه في متغير Max.
- نعمل حلقة تكرارية على بقية الارقام وكلما وجدنا رقما اكبر من Maxنضعه في Max.

نكتب الاجراء التالي على مفتاح Max وكمايلي:

```
Private Sub Command2_Click()
Dim i As Integer
Dim Max As Single
Max = Val(M(0))
For i = 1 To M.Count - 1
If Val(M(i)) > Max Then
   Max = Val(M(i))
End If
Next i
TxtMax.Text = Max
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- **برمجة المفتاح Min**

بنفس المبدأ المعتمد في حساب القيمة العظمى, نكتب الاجراء التالي على المفتاح Min وكمايلي:

```
Private Sub Command3_Click()
Dim i As Integer
Dim Min As Single
Min = Val(M(0))
For i = 1 To M.Count - 1
If Val(M(i)) < Min Then
   Min = Val(M(i))
End If
Next i
TxtMin.Text = Min
End Sub
```

- **برمجة المفتاح تلوين Color**

لاختبار ان عددا ما زوجي ام فردي نختبر باقي قسمتة على 2 باستخدام Mod فاذا كان باقي القسمة صفرا يكون العدد زوجيا والا كان فرديا.

نكتب الاجراء التالي على المفتاح تلوين Color

```
Private Sub Command5_Click()
Dim i As Integer
Dim j As Integer
Dim x As Single
For i = 0 To M.Count - 1
   For j = i + 1 To M.Count - 1
      If Val(M(j)) < Val(M(i)) Then
         x = Val(M(j))
         M(j) = M(i)
         M(i) = x
      End If
   Next j
Next i
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

v برمجة المفتاح ترتيب تصاعدي Ordered

يوجد العديد من الطرق لترتيب مجموعة من الارقام في مصفوفة من احد هذه الطرق:

- نبدأ بأول عنصر في المصفوفة.
- نعمل حلقة تكرارية على باقي عناصر المصفوفة (من العنصر الثاني الى اخر عنصر) وكلما وجدنا عنصرا اصغر من اول عنصر نقوم بتبديله مع اول عنصر, وفي نهاية هذه الدوره سيكون اول عنصر هو العدد الاصغر في المصفوفة.
- ننتقل الى العنصر الثاني في المصفوفة.
- نعمل حلقة تكرارية على باقي عناصر المصفوفة (من العنصر الثالث الى اخر عنصر) وكلما وجدنا عنصرا اصغر من العنصر الثاني بادلنا بينه وبين ثاني عنصر. في نهاية هذه الحلقة سيكون اول عنصر هو العدد الاصغر في المصفوفة وثاني عنصر هو العدد الذي يلية وهكذا حتى الوصول الى اخر عنصر في المصفوفة.
- نحتاج في هذا الاجراء الى حلقتي For وهما

```
For i=0 to M.Count-1
    For j=i+1 to M.Count
```

الحلقة الاولى I للدوران على كل عناصر المصفوفة, والحلقة الثانية j للدوران من اجل كل I على عناصر المصفوفة من i+1 الى اخر عنصر في المصفوفة.

- يصبح الاجراء الخاص بالمفتاح ترتيب تصاعدي Ordered كمايلي:

```
Private Sub Command5_Click()
Dim i As Integer
Dim j As Integer
Dim x As Single
For i = 0 To M.Count - 1
   For j = i + 1 To M.Count - 1
      If Val(M(j)) < Val(M(i)) Then
         x = Val(M(j))
         M(j) = M(i)
         M(i) = x
      End If
   Next j
Next i
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

عملية التنفيذ لكافة الاجراءات موضحه في الشكل التالي:

PDF created with pdfFactory trial version www.pdffactory.com

المشروع الرابع عشر

-:----------

التحريـــــــك

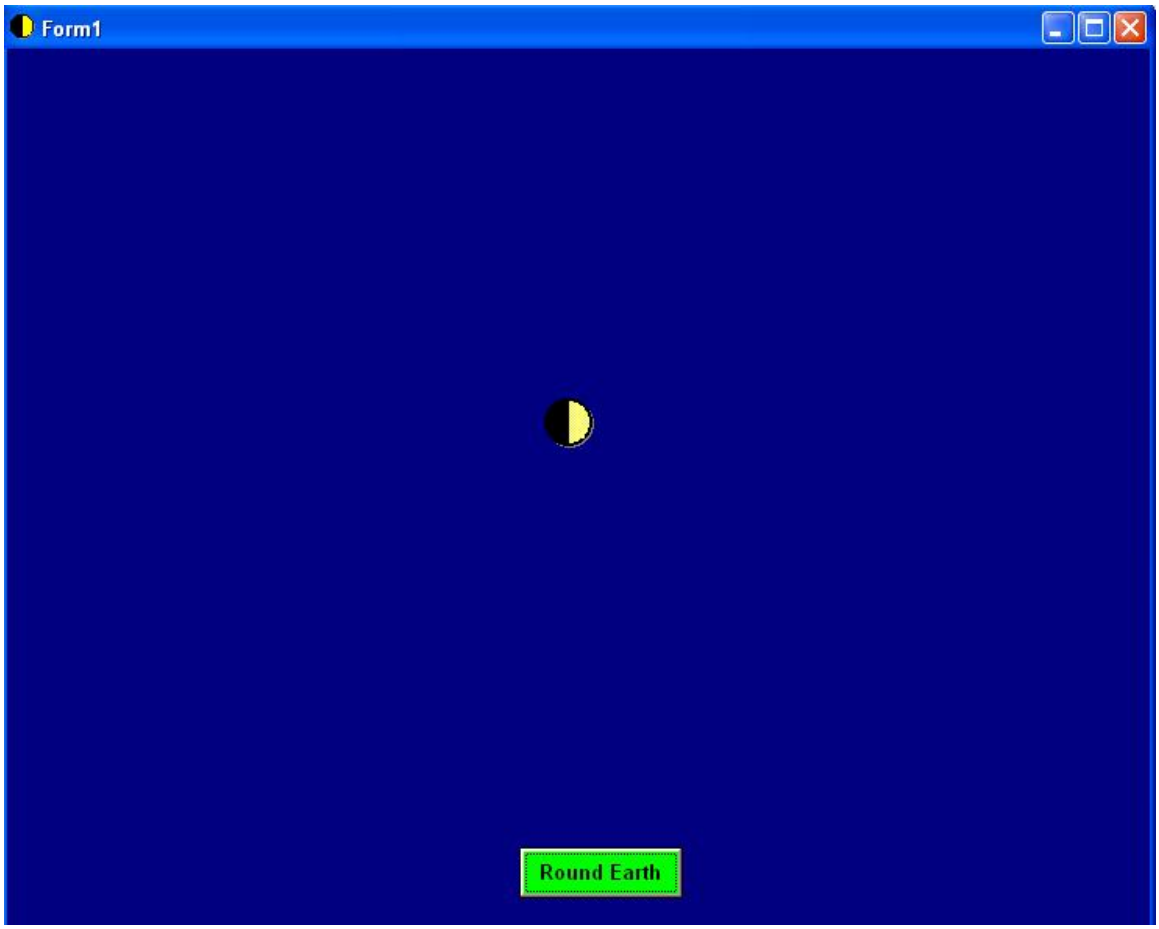
سنتعلم من خلال هذا المشروع :-

- التعامل مع المؤقت Timer.
- التعامل مع مصفوفات عناصر التحكم.
- المتغيرات الساكنة Static .
- اشرطة الانزلاق Scroll Bars.
- توليد رقم عشوائي Rnd.

الهدف من المشروع

-:----------

- عند تنفيذ هذا المشروع, يظهر النموذج الاول من المشروع والذي يحتوي على صورة قمر يدور حول نفسة (كذلك نلاحظ دوران ايكونة النموذج). كما يتغير لون خلفية النموذج كل نصف ثانية.



PDF created with pdfFactory trial version www.pdffactory.com

- عند النقر على مفتاح حول الارض Round Earth في النموذج الاول, يظهر النموذج الثاني والذي يحوي قمرا يدور حول الارض, يمكن التلاعب بسرعة دوران القمر حول الارض عن طريق شريط الانزلاق العمودي, كذلك يمكن عكس اتجاه دوران القمر بالنقر على مفتاح Right/Left.

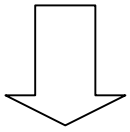
مراحل التنفيـــذ

-:----------

- نفتح مشروعا قياسيا جديدا.
- ننشيء نموذجين.
- ننشيء في النموذج الاول مصفوفة ونسميها (Moon) من ثمانية صور Image وكمايلي:

MOON (0) IMAGE	
Name	M
Index	0
Picture	Icon

MOON (1) IMAGE	
Name	M
Index	1
Picture	Icon



MOON (7) IMAGE	
Name	M
Index	7
Picture	Icon

- نضع في عناصر المصفوفة Moon صور الاقمار باوضاعها المختلفة والتي نجدها في المجلد التالي:

C:\Program Files\Microsoft Visual Studio\Common\Graphics\Icons\Elements

- ننشيء صورة (التي سوف تدور فيها الاقمار) ونسميها ImgMoon.

PDF created with pdfFactory trial version www.pdffactory.com

- ننشيء مؤقتا Timer ونضبط مجالة الزمني Interval على 500 ميلي ثانية مثلا.
- المؤقت Timer يستخدم لتنفيذ اجراء كل فترة زمنية محددة.
- تستخدم الخاصية Interval لتحديد الفترة الزمنية للمؤقت, أي الفترة الزمنية الفاصلة بين تنفيذ اجراء المؤقت. يقاس عادة المجال بالميلي ثانية (اي اذا وضعنا القيمة 1000 مثلا, يكون الزمن للمؤقت عبارة عن ثانية واحدة, اذا كانت قيمة المجال صفرا لن يعمل المؤقت, تتراوح قيمة المجال الزمني بين الصفر و القيمة 65535 ميلي ثانية اي اكثر من الدقيقة بقليل.
- مثال: ضع مؤقتا على النموذج, واضبط مجالة الزمني على 500 ميلي ثانية, واكتب الامر التالي على اجراء المؤقت:

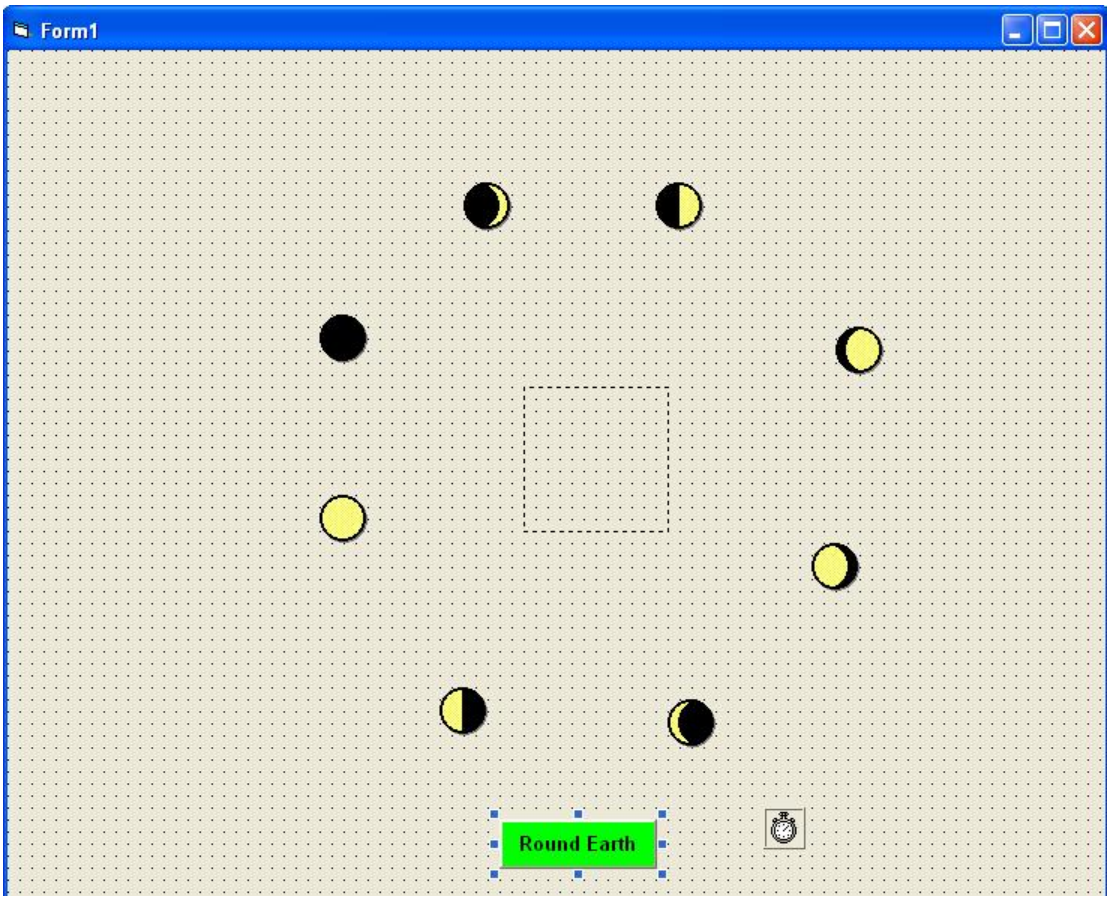
Me.BackColor = QBColor(15*Rnd)

- عند التنفيذ سيتغير لون النموذج بشكل عشوائي كل نصف ثانية.
- برمجة الدالة Rnd تعيد عند كل استدعاء رقما عشوائيا محصورا بين الصفر والواحد , في الامر السابق نستخدم لتوليد لون الخلفية الدالة QBColor التي تأخذ شكل متغير أدخال رقما محصورا بين قيمة الصفر والقيمة 15, القيمة الناتجه من حاص الضرب (Rnd * 15) تعيد عند كل أستدعاء رقما عشوائيا محصورا بين قيمة الصفر والقيمة 15.
- برمجة المؤقت Timer في النموذج الاول : من خلال أستخدام المؤقت نحصل على صور متحركة مختلفة ومتسلسلة بفترة زمنية قصيرة نسبيا, تبدو الصورة وكانها تتحرك.
  - في مثالنا هذا نقوم بعرض صور الاقمار الثمانية في ImgMoon بشكل مستمر:
  - نضع اولا الصورة الاولى Moon(0) في ImgMoon.
  - بعد نصف ثانية نضع الصورة الثانية Moon(1) في ImgMoon.
  - بعد نصف ثانية ايضا نضع الصورة الثالثة Moon(2) في ImgMoon.
  - وهكذا حتى عرض الصورة الاخيرة Moon(7) عندها نحتاج الى الرجوع الى الصورة الاولى Moon(0) , نحتاج الى ذلك متغيير ثابت Static أسمة I كي يحافظ على قيمتة من استدعاء لاخر , سنزيد هذا المتغيير بمقدار واحد في كل مرة وعندما يصل I الى ثمانية نعيدة الى الصفر.
  - يكون اجراء مؤقت النموذج الاول كمايلي:

```
Private Sub Timer1_Timer()
Me.BackColor = QBColor(15 * Rnd)
Static i As Integer
ImgMoon.Picture = M(i).Picture
Me.Icon = M(i).Picture
i = i + 1
If i = 8 Then i = 0
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

الخطوات في الاعلى موضحة في النموذج التالي:

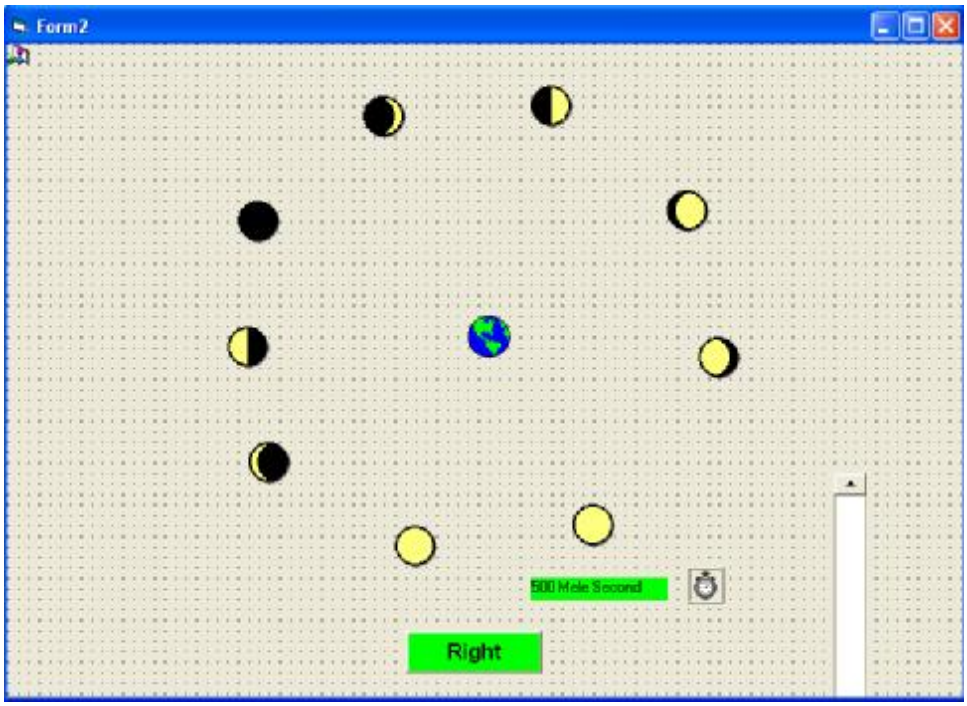


- ننشيء مفتاح حول الارض Round Earth على النموذج الاول ونكتب في أجرائه امر اضهار النموذج الثاني:

```
Private Sub Command1_Click()
Me.Hide
Form2.Show
End Sub
```

- ننشيء صورة الارض على النموذج الثاني ImgEarth ونضع حولها مصفوفة من ثمانية صور للقمر Moon بشكل دائري, نجعل جميع عناصر هذه المصفوفة مخفية عن طريق الخاصية Visible = False, وكما موضح في الشكل التالي:

PDF created with pdfFactory trial version www.pdffactory.com



- ننشيء مفتاح عنوانة الابتدائي Right , كلما نقرنا على هذا المفتاح يتغير عنوانة Right/Left الاجراء التالي يوضح ذلك:

```
Private Sub Command1_Click()
If Command1.Caption = "Right" Then
Command1.Caption = "Left"
Else
Command1.Caption = "Right"
End If
End Sub
```

- ننشيء مؤقتا على النموذج الثاني, ونضبط مجالة الزمني على 500 ميلي ثانية.
- ننشيء تسمية Label ونضع في خاصية العنوان له قيمة مجال المؤقت 500 Mele Second.
- ننشيء شريط الانزلاق العمودي Vertical Scroll Bar ونضبط خصائصة كالاتي:

PDF created with pdfFactory trial version www.pdffactory.com

VSCROLL1 VSCROLLBAR	
Name	VScroll1
Max	10000
Min	1
Value	500

- يستخدم شريط الانزلاق Scroll Bar لتوليد قيمة Value محصورة بين قيمة دنيا Min وقيمة عظمى Max.
- في مثالنا هذا نستخدم شريط الانزلاق لتغيير مجال مؤقت النموذج الثاني من 1 ميلي ثانية الى 10000 ميلي ثانية أي (10) ثواني.
- نكتب الاجراء التالي على حدث تغيير Change قيمة شريط الانزلاق:

```
Private Sub VScroll1_Change()
Label1.Caption = VScroll1.Value
Timer1Sub.Interval = VScroll1.Value
End
```

- برمجة مؤقت النموذج الثاني: لجعل الاقمار تبدو وكأنها تدور حول الارض:
- في البداية تكون الاقمار مخفية.
- بعد نصف ثانية يظهر القمر الاول Moon(0).
- بعد نصف ثانية يختفي القمر الظاهر Moon(0) ويظهر القمر Moon(1).
- وهكذا حتى الوصول الى القمر Moon(7) بعدها نعود الى القمر الاول Moon(0).
- نحتاج في أجراء المؤقت الى متغيير ساكن I كي يحافظ على قيمتة من أستدعاء لاستدعاء اخر, سيزيد هذا المتغيير بقيمة واحد في كل مرة اذا كان عنوان المفتاح هو Right وعندما نصل الى ثمانية نعيدة للصفر, اما أذا كان عنوان المفتاح Left ينقص من المتغير بمقدار واحد وعندما يصل هذا المتغيير الى -1 نعيده الى سبعة, في كل مرة نخفي جميع الاقمار ونظهر القمر الذي رقمة i . يكون اجراء المؤقت كمايلي:

```
Private Sub Timer1_Timer()
Static i As Integer
Dim j As Integer
For j = 0 To 7
   M1(j).Visible = False
Next j
```


PDF created with pdfFactory trial version www.pdffactory.com

```
    M1(i).Visible = True
If Command1.Caption = "Right" Then
i = i + 1
Else
i = i - 1
End If
If i = 8 Then i = 0
If i = -1 Then i = 7
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الخامس عشر

-:----------

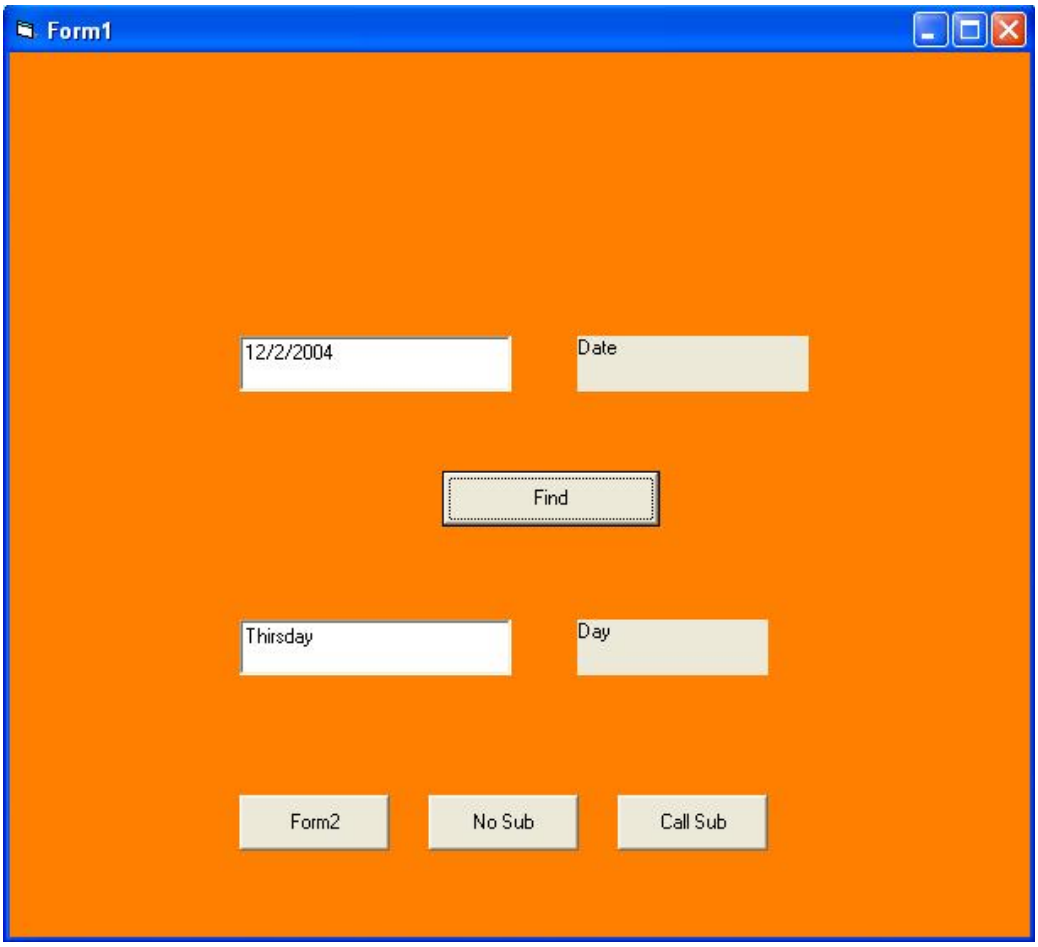
سوف نتعلم من خلال هذا المشروع مايلي:

- بناء البرامج الفرعية.
- الاجراءات Sub.
- الدوال Function.

الهدف من المشروع

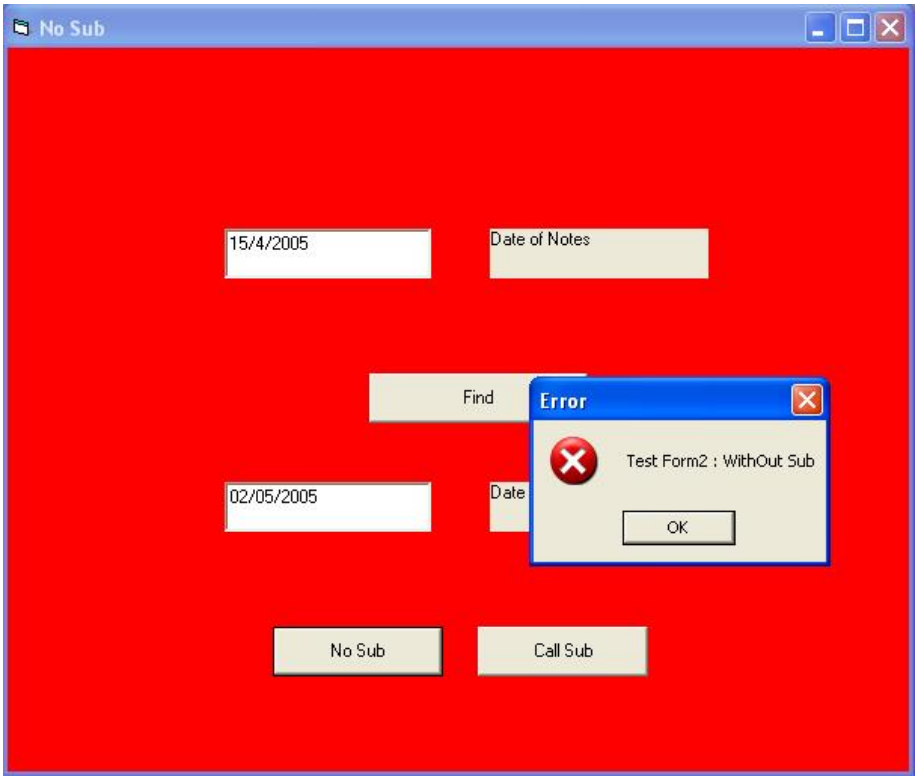
-:----------

- عند تنفيذ هذا المشروع سوف يظهر النموذج الاول من المشروع, وعند النقر على مفتاح Form2 يظهر النموذج الثاني, كما موضح في الاشكال التالية:



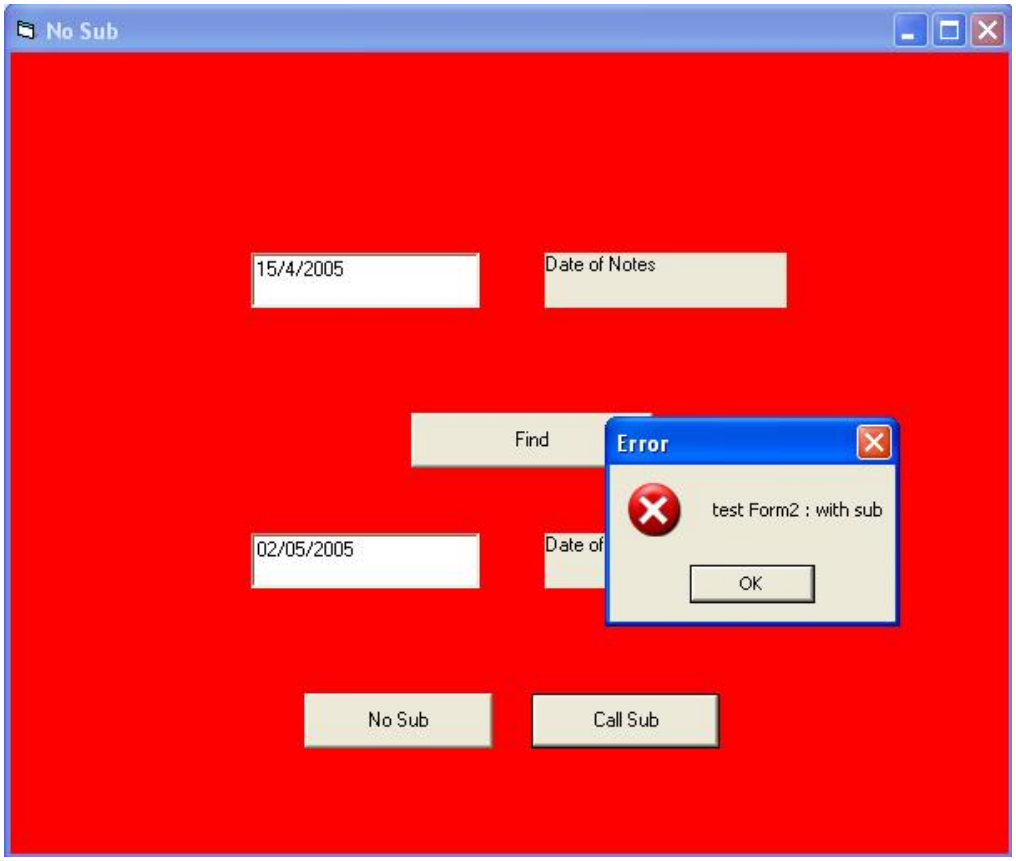
PDF created with pdfFactory trial version www.pdffactory.com

- عند النقر على مفتاح **No Sub** في احد النموذجين, يتلون النموذج باللون الاحمر ويظهر صندوق رسالة خطأ كما في الشكل التالي:



PDF created with pdfFactory trial version www.pdffactory.com

- عند النقر على مفتاح **Call Sub** في احد النموذجين, يتلون النموذج أيضا باللون الاحمر ويظهر صندوق رسالة خطأ كما في الشكل التالي:



- بعد كتابة تاريخ في صندوق النص الاول في النموذج الاول, ومن ثم النقر على المفتاح أوجد Find , يظهر في صندوق النص الثاني أسم اليوم الموافق للتاريخ.
- بعد كتابة تاريخ في صندوق النص الاول في النموذج الثاني, ومن ثم النقر على مفتاح أوجد Find يظهر في صندوق النص الثاني تاريخ اول يوم ليس عطلة (سبت او أحد) من الشهر التالي:

مراحل التنفيذ

----------:-

- نفتح مشروعا قياسيا جديدا.
- ننشيء نموذجين.
- ننشيء على النموذجين عناصر التحكم الظاهره في الاشكال السابقة.
- برمجة المفتاح No Sub على النموذج الاول:

PDF created with pdfFactory trial version www.pdffactory.com

الفكرة الرئيسية من خلال تنفيذ هذا الاجراء هو اننا نريد عند كل ظهور لصندوق رسالة MsgBox ان نلون النموذج باللون الاحمر لشد انتباه المستخدم, وبعد أغلاق الصندوق يعود النموذج الى اللون الاول.

- **نكتب الاجراء التالي على مفتاح No Sub في النموذج الاول:**

```
Private Sub Command3_Click()
Dim c As Long
c = Me.BackColor
Me.BackColor = vbRed
MsgBox "Test Form1 : WithOut Sub", vbCritical, "Error"
Me.BackColor = c
End Sub
```

- **برمجة المفتاح No Sub على النموذج الثاني:**

نكتب الاجراء التالي الذي يكون شبيها بالاجراء السابق:

```
Private Sub Command2_Click()
Dim c As Long
c = Me.BackColor
Me.BackColor = vbRed
MsgBox "Test Form2 : WithOut Sub", vbCritical, "Error"
Me.BackColor = c
End Sub
```

- **كتابة الاجراء MyError**

نلاحظ أننا في الاجرائين السابقين استخدمنا نفس التعليمات او الاوامر, مايختلف في التعليمات هو أسم الاجراء ورسالة الخطأ.

سوف نقوم عوظا عن كتابة نفس هذه التعليمات في اكثر من مكان, بكتابة اجراء عام ندعوه MyError ومن ثم نستعدي هذا الاجراء حيثما نريد في المشروع, سنمرر لهذا الاجراء عند كل أستدعاء النموذج المنفذ علية ورسالة الخطأ التي نريد أظهارها في صندوق الرسالة.

- **برمجة البرامج الفرعية**

تستخدم البرامج الفرعية عادة لاتمام المهام التي نقوم بها بشكل متكرر, مثلا يمكن أن ننشيء دالة ونستدعيها في عدة تعابير في أماكن متعددة.

PDF created with pdfFactory trial version www.pdffactory.com

يمكن أن نستخدم البرامج الفرعية أيضا في حال أن عدة أجراءات (أحداث Events) تقوم بنفس العمليات, عندها يمكن ان نضع هذه العمليات في برنامج فرعي واحد ونجعل جميع أجراءات الاحداث تستدعي هذا البرنامج الفرعي, مماسيؤدي الى الغاء التعليمات المتكررة كما سيجعل التعديل أسهل فيما بعد, أذ أن التعديل سيكون في البرنامج الفرعي فقط وليس في كل أجراءات الاحداث, تكون البرامج الفرعية اما اجراءات Sub او دوال Function.

- **برمجة الاجراءات Sub**

الشكل العام للاجراء هو

```
[Public| Privates] Sub name [(arglist)]
[statements]
[Exit Sub]
[statements]
End Sub
```

حيث ان :

- **Public** تحدد ان الاجراء هو عام, أي قابل للاستدعاء من جميع أجزاء المشروع(وهي الحالة الافتراضية أن لم نحدد نوع الاجراء عاما ام خاصا).
- **Private** تحدد أن الاجراء خاص, لايستدعى الا في نفس مكان تعريفة(صفحة البرمجة لنموذج أو وحدة نمطية).
- **Name** اسم الاجراء.
- **Statements** وهي الاوامر او التعليمات الخاصة بالاجراء التي ستنفذ عند أستدعاء الاجراء.
- **Exit Sub** وهي أمر الخروج الفوري من الاجراء (كسر الاجراء).
- **Arglist** هي أسماء متغيرات الادخال للاجراء, ويفصل بينهما بفاصلة (,) في حالة وجود أكثر من متغير أدخال, كل متغير أدخال يعرف كتعريف أي متغير وسيعامل كمتغير في تعليمات الاجراء, الشكل العام لتعريف أي متغير أدخال هي:

```
[By Val]|By Ref] varname[()] [As type]
```

حيث أن:

- **By Val** تحدد أن تمرير المتغير هي التمرير بالقيمة, أي لاتتغير قيمة المتغير الممرر للاجراء.
- **By Ref** تحدد أن طريقة تمرير المتغير هي التمرير بالعنوان, أي يمكن أن تتغير قيمة المتغير الممرر للاجراء (وهي الطريقة الافتراضية أن لم نحدد طريقة التمرير).

PDF created with pdfFactory trial version www.pdffactory.com

- **Varname** أسم المتغيير , حيث أن الاقواس بعد اسم المتغير () تدل على أنه مصفوفة (Array) . عند أستدعاء الاجراء نعطي قيما معينة لمتغيرات الدخل, يتم تنفيذ تعليمات الاجراء من أجلها(أي سيتم أستبدال كل متغير أدخال في تعليمات الاجراء بالقيمة المررخ).
- **Type** يمكن أن يكون النوع أي نوع بيانات اساسي:

(Integer, Long, Single, Double, Currency, String, …)

أو **Variant** او نوع كائن Object Type , أذا لم نحدد نوع البيانات الخاصة بمتغير الادخال, فأنه سيأخذ النوع variant أي أنه أي نوع من البيانات .

- يتم أستدعاء الاجراء بأستخدام الامر

[Call] nme [argumentlist]

- **Name** أسم الاجراء.
- **Argmentlist** قائمة قيم متغيرات الادخال للاجراء.
- **Call** خيارية, أذا وضعت يجب أن نضع قائمة متغيرات الدخل ضمن قوسين () و الا فلا يجب وضع الاقواس.
- **برمجة**: يكتب البرنامج الفرعي العام في وحدة مطية Module ويصبح قابلا للاستدعاء من مختلف نماذج المشروع, يمكن في الوحدة النمطية كتابة عدد من الاجراءات والدوال.
- **تصميم**: لاضافة وحدة نمطية الى المشروع نختار القائمة Project/Add Module فتظهر صفحة تحرير, يمكن كتابة الاجاء فيها. تحفظ الوحدة النمطية في ملف مستقل ينتهي بالامتداد (.bas) , ننشيء وحدة نمطية جديده Module1 ونكتب الاجراء MyError التالي فيها. يأخذ هذا الاجراء متغيرات الادخال التالية:

F as Form نمرر في هذا المتغير النموذج المراد تلوينة.

S As String نمرر في هذا المتغير رسالة الخطأ.

```
Sub MyError(f As Form, s As String)
Dim c As Long
   c = f.BackColor
   f.BackColor = vbRed
   MsgBox s, vbCritical, "Error"
   f.BackColor = c
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- **برمجة المفتاح Call Sub على النموذج الاول:**

نستدعي في أجراء هذا المفتاح الاجراء العام MyError , ونمرر له النموذج الاول Form1 ورسالة الخطأ

```
Private Sub Command2_Click()
    MyError Form1, "Test Form1 : with Sub"
End Sub
```

- **برمجة المفتاح "Call Sub" على النموذج الثاني**

نستدعي في أجراء هذا المفتاح الاجراء العام MyError ونمرر له النموذج الثاني Form2 ورسالة الخطأ.

```
Private Sub Command2_Click()
    Call MyError(Form2, "Test Form2: with Sub")
End Sub
```

- **كتابة الدالة DateDay في وحدة نمطية Module2**
- **برمجة الدوال Function**

الشكل العام لتعريف الدالة:

```
[Public| Private] Function name [(arglist)]
[statements]
[name = expression]
[Exit Function]
[statements]
[name = expression]
End Function
```

تعمل متغيرات الادخال في الدالة تماما كما تعمل متغيرات الادخال في الاجراء. هناك ثلاثة فروق اساسية بين الاجراء والدالة:

- يجب أن نضع دائما أقواسا بعد أسم الدالة عند أستدعاها .
- للدوال متغيرات نوع بيانات يحدد نوع القيمة المرجعة.
- تنسب القيمة المرجعة في الدالة لاسم الدالة, وبالتالي يمكن أستخدام الدالة في التعابير.
- في مشروعنا هذا نريد كتابة دالة نمرر لها تاريخا , فتعيد لنا أسم اليوم الموافق لهذا التاريخ, نكتب الدالة DateDay التالية في وحدة نمطية ثانية Module2.

```
Function DateDay(d As Date) As String
```


PDF created with pdfFactory trial version www.pdffactory.com

```
Select Case Weekday(d)
Case 1: DateDay = "Sanday"
Case 2: DateDay = "Monday"
Case 3: DateDay = "Tuesday"
Case 4: DateDay = "Wensday"
Case 5: DateDay = "Thirsday"
Case 6: DateDay = "Friday"
Case 7: DateDay = "Saturday"
End Select
End Function
```

تأخذ الدالة DateDay متغير أدخال وحيد هو D من النوع date , وتعيد سلسلة نصية String .

- برمجة الدالة WeekDay (Date) من دوال VB تأخذ تاريخا Date وتعيد رقم اليوم في الاسبوع الموافق لهذا التاريخ (1 الاحد, 2 الاثنين, ...).
- أستدعاء الدالة DateDay على المفتاح اوجد Find في النموذج الاول:

المطلوب أن يكتب المستخدم تاريخا ما في صندوق النص Text1. بعد النقر على المفتاح Find, يظهر في صندوق النص Text2 أسم اليوم الموافق للتاريخ المدخل. نستدعي في أجراء المفتاح أوجد Find الدالة السابقة DateDay , نمرر لها Text1.Text كقيمة متغير أدخال للدالة ونسند نتيجة الدالة الى Text2. نختبر قبل أستدعاء الدالة النص المدخل في Text1 هو تاريخ مقبول بأستخدام الدالة IsDate والا فتظهر عبارة خطأ في صندوق النص Text2.

- **نكتب على مفتاح أوجد Find الاجراء التالي:**

```
Private Sub Command1_Click()
If IsDate(Text1.Text) = True Then
   Text2 = DateDay(Text1.Text)
Else
   Text2 = "Error"
End If
End Sub
```

- **برمجة الدالة IsDate(expr) من دوال VB** تختبر أن تعبيرا ما expr هو تاريخ فتعيد True والا فتعيد False .

PDF created with pdfFactory trial version www.pdffactory.com

- **كتابة الدالة DueDate** في وحد نمطية Module2:

- المطلوب أن نكتب دالة DueDate نمرر لها تاريخا (تاريخ فاتورة مثلا) فتعيد تاريخ أول يوم من الشهر القادم (تاريخ وجوب الدفع) مع مراعاة مايلي:
- أذا كان أول يوم من الشهر القادم يوم أحد, فستعيد عوظا عنه يوم الاثنين التالي:
- أذا كان أول يوم من الشهر القادم يوم سبت, فستعيد عوظا عنه يوم الاثنين التالي.
- أذا كان أول يوم من الشهر القادم يوم جمعة, فستعيد عوظا عنة يوم الخميس السابق.

```
Function DueDate(d As Date) As Date
Dim Result As Date
   Result = DateSerial(Year(d), Month(d) + 1, 1)
   Select Case Weekday(Result)
      Case 1: DueDate = Result + 1
      Case 7: DueDate = Result + 2
      Case 6: DueDate = Result - 1
   Case Else
      DueDate = Result
   End Select
End Function
```

- برمجة الدالة Year(Date) من دوال VB نمرر لها تاريخا Date فتعيد عام هذا التاريخ . الدالة Month(Date) نمرر لها تاريخا Date فتعيد شهر ذلك التاريخ.
- برمجة الدالة DateSerial(year, month, day) من دوال VB نمرر لها عام year وشهرا month ويوما day , فتعيد التاريخ الموافق.
- أستدعاء الدالة DueDate على المفتاح أوجد Find في النموذج الثاني:

المطلوب أن يكتب المستخدم تاريخا ما في صندوق النص Text1 (تاريخ الفاتورة) . بعد النقر على مفتاح أوجد Find يظهر في صندوق النص الثاني Text2 تاريخ أول يوم من الشهر القادم (تاريخ وجود الدفع). نستدعي في أجراء المفتاح أوجد Find الدالة السابقة DueDate , نمرر لها Text1.Text كقيمة متغير أدخال الدالة ونسند ناتج الدالة الى Text2. نختبر قبل أستدعاء الدالة أن النص المدخل في Text1 هو تاريخ مقبول بأستخدام الدالة Isdate , والا فتظهر عبارة خطأ في صندوق النص Text2.

- نكتب على المفتاح أوجد Find الاجراء التالي:

```
Private Sub Command1_Click()
If IsDate(Text1.Text) Then
   Text2 = DueDate(Text1.Text)
Else
   Text2 = "Error"
End If
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع السادس عشر

-:----------

ربط الفيجوال بسيك بقاعدة البيانات

سنتعلم من خلال هذا المشروع مايلي:

- الاضافة Add
- البحث Search
- الحذف Delete
- العرض Display
- الترتيب Ordere

الهدف من المشروع

-:----------

ربط قاعدة البيانات مع الفيجوال بيسك ضمن برنامج خاص بمعلومات الموظفين, كما مبين في النموذج التالي:

PDF created with pdfFactory trial version www.pdffactory.com

مراحل التنفيذ

-:----------

- نفتح مشروعا قياسيا جديدا.
- نضع على النموذج عناصر التحكم المبينة في الشكل أعلاه :
- برمجة الدالة Data1

DATABASENAME	C:\DOCUMENTS AND SETTINGS\COMPUTER SYSTEM\MY DOCUMENTS\DB297.MDB
RecordSource	Table1
Visible	False

- النص الاول TextBox1 نحدد خصائصة كمايلي:

DATA FIELD	ID
Data Source	Data1
Text	فراغ

- النص الثاني TextBox2 نحدد خصائصة كمايلي:

DATA FIELD	NAME
Data Source	Data1
Text	فراغ

- النص الثالث TextBox3 نحدد خصائصة كمايلي:

DATA FIELD	GENDER
Data Source	Data1
Text	فراغ

PDF created with pdfFactory trial version www.pdffactory.com

- النص الرابع TextBox4 نحدد خصائصة كمايلي:

DATA FIELD	AGE
Data Source	Data1
Text	فراغ

- النص الخامس TextBox5 نحدد خصائصة كمايلي:

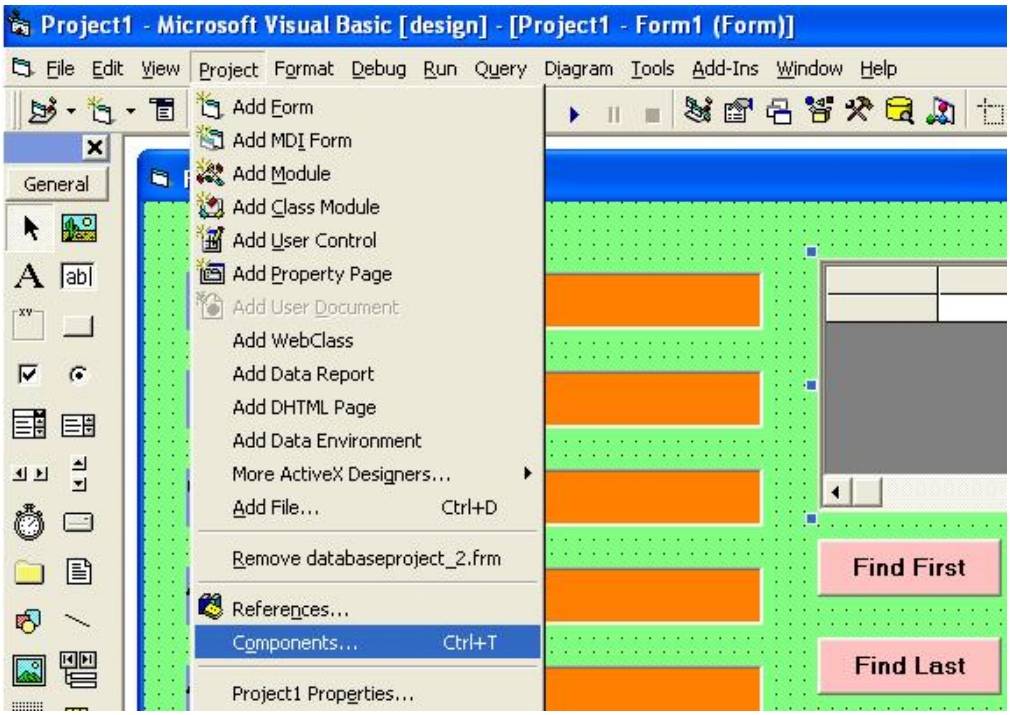
DATA FIELD	ADDRESS
Data Source	Data1
Text	فراغ

- النص السادس TextBox6 نحدد خصائصة كمايلي:

DATA FIELD	SALARY
Data Source	Data1
Text	فراغ

- برمجة الاداة Microsoft FlexGrid Control 6.0

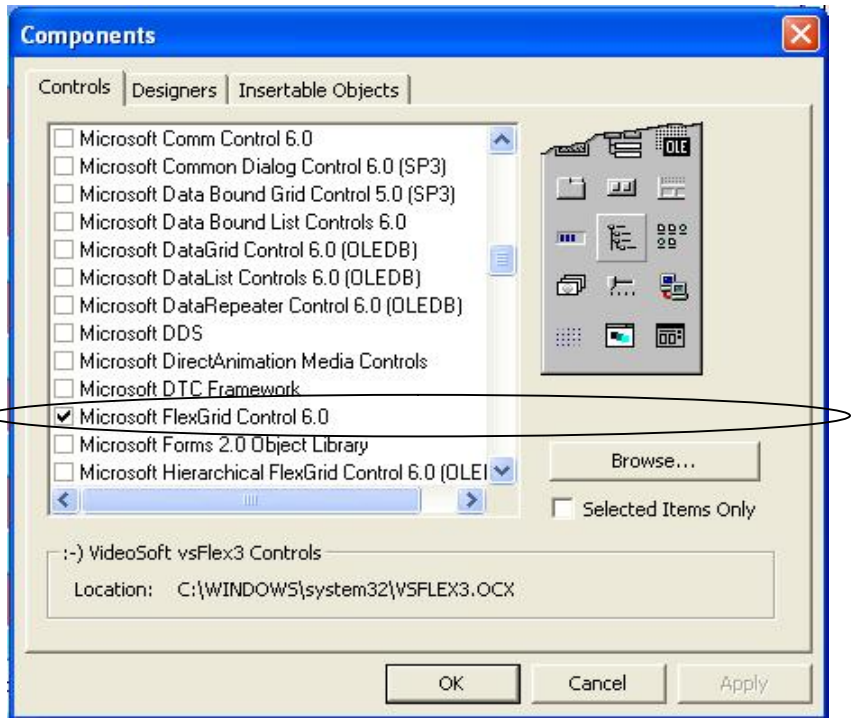
نفعل هذه الاداة MsFlexGrid عن طريق الدخول الى الامر Project ثم الامر Components عند ذلك نفعل MsFlexGrid كما موضح في الشكل التالي:



PDF created with pdfFactory trial version www.pdffactory.com

- ومن ثم تظهر الشاشة التالية:

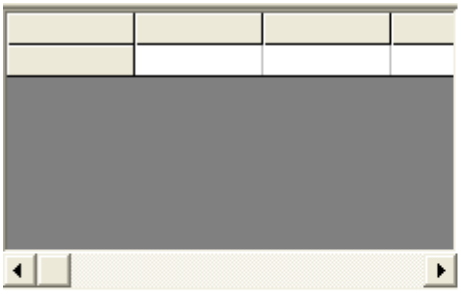
نؤشر الاختيار المطلوب بعلامة صح وكما موضح في الشكل التالي:

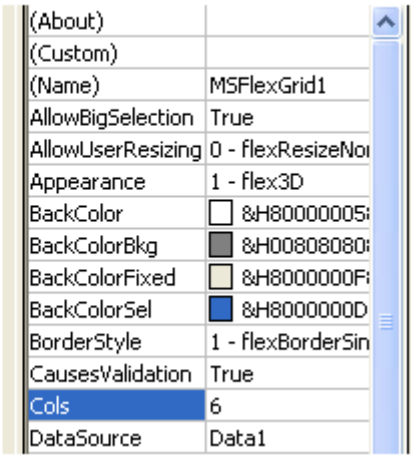


- ومن ثم نحدد خصائص هذه الدالة بتحديد الخصائص التالية:

COLS	6
DataSource	Data1

وكما موضح في الشكل التالي:





PDF created with pdfFactory trial version www.pdffactory.com

- برمجة المفتاح التالي Next : عند النقر على المفتاح Next نكتب الاجراء التالي:

```
Private Sub Command1_Click()
On Error Resume Next
Data1.Recordset.MoveNext
If Data1.Recordset.EOF Then
Data1.Recordset.MoveLast
End If
End Sub
```

- برمجة المفتاح السابق Previous: عند النقر على المفتاح Previous نكتب الاجراء التالي:

```
Private Sub Command2_Click()
On Error Resume Next
Data1.Recordset.MovePrevious
If Data1.Recordset.BOF Then
   Data1.Recordset.MoveNext
End If
End Sub
```

- برمجة مفتاح الانتقال للسجل الاول First Record : عند النقر على مفتاح First Record نكتب الاجراء التالي:

```
Private Sub Command3_Click()
Data1.Recordset.MoveFirst
End Sub
```

- برمجة مفتاح الانتقال الى السجل الاخير Last Record: عند النقر على مفتاح Last Record اي الانتقال الى السجل الاخير نكتب الاجراء التالي:

```
Private Sub Command4_Click()
Data1.Recordset.MoveLast
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- أضافة سجل Add: عند النقر على مفتاح أضافة (Add) نكتب الاجراء التالي:

```
Private Sub Command5_Click()
Data1.Recordset.AddNew
End Sub
```

- عملية الخزن Save عند النقر على مفتاح save سوف تظهر الواجهة التالية :

نكتب الاجراء التالي عند النقر على مفتاح Save

```
Private Sub Command6_Click()
On Error Resume Next
Data1.Recordset.Update
MsgBox "Data already saved", vbOKOnly + vbMsgBoxRight + vbMsgBoxRtlReading,
"Saved the change"
End Sub
```

- برمجة مفتاح الحذف Delete : عند النقر على مفتاح Delete نكتب الاجراء التالي :

```
Private Sub Command7_Click()
On Error Resume Next
x = MsgBox("Are you want to delete this record", vbOKCancel)
If x = vbOK Then
Data1.Recordset.Delete
End If
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- برمجة المفتاح Count اي حساب عدد القيود الموجوده في ملف قاعدة البيانات , عند النقر على مفتاح Count مرتين متتاليتين نكتب الاجراء التالي:

```
Private Sub Command8_Click()
On Error Resume Next
MsgBox Data1.Recordset.RecordCount
End Sub
```

- برمجة المفتاح اوجد السجل الاول Find First: عند النقر على مفتاح Find First نكتب الاجراء التالي:

```
Private Sub Command9_Click()
xx = InputBox("Enter name of the employment to be search")
yy = "Name = '" & xx & "'"
Data1.Recordset.FindFirst yy
If Data1.Recordset.NoMatch Then
   MsgBox ("The name is not found")
End If
End Sub
```

- برمجة المفتاح اوجد السجل الاخير Find last:

```
Private Sub Command10_Click()
On Error Resume Next
xx = InputBox("Enter name of the employment to be search")
yy = "Name = '" & xx & "'"
Data1.Recordset.FindLast yy
If Data1.Recordset.NoMatch Then
   MsgBox ("The name is not found")
End If
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- اوجد السابق Find Previous : عند النقر على مفتاح اوجد السابق نكتب الاجراء التالي:

```
Private Sub Command11_Click()
On Error Resume Next
xx = InputBox("Enter name of the employment to be search")
yy = "Name = '" & xx & "'"
Data1.Recordset.FindPrevious yy
If Data1.Recordset.NoMatch Then
   MsgBox ("The name is not found")
End If
End Sub
```

- برمجة المفتاح اوجد اللاحق Find Next: عند النقر على مفتاح اوجد اللاحق Find Next نكتب الاجراء التالي:

```
Private Sub Command12_Click()
On Error Resume Next
xx = InputBox("Enter name of the employment to be search")
yy = "Name = '" & xx & "'"
Data1.Recordset.FindNext yy
If Data1.Recordset.NoMatch Then
   MsgBox ("The name is not found")
End If
End Sub
```

- برمجة المفتاح عرض السجلات Display Record: عند النقر على مفتاح عرض السجلات Display Record مرتين متتاليتين نكتب الاجراء التالي:

```
Private Sub Command13_Click()
Dim xxx
xxx = InputBox("Enter name of employment to be search")
Data1.RecordSource = " select * from table1 where name like '" & xxx & "*'"
Data1.Refresh
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- برمجة المفتاح اصغر راتب Ordered Min: عند النقر على هذا المفتاح مرتين متتاليتين نكتب الاجراء التالي:

```
Private Sub Command14_Click()
On Error Resume Next
Data1.RecordSource = " select * from table1 order by salary "
Data1.Refresh
End Sub
```

- برمجة المفتاح البحث عن الراتب Search salary: عند النقر على مفتاح البحث عن الراتب Search Salary نكتب الاجراء التالي:

```
Private Sub Command15_Click()
On Error Resume Next
Dim x, y
x = InputBox("Enter salary to be search")
y = " select * from table1 where salary like '" & x & "*'"
Data1.RecordSource = y
Data1.Refresh
End Sub
```

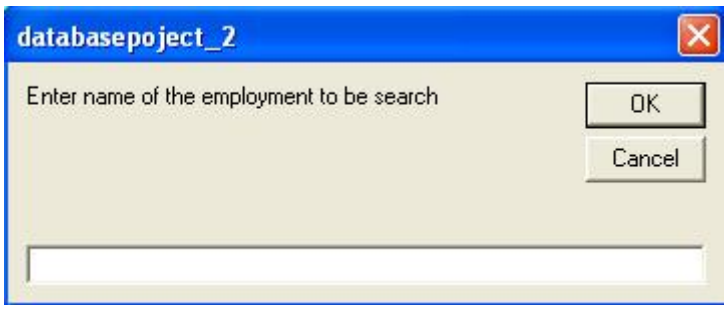
- برمجة المفتاح اوجد اسم الموظف الذي يمتلك اقل راتب Search Name: عند النقر على مفتاح Search Name نكتب الاجراء التالي:

```
Private Sub Command16_Click()
On Error Resume Next
Data1.RecordSource = "select * from table1 order by salary"
Data1.Refresh
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- عند النقر على احد هذه المفاتيح

- سوف تظهر الشاشة التالية:

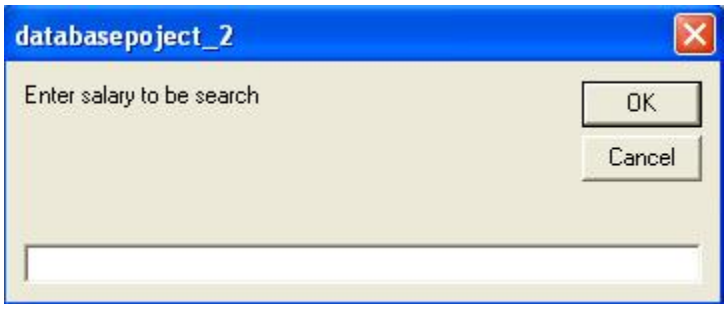


- اذا كان الاسم غير موجود تظهر الشاشة التالية:

PDF created with pdfFactory trial version www.pdffactory.com

- وعند النقر على المفاتيح التالية:

- سوف تظهر الشاشة التالية:



PDF created with pdfFactory trial version www.pdffactory.com

المشروع السابع عشر

-:----------

أنشاء الة حاسبة علمية Scientific

سوف نتعلم من خلال هذا المشروع مايلي:-

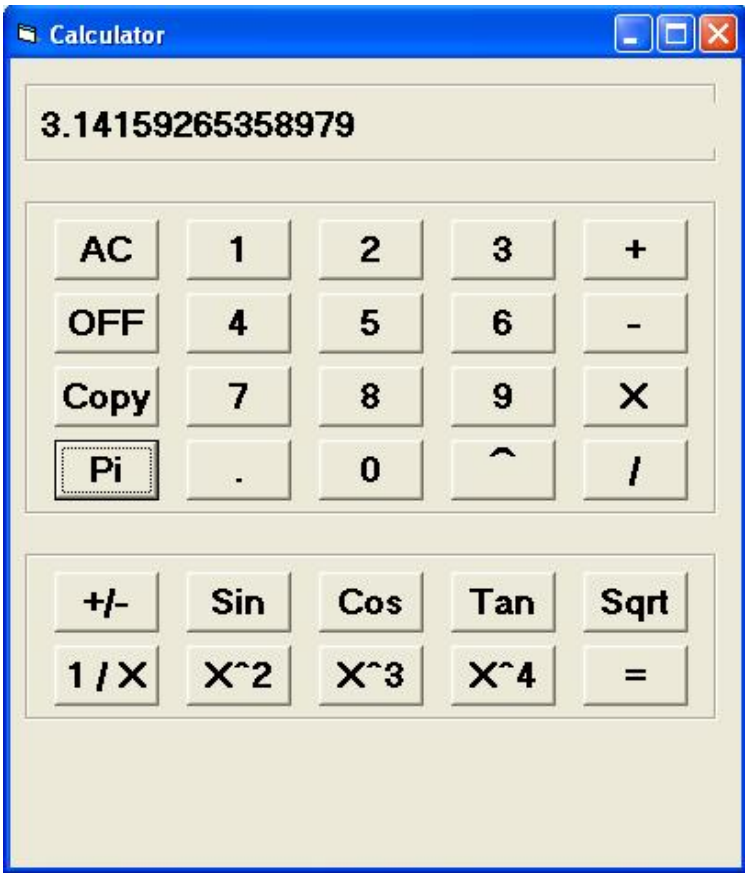
4. التعامل مع عناصر التحكم التاليه:-

- النموذج Form
- التسمية Label
- زر الاوامر Command Button
- الشكل Shape
- أستخدام الدوال الجاهزه مثل Sin, Cos, Tan
- حساب الدوال التالية X^4 , X^3 , X^2 , 1 / X, +/-
- معرفة قيمة الباي Pi=3.141
- مفتاح الاطفاء OFF
- الجذر التربيعي Sqrt
- مفتاح الاستنساخ Copy

الهدف من المشروع

-:----------

بناء واجهة لتحويل حاسبة علمية , عند تنفيذ هذا المشروع تظهر الواجه التاليه:



PDF created with pdfFactory trial version www.pdffactory.com

خطوات تنفيذ المشروع

----------:-

- أنشاء ثلاثة عناصر تحكم من نوع Shape لتقسيم شكل الحاسبة كما هو ظاهر في الشكل أعلاه, حيث ان Shape1 يضم الامر Label1, بينما Shape2 يضم المفاتيح من (1, 2, 3, 4, 5, 6, 7, 8, 9, 0, +, -, *, /, AC, OFF, Copy, Pi) , بينما Shape3 يضم المفاتيح (Sin, Cos, Tan, Sqrt, 1/X, X^2, X^3, X^4 -/+) .
- أنشاء الامر Label1 في عنصر التحكم Shape1
- أنشاء (30) مفتاح من نوع Command, نضع فيها المفاتيح الظاهره في الشكل أعلاه.
- التصريحات العامة في البرنامج هي:

```
Dim n1 As Single
Dim n2 As Single
Dim op As String
```

- برمجة المفتاح AC او مفتاح المسح Clear تتم كالاتي:

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

- برمجة مفتاح عملية الطرح

```
Private Sub Command10_Click()
n1 = Val(Label1.Caption)
op = "-"
Label1.Caption = ""
End Sub
```

- برمجة عملية الاستنساخ Copy

```
Private Sub Command11_Click()
Clipboard.SetText Label1.Caption
End Sub
```

- برمجة مفتاح الارقام 7

```
Private Sub Command12_Click()
Label1.Caption = Label1.Caption + "7"
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- برمجة مفتاح الارقام 8

```
Private Sub Command13_Click()
Label1.Caption = Label1.Caption + "8"
End Sub
```

- برمجة مفتاح الارقام 9

```
Private Sub Command14_Click()
Label1.Caption = Label1.Caption + "9"
End Sub
```

- برمجة مفتاح عملية الضرب X

```
Private Sub Command15_Click()
n1 = Val(Label1.Caption)
op = "*"
Label1.Caption = ""
End Sub
```

- برمجة مفتاح العملية -/+

```
Private Sub Command16_Click()
Label1.Caption = Val(Label1.Caption) - (Val(Label1.Caption) + Val(Label1.Caption))
End Sub
```

- برمجة الفارزة العشرة (.)

```
Private Sub Command17_Click()
If InStr(1, Label1.Caption, ".") = 0 Then
Label1.Caption = Label1.Caption + "."
End If
End Sub
```

- برمجة مفتاح الارقام 0

```
Private Sub Command18_Click()
Label1.Caption = Label1.Caption + "0"
End Sub
```

- برمجة مفتاح المساواة او ناتج العملية (=)

```
Private Sub Command19_Click()
   n2 = Val(Label1.Caption)
   If op = "+" Then
      Label1.Caption = n1 + n2
   ElseIf op = "-" Then
      Label1.Caption = n1 - n2
   ElseIf op = "*" Then
```


PDF created with pdfFactory trial version www.pdffactory.com

```
    Label1.Caption = n1 * n2
  ElseIf op = "^" Then
    Label1.Caption = n1 ^ n2
  ElseIf op = "/" Then
  If n2 = 0 Or Label1.Caption = "" Then
    MsgBox "dont allow to divide by zero", vbCritical, "Error"
  Else
    Label1.Caption = n1 / n2
  End If
  End If
End Sub
```

- برمجة مفتاح الارقام 1

```
Private Sub Command2_Click()
Label1.Caption = Label1.Caption + "1"
End Sub
```

- برمجة عملية القسمة /

```
Private Sub Command20_Click()
n1 = Val(Label1.Caption)
op = "/"
Label1.Caption = ""
End Sub
```

- برمجة النسبة الثابتة Pi

```
Private Sub Command21_Click()
Label1.Caption = "3.14159265358979"
End Sub
```

- برمجة العملية X^2

```
Private Sub Command22_Click()
Label1.Caption = Val(Label1.Caption) ^ 2
End Sub
```

- برمجة العملية X^3

```
Private Sub Command23_Click()
Label1.Caption = Val(Label1.Caption) ^ 3
End Sub
```

- برمجة العملية X^4

```
Private Sub Command24_Click()
Label1.Caption = Val(Label1.Caption) ^ 4
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- برمجة دالة أيجاد الجذر التربيعي Sqrt

```
Private Sub Command25_Click()
If Val(Label1.Caption) >= 0 Then
Label1.Caption = Sqr(Val(Label1.Caption))
Else
MsgBox "There is not squart to this number", vbCritical, "Error"
End If
End Sub
```

- برمجة العملية 1/X

```
Private Sub Command26_Click()
Label1.Caption = 1 / (Val(Label1.Caption))
End Sub
```

- برمجة الدالة Sin

```
Private Sub Command27_Click()
Label1.Caption = Sin(Val(Label1.Caption))

End Sub
```

- برمجة الدالة Cos

```
Private Sub Command28_Click()
Label1.Caption = Cos(Val(Label1.Caption))
End Sub
```

- برمجة الدالة Tan

```
Private Sub Command29_Click()
Label1.Caption = Tan(Val(Label1.Caption))
End Sub
```

- برمجة مفتاح الارقام (2)

```
Private Sub Command3_Click()
Label1.Caption = Label1.Caption + "2"
End Sub
```

- برمجة عملية ايجاد الاس

```
Private Sub Command30_Click()
n1 = Val(Label1.Caption)
op = "^"
Label1.Caption = ""
End Sub
```

- برمجة مفتاح الارقام (3)

```
Private Sub Command4_Click()
Label1.Caption = Label1.Caption + "3"
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- برمجة عملية ايجاد المجموع

```
Private Sub Command5_Click()
n1 = Val(Label1.Caption)
op = "+"
Label1.Caption = ""
End Sub
```

- برمجة مفتاح الاطفاء OFF

```
Private Sub Command6_Click()
Label1.BackColor = vbBlack
Timer1.Enabled = True
End Sub
```

- برمجة مفتاح الارقام (4)

```
Private Sub Command7_Click()
Label1.Caption = Label1.Caption + "4"
End Sub
```

- برمجة مفتاح الارقام (5)

```
Private Sub Command8_Click()
Label1.Caption = Label1.Caption + "5"
End Sub
```

- برمجة مفتاح الارقام (6)

```
Private Sub Command9_Click()
Label1.Caption = Label1.Caption + "6"
End Sub
```

- برمجة عملية تصغير النافذة Minimize عند عملية الاطفاء OFF

```
Private Sub Timer1_Timer()
Form1.WindowState = 1
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الثامن عشر

-:----------

ربط الكائنات مع الفيجوال بيسك (Object Linking Embedded, OLE)

سوف نتعلم من خلال هذا المشروع مايلي:

- ربط الصوت Sound مع الفيجوال بيسك Visual Basic .
- ربط Microsoft Word مع Visual Basic
- ربط Microsoft Power Point مع Visual Basic
- ربط Microsoft Excel مع Visual Basic
- ربط الصورة من الرسام Paint Brush مع Visual Basic.

الهدف من المشروع

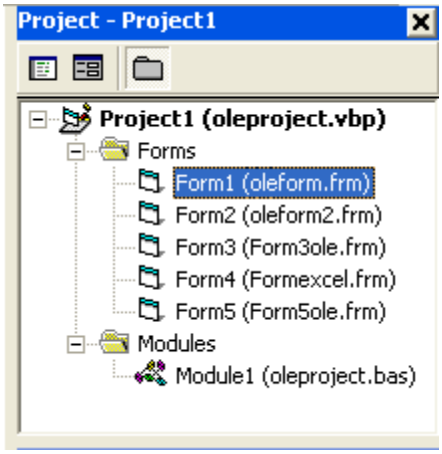
-:----------

عند تنفيذ هذا المشروع سوف نسمع صوتا في النموذج الاول, أضافة الى وجود مفتاح يقودنا الى النموذج الثاني الذي يظهر فية Microsoft Word للدلالة على الربط الصحيح بأستخدام OLE.

مراحل تنفيذ المشروع

-:----------

- نفتح خمسة نماذج قياسية جديدة ووحدة نمطية واحدة كما في الشكل التالي:



- نضع في النموذج الاول ملف صوتي أسمه (chimes.wav) مثلا ونضعة في نفس مسار المشروع أي في الدليل VB98 الموجود في المسار

C:\Program Files\Microsoft Visual studio\vb98

PDF created with pdfFactory trial version www.pdffactory.com

- ننقر على أي موقع في النموذج ونكتب الاجراء التالي:

```
Private Sub Form_Load()
   f = App.Path + "\chimes.wav"
   OLE1.CreateEmbed (f)
   OLE1.DoVerb
End Sub
```

- الغرض من الاجراء السابق هو لتحميل الملف الصوتي chimes.wav الموجود في نفس مسار المشروع عن طريق المتغير العام Public المصرح عنه في الوحدة النمطية Module كما في الايعاز التالي:

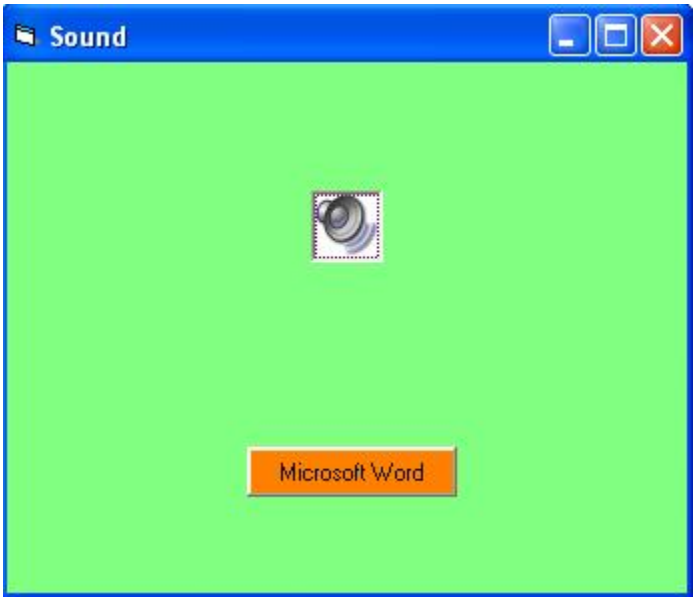
```
Dim f  As String
```

- يتم تحميل الملف chimes.wav عن طريق المتغير (f) الذي يخلق عن طريق العنصر OLE1 كما في الايعاز التالي OLE1.CreateEmbed (f) ومن ثم يفتح هذا الملف عن طريق الايعاز OLE1.DoVerb.

- عند النقر على مفتاح Microsoft Word مرتيين متتاليتين نكتب الاجراء التالي الذي يقوم بأخفاء النموذج الاول وأظهار النموذج الثاني الخاص ب Microsoft Word وكمايلي:

```
Private Sub Command1_Click()
Me.Hide
Form2.Show
End Sub
```

- النموذج التالي يوضح العملية أعلاه.



PDF created with pdfFactory trial version www.pdffactory.com

- نقوم بضبط خصائص عنصر التحكم OLE1 الموجود في النموذج أعلاه كما موضح في الجدول التالي:

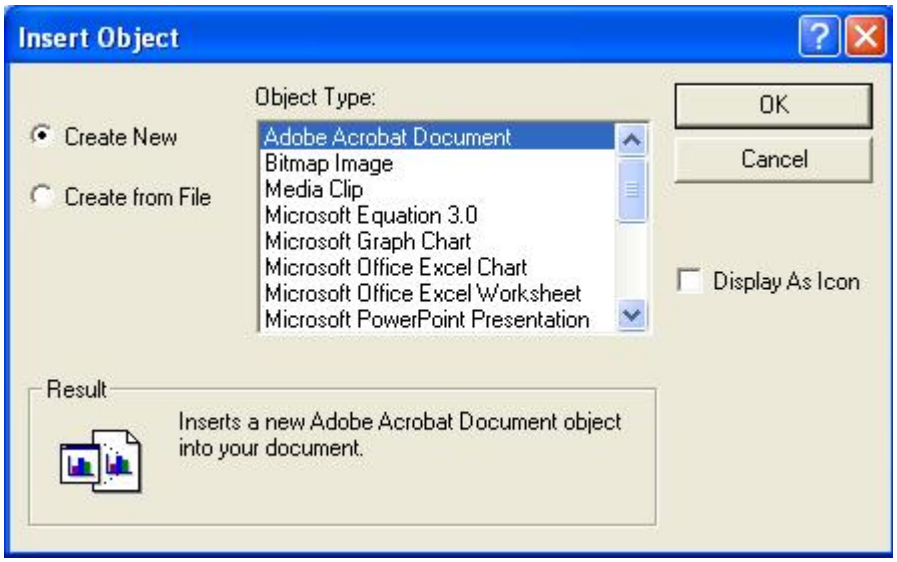
PROPERTIES OF OLE1 خصائص عنصر التحكم نوع OLE1	
Name	OLE1
Size Mode	2 – Auto Size
Source Doc	C:\Program Files\Microsoft Visual Studio\VB98\lijnah.doc

- أي نضع الملف lijnah.doc في نفس مسار المشروع (أي في الدليل VB98)
- أو بنفس الطريقة نكتب الاجراء التالي الذي يقوم بربط Microsoft Word مع Visual Basic

```
Private Sub Form_Load()
   f = App.Path + "\lijnah.doc"
   OLE1.CreateEmbed (f)
   OLE1.DoVerb
End Sub
```

- تصميم العنصر OLE (Object Linking and Embedding)

يستخدم هذا العنصر كحاوية للكائنات, يمكن أن يضم Embedded الكائن في النموذج, أي يقوم بحفظ نسخة ثانية من الكائن في النموذج ولن يكون هناك أي أرتباط مع الكائن الاصلي, كما يمكن أن ترتبط Linked مع كائن بحيث أن أي تعديل في الكائن ينعكس على الكائن المرتبط بالنموذج.
يمكن تضمين أو أرتباط الكائنات عن طريقين, أحدهما في مرحلة التصميم (عن طريق النافذة الحوارية Insert Object التي تظهر عند وضع عنصر التحكم OLE على النموذج, كما في الشكل التالي:



أو في مرحلة التنفيذ بأستخدام الطرق المعرفة على العنصر OLE.

- من أهم خصائص العنصر OLE هي الخاصية SizeMode والتي تأخذ القيم الموضحه في الجدول التالي:

PDF created with pdfFactory trial version www.pdffactory.com

0 - CLIP	يظهر الكائن بحجمه الطبيعي (أذا كان أكبر من العنصر فان يظهر كله)
1 - Stretch	حجم الكائن يطابق حجم العنصر وقد تتغير أبعاد الكائن
2 - AutoSize	حجم العنصر يطابق حجم الكائن بحيث يظهر كل الكائن
3 - Zoom	مطابقة حجم الكائن مع حجم العنصر دون التغيير في أبعاد الكائن

- من أهم طرق العنصر OLE هي:

CREATEEMBED *SOURCEDOC*	تضمين كائن من الملف *SOURCEDOC*
CreateLink *sourcedoc*	الارتباط مع كائن من الملف *sourcedoc*
DoVerb	فتح الكائن

- عند النقر على مفتاح Microsoft Word مرتيين متتاليتين نكتب الاجراء التالي الذي يقوم بأخفاء النموذج الاول وأظهار النموذج الثاني الخاص ب Microsoft Word وكمايلي:

```
Private Sub Command1_Click()
Me.Hide
Form2.Show
End Sub
```

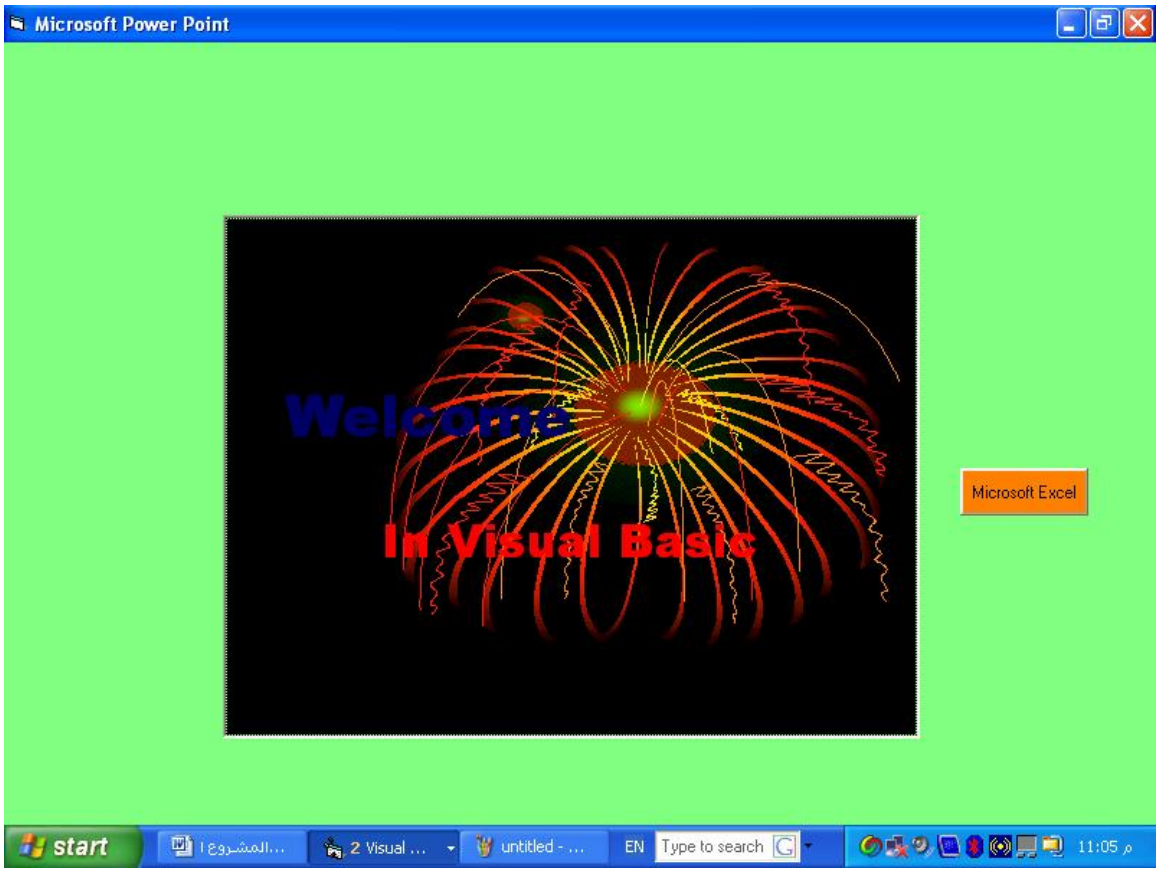
- عند النقر على مفتاح Microsoft Word في عملية تنفيذ المشروع يقوم بنقلنا الى النموذج الخاص بربط Microsoft Word مع Visual Basic وكما موضح في النموذج التالي:

PDF created with pdfFactory trial version www.pdffactory.com

- عند النقر على مفتاح Microsoft Power Point مرتيين متتاليتين نكتب الاجراء التالي الذي يقوم بأخفاء النموذج الثاني وأظهار النموذج الثالث الخاص ب Microsoft Power Point وكمايلي:

```
Private Sub Command1_Click()
Me.Hide
Form3.Show
End Sub
```

- عند النقر على مفتاح Microsoft Power Point في عملية تنفيذ المشروع يقوم بنقلنا الى النموذج الخاص بربط Microsoft Power Point مع Visual Basic وكما موضح في النموذج التالي:



- نكتب الاجراء التالي الذي يقوم بربط ملف Microsoft Power Point الموجود في نفس دليل المشروع أيضا مع Visual Basic

```
Private Sub Form_Load()
   f = App.Path + "\Welcome.ppt"
   OLE1.CreateEmbed (f)
   OLE1.DoVerb
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

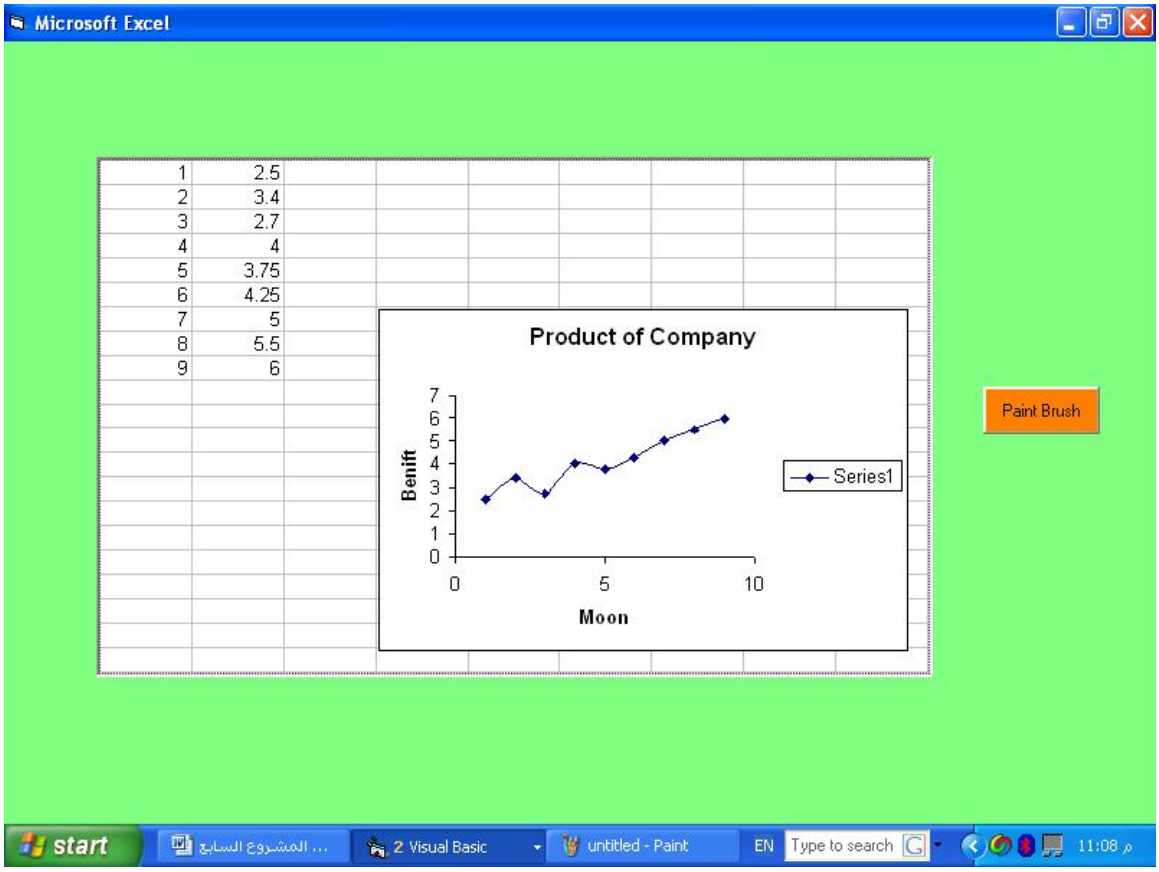
- أوبنفس الطريقة نقوم بضبط خصائص عنصر التحكم OLE1 الموجود في النموذج أعلاه كما موضح في الجدول التالي:

PROPERTIES OF OLE1 خصائص عنصر التحكم نوع OLE1	
Name	OLE1
Size Mode	2 – Auto Size
Source Doc	C:\Program Files\Microsoft Visual Studio\VB98\Welcome.ppt

- عند النقر على مفتاح Microsoft Excel مرتيين متتاليتين نكتب الاجراء التالي الذي يقوم بأخفاء النموذج الثالث وأظهار النموذج الرابع الخاص ب Microsoft Excel وكمايلي:

```
Private Sub Command1_Click()
Me.Hide
Form4.Show
End Sub
```

- عند النقر على مفتاح Microsoft Excel في عملية تنفيذ المشروع يقوم بنقلنا الى النموذج الخاص بربط Microsoft Excel مع Visual Basic وكما موضح في النموذج التالي:



PDF created with pdfFactory trial version www.pdffactory.com

- لاظهار النموذج أعلاه نكتب الاجراء التالي الذي يقوم بربط ملف Microsoft Excel مع Visual Basic

```
Private Sub Form_Load()
   f = App.Path + "\Chart1.xls"
   OLE1.CreateEmbed (f)
   OLE1.DoVerb
End Sub
```

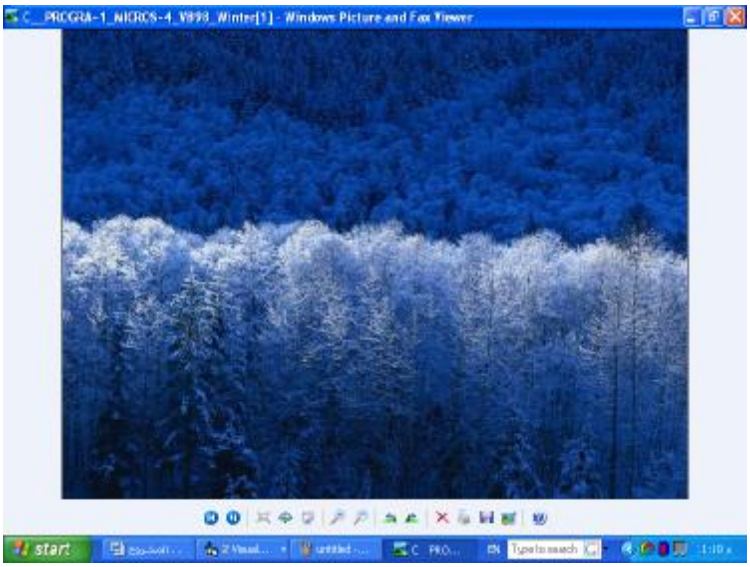
- أوبنفس الطريقة نقوم بضبط خصائص عنصر التحكم OLE1 الموجود في النموذج أعلاه كما موضح في الجدول التالي:

PROPERTIES OF OLE1 خصائص عنصر التحكم نوع OLE1	
Name	OLE1
Size Mode	2 – Auto Size
Source Doc	C:\Program Files\Microsoft Visual Studio\VB98\Chart1.xls

- عند النقر على مفتاح Paint Brush مرتيين متتاليتين نكتب الاجراء التالي الذي يقوم بأخفاء النموذج الرابع وأظهار النموذج الخامس الخاص ب Paint Brush وكمايلي:

```
Private Sub Command1_Click()
Me.Hide
Form5.Show
End Sub
```

- عند النقر على مفتاح Paint Brush في عملية تنفيذ المشروع يقوم بنقلنا الى النموذج الخاص بربط Paint Brush مع Visual Basic , أي تحميل الصورة (Winter.jpg) الموجودة أيضا في نفس مسار المشروع VB98 , وكما موضح في النموذج التالي:



PDF created with pdfFactory trial version www.pdffactory.com

- نكتب الاجراء التالي الذي يقوم بربط ملف Paint Brush مع Visual Basic

```
Private Sub Form_Load()
   f = App.Path + "\Winter.jpg"
   OLE1.CreateEmbed (f)
   OLE1.DoVerb
End Sub
```

- أوبنفس الطريقة نقوم بضبط خصائص عنصر التحكم OLE1 الموجود في النموذج أعلاه كما موضح في الجدول التالي:

PROPERTIES OF OLE1 خصائص عنصر التحكم نوع OLE1	
Name	OLE1
Size Mode	2 – Auto Size
Source Doc	C:\Program Files\Microsoft Visual Studio\VB98\Winter.jpg

PDF created with pdfFactory trial version www.pdffactory.com

المشروع التاسع عشر

-:----------

رسم الخط على النموذج

سوف نتعلم من خلال هذا المشروع مايلي:

- رسم الخط على النموذج بشكل الي أو تلقائي.

الهدف من المشروع

-:----------

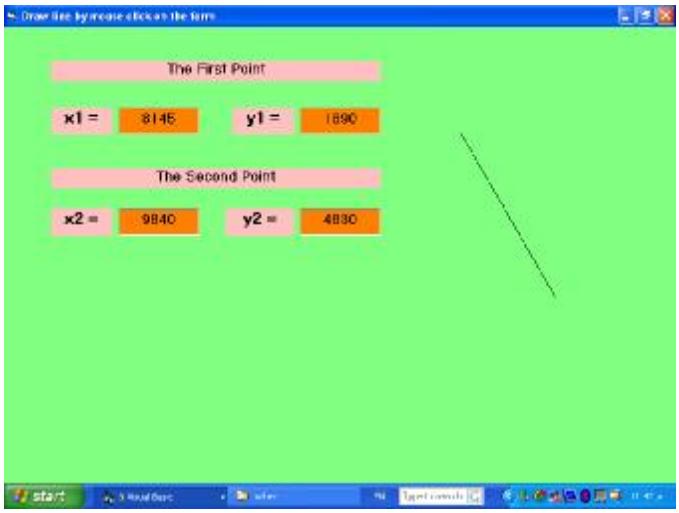
- رسم الخط عن طريق النقر بالماوس على أي موقع في النموذج لغرض تحديد النقطة الاولى, ومن ثم سحب الماوس الى الموقع الثاني على النموذج لغرض تحديد النقطة الثانية.
- يظهر شكل الخط على النموذج بحسب الاتجاة الذي يحدده المستخدم نفسة على النموذج.

تنفيذ المشروع

-:----------

- عند تنفيذ المشروع تظهر الواجهة التالية, التي يتم فيها رسم الخط عن طريق تحديد النقطة الاولى وذلك بالنقر على أي موقع في النموذج, حيث يتم فيه تحديد أحداثيات النقطة الاولى مثلا (x1 = 8145) الذي يمثل الاحداثي السيني للنقطة الاولى و (y1 = 1890) الذي يمثل الاحداثي الصادي للنقطة الاولى أيضا, حيث أنه من المعروف أن تحديد النقطة يتم عن طريق تقاطع الاحداثي السيني مع الاحداثي الصادي.
- نسحب الماوس الى أي موقع أخر على النموذج لغرض تحديد النقطة الثانية, في مثالنا هذا حيث يمثل (x2 = 9840) أحداثي السيني للنقطة الثانية, وأن (y2 = 4830) يمثل الاحداثي الصادي للنقطة الثانية أيضا, فيكون رسم الخط الاول محدد بين النقطتين (8145, 1890) – (9840, 4830) , عند ذلك يتم رسم الخط حسب الايعاز التالي الخاص برسم الخط (الذي يمثل معادلة الخط المستقيم).

Line (x1, y1) – (x2, y2)
Line (8145, 1890) – (9840, 4830)



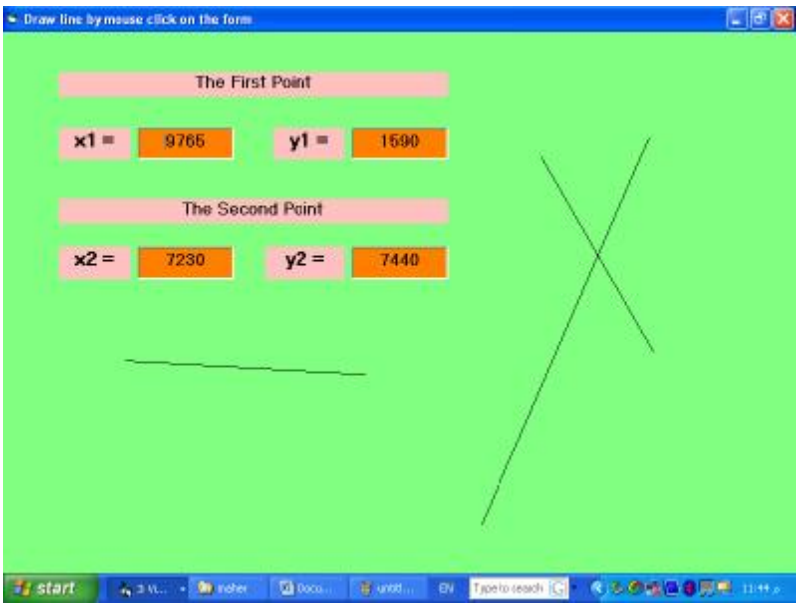
PDF created with pdfFactory trial version www.pdffactory.com

- بنفس الطريقة يتم رسم الخط الثاني على النموذج, حيث يتم تحديد أحداثياته بشكل الي أوتلقائي وكما يلي:

Line (1860, 4950) – (5490, 5160)

- أيضا بنفس الطريقة يتم رسم الخط الثالث على النموذج, حيث تتحدد أحداثياته أيضا بشكل الي أوتلقائي وحسب موقع الماوس على النموذج وكما يلي:

Line (9765, 1590) – (7230, 7440)



PDF created with pdfFactory trial version www.pdffactory.com

- وهكذا بالنسبة لرسم بقية الخطوط الاخرى.

مراحل تنفيذ المشروع

-:----------

- نفتح مشروعا قياسيا جديدا.
- نضع على النموذج ستة عناصر تحكم من نوع label, ونضبط خصائص العنوان Caption لها كمايلي:

LABEL1	THE FIRST POINT
Label2	X1 =
Label3	Y1 =
Label4	The Second Point
Label 5	X2 =
Label6	Y2 =

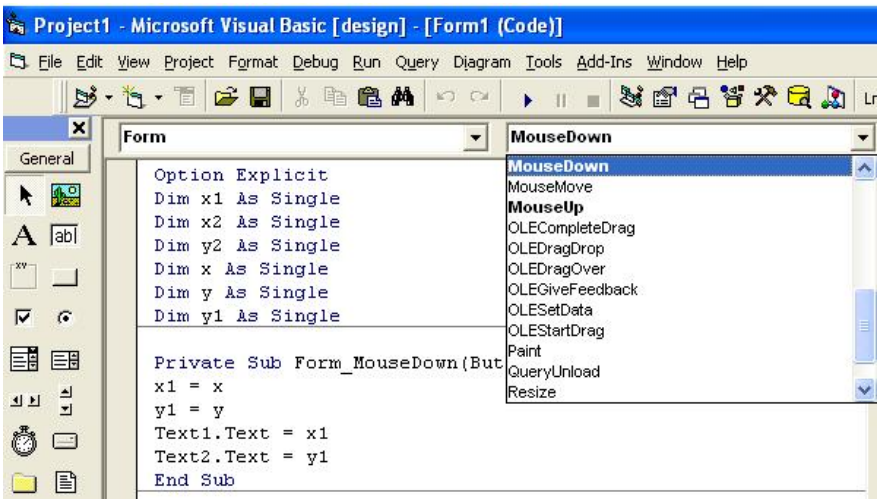
- نضع على النموذج أربعة عناصر تحكم من نوع Text Box كما موضح في الجدول التالي:

TEXT1	لتحديد الاحداثي السيني للنقطة الاولى
Text2	لتحديد الاحداثي الصادي للنقطة الاولى
Text3	لتحديد الاحداثي السيني للنقطة الثانية
Text4	لتحديد الاحداثي الصادي للنقطة الثانية

- يتم تعريف المتغيرات التالية الخاصة بالنموذج وكمايلي:

```
Option Explicit
Dim x1 As Single
Dim x2 As Single
Dim y2 As Single
Dim x As Single
Dim y As Single
Dim y1 As Single
```

- ننقر مرتين متتاليتين على النموذج ونقوم بتغير الحدث الى MouseDown كما في الشكل التالي:

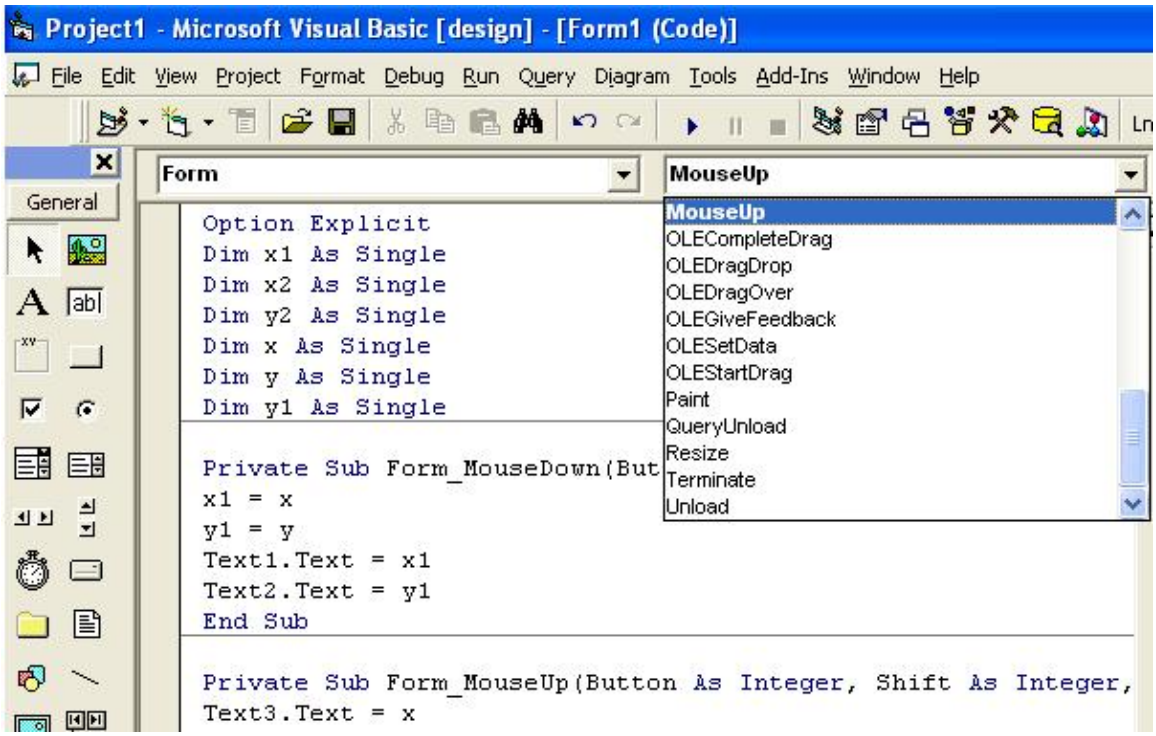


PDF created with pdfFactory trial version www.pdffactory.com

- عند ذلك نقوم بكتابة الاجراء التالي:

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, x As Single, y As Single)
x1 = x
y1 = y
Text1.Text = x1
Text2.Text = y1
End Sub
```

- بنفس الطريقة ننقر مرتين متتاليتين على النموذج ونقوم بتغير الحدث الى MouseUp كمايلي:



- ومن ثم نكتب الاجراء التالي:

```
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, x As Single, y As Single)
Form1.AutoRedraw = True
Text3.Text = x
Text4.Text = y
x2 = x
y2 = y
Line (x1, y1)-(x2, y2), 1
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- حيث أن الاجراء الاول يقوم بتحديد النقطة الاولى, والاجراء الثاني يقوم بتحديد النقطة الثانية لرسم الخط.

المشروع العشرون

-:----------

رسم المستطيل على النموذج

سوف نتعلم من خلال هذا المشروع مايلي:

- رسم المستطيل وتحديد نقاطه بشكل الي أو تلقائي عن طريق النقر بالماوس على النموذج لتحديد النقطة الاولى التي تمثل رأس المستطيل والنقطة الثانية التي تمثل الرأس المقابل لها.

الهدف من المشروع

-:----------

- رسم المستطيل وبأبعاد مختلفة عن طريق النقر على أي موقع في النموذج لغرض تحديد النقطة الاولى, ومن ثم سحب الماوس الى الموقع الثاني على النموذج لغرض تحديد النقطة الثانية.
- يظهر شكل المستطيل على النموذج بحسب المقدار الذي يحدده المستخدم نفسة على النموذج.

تنفيذ المشروع

-:----------

- عند تنفيذ المشروع تظهر الواجهة التالية, التي يتم فيها رسم المستطيل عن طريق تحديد رأس المستطيل الذي يمثل النقطة الاولى وذلك بالنقر على أي موقع في النموذج, حيث يتم فيه تحديد أحداثيات النقطة الاولى عن طريق مثلا (x1 = 4410) الذي يمثل الاحداثي السيني للنقطة الاولى و (y1 = 4365) الذي يمثل الاحداثي الصادي للنقطة الاولى أيضا, حيث أنه من المعروف أن تحديد النقطة يتم عن طريق تقاطع الاحداثي السيني مع الاحداثي الصادي.
- نسحب الماوس الى أي موقع أخر على النموذج لغرض تحديد النقطة الثانية, حيث يمثل (x2 =8055) أحداثي السيني للنقطة الثانية, وأن (y2 =5220) يمثل الاحداثي الصادي للنقطة الثانية أيضا, فيكون رسم المستطيل الاول محدد بين النقطتين (4410, 4365) – (8055, 5220) , عند ذلك يتم رسم المستطيل حسب الايعاز التالي الخاص به (الذي يمثل معادلة الخط المستقيم, مضافا اليها الحرف B الذي يمثل الايعاز Box أو الصندوق).

```
Line (x1, y1) – (x2, y2), B
Line (4410, 4365) – (8055, 5220), B
```

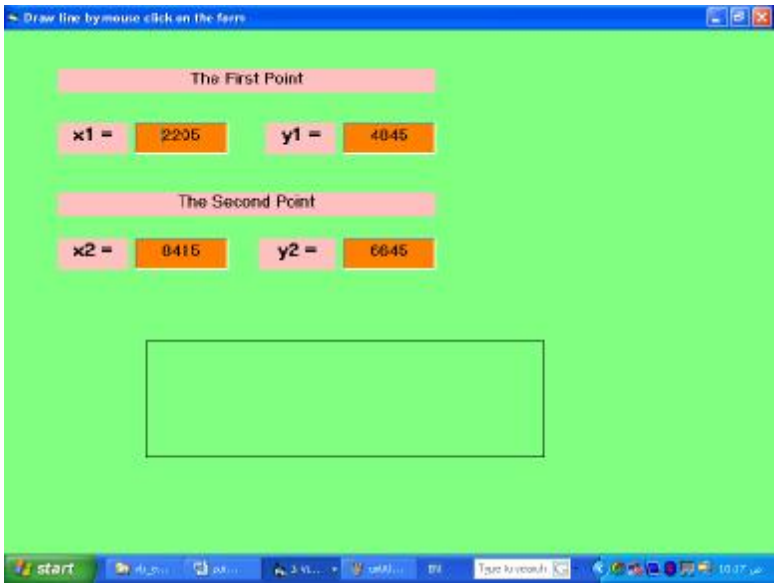
PDF created with pdfFactory trial version www.pdffactory.com

- بنفس الطريقة يتم رسم المستطيل الثاني على النموذج, حيث يتم تحديد أحداثياته بشكل الي أوتلقائي وكما يلي:

Line (7575,1995) – (11145,3720), B

- بنفس الطريقة أيضا يتم رسم المستطيل الثالث على النموذج, حيث تتحدد أحداثياته أيضا بشكل الي أوتلقائي أيضا وحسب موقع النقر على النموذج وكما يلي:

Line (2205,4845) – (8415,6645), B



PDF created with pdfFactory trial version www.pdffactory.com

- وهكذا بالنسبة لرسم بقية المستطيلات الاخرى.

مراحل تنفيذ المشروع

-:----------

- نفتح مشروعا قياسيا جديدا.
- نضع على النموذج ستة عناصر تحكم من نوع label, ونضبط خصائص العنوان Caption لها كمايلي:

LABEL1	THE FIRST POINT
Label2	X1 =
Label3	Y1 =
Label4	The Second Point
Label 5	X2 =
Label6	Y2 =

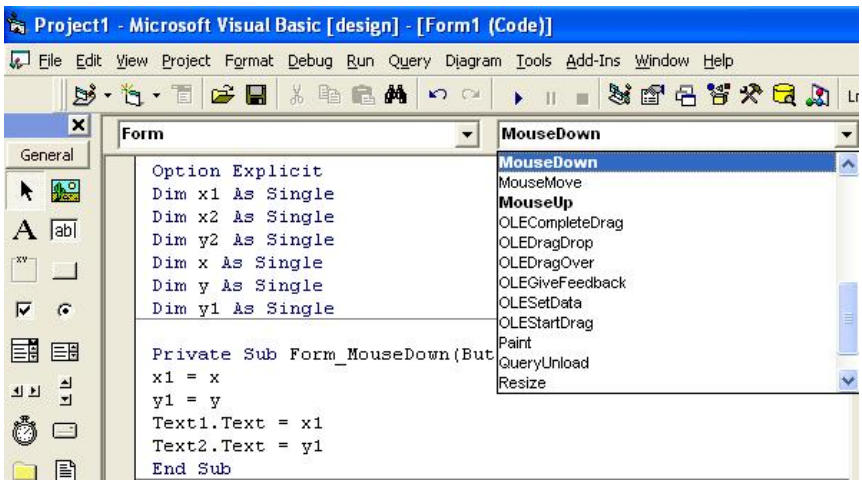
- نضع على النموذج أربعة عناصر تحكم من نوع Text Box كما موضح في الجدول التالي:

TEXT1	لتحديد الاحداثي السيني للنقطة الاولى
Text2	لتحديد الاحداثي الصادي للنقطة الاولى
Text3	لتحديد الاحداثي السيني للنقطة الثانية
Text4	لتحديد الاحداثي الصادي للنقطة الثانية

- يتم تعريف المتغيرات التالية الخاصة بالنموذج وكمايلي:

```
Option Explicit
Dim x1 As Single
Dim x2 As Single
Dim y2 As Single
Dim x As Single
Dim y As Single
Dim y1 As Single
```

- ننقر مرتين متتاليتين على النموذج ونقوم بتغير الحدث الى MouseDown كما في الشكل التالي:

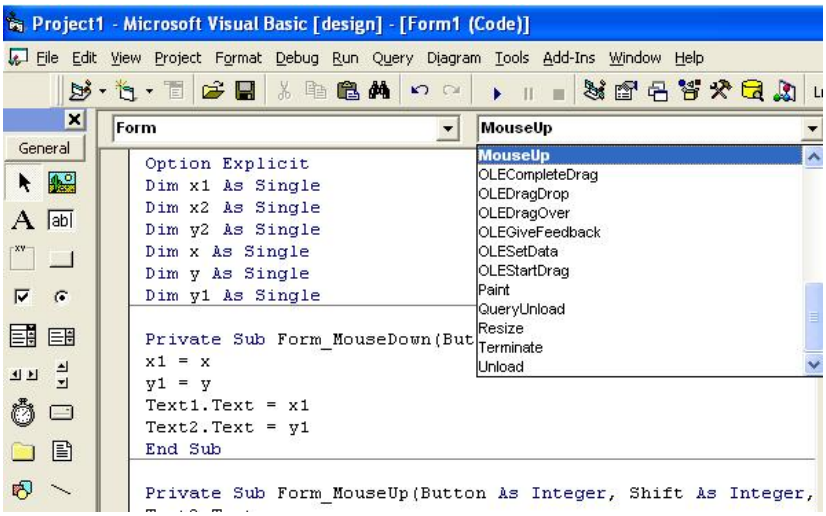


PDF created with pdfFactory trial version www.pdffactory.com

- عند ذلك نقوم بكتابة الاجراء التالي:

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, x As Single, y As Single)
x1 = x
y1 = y
Text1.Text = x1
Text2.Text = y1
End Sub
```

- بنفس الطريقة ننقر مرتين متتاليتين على النموذج ونقوم بتغير الحدث الى MouseUp كما في الشكل التالي:



- ومن ثم نكتب الاجراء التالي:

```
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, x As Single, y As Single)
Text3.Text = x
Text4.Text = y
x2 = x
y2 = y
Line (x1, y1)-(x2, y2), 1, B
End Sub
```

- حيث أن الاجراء الاول يقوم بتحديد النقطة الاولى, والاجراء الثاني يقوم بتحديد النقطة الثانية لرسم المستطيل.

PDF created with pdfFactory trial version www.pdffactory.com

المشروع الحادي والعشرون

-:----------

رسم الدائرة

سوف نتعلم من هذا المشروع رسم الدائرة الالي عن طريق النقر بالماوس على النموذج.

الهدف من المشروع

-:----------

رسم الدائرة على النموذج تلقائيا عن طريق النقر بالماوس .

تنفيذ المشروع

-:----------

- عند النقر بالماوس على أي موقع في النموذج فأنه سيمثل مركز الدائرة الجديد الذي نصف قطرها مقدار ثابت يساوي 1000.

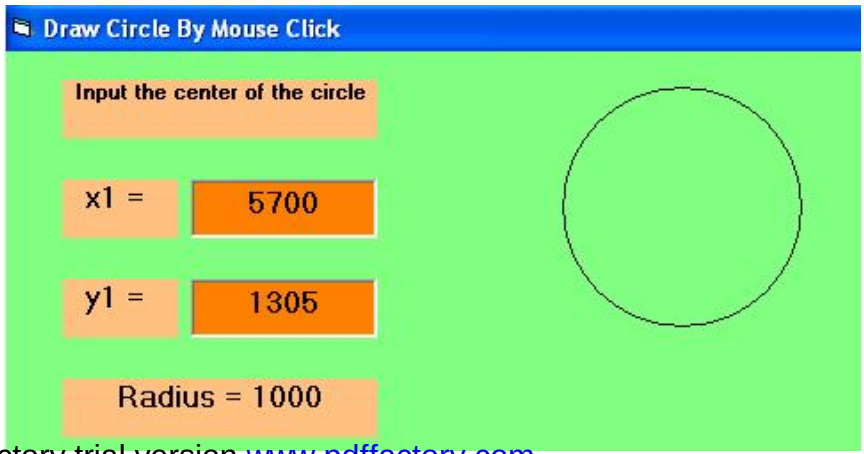
مراحل تنفيذ المشروع

-:----------

- نفتح مشروعا قياسيا جديدا.
- ننشيء أربعة عناصر تسمية من نوع Label ونغير العنوان (Caption) لها كما موضح في الجدول التالي:

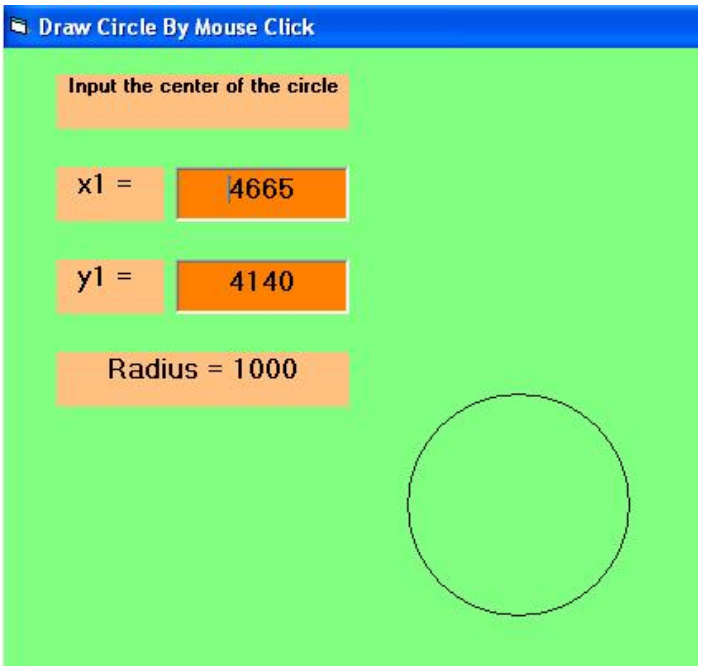
LABEL1	INPUT THE CENTER OF THE CIRCLE
Label2	x1 =
Label3	y1 =
Label4	Radius

- ننشيء عنصري تحكم من Text Box , حيث أن Text1 تمثل قيمة المحور السيني للنقطة التي تمثل مركز الدائرة, بينما Text2 تمثل قيمة المحور الصادي للنقطة نفسها, وبذلك عن طريق تحديد قيمة النقطة (Text1, Text2) سوف يتحدد مركز الدائرة كما في المثال الظاهر في النموذج التالي:



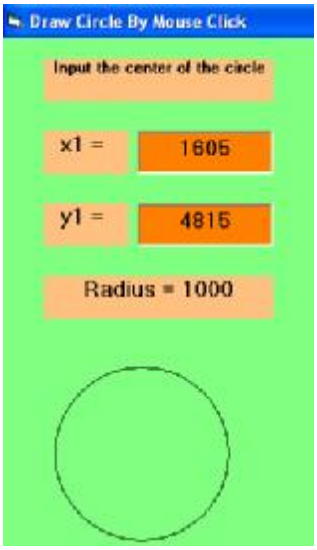
PDF created with pdfFactory trial version www.pdffactory.com

- حيث أن مركز الدائرة الموضح في الشكل الاعلى هو النقطة التي تمثل (x1, y1) أي النقطة التي أحداثياتها هي (5700, 1305) , حيث أن هذه القيم تتحدد تلقائيا عند النقر بالماوس على أي موقع في النموذج, وهذه القيم تتغير في كل مرة يتغير فيها موقع النقر على النموذج, كما في المثال التالي:



- حيث أن موقع مركز الدائرة الجديد هو (4665, 4140) , وفي كل مرة ننقر بالماوس على النموذج سوف يتحدد موقع مركز الدائرة تبعا للموقع الجديد, ألا أن حجم هذه الدوائر يكون ثابت بسبب ثبوت قيمة نصف قطر الدائرة والذي يساوي في مثالنا هذا المقدار (1000)

- في الشكل التالي يكون موقع مركز الدائرة الجديد هو (1605, 4815)



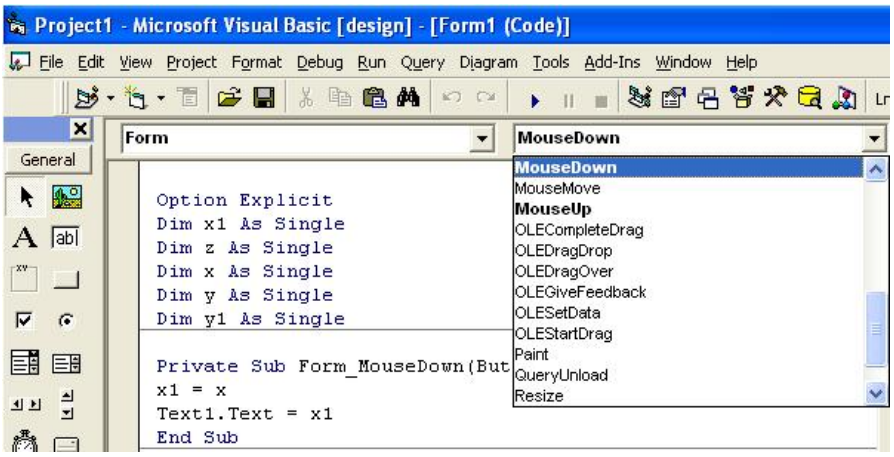
PDF created with pdfFactory trial version www.pdffactory.com

يتم رسم الدائرة بالاعتماد على المعادلة التالية الخاصة بالدائرة وهي Circle (x1, x2), Radius حيث أن النقطة (x1, x2) تمثل مركز الدائرة بينما Radius يمثل نصف قطر الدائرة.

- يتم تعريف المتغيرات المستخدمة في النموذج كمايلي:

```
Option Explicit
Dim x1 As Single
Dim z As Single
Dim x As Single
Dim y As Single
Dim y1 As Single
```

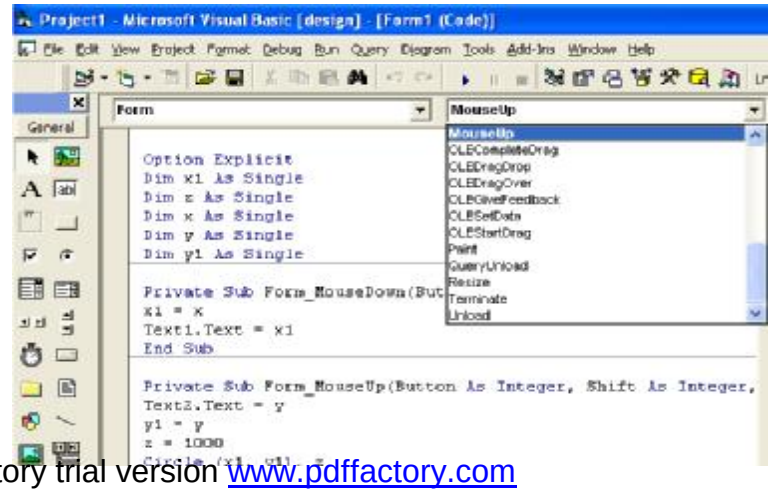
- عند النقر بالماوس مرتتين متتاليتين على النموذج, نقوم بتغير الحدث الى MouseDown كما في الشكل التالي:



- ثم نكتب الاجراء التالي:

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, x As Single, y As Single)
x1 = x
Text1.Text = x1
End Sub
```

- بنفس الطريقة ننقر مرتين متتاليتين على النموذج ونغير الحدث الى MouseUp كما موضح في الشكل التالي:



PDF created with pdfFactory trial version www.pdffactory.com

- ثم نكتب الاجراء التالي الخاص الذي يمثل موقع الحدث MouseUp

```
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, x As Single, y As Single)
Text2.Text = y
y1 = y
z = 1000
Circle (x1, y1), z
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الثاني والعشرون

-:----------

تغيير نوع النص, حجم النص, لون النص, وطبيعة النص

سوف نتعلم من خلال تنفيذ هذا المشروع مايلي:

- تغيير نوع كتابة النص.
- تغيير حجم النص.
- تغيير لون النص.
- تغير نمط النص, غامق Bold, مائل Italic, وضع خط تحت النص Under Line.

الهدف من المشروع

-:----------

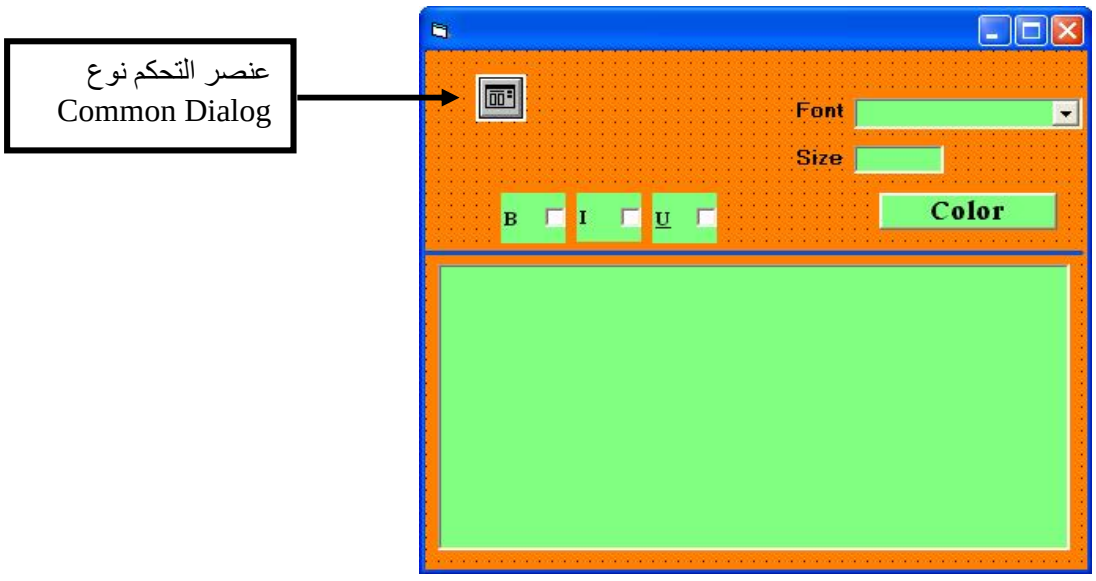
كيفية ترتيب النص من حيث نوع الخط, الحجم, اللون, وطبيعة كتابة النص.

مراحل تنفيذ المشروع

-:----------

- نفتح مشروعا قياسيا جديدا, ونضع فيه عناصر التحكم التالية:
- ComboBox
- صندوق نص عدد أثنان واحد لتحديد حجم الخط, والثاني لكاتبة النص ومن ثم معالجته.
- مفتاح الامر CommandButton1 , تغيير عنوانه (Caption) الى Color.
- ثلاثة عناصر تحكم من نوع Check, ونغير عناوينها الى B ويعني Bold, و I ويعني Italic , و U ويعني UnderLine.
- عناصر تحكم نوع Label عدد أثنان , نغيير العناوين (Caption) لها Font, Size على التوالي:

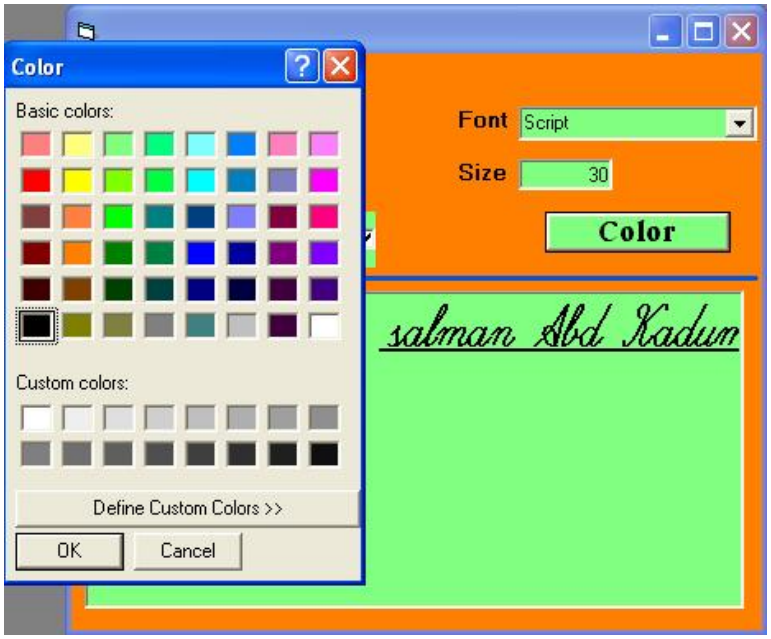
كما موضح في الشكل التالي:



PDF created with pdfFactory trial version www.pdffactory.com

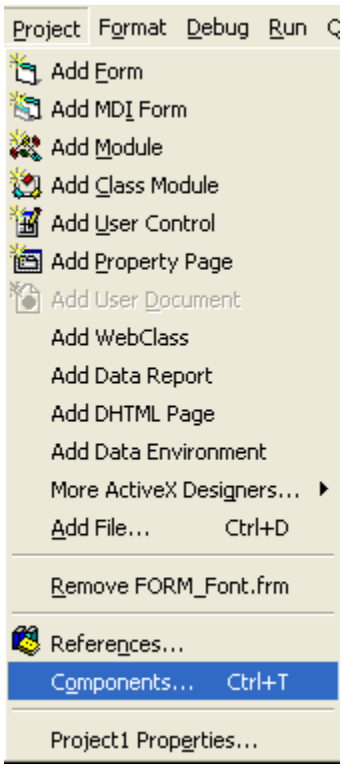
- نكتب النص المطلوب كما هو مبين في الشكل التالي, نقوم بتغيير نوع الخط مثلا الى Script من خلال Combo Box وحجم الخط من خلال Text Box , وطبيعة الخط من حيث وضع خط تحت الكلمة, مائل, أوغامق وذلك من خلال تفعيل U, I, B على التوالي.

- تغيير لون النص يتم من خلال النقر على مفتاح Color , كما مبين في الشكل التالي:

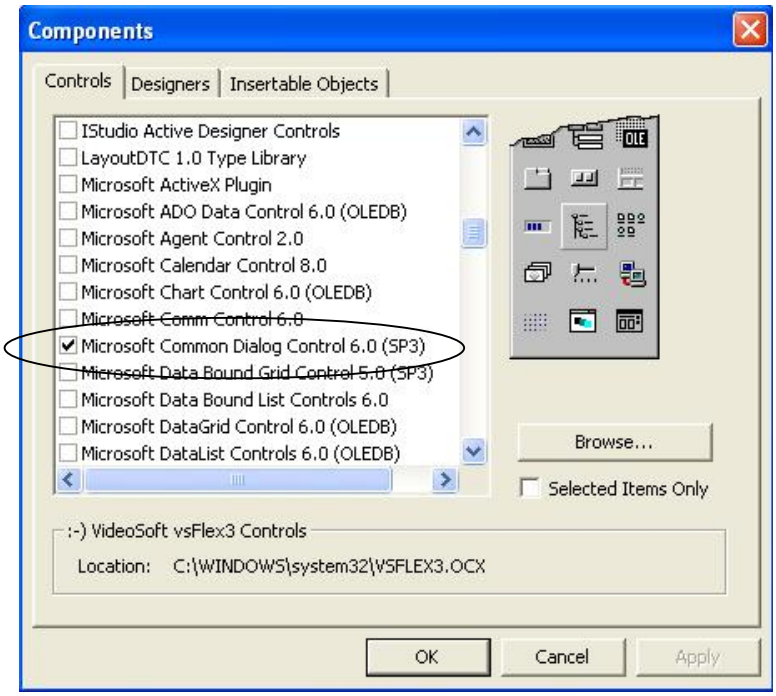


PDF created with pdfFactory trial version www.pdffactory.com

- تم أضافة عنصر جديد New Components من نوع CommonDialog لغرض تفعيل نافذة حوار الالون من خلال قائمة المشروع Project ومن ثم نختار Component كما موضح في الشكل التالي:



- عند أختيار Components تظهر النافذة التالية التي نختار منها الاختيار المؤشر بالشكل البيضوي الموضح بالشكل التالي لغرض تفعيل خاصية أو عنصر التحكم CommonDialog



PDF created with pdfFactory trial version www.pdffactory.com

- أي نقوم بتفعيل Microsoft Common Dialog Control 6.0 (SP3) كما هو مبين في الشكل الموجود في الاعلى.
- برمجة المفتاح Check1 الخاص بنمط الخط الايطالي (I), عند النقر على مفتاح Check1 مرتين متتاليتين نكتب الاجراء التالي:

```
Private Sub Check1_Click()
Text1.FontItalic = Check1.Value
End Sub
```

- برمجة المفتاح Check2 الخاص بوضع خط تحت النص (U) أو مايسمي ب Under Line, ننقر على مفتاح Check2 مرتين متتاليتين ونكتب الاجراء التالي:

```
Private Sub Check2_Click()
Text1.FontUnderline = Check2.Value
End Sub
```

- برمجة المفتاح Check3 الخاص بجعل الخط غامق (B), ننقر على مفتاح Check3 مرتين متتاليتين ونكتب الاجراء التالي:

```
Private Sub Check3_Click()
Text1.FontBold = Check3.Value
End Sub
```

- برمجة عنصر التحكم Combo Box : عند النقر على الكائن Combo Box مرتيين متتاليتين, نكتب الاجراء التالي:

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

- برمجة المفتاح Color: عند النقر على المفتاح Color مرتين متتاليتين نكتب لاجراء التالي:

```
Private Sub Command1_Click()
CommonDialog1.ShowColor
Text1.ForeColor = CommonDialog1.Color
End Sub
```

- عند النقر على أي موقع في النموذج نكتب الاجراء التالي:

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


PDF created with pdfFactory trial version www.pdffactory.com

```
Combo1.Text = Combo1.List(0)
End Sub
```
الغرض من الاجراء أعلاه هو لاظهار كافة أنواع الخطوط المخصصة للشاشة وحسب الانواع المتوفرة في الوندوز أو الاوفس.

- برمجة المفتاح Text2 : عند النقر مرتين متتاليتين على Text2 نكتب الاجراء التالي لغرض أظهار حجم النصوص Size.

```
Private Sub Text2_Change()
On Error GoTo 11
Text1.FontSize = Text2.Text
Exit Sub
11:
Text2.Text = ""
MsgBox "ÚÝæÇð ÃÏÎá ÑÞãÇð ÝÞØ", vbMsgBoxRight, "aMbIgUoUs"
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الثالث والعشرون

-:----------

رسم الاشكال الهندسية (المضلع, النجمة)

سوف نتعلم من خلال تنفيذ هذا المشروع مايلي:

- رسم مضلع Polygon متعدد الاضلاع.
- رسم نجمة Star متعددة الاضلاع.

الهدف من المشروع

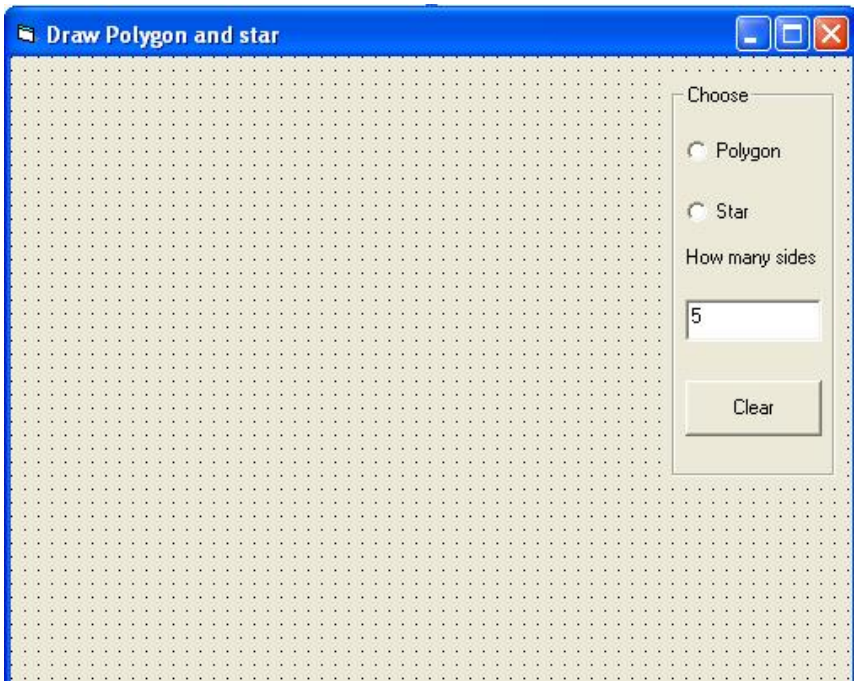
-:----------

رسم الاشكال الهندسية مثل المضلع والنجمة وبأضلاع مختلفة يحددها المستخدم نفسة.

مراحل تنفيذ المشروع

-:----------

- نفتح مشروعا قياسيا جديدا.
- نضع عناصر التحكم التالية على النموذج مثل الاطار Frame الذي يستخدم كحاوية لبقية العناصر المستخدمة في المشروع, كالخيار الاول Option1 مثلا الذي نقوم بتغيير عنوانه Caption الى مضلع (Polygon), والخيار الثاني Option2 الذي يكون عنوانة نجمة (Star), وتسمية واحدة Label1 الذي يكون عنوانه How many sides الذي يستخدم لتحديد عدد الاضلاع المستخدمة في رسم المضلع أو النجمة, حيث أن عدد الاضلاع يتم تحديدها في صندوق النص Text1 الذي تكون القيمة الافتراضية له (5) أضلاع, قابلة للتغير حسب رغبة المستخدم, ومفتاح من نوع Command Button الذي نقوم بتغيير عنوانة الى Clear لغرض مسح النموذج, كما مبين في الشكل التالي:



PDF created with pdfFactory trial version www.pdffactory.com

- نقوم بتعريف المتغيرات التالية الخاص بالنموذج وكمايلي:

```
Dim sta As Double
Dim color As Variant
Dim a As Variant
Dim r As Variant
Dim r2 As Variant
Dim cc As Double
Dim n As Double
Dim x1 As Double
Dim x2 As Double
Dim yc As Double
Dim xc As Double
Dim ycc As Double
Dim xcc As Double
Dim y1 As Double
Dim y2 As Double
Dim m As Double
Dim pi As Double
Dim Y As Double
Dim X As Double
```

- برمجة الخيار الاول Option1 الخاص برسم المضلع أي Polygon
عند النقر على هذا الخيار نكتب الاجراء التالي الذي يقوم برسم المضلع:

```
Public Sub poly(X, Y)
If (cc = 1) Then
n = Val(Text1.Text)
  pi = 44 / 7
  For i = 1 To n
    x1 = xc + r * Cos(a + i * pi / n)
    y1 = yc + r * Sin(a + i * pi / n)
    x2 = xc + r * Cos(a + (i + 1) * pi / n)
    y2 = yc + r * Sin(a + (i + 1) * pi / n)
    Line (x1, y1)-(x2, y2), RGB(255, 255, 255)
  Next
  r = ((xc - X) ^ 2 + (yc - Y) ^ 2) ^ (0.5)
  If ((xc - X) = 0 And (yc < Y)) Then
     a = (0.25) * pi
  Else
    If ((xc - X) = 0 And (yc > Y)) Then
      a = (0.75) * pi
    Else
      m = (yc - Y) / (xc - X)
      a = Atn(m)
```


PDF created with pdfFactory trial version www.pdffactory.com

```
      a = I
          If(xc < X, a, a - (pi / (n * (1 + (n Mod 2)))))
    End If
  End If
  For i = 1 To n
    x1 = xc + r * Cos(a + i * pi / n)
    y1 = yc + r * Sin(a + i * pi / n)
    x2 = xc + r * Cos(a + (i + 1) * pi / n)
    y2 = yc + r * Sin(a + (i + 1) * pi / n)
    Line (x1, y1)-(x2, y2), color
  Next
End If
End Sub
```

- برمجة الخيار الثاني Option2 الخاص برسم النجمة Star

عند النقر مرتين متتاليتين على الخيار الثاني نكتب الاجراء التالي الخاص برسم النجمة التي يتحدد عدد أضلاعها من قبل المستخدم.

```
Public Sub star(X, Y)
If (sta <> 0) Then
n = Val(Text1.Text)
n = 2 * n
pi = 44 / 7

  For i = 1 To n
    z = r
    r = r2
    r2 = z
    x1 = xcc + r * Cos(a + i * pi / n)
    y1 = ycc + r * Sin(a + i * pi / n)
    x2 = xcc + r2 * Cos(a + (i + 1) * pi / n)
    y2 = ycc + r2 * Sin(a + (i + 1) * pi / n)
    Line (x1, y1)-(x2, y2), RGB(255, 255, 255)
  Next
  r = ((xcc - X) ^ 2 + (ycc - Y) ^ 2) ^ (0.5)
  If (sta = 1) Then
 r2 = r
End If
  If ((xcc - X) = 0 And (ycc < Y)) Then
    a = (0.25) * pi
  Else
    If ((xcc - X) = 0 And (ycc > Y)) Then
      a = (0.75) * pi
    Else
      m = (ycc - Y) / (xcc - X)
      a = Atn(m)
      a = IIf(xcc < X, a, a - (pi / (n * (1 + (n Mod 2)))))
```


PDF created with pdfFactory trial version www.pdffactory.com

```
    End If
  End If
  For i = 1 To n
  z = r
    r = r2
    r2 = z
    x1 = xcc + r * Cos(a + i * pi / n)
    y1 = ycc + r * Sin(a + i * pi / n)
    x2 = xcc + r2 * Cos(a + (i + 1) * pi / n)
    y2 = ycc + r2 * Sin(a + (i + 1) * pi / n)
    Line (x1, y1)-(x2, y2), color
  Next
End If
End Sub
```

- برمجة المفتاح Clear
  يستخدم هذا المفتاح لغرض مسح النموذج , عند النقر على هذا المفتاح مرتيين متتاليتين نكتب الاجراء التالي الذي يقوم بمسح النموذج.

```
Private Sub Command1_Click()
Me.Cls
End Sub
```

- ننقر مرتيين متتاليتين على أي موقع في النموذج ونغيير الحدث الى MouseDown الذي يقوم بتحديد الموقع الاول لرسم الشكل الهندسي, بعد ذلك نكتب الاجراء التالي:

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
If (sta = 0) Then
  xcc = X
  ycc = Y
End If
  xc = X
  yc = Y
  cc = 1
End Sub
```

- بنفس الطريقة ننقر مرتيين متتاليتين على أي موقع في النموذج ونغير الحدث الى MouseUp الذي يقوم بتحديد الموقع الثاني لرسم الشكل الهندسي بعد رفع اليد من الماوس, بعد ذلك نكتب الاجراء التالي:

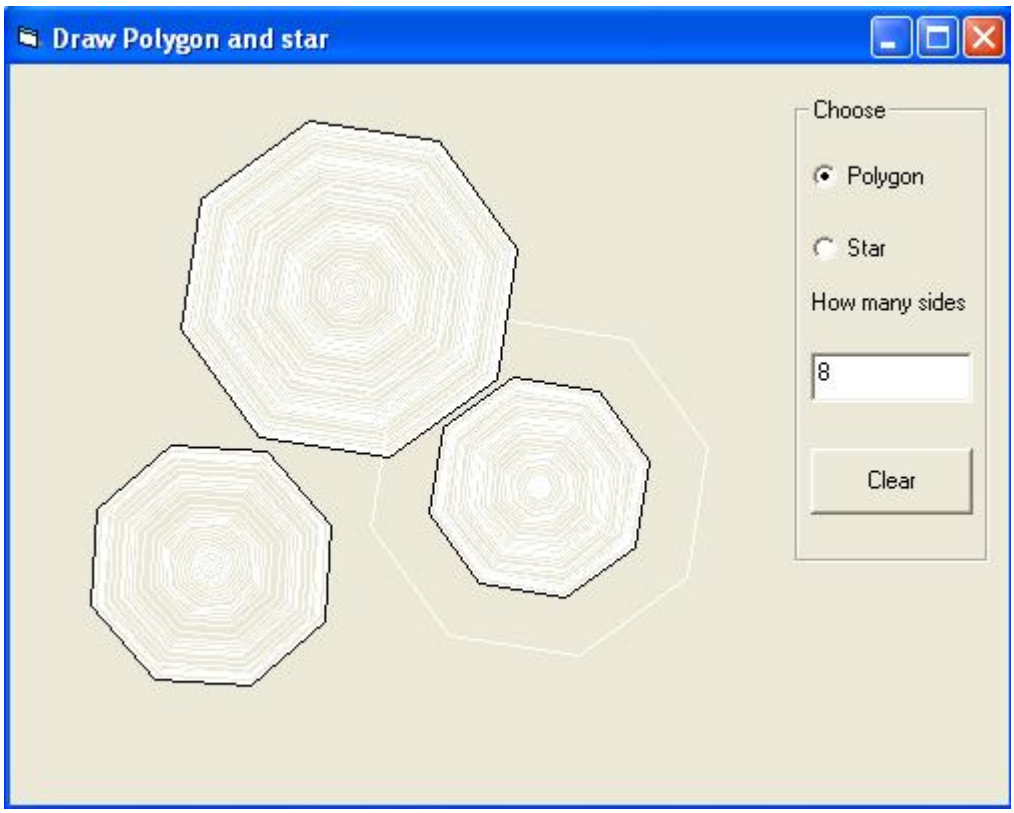
```
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
cc = 0
sta = I
If(sta = 2, 0, sta + 1)
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

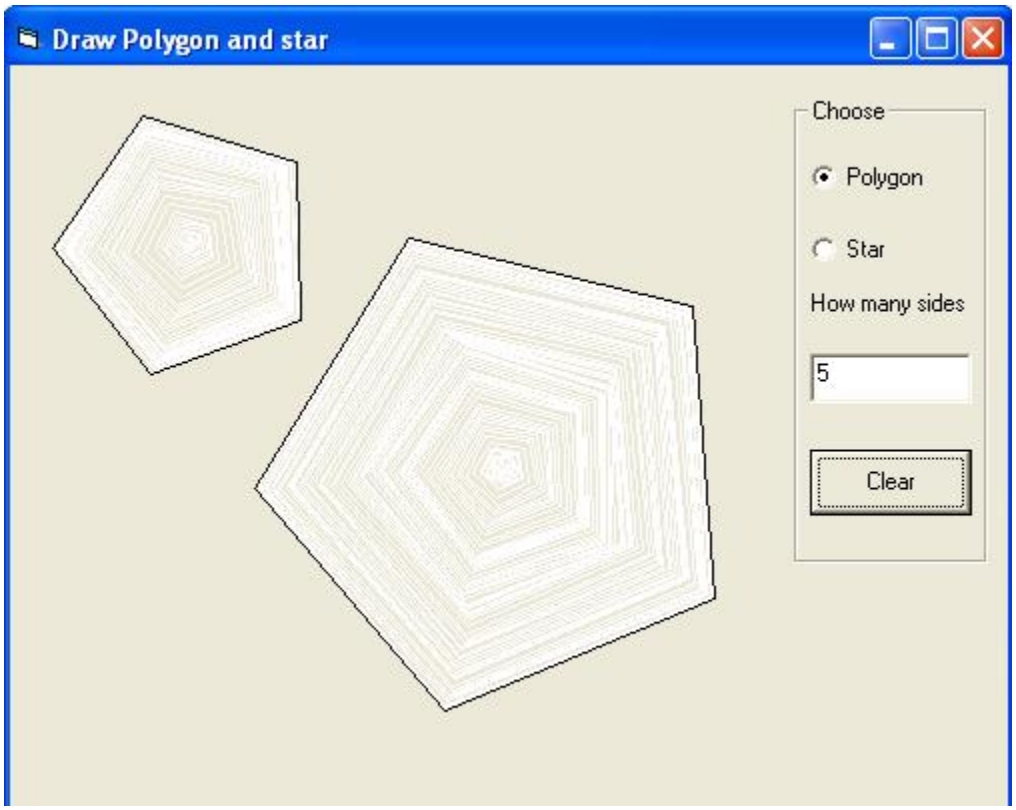
- وبنفس الطريقة ننقر مرتيين متتاليتين على أي موقع في النموذج ونغير الحدث الى MouseMove الذي يقوم بحرية تحريك الشكل الهندسي حسب رغبة المستخدم لتحديد شكلة, بعد ذلك نكتب الاجراء التالي

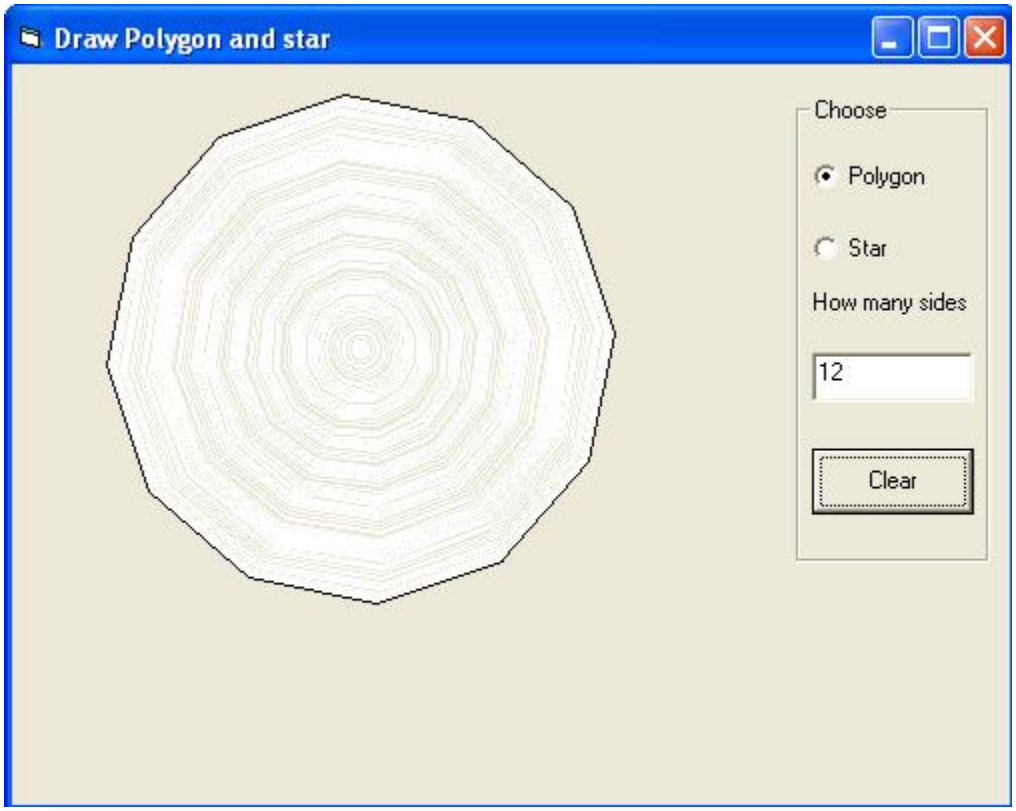
```
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
If (Option1.Value = True) Then
Call poly (X, Y)
End If
If (Option2.Value = True) Then
Call star(X, Y)
End If
End Sub
```

- الاشكال التالية تمثل النماذج التي تم رسمها خلال تنفيذ المشروع الخاصة برسم المضلع الذي يتم تحديد عدد أضلاعة من قبل المستخدم.

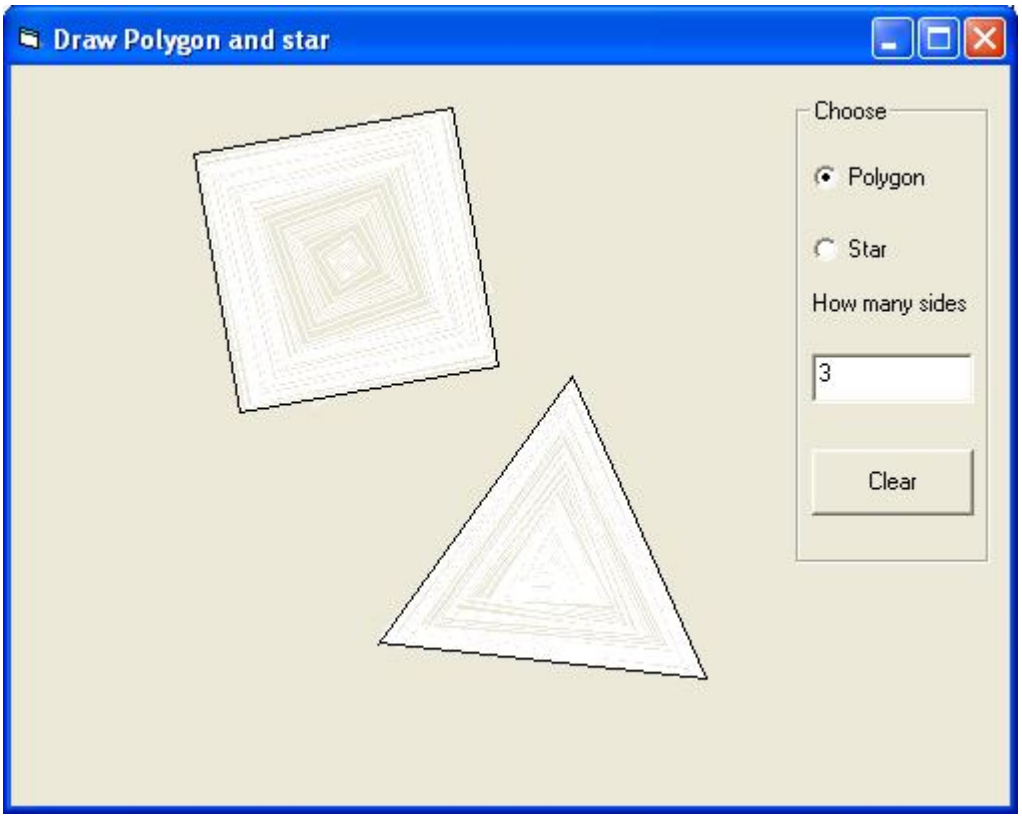


PDF created with pdfFactory trial version www.pdffactory.com

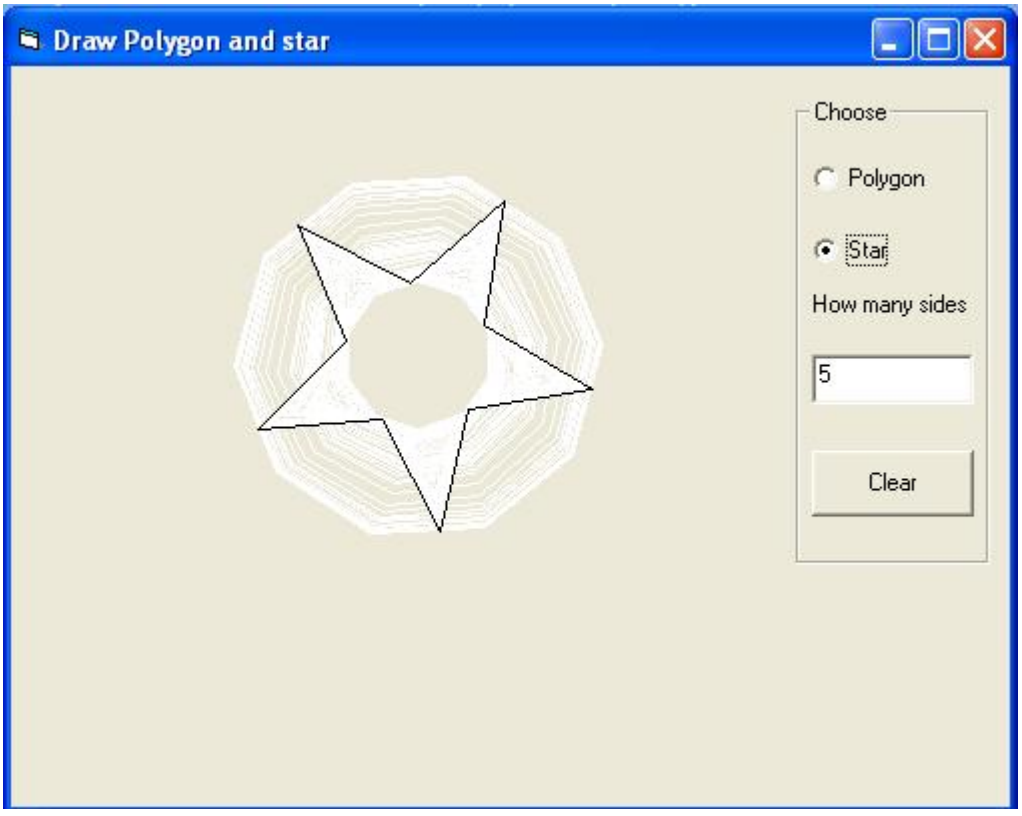




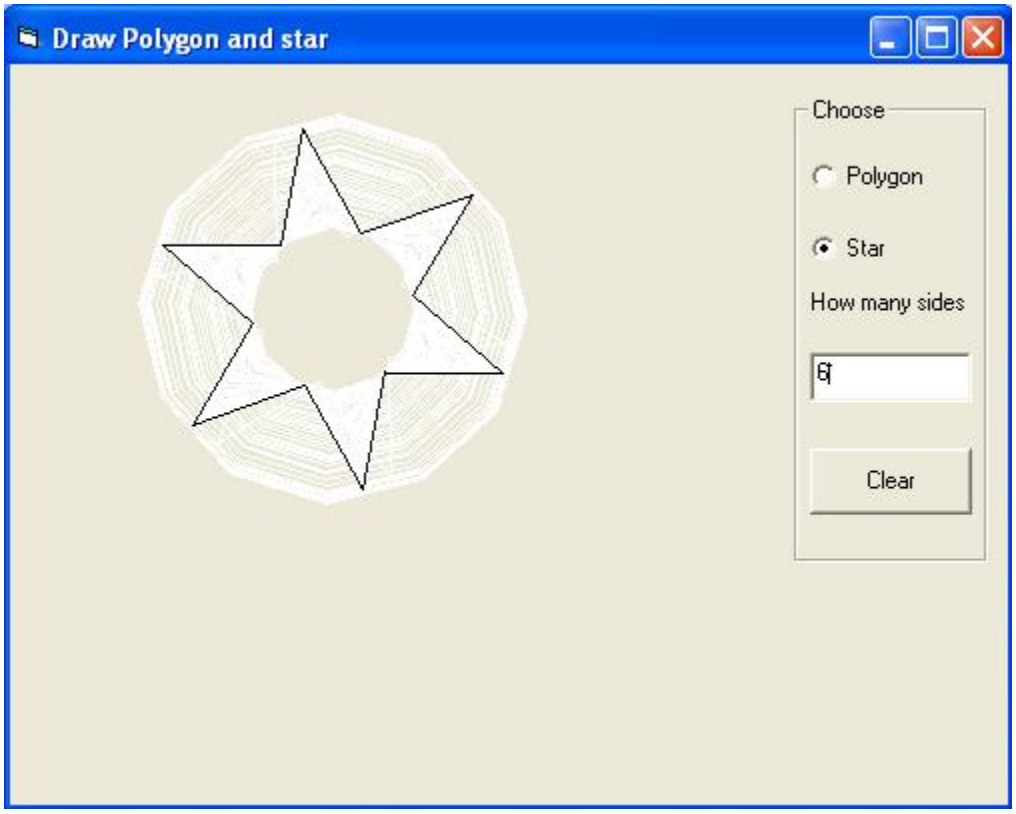
PDF created with pdfFactory trial version www.pdffactory.com

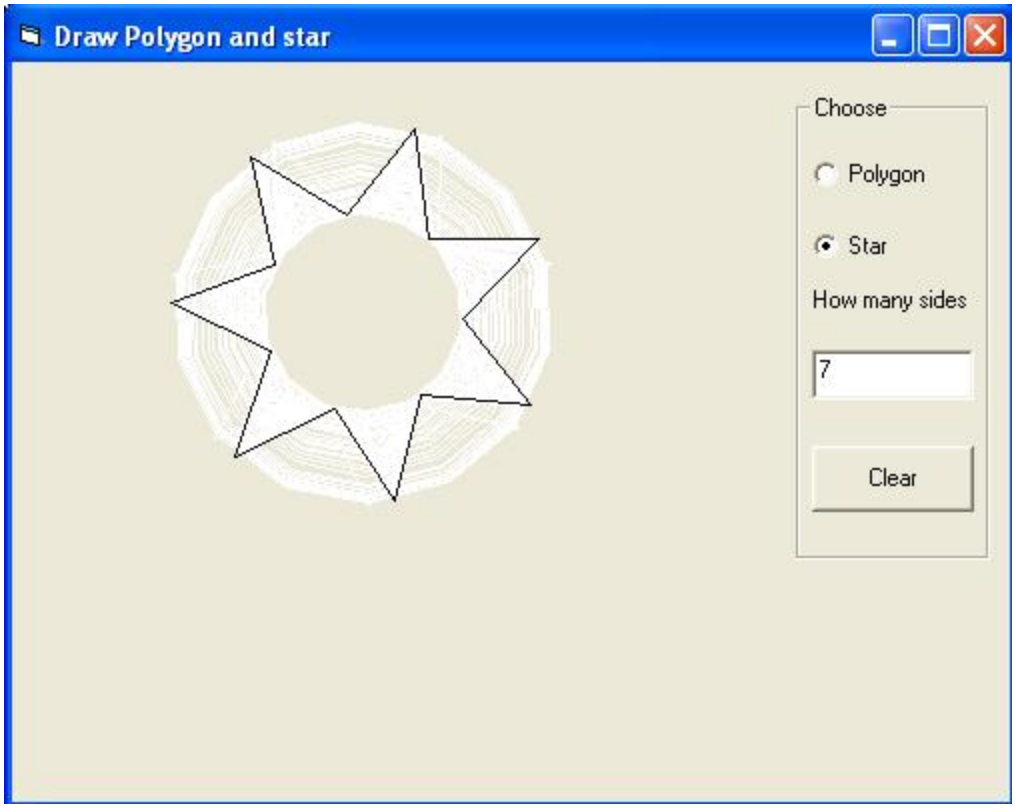


- الاشكال التالية تمثل النماذج التي تم رسمها خلال تنفيذ المشروع الخاصة برسم النجمة الذي يتم تحديد عدد أضلاعها من قبل المستخدم.

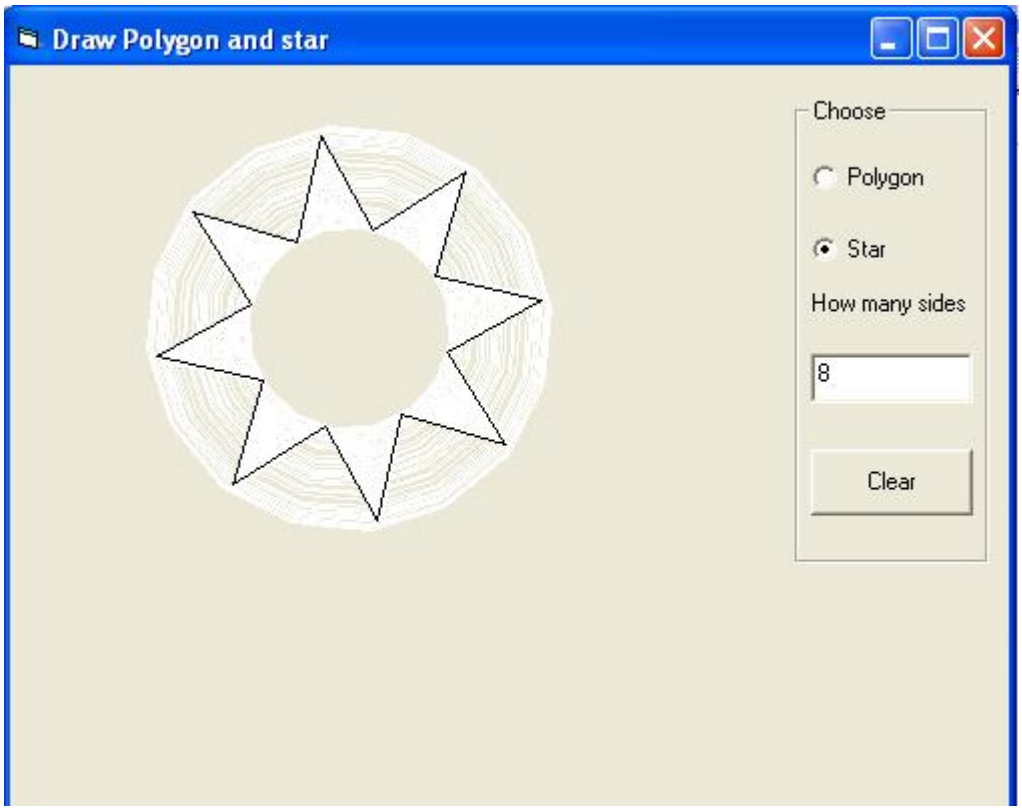


PDF created with pdfFactory trial version www.pdffactory.com





PDF created with pdfFactory trial version www.pdffactory.com



PDF created with pdfFactory trial version www.pdffactory.com

المشروع الرابع والعشرون

-:----------

رسم الاشكال الهندسية (كالمضلع Polygon والنجمة Star)

سوف نتعلم من خلال هذا المشروع مايلي:

- رسم المضلع Polygon متعدد الاضلاع.
- رسم النجمة Star متعددة الاضلاع.

الهدف من المشروع

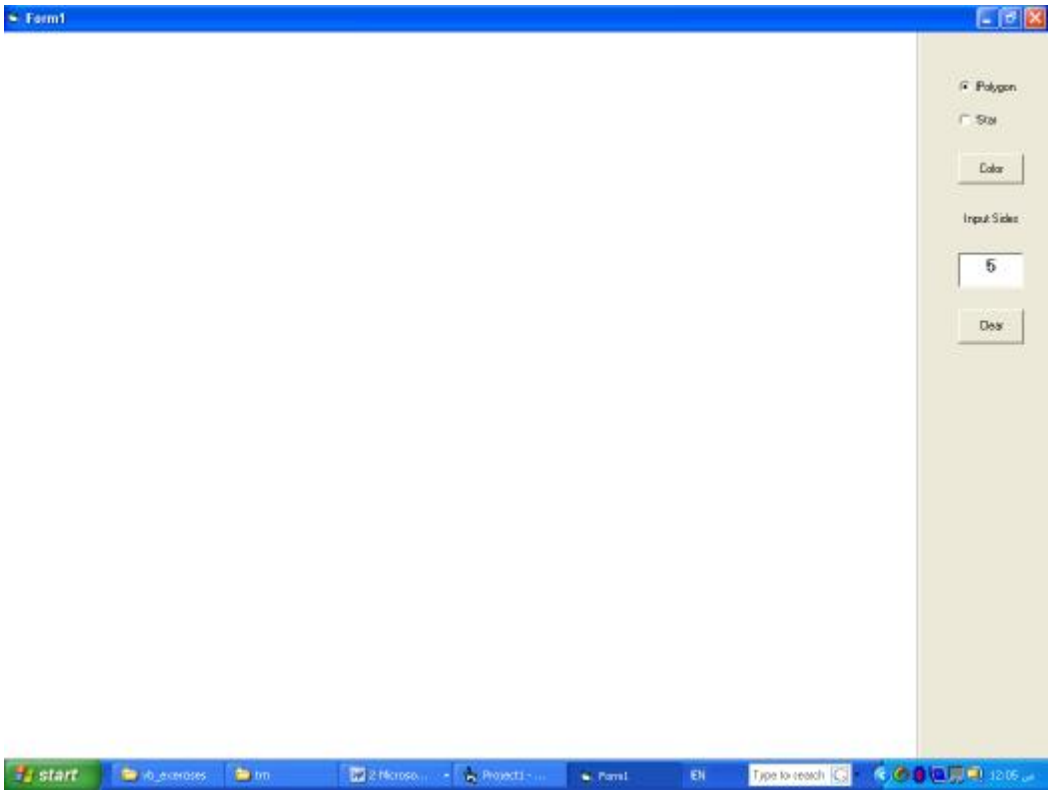
-:----------

رسم الاشكال الهندسية المتعددة الاضلاع كالمضلع والنجمة.

مراحل تنفيذ المشروع

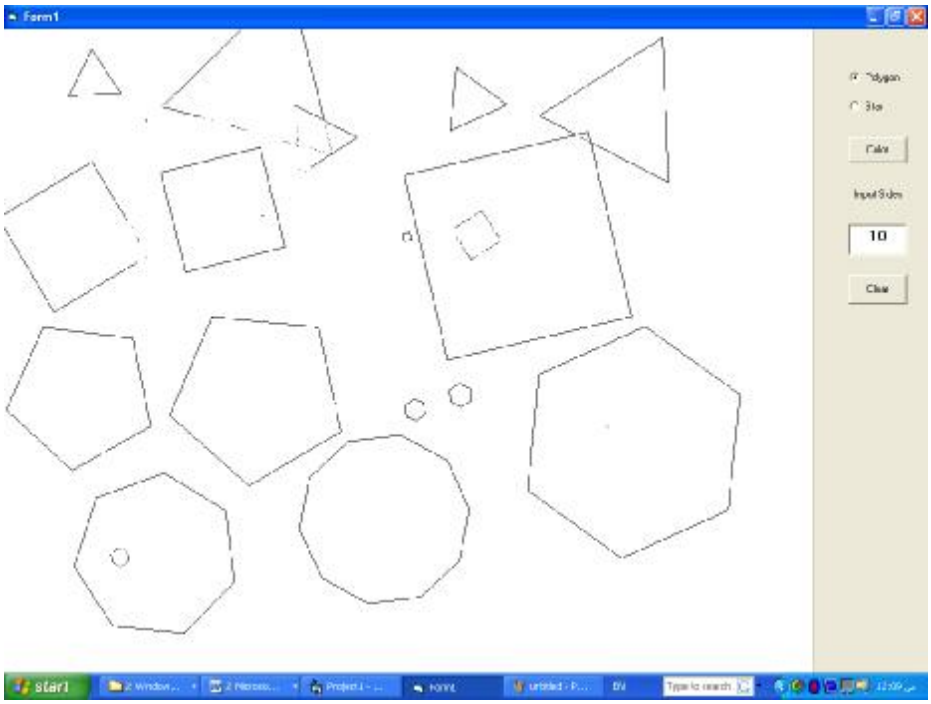
-:----------

- نفتح مشروعا قياسيا جديدا.
- نضع عناصر التحكم التالية على النموذج مثل الاطار Frame الذي يعتبر كحاوية للعناصر المستخدمة كالخيار الاول Option1 الذي يكون عنوانه Polygon, و الخيار الثاني Option2 الذي يكون عنوانة Star, أضافة الى مفتاح من نوع Command Button الذي يكون عنوانه Color, وصندوق نص Text Box لغرض تحديد عدد أضلاع المضلع أو النجمة كما هو مبين في النموذج التالي:

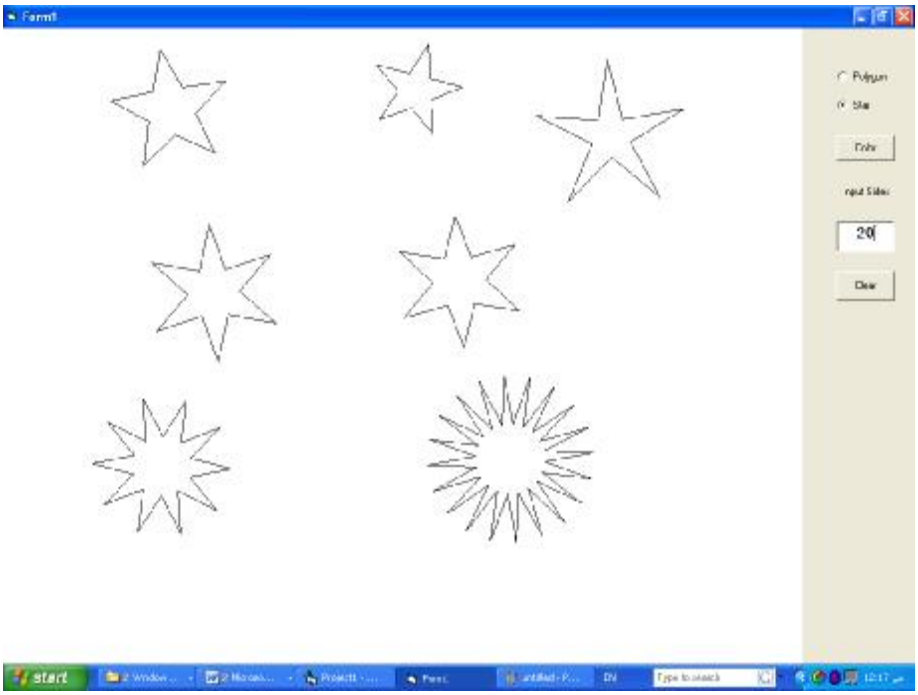


PDF created with pdfFactory trial version www.pdffactory.com

- الاشكال التالية تمثل النماذج المرسومة عند تنفيذ المشروع, حيث أن رسم هذه الاشكال يتحدد عن طريق تحديد هل الرسم المطلوب هو مضلع أو نجمة, أضافة الى تحديد عدد الاضلاع المطلوبة لرسم الشكل المرغوب, فاذا تم أختيار المضلع وكان عدد الاضلاع ثلاثة فسوف يرسم شكل المثلث, واذا كان عدد الاضلاع أربعة فسوف نحصل على شكل مربع, واذا كان عدد الاضلاع خمسة واكثر فسوف يرسم مضلع متعدد الاضلاع بحسب عدد الاضلاع التي يحددها المستخدم نفسه, كما هو مبين في الشكل التالي:



- فاذا تم أختيار النجمة, نحدد بعدها عدد الاضلاع, الاشكال التالية تمثل رسم النجوم وبأضلاع مختلفة وحسب تحديد المستخدم نفسه.



PDF created with pdfFactory trial version www.pdffactory.com

- نكتب المتغيرات التالية في بداية المشروع:

```
Dim sta As Double
Dim color As Variant
Dim a As Variant
Dim r As Variant
Dim r2 As Variant
Dim cc As Double
Dim n As Integer
Dim x1 As Double
Dim x2 As Double
Dim yc As Double
Dim xc As Double
Dim ycc As Double
Dim xcc As Double
Dim y1 As Double
Dim y2 As Double
```

- عند النقر مرتين متتاليتين على الخيار Polygon نكتب الاجراء العام Public التالي, الذي يقوم برسم المضلع العام المتعدد الاضلاع.

```
Public Sub poly(x, y)
If (cc = 1) Then
n = Val(Text1.Text)
  pi = 44 / 7
  For i = 1 To n
    x1 = xc + r * Cos(a + i * pi / n)
    y1 = yc + r * Sin(a + i * pi / n)
    x2 = xc + r * Cos(a + (i + 1) * pi / n)
    y2 = yc + r * Sin(a + (i + 1) * pi / n)
    Line (x1, y1)-(x2, y2), RGB(255, 255, 255)
  Next
  r = ((xc - x) ^ 2 + (yc - y) ^ 2) ^ (0.5)
  If ((xc - x) = 0 And (yc < y)) Then
     a = (0.25) * pi
  Else
    If ((xc - x) = 0 And (yc > y)) Then
      a = (0.75) * pi
    Else
      m = (yc - y) / (xc - x)
      a = Atn(m)
      a = IIf(xc < x, a, a - (pi / (n * (1 + (n Mod 2)))))
    End If
  End If
  For i = 1 To n
    x1 = xc + r * Cos(a + i * pi / n)
```


PDF created with pdfFactory trial version www.pdffactory.com

```
    y1 = yc + r * Sin(a + i * pi / n)
    x2 = xc + r * Cos(a + (i + 1) * pi / n)
    y2 = yc + r * Sin(a + (i + 1) * pi / n)
    Line (x1, y1)-(x2, y2), color
  Next
End If
End Sub
```

- عند النقر مرتين متتاليتين على الخيار الثاني Star سوف نكتب الاجراء العام Public التالي, الذي يقوم برسم النجوم وبأضلاع مختلفة أيضا.

```
Public Sub star(x, y)
If (sta <> 0) Then
n = Val(Text1.Text)
n = 2 * n
pi = 44 / 7

  For i = 1 To n
    z = r
    r = r2
    r2 = z
    x1 = xcc + r * Cos(a + i * pi / n)
    y1 = ycc + r * Sin(a + i * pi / n)
    x2 = xcc + r2 * Cos(a + (i + 1) * pi / n)
    y2 = ycc + r2 * Sin(a + (i + 1) * pi / n)
    Line (x1, y1)-(x2, y2), RGB(255, 255, 255)
  Next
  r = ((xcc - x) ^ 2 + (ycc - y) ^ 2) ^ (0.5)
  If (sta = 1) Then
 r2 = r
End If
  If ((xcc - x) = 0 And (ycc < y)) Then
     a = (0.25) * pi
  Else
    If ((xcc - x) = 0 And (ycc > y)) Then
      a = (0.75) * pi
    Else
      m = (ycc - y) / (xcc - x)
      a = Atn(m)
      a = IIf(xcc < x, a, a - (pi / (n * (1 + (n Mod 2)))))
    End If
  End If
  For i = 1 To n
  z = r
    r = r2
    r2 = z
```


PDF created with pdfFactory trial version www.pdffactory.com

```
    x1 = xcc + r * Cos(a + i * pi / n)
    y1 = ycc + r * Sin(a + i * pi / n)
    x2 = xcc + r2 * Cos(a + (i + 1) * pi / n)
    y2 = ycc + r2 * Sin(a + (i + 1) * pi / n)
    Line (x1, y1)-(x2, y2), color
  Next
End If
End Sub
```

- عند النقر على مفتاح Color مرتين متتاليتين نكتب الاجراء التالي الخاص بأستدعاء حوار الالوان وكمايلي:

```
Private Sub Command1_Click()
 CommonDialog1.ShowColor
 color = CommonDialog1.color
 Command1.BackColor = color
End Sub
```

- عند النقر على مفتاح Clear مرتين متتاليتين نكتب الاجراء التالي, الخاص بمسح النموذج.

```
Private Sub Command2_Click()
Me.Cls
End Sub
```

- عند النقر على أي موقع في النموذج , نقوم بتغير الحدث الى MouseDown , عندها نكتب الاجراء التالي, الغرض من هذا الاجراء هو لتحديد الموقع الاول للشكل المطلوب.

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, x As Single, y As
Single)
If (sta = 0) Then
  xcc = x
  ycc = y
End If
  xc = x
  yc = y
  cc = 1
End Sub
```

- عند النقر على أي موقع في النموذج نقوم أيضا بتغير الحدث الى MouseUp ونكتب الاجراء التالي, الغرض من هذا الاجراء هو لتحديد الموقع الاخير للشكل المطلوب.

```
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, x As Single, y As
Single)
cc = 0
sta = IIf(sta = 2, 0, sta + 1)
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

- عند النقر على أي موقع في النموذج نقوم بتغير الحدث الى MouseMove عندها نكتب الاجراء التالي, الغرض من هذا الاجراء هو لازاحة الشكل او لتحريكة.

```
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, x As Single, y As Single)
If (Option1.Value = True) Then
Call poly(x, y)
End If
If (Option2.Value = True) Then
Call star(x, y)
End If
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع الخامس والعشرون
-:----------

رسم الكرة الارضية

سوف نتعلم من هذا المشروع مايلي:

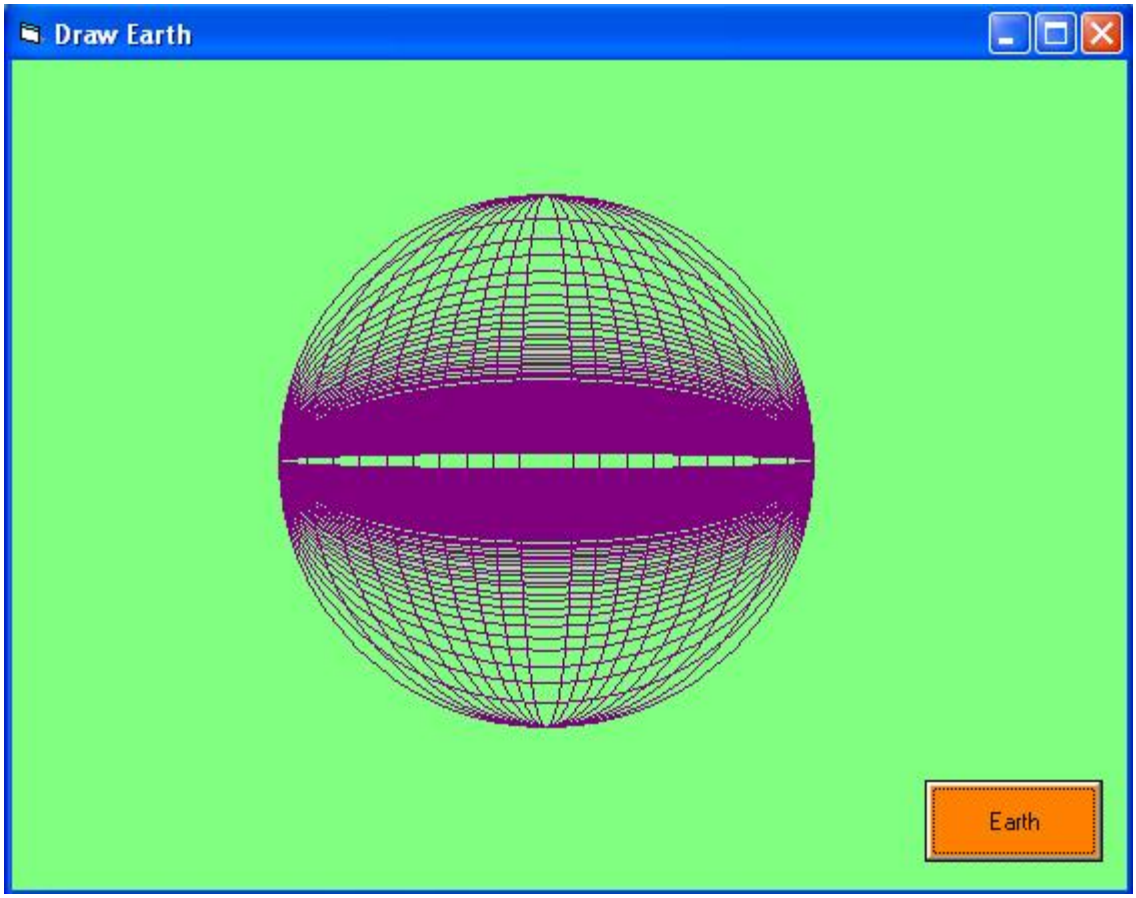
- كيفية رسم الدوائر المتداخلة للحصول على شكل يشبة الى حدما شكل الكرة الارضية.

الهدف من المشروع
-:----------

رسم الكرة الارضية.

مراحل تنفيذ المشروع
-:----------

- نفتح مشروعا قياسيا جديدا, ثم نغير عنوان النموذج الى Draw Earth, نغير لون واجهة النموذج الى اللون الاخضر كما في الشكل التالي:



PDF created with pdfFactory trial version www.pdffactory.com

- نضع عنصر التحكم Command1 على النموذج ونغير عنوانه Caption الى Earth, ولونه الى اللون البرتقالي.
- عند النقر مرتيين متتاليتين على المفتاح Earth نكتب الاجراء التالي الذي يقوم برسم دوائر متداخلة تظهر وكأنه خطوط الطول والعرض موضوعه عليها بشكل هندسي صحيح.

```
Private Sub Command1_Click()
Dim i As Integer
For i = 1 To 300
Me.Circle (4000, 3000), 2000, QBColor(5), , , 10 / i
Next i
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com

المشروع السادس والعشرون

-:----------

رسم الشكل البيضوي

سوف نتعلم من هذا المشروع مايلي:

- كيفية رسم الشكل البيضوي Ellipse للحصول على شكل بيضوي أفقي أو عمودي.

الهدف من المشروع

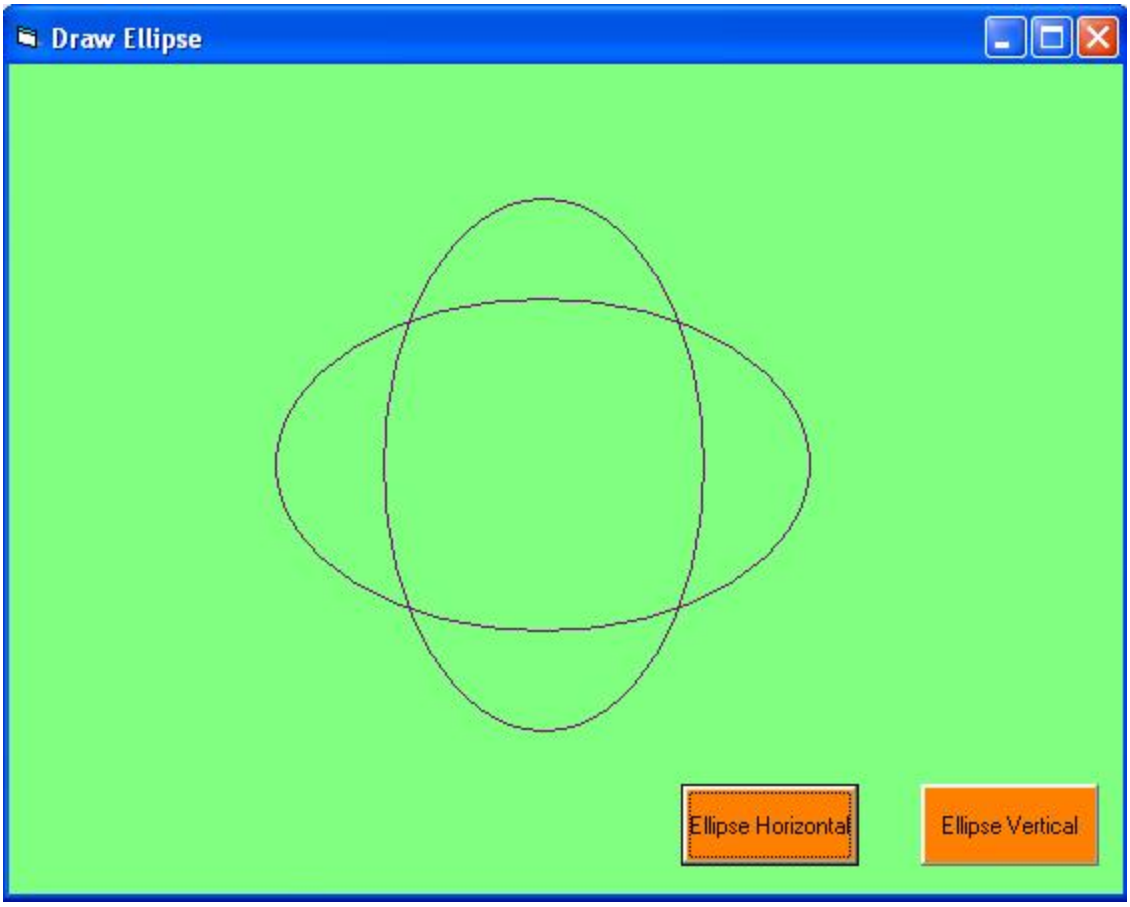
-:----------

رسم الشكل البيضوي.

مراحل تنفيذ المشروع

-:----------

- نفتح مشروعا قياسيا جديدا, ثم نغير عنوان النموذج الى Draw Ellipse, نغير لون واجهة النموذج الى اللون الاخضر كما في الشكل التالي:



PDF created with pdfFactory trial version www.pdffactory.com

- نضع عنصر التحكم Command1 على النموذج ونغير عنوانه Caption الى Ellipse Vertical, ولونه الى اللون البرتقالي.
- عند النقر مرتيين متتاليتين على المفتاح Ellipse Vertical نكتب الاجراء التالي الذي يقوم برسم الشكل البيضوي العمودي وكمايلي:

```
Private Sub Command1_Click()
Dim i As Integer
Me.Circle (4000, 3000), 2000, QBColor(5), , , 10 / 6
End Sub
```

- نضع عنصر التحكم Command2 على النموذج ونغير عنوانه Caption الى Ellipse Horizontal ولونه الى اللون البرتقالي أيضا.
- عند النقر مرتيين متتاليتين على المفتاح Ellipse Horizontal نكتب الاجراء التالي الذي يقوم برسم الشكل البيضوي الافقي وكمايلي:

```
Private Sub Command1_Click()
Dim i As Integer
Me.Circle (4000, 3000), 2000, QBColor(5), , , 10 / 16
End Sub
```


PDF created with pdfFactory trial version www.pdffactory.com