

الفصل السادس

مربعات الحوار

يوضح هذا الفصل كيفية دمج واستخدام مربعات الحوار في برنامجك، تستخدم مربعات الحوار لإظهار والحصول على المعلومات من المستخدم. توجد ثلاثة أنواع من مربعات الحوار في فيجول بيسك، وهذه الأنواع هي:

مربعات الحوار مسبقة التعريف Predefined Dialog Boxes أو (مربعات الحوار الجاهزة).

مربعات الحوار المخصصة Custom Dialog Boxes.

مربعات الحوار الشائعة Common Dialog Boxes.

مربعات الحوار مسبقة التعريف

حسب ما يتبدى من الاسم، فمربعات الحوار مسبقة التعريف، عبارة عن مربعات معرفة مسبقاً بواسطة فيجول بيسك. ولإظهار مربع مسبق التعريف، تستخدم عبارة ذات وسائط معينة تحدد كيف ومتى يتوجب إظهار المربع الحواري. يمكنك إظهار مربع حوار مسبق التعريف بواسطة ما يلي:

□ عبارة MsgBox() والتابع MsgBox()

□ التابع الوظيفي InputBox().

برنامج الرسالة

يسمح استخدام العبارة MsgBox والتابع الوظيفي MsgBox() بإظهار الرسائل للمستخدم والحصول على استجابته (الاستجابة تكون إما نعم Yes أو لا No). يوضح برنامج الرسالة كيف تستخدم العبارة MsgBox والتابع الوظيفي MsgBox() في البرنامج.

التمثيل المرئي لبرنامج الرسالة

سنبدأ كعادتنا بالتمثيل المرئي لنموذج البرنامج:

□ أنشئ الدليل C:\VB5Prg\Ch06.

□ ابدأ مشروعاً من النوع التنفيذي القياسي Standard EXE.

□ احفظ نموذج المشروع بالاسم Message.Frm في الدليل C:\VB5Prg\Ch06،

واحفظ ملف المشروع بالاسم Message.Vbp في الدليل ذاته.

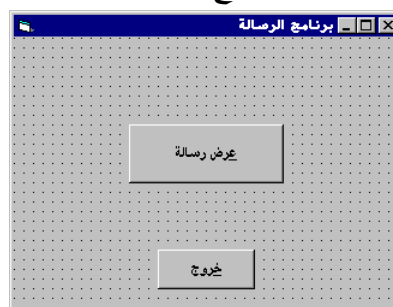
□ أنشئ النموذج تبعاً للجدول ٦-١.

يفترض أن يظهر النموذج المكتمل مثل المبين في الشكل ٦-١.

الشكل ٦-١

النموذج frmMessage

(طور التصميم).



الجدول ٦-١. جدول خواص النموذج frmMessage.

القيمة	الخاصية	الكائن
FrmMessage	Name	Form

	Caption	برنامج الرسالة
	RightToLeft	True
CommandButton	Name	cmdExit
	Caption	&خروج
	RightToLeft	True
CommandButton	Name	cmdMessage
	Caption	إظهار رسالة
	RightToLeft	True

إدخال نص برنامج الرسالة

سندخل الآن نص برنامج الرسالة:

□ تحقق أن قسم التصاريح العامة من النموذج frmMessage يحوي العبارة Option Explicit، أي:

```
يجب التصريح عن كل المتحولات'
Option Explicit
```

□ أدخل النص التالي ضمن الإجراء :cmdMessage_Click()

```
Private Sub cmdMessage_Click()
    Dim Message As String
    Dim Title As String
    Dim ButtonsAndIcon As Long

    تحديد الرسالة التي ستظهر في المربع الحواري'
    Message = "هذا مثال على رسالة بسيطة"

    تحديد نوع الأزرار المفترض ظهورها والصورة أيضا'
    ButtonsAndIcon = vbOKOnly + vbExclamation + _
        vbMsgBoxRight + vbMsgBoxRtlReading

    تحديد العنوان'
    Title = "عنوان المربع الحواري"

    إظهار المربع الحواري'
```

```
MsgBox Message, ButtonsAndIcon, Title
```

```
End Sub
```

□ أدخل النص التالي ضمن الإجراء cmdExit_Click():

```
Private Sub cmdExit_Click()
```

```
Dim Message As String
```

```
Dim Title As String
```

```
Dim ButtonsAndIcon As Long
```

```
Dim Response As String
```

```
' تحديد الرسالة التي ستظهر في المربع الحواري
```

```
Message = "هل أنت متأكد أنك تريد الخروج؟"
```

```
' تحديد نوع الأزرار المفترض ظهورها والصورة أيضا
```

```
ButtonsAndIcon = vbYesNo + vbQuestion + _  
vbMsgBoxRight + vbMsgBoxRtlReading
```

```
' تحديد العنوان
```

```
Title = "برنامج الرسالة"
```

```
' إظهار المربع الحواري وانتظار رد المستخدم
```

```
Response = MsgBox(Message, ButtonsAndIcon, Title)
```

```
' تقييم استجابة المستخدم
```

```
If Response = vbYes Then
```

```
End
```

```
End If
```

```
End Sub
```

□ اختر البند Save Project من القائمة File لفيجول بيسك لحفظ عملك.

تنفيذ برنامج الرسالة

دعنا ننفذ برنامج الرسالة، ونرى نتيجة ما كتبناه على أرض الواقع.

□ نفذ برنامج الرسالة.

لاحظ الظاهرة التالية أثناء تنفيذ البرنامج: يظهر مربع حوار مع رسالة عندما تضغط

على الزر عرض رسالة، كما يحمل المربع الزر موافق ورمز إشارة التعجب Exclamation Icon، انظر الشكل ٦-٢. مجال هذا المربع الحواري محدد (أو محصور) بمعنى أن البرنامج لا يستطيع استئناف التنفيذ ما لم تغلق مربع الحوار هذا. فمثلاً جرب النقر بواسطة الفأرة على الزر خروج أثناء ظهور مربع الحوار هذا، وستجد أن البرنامج لن ينتهي (أي لم ينفذ الإجراء `cmdExit_Click()`).

□ انقر على الزر موافق في مربع الحوار.

يغلق مربع الحوار ذاته، استجابة لذلك.

□ انقر على الزر خروج.

يظهر نتيجة لذلك مربع حوارى يحمل زراً يدعى نعم وآخر لا، ورمز إشارة الاستفهام (انظر الشكل ٦-٣). يطلب منك هذا المربع تثبيت رغبتك في الخروج. يؤدي الضغط على نعم إلى إنهاء البرنامج، أما الضغط على الزر لا فيؤدي إلى إغلاق المربع الحوارى ولكن بدون انتهاء البرنامج. يعتبر مربع الحوار هذا أيضاً محدداً (مشروطاً Modal) بمعنى أنك لا تستطيع استئناف البرنامج ما لم تستجب لهذا المربع بالنقر على أحد الزرين نعم أو لا.

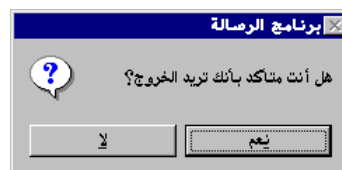
الشكل ٦-٢

مربع حوار مع الزر موافق وإشارة التعجب.



الشكل ٦-٣

مربع حوار مع الزرين نعم و لا وإشارة استفهام.



ملاحظة

تدعى مربعات الحوار التي تناولها مثالنا هذا، مربعات حوار محددة Modal Dialog Boxes. يمكن أثناء عرض مربع الحوار من هذا النوع (مثل مربع الرسالة) الانتقال إلى برامج ويندوز الأخرى، لكن ما لا يمكن هو، العودة إلى نافذة برنامج الرسالة

دون إغلاق مربع الحوار هذا.

نوع آخر من مربعات الحوار المحددة، هو مربع حوار النظام المحدد، الذي لا يسمح لك بالرجوع إلى التطبيق، ولا يمكنك من الانتقال إلى تطبيقات ويندوز أخرى. أي بمعنى آخر، لا تستطيع العمل خارج هذا المربع حتى تجيب عليه.

كيف يعمل برنامج الرسالة؟

يستخدم برنامج الرسالة العبارة MsgBox والتابع الوظيفي MsgBox() لإظهار مربعات حوار الرسائل.

إظهار مربع حوار بواسطة العبارة MsgBox

ينفذ الإجراء cmdMessage_Click() آلياً عند الضغط على الزر عرض رسالة. يستخدم البرنامج المكتوب داخل هذا الإجراء العبارة MsgBox لإظهار مربع حوار (مربع رسالة) يحمل زراً يدعى موافق ورمز لإشارة التعجب:

```
Private Sub cmdMessage_Click()

    Dim Message As String
    Dim Title As String
    Dim ButtonsAndIcon As Long

    ' تحديد الرسالة التي ستظهر في المربع الحواري
    Message = "هذا مثال على رسالة بسيطة"

    ' تحديد نوع الأزرار المفترض ظهورها والصورة أيضاً
    ButtonsAndIcon = vbOKOnly + vbExclamation + _
        vbMsgBoxRight + vbMsgBoxRtlReading

    ' تحديد العنوان
    Title = "عنوان المربع الحواري"

    ' إظهار المربع الحواري
```

```
MsgBox Message, ButtonsAndIcon, Title
```

End Sub

لاحظ أننا مررنا ثلاثة وسائط إلى العبارة MsgBox:

الرسالة المراد إظهارها (سلسلة كتابية String).

الأزرار والرموز المراد ظهورها في مربع الرسالة (قيمة عددية).

عنوان مربع الحوار (سلسلة كتابية String).

يجب إجراء cmdMessage_Click() المتحولات الثلاثة اللازمة كوسائط في عبارة

MsgBox قبل استخدام هذه العبارة. وأول متحول قام بالإجراء بتكليفه كما رأينا هو

المتحول Message الذي يحمل الرسالة التي ستظهر في مربع الحوار:

```
Message = "هذا مثال على رسالة بسيطة"
```

يستخدم المتحول Message كأول وسيط في العبارة MsgBox.

المتحول الثاني عبارة عن متحول من النوع الصحيح Integer ويدعى ButtonsAndIcon.

ويتم تكليفه بعدد يحدد نوع الأزرار وشكل الرموز التي ستظهر في مربع الرسالة،

بالإضافة إلى الثابتين vbMsgBoxRight و vbMsgBoxRtlReading اللذين يقومان بتعريب

مربع الرسالة:

```
ButtonsAndIcon = vbOKOnly + vbExclamation + _  
vbMsgBoxRight + vbMsgBoxRtlReading
```

يستخدم المتحول ButtonsAndIcon كثاني وسيط في العبارة MsgBox كما أن vbOKOnly و

vbExclamation عبارة عن ثابتين من ثوابت فيجول بيسك. يمثل الثابت vbOKOnly الزر

موافق، بينما يمثل الثابت vbExclamation رمز إشارة التعجب.

وإسناد مجموع الثابتين (vbExclamation و vbOKOnly) إلى المتحول ButtonsAndIcon،

يفرض على المربع الحوار أن يمتلك الزر موافق ورمز إشارة التعجب، حسب ما

يوضحه الشكل ٢-٦.





يعطي الجدولان ٢-٦ و ٣-٦ ثوابت الأزرار وثوابت الأيقونات التي يمكنك

استخدامها من أجل الوسيط الثاني في عبارة MsgBox.

الجدول ٦-٢. ثوابت أزرار مربع حوار الرسالة.

اسم الثابت	القيمة	الأزرار التي ستظهر
vbOKOnly	٠	موافق
vbOKCancel	١	موافق و إلغاء
vbAbortReryIgnore	٢	إحباط و إعادة المحاولة و تجاهل
vbYesNoCancel	٣	نعم و لا و إلغاء
vbYesNo	٤	نعم و لا
vbRetryCancel	٥	إعادة المحاولة و إلغاء

الجدول ٦-٣. ثوابت رموز مربع حوار الرسالة.

اسم الثابت	القيمة	الرمز التي ستظهر
vbCirtical	١٦	 رمز الخطأ الحرج
vbQuestion	٣٢	 رمز إشارة الاستفهام
vbExclamation	٤٨	 رمز التعجب
vbInformation	٦٤	 رمز المعلومات

المتحول الثالث الذي يعمل الإجراء cmdMessage_Click() على إعداده، هو المتحول Title ، ويتم إسناد عنوان مربع الحوار إلى ذلك المتحول:

```
Title = "عنوان المربع الحواري"
```


وطبعاً يُستخدم Title كثالث وسيط في العبارة MsgBox، وبنهاية الأمر، ينتج عن استخدام الوسائط الثلاثة Message و ButtonsAndIcon و Title في عبارة المربع الحوارى ما هو مبين في الشكل ٦-٢.

MsgBox Message, ButtonsAndIcon, Title

إظهار مربع حوار بواسطة التابع الوظيفي MsgBox()

ينفذ الإجراء cmdExit_Click() آلياً، لدى النقر على الزر خروج. يستخدم هذا الإجراء، التابع الوظيفي MsgBox() لإظهار مربع حوار يحمل إشارة استفهام وزرين هما نعم و لا:

```
Private Sub cmdExit_Click()
    Dim Message As String
    Dim Title As String
    Dim ButtonsAndIcon As Long
    Dim Response As String

    تحديد الرسالة التي ستظهر في المربع الحوارى
    Message = "هل أنت متأكد بأنك تريد الخروج؟"

    تحديد نوع الأزرار المفترض ظهورها والصورة أيضاً
    ButtonsAndIcon = vbYesNo + vbQuestion + _
        vbMsgBoxRight + vbMsgBoxRtlReading

    تحديد العنوان
    Title = "برنامج الرسالة"

    إظهار المربع الحوارى وانتظار رد المستخدم
    Response = MsgBox(Message, ButtonsAndIcon, Title)

    تقييم استجابة المستخدم
    If Response = vbYes Then
        End
```

```
End If
```

```
End Sub
```

يأخذ التابع الوظيفي MsgBox() نفس الوسائط التي تأخذها العبارة MsgBox، والاختلاف الوحيد بينهما، أن التابع الوظيفي MsgBox() يعيد قيمةً تشير إلى الزر الذي تم النقر عليه في مربع الحوار.

يعمل الإجراء cmdExit_Click() على تكليف المتحولات الثلاثة Message و ButtonsAndIcon و Title قبل تنفيذ التابع الوظيفي MsgBox() لاستخدامها فيه:

```
Message = "هل أنت متأكد أنك تريد الخروج؟"
ButtonsAndIcon = vbYesNo + vbQuestion + _
    vbMsgBoxRight + vbMsgBoxRtlReading
Title = "برنامج الرسالة"
```

تم إسناد مجموع الثابتين vbYesNo و vbQuestion إلى المتحول ButtonsAndIcon لأن مربع الحوار يجب أن يمتلك الزرين نعم و لا ورمز إشارة الاستفهام. يُنفذ التابع الوظيفي حال الانتهاء من إعداد وسائطه، ويعيد قيمةً يتم إسنادها إلى المتحول Response.

```
Response = MsgBox(Message, ButtonsAndIcon, Title)
```

تدل القيمة التي يعيدها التابع MsgBox() على الزر الذي تم النقر أو الضغط عليه. يبين الجدول ٤-٦، كل الثوابت المحتملة التي يتمكن التابع MsgBox() من إعادتها. فمثلاً إذا كانت القيمة المعادة من قبل التابع الوظيفي MsgBox() هي الثابت vbYes، فهذا يعني أن المستخدم قام بالنقر على الزر نعم في مربع الرسالة.

الجدول ٤-٦. القيم المحتملة المعادة من قبل التابع الوظيفي MsgBox():

اسم الثابت	القيمة	الزر الذي تم نقره
vbOK	١	الزر موافق

الزر إلغاء	٢	vbCancel
الزر إحباط	٣	vbAbort
الزر إعادة المحاولة	٤	vbRetry
الزر تجاهل	٥	vbIgnore
الزر نعم	٦	vbYes
الزر لا	٧	vbNo

تستخدم عبارة If مع قيمة المتحول Response للتحقق من الزر الذي تم النقر عليه (أي للتعرف عليه):

```
If Response = vbYes Then
    End
End If
```

عند الضغط على الزر نعم فإن قيمة Response تساوي vbYes، وينتهي البرنامج نتيجة لتنفيذ عبارة End. وهكذا تكون قد أجبت على السؤال "هل أنت متأكد بأنك تريد الخروج؟"، بالموافقة، وبذلك ينتهي البرنامج.

ملاحظة

تنفذ العبارة MsgBox في الإجراء cmdMessage_Click() كالتالي:
MsgBox Message , ButtonsAndIcon , Title
وينفذ التابع الوظيفي في الإجراء cmdExit_Click() كالتالي:
Response = MsgBox(Message , DialogType , Title)
فعند استخدام العبارة MsgBox لا يتم إحاطة الوسائط بقوسين، بعكس التابع الوظيفي MsgBox() الذي يتوجب فيه إغلاق الوسائط بقوسين، ولا تنسى وجوب استخدام القيمة المعادة من قبل التابع الوظيفي لإسنادها إلى متحول.

برنامج مربعات الحوار

أوضح برنامج الرسالة السابق، كيفية استخدام العبارة MsgBox والتابع الوظيفي MsgBox() لإظهار مربع حوار يحمل الزر موافق، ومربع حوار يحمل الزرين نعم ولا.

تستطيع حسب ما يوضحه الجدول ٦-٢ إظهار أزرار قياسية أخرى في مربعات الحوار، بالاستعانة بالعبارة والتابع الوظيفي المذكورين. يوضح برنامج مربعات الحوار كيفية استخدام التابع الوظيفي MsgBox() لإظهار مختلف أنواع الأزرار وكيفية معرفة القيمة المعادة من التابع MsgBox() لتحديد الزر الذي تم النقر عليه.

التمثيل المرئي لبرنامج مربعات الحوار

كعادتنا سنبدأ بخطوة التمثيل المرئي للنموذج frmDialogs.

□ ابدأ مشروعاً قياسياً جديداً Standard EXE.

□ احفظ نموذج المشروع بالاسم Dialogs.Frm في الدليل C:\VB5Prg\Ch06، ثم

احفظ ملف المشروع في الدليل ذاته، بالاسم Dialoge.VBp.

□ أنشئ النموذج وفقاً للجدول ٥-٦ والجدول ٦-٦ (الجدول ٦-٦ هو الجدول

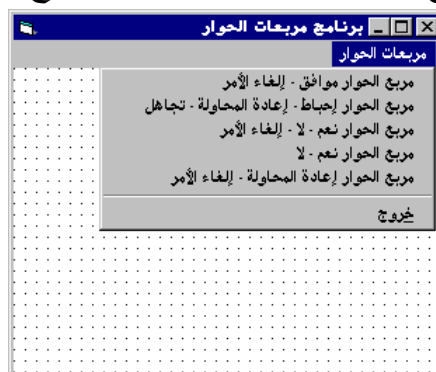
المستخدم لبناء قائمة النموذج frmDialogs).

يوضح الشكل ٦-٤ كيف سيبدو النموذج بعد اكتماله.

الشكل ٦-٤

النموذج frmDialogs

(طور التصميم).



الجدول ٥-٦. جدول خصائص النموذج frmDialogs.

القيمة	الخاصية	الكائن
frmDialogs	Name	Form
برنامج مربعات الحوار	Caption	
True	RightToLeft	
White	BackColor	
(انظر الجدول ٦-٦)	(انظر الجدول ٦-٦)	Menu

الجدول ٦-٦. جدول قائمة النموذج frmDialogs.

العنوان	الخاصية Name
&مربعات الحوار	mnuDialogs
...مربع الحوار موافق - إلغاء الأمر	mnuOKCancel
...مربع الحوار إحباط - إعادة المحاولة - تجاهل	mnuAbortRetryIgnore
...مربع الحوار نعم - لا - إلغاء الأمر	mnuYesNoCancel
...مربع الحوار نعم - لا	mnuYesNo
...مربع الحوار إعادة المحاولة - إلغاء الأمر	mnuRetryCancel
...	mnuSep1
&...خروج	mnuExit

إدخال نص برنامج مربعات الحوار

سندخل الآن نص برنامج مربعات الحوار:

□ تحقق بأن العبارة Option Explicit تقع في قسم التصاريح العامة للنموذج

:frmDialogs

```
'All Variables Must Be Declared
Option Explicit
```

□ أدخل النص التالي ضمن الإجراء :mnuAbortRetryIgnore_Click()

```

Private Sub mnuAbortRetryIgnore_Click()
    Dim DialogType As Long
    Dim DialogTitle As String
    Dim DialogMsg As String
    Dim Response As Integer
    ' تحديد الأزرار الواجب إظهارها
    DialogType = vbAbortRetryIgnore + vbExclamation
    ' تحديد الصفات العربية لمربع الحوار
    DialogType = DialogType + vbMsgBoxRight + _
        vbMsgBoxRtlReading
    ' تحديد عنوان مربع الحوار
    DialogTitle = "عنوان مربع الحوار"
    ' تحديد رسالة مربع الحوار
    DialogMsg = "هذا مثال على مربع حوار يظهر الأزرار التالية "
    ' إظهار مربع الحوار والحصول على رد المستخدم
    Response = MsgBox(DialogMsg, DialogType, DialogTitle)

    ' تقييم رد المستخدم
    Select Case Response
        Case vbAbort
            MsgBox "لقد نقرت الزر إحباط"
        Case vbRetry
            MsgBox "لقد نقرت الزر إعادة المحاولة"
        Case vbIgnore
            MsgBox "لقد نقرت الزر تجاهل"
    End Select
End Sub

```

□ أدخل النص التالي ضمن الإجراء mnuExit_Click():

```

Private Sub mnuExit_Click()
    Dim DialogType As Long
    Dim DialogTitle As String
    Dim DialogMsg As String
    Dim Response As Integer
    ' تحديد الأزرار الواجب إظهارها

```

```

DialogType = vbYesNo + vbCritical
' تحديد الصفات العربية لمربع الحوار
DialogType = DialogType + vbMsgBoxRight + vbMsgBoxRtlReading
' تحديد عنوان مربع الحوار
DialogTitle = "برنامج مربعات الحوار"
' تحديد رسالة مربع الحوار
DialogMsg = "هل أنت متأكد بأنك تريد الخروج؟"
' إظهار مربع الحوار والحصول على رد المستخدم
Response = MsgBox(DialogMsg, DialogType, DialogTitle)

' تقييم رد المستخدم
If Response = vbYes Then
    End
End If
End Sub

```

□ أدخل النص التالي ضمن الإجراء :mnuOKCancel_Click()

```

Private Sub mnuOKCancel_Click()
    Dim DialogType As Long
    Dim DialogTitle As String
    Dim DialogMsg As String
    Dim Response As Integer
    ' تحديد الأزرار الواجب إظهارها
    DialogType = vbOKCancel + vbExclamation
    ' تحديد الصفات العربية لمربع الحوار
    DialogType = DialogType + vbMsgBoxRight + vbMsgBoxRtlReading
    ' تحديد عنوان مربع الحوار
    DialogTitle = "عنوان مربع الحوار"
    ' تحديد رسالة مربع الحوار
    DialogMsg = "هذا مثال على مربع حوار يظهر الأزرار التالية"
    ' إظهار مربع الحوار والحصول على رد المستخدم
    Response = MsgBox(DialogMsg, DialogType, DialogTitle)

    ' تقييم رد المستخدم
    Select Case Response

```

```

Case vbOK
    MsgBox "لقد نقرت الزر موافق"
Case vbCancel
    MsgBox "لقد نقرت الزر إلغاء الأمر"
End Select

```

End Sub

□ أدخل النص التالي ضمن الإجراء :mnuRetryCancel_Click()

```

Private Sub mnuRetryCancel_Click()
    Dim DialogType As Long
    Dim DialogTitle As String
    Dim DialogMsg As String
    Dim Response As Integer

    ' تحديد الأزرار الواجب إظهارها
    DialogType = vbRetryCancel + vbExclamation
    ' تحديد الصفات العربية لمربع الحوار
    DialogType = DialogType + vbMsgBoxRight + vbMsgBoxRtlReading
    ' تحديد عنوان مربع الحوار
    DialogTitle = "عنوان مربع الحوار"
    ' تحديد رسالة مربع الحوار
    DialogMsg = "هذا مثال على مربع حوار يظهر الأزرار التالية"
    ' إظهار مربع الحوار والحصول على رد المستخدم
    Response = MsgBox(DialogMsg, DialogType, DialogTitle)

    ' تقييم رد المستخدم
    Select Case Response
        Case vbRetry
            MsgBox "لقد نقرت الزر إعادة المحاولة"
        Case vbCancel
            MsgBox "لقد نقرت الزر إلغاء الأمر"
    End Select

```

End Sub

□ أدخل النص التالي ضمن الإجراء :mnuYesNo_Click()


```

Private Sub mnuYesNo_Click()
    Dim DialogType As Long
    Dim DialogTitle As String
    Dim DialogMsg As String
    Dim Response As Integer

    ' تحديد الأزرار الواجب إظهارها
    DialogType = vbYesNo + vbQuestion
    ' تحديد الصفات العربية لمربع الحوار
    DialogType = DialogType + vbMsgBoxRight + vbMsgBoxRtlReading
    ' تحديد عنوان مربع الحوار
    DialogTitle = "عنوان مربع الحوار"
    ' تحديد رسالة مربع الحوار
    DialogMsg = "هذا مثال على مربع حوار يظهر الأزرار التالية"
    ' إظهار مربع الحوار والحصول على رد المستخدم
    Response = MsgBox(DialogMsg, DialogType, DialogTitle)

    ' تقييم رد المستخدم
    Select Case Response
        Case vbYes
            MsgBox "لقد نقرت الزر نعم"
        Case vbNo
            MsgBox "لقد نقرت الزر إلغاء لا"
    End Select
End Sub

```

□ أدخل النص التالي ضمن الإجراء :mnuYesNoCancel_Click()

```

Private Sub mnuYesNoCancel_Click()
    Dim DialogType As Long
    Dim DialogTitle As String
    Dim DialogMsg As String
    Dim Response As Integer

    ' تحديد الأزرار الواجب إظهارها
    DialogType = vbYesNoCancel + vbExclamation

```

```

' تحديد الصفات العربية لمربع الحوار'
DialogType = DialogType + vbMsgBoxRight + vbMsgBoxRtlReading
' تحديد عنوان مربع الحوار'
DialogTitle = "عنوان مربع الحوار"
' تحديد رسالة مربع الحوار'
DialogMsg = "هذا مثال على مربع حوار يظهر الأزرار التالية"
' إظهار مربع الحوار والحصول على رد المستخدم'
Response = MsgBox(DialogMsg, DialogType, DialogTitle)

' تقييم رد المستخدم'
Select Case Response
    Case vbYes
        MsgBox "لقد نقرت الزر نعم"
    Case vbNo
        MsgBox "لقد نقرت الزر لا"
    Case vbCancel
        MsgBox "لقد نقرت الزر إلغاء الأمر"
End Select
End Sub

```

□ احفظ المشروع باختيار البند **Save Project** من القائمة **File**.

تنفيذ برنامج مربعات الحوار

دعنا نشاهد نتيجة ما كتبناه قيد التنفيذ:

□ نفذ برنامج مربعات الحوار.

□ اختبر وتمرن على مختلف مربعات الحوار، فمثلاً أنجز ما يلي لإظهار مربع

الحوار إحباط-إعادة المحاولة-تجاهل:

□ اختر البند إحباط-إعادة المحاولة-تجاهل من قائمة مربعات الحوار.

يستجيب البرنامج بإظهار مربع حوار يحوي الأزرار إحباط-إعادة المحاولة-تجاهل.
(انظر الشكل ٦-٥).

الشكل ٥-٦

مربع حوار يحوي الأزرار
الثلاثة:

إحباط و إعادة المحاولة و
تجاهل.



استخدم الخطوتين التاليتين لإنهاء برنامج مربعات الحوار:

❑ اختر البند خروج من قائمة مربعات الحوار.

يستجيب البرنامج بإظهار مربع الحوار نعم/لا، ويتحقق من رغبتك بالخروج.

❑ انقر فوق الزر نعم.

يستجيب برنامج مربعات الحوار بإنهاء نفسه.

كيف يعمل برنامج مربعات الحوار

يستخدم برنامج مربعات الحوار التابع الوظيفي MsgBox() لإظهار شتى أشكال مربعات الحوار مع مختلف الأزرار. ويستخدم البرنامج القيمة المعادة من التابع الوظيفي MsgBox() لتحديد الزر الذي تم النقر عليه.

الإجراء mnuAbortRetryIgnore_Click()

ينفذ هذا الإجراء عند اختيار البند إحباط-إعادة المحاولة-تجاهل، من قائمة مربعات الحوار. يستخدم نص برنامج الإجراء هذا التابع الوظيفي MsgBox() لإظهار مربع حوار يحمل الأزرار الثلاثة: إحباط-إعادة المحاولة-تجاهل:

```
Private Sub mnuAbortRetryIgnore_Click()
```

```
Dim DialogType As Long
```

```
Dim DialogTitle As String
```

```
Dim DialogMsg As String
```

```
Dim Response As Integer
```

```
' تحديد الأزرار الواجب إظهارها'
```

```

DialogType = vbAbortRetryIgnore + vbExclamation
' تحديد الصفات العربية لمربع الحوار
DialogType = DialogType + vbMsgBoxRight + _
                vbMsgBoxRtlReading
' تحديد عنوان مربع الحوار
DialogTitle = "عنوان مربع الحوار"
' تحديد رسالة مربع الحوار
DialogMsg = "هذا مثال على مربع حوار يظهر الأزرار التالية"
' إظهار مربع الحوار والحصول على رد المستخدم
Response = MsgBox(DialogMsg, DialogType, DialogTitle)

' تقييم رد المستخدم
Select Case Response
    Case vbAbort
        MsgBox "لقد نقرت الزر إحباط"
    Case vbRetry
        MsgBox "لقد نقرت الزر إعادة المحاولة"
    Case vbIgnore
        MsgBox "لقد نقرت الزر تجاهل"
End Select

```

End Sub

يعدل الإجراء mnuAbortRetryIgnore_Click() المتحولات DialogType و DialogMsg و DialogTitle قبل أن ينفذ التابع الوظيفي MsgBox():

```

DialogType = DialogType + vbMsgBoxRight + _
                vbMsgBoxRtlReading
DialogTitle = "عنوان مربع الحوار"
DialogMsg = "هذا مثال على مربع حوار يظهر الأزرار التالية"
Response = MsgBox(DialogMsg, DialogType, DialogTitle)

```

أسند مجموع كل من vbExclamation و vbAbortRetryIgnore إلى المتحول DialogType، لأن مربع الحوار يفترض أن يحمل الأزرار إحباط-إعادة المحاولة-تجاهل إلى جانب إشارة التعجب.

ينفذ التابع الوظيفي MsgBox() حال الانتهاء من إعداد المتحولات المذكورة، ويسند القيمة المعادة من التابع إلى المتحول Response:

```
Response = MsgBox(DialogMsg, DialogType, DialogTitle)
```

تختبر قيمة المتحول Response بواسطة عبارة Select Case لتحديد الزر الذي تم النقر عليه (أي أحد الأزرار الثلاثة إحباط-إعادة المحاولة-تجاهل)، باستخدام التعبير التالي:

```
Select Case Response
Case vbAbort
    MsgBox "لقد نقرت الزر إحباط"
Case vbRetry
    MsgBox "لقد نقرت الزر إعادة المحاولة"
Case vbIgnore
    MsgBox "لقد نقرت الزر تجاهل"
End Select
```

فإذا نقرت بواسطة الفأرة على الزر تجاهل على سبيل المثال، فسوف تساوي قيمة المتحول Response إلى vbIgnore، وبالتالي سوف تنفذ العبارة الواقعة تحت Case vbIgnore (يوضح الجدول ٦-٤ معاني الثوابت vbAbort و vbRetry و vbIgnore).

تأخذ عبارات MsgBox الواردة تحت عبارة Select Case وسيطاً واحداً فقط وهو الرسالة المراد عرضها. مثال ذلك، عبارة MsgBox الواردة تحت السطر Case vbAbort، أي:

```
MsgBox "لقد نقرت الزر إحباط"
```

يؤدي عدم تحديد الوسيطين الثاني والثالث لعبارة MsgBox إلى ظهور مربع حوار يحمل زر موافق فقط، ودون أي رمز. كما يطلق على هذا المربع عنواناً افتراضياً وهو اسم البرنامج (اسم ملف المشروع).

تتشابه باقي الإجراءات في هذا البرنامج مع الإجراءات mnuAbortRetryIgnre_Click() فتقوم بإعداد المتحولات DialogMsg و DialogType و DialogTitle، ثم تستخدم التابع الوظيفي MsgBox() لإظهار مربع الحوار، ثم تستخدم بعد ذلك القيمة المعادة من التابع الوظيفي MsgBox() لتحديد الزر الذي تم النقر عليه.

التابع الوظيفي InputBox

يمكن استخدام التابع الوظيفي InputBox() لتلقي معلومات كتابية من قبل المستخدم. يُظهر التابع الوظيفي InputBox() مربع حوار مع رسالة وزرين هما: OK و Cancel، ويستطيع المستخدم إدخال نص ما في الحقل النصي وإغلاق مربع الحوار بالنقر على الزر OK.

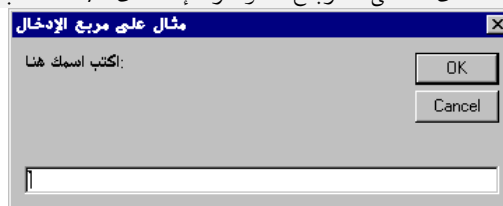
الوسيط الأول من التابع الوظيفي InputBox() هو رسالة مربع الحوار، والوسيط الثاني هو عنوان مربع الحوار. يعيد التابع الوظيفي InputBox() ما أدخله المستخدم في الحقل النصي. فمثلاً تظهر العبارة التالية، مربع الحوار المبين في الشكل ٦-٦ والذي يطلب من المستخدم إدخال اسم:

```
Name = InputBox("مثال على مربع حوار إدخال", "اكتب اسمك هنا:")
```

الشكل ٦-٦

مربع حوار إدخال يطلب

من المستخدم إدخال اسم.



يُدخل المستخدم اسماً ما في الحقل النصي. ثم ينقر الزر OK لإغلاق مربع الحوار. وفي مثالنا هذا نستخدم القيمة العائدة من التابع الوظيفي InputBox() لإسنادها إلى المتحول Name.

فيتم إسناد الاسم الذي أدخله المستخدم للمتحول Name فور نقر الزر OK.

لنحسن الآن برنامج مربعات الحوار بحيث يُستخدم التابع الوظيفي InputBox()، لتلقي سلسلة أحرف أو عدد أو تاريخ من المستخدم.

□ أضف البنود الأربعة الجديدة التالية، إلى قائمة برنامج مربعات الحوار وضعها فوق بند قائمة خروج.

العنوان	الخاصية Name
---------	--------------

mnuGetString	...إدخال نص
mnuGetNumber	...إدخال رقم
mnuGetDate	...إدخال تاريخ
mnuSep2	...

وبعد إضافة هذه العناصر يفترض أن يكون جدول القائمة مطابقاً للجدول ٧-٦.

جدول ٧-٦. جدول قائمة البرنامج الجديدة.

العنوان	الخاصية Name
&مربعات الحوار	mnuDialogs
...مربع الحوار موافق - إلغاء الأمر	mnuOKCancel
...مربع الحوار إحباط - إعادة المحاولة - تجاهل	mnuAbortRetryIgnore
...مربع الحوار نعم - لا - إلغاء الأمر	mnuYesNoCancel
...مربع الحوار نعم - لا	mnuYesNo
...مربع الحوار إعادة المحاولة - إلغاء الأمر	mnuRetryCancel
...	mnuSep1
...إدخال نص	mnuGetString
...إدخال رقم	mnuGetNumber
...إدخال تاريخ	mnuGetDate
...	mnuSep2
&خروج	mnuExit

□ اكتب النص التالي ضمن الإجراء mnuGetString_Click():

```
Private Sub mnuGetString_Click()  
    Dim UserInput
```

```

الحصول من المستخدم على إدخال نصي'
UserInput = _
    InputBox("مثال على مربع الإدخال" , "اكتب هنا أي نص:")
'إذا لم يكتب المستخدم أي شيء أنه ضغط الزر لإلغاء الأمر'
If UserInput = "" Then
    MsgBox "أنت لم تكتب أي نص أو أنك ضغطت الزر لإلغاء الأمر"
    Exit Sub
End If
'إظهار رسالة بما كتبه المستخدم'
MsgBox " & UserInput :لقد كتبت ما يلي"
End Sub

```

□ اكتب النص التالي ضمن الإجراء :mnuGetNumber_Click()

```

Private Sub mnuGetNumber_Click()
    Dim UserInput

    'الحصول من المستخدم على إدخال رقمي'
    UserInput = _
        InputBox("مثال على مربع الإدخال" , "اكتب هنا أي رقم:")
    'إذا لم يكتب المستخدم أي شيء أنه ضغط الزر لإلغاء الأمر'
    If UserInput = "" Then
        MsgBox "أنت لم تكتب أي شيء أو أنك ضغطت الزر لإلغاء الأمر"
        Exit Sub
    End If
    'إذا لم يكن ما كتبه المستخدم رقما فاخرج من الإجراء'
    If Not IsNumeric(UserInput) Then
        MsgBox "الرقم الذي كتبتة غير صحيح"
        Exit Sub
    End If
    'إظهار رسالة بالرقم الذي كتبه المستخدم'
    MsgBox " & UserInput :لقد كتبت الرقم التالي"
End Sub

```

□ اكتب النص التالي ضمن الإجراء :mnuGetData_Click()

```

Private Sub mnuGetData_Click()

```



```
Dim UserInput, DayOfWeek, Msg

'الحصول من المستخدم على إدخال زمني'
UserInput = _
    InputBox("مثال على مربع الإدخال" , " اكتب هنا أي تاريخ:" )
'إذا لم يكتب المستخدم أي شيء أنه ضغط الزر لإلغاء الأمر'
If UserInput = "" Then
    MsgBox "أنت لم تكتب أي شيء أو أنك ضغطت الزر لإلغاء الأمر"
    Exit Sub
End If
'إذا لم يكن ما كتبه المستخدم تاريخاً فإخرج من الإجراء'
If Not IsDate(UserInput) Then
    MsgBox "التاريخ الذي كتبته غير صحيح"
    Exit Sub
End If

DayOfWeek = Format(UserInput, "dddd")
'إظهار رسالة بالتاريخ الذي أدخله المستخدم'
'وإظهار النتيجة أيضاً'
Msg = "لقد أدخلت التاريخ: " & UserInput
Msg = Msg + vbCrLf + " _ :يوم التاريخ الذي أدخلته هو"
    & DayOfWeek

MsgBox Msg

End Sub
```

□ احفظ المشروع باختيار البند **Save Project** من قائمة **File**.

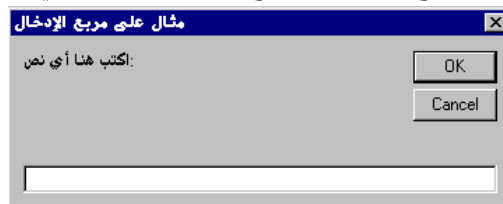
□ نفذ برنامج مربعات الحوار وتمرن على عناصر القائمة الجديدة.

□ اختر البند **إدخال نص** من قائمة **مربعات الحوار**.

يستجيب البرنامج بإظهار مربع الحوار المبين في الشكل ٦-٧.

الشكل ٦-٧

مربع حوار إدخال نص.



- اكتب أي نص في مربع الحوار هذا، ثم انقر على الزر موافق.
يستجيب البرنامج بإظهار رسالة تخبرك ماذا كتبت.
- اختر البند إدخال نص، ولكن هذه المرة انقر على الزر إلغاء الأمر بدلاً من الزر موافق.
يستجيب البرنامج بإظهار رسالة تخبرك بأنك لم تكتب شيئاً، أو أنك نقرت الزر إلغاء الأمر.
- اختر البند إدخال رقم من قائمة البرنامج.
يستجيب البرنامج بإظهار مربع حوار يطلب منك إدخال رقم.
- اكتب أي رقم صحيح مثل ١٢٣٤، ثم انقر على الزر موافق.
يستجيب البرنامج بإظهار رسالة بالرقم الذي كتبت.
- اختر البند إدخال رقم مجدداً، ولكن لا تكتب رقماً هذه المرة مثل ABCD.
يستجيب البرنامج بإظهار رسالة تخبرك بأنك أدخلت رقماً غير صحيح.
- تمرن الآن على البند إدخال تاريخ، بإدخال صيغتي تاريخ إحداهما صحيحة والأخرى غير صحيحة في مربع الحوار إدخال تاريخ. التاريخ الصحيح يمكن أن يحمل صيغة مشابهة للصيغ التالية: July 4,1776 و January 1,1994 و 01/01/1994. أما الأمثلة على الصيغ الغير صحيحة فكثيرة منها: 1234,ABCD و 14/77/93.
- لاحظ أنه بعد إدخالك للتاريخ الصحيح في مربع الحوار إدخال تاريخ فإن البرنامج يُظهر رسالة تحمل التاريخ الذي أدخلته إضافة إلى اليوم من ذلك الأسبوع التابع لهذا التاريخ. فمثلاً يستجيب البرنامج عند إدخال التاريخ ٢٠٠٠/١/١ بإظهار الرسالة: لقد أدخلت التاريخ ٢٠٠٠/١/١ ويوم التاريخ الذي أدخلته هو: السبت.

نص الإجراء mnuGetString_click():

ينفذ الإجراء mnuGetString_click() عند اختيار البند إدخال نص من القائمة. ويبدأ الإجراء بتنفيذ التابع الوظيفي InputBox():

```
UserInput = _
    InputBox("مثال على مربع الإدخال" , "اكتب هنا أي نص:")
```

تُظهر هذه العبارة، مربع الحوار المبين في الشكل ٦-٧، ويتم إسناد القيمة المعادة من التابع الوظيفي InputBox() إلى المتحول UserInput، تساوي القيمة المعادة من التابع الوظيفي "لا شيء" null، إذا لم يكتب المستخدم أي شيء في مربع الإدخال. أما إذا كتبت أي شيء ونقرت على الزر OK فالقيمة المعادة من التابع الوظيفي سوف تساوي الشيء الذي كتبتّه.

تستخدم العبارة If التالية للتحقق من الضغط على الزر Cancel:

```
If UserInput = "" Then
    MsgBox "أنت لم تكتب أي شيء أو أنك ضغطت الزر إلغاء الأمر"
    Exit Sub
End If
```

يتحقق الشرط UserInput = "" عند عدم إدخال أي شيء أو عند ضغط الزر Cancel، ونتيجة ذلك تظهر الرسالة "أنت لم تكتب أي شيء أو أنك ضغطت الزر إلغاء الأمر". وينتهي الإجراء بواسطة العبارة Exit Sub. أما إذا لم يتحقق الشرط UserInput = "" فسوف تنفذ آخر عبارة في الإجراء:

```
MsgBox UserInput & " :لقد كتبت ما يلي"
```

نص الإجراء mnuGetNumber_Click()

يُنفيذ الإجراء mnuGetNumber_Click() عند اختيار البند إدخال رقم من قائمة مربعات الحوار، يطلب منك إدخال رقم ما، ثم يتحقق بأن الرقم صحيح.

يبدأ الإجراء بتنفيذ التابع الوظيفي InputBox():

```
UserInput = _
    InputBox("مثال على مربع الإدخال" , "اكتب هنا أي رقم:")
```

تستخدم العبارة If للتحقق من النقر على الزر Cancel:

```
If UserInput = "" Then
    MsgBox "أنت لم تكتب أي شيء أو أنك ضغطت الزر إلغاء الأمر"
    Exit Sub
```

```
End If
```

يتحقق الشرط `userInput = ""` عند نقر الزر `Cancel`، أو عند عدم إدخال أي شيء ونتيجة ذلك تظهر العبارة: "أنت لم تكتب أي شيء أو أنك ضغطت الزر لإلغاء الأمر". وينتهي تنفيذ الإجراء بواسطة العبارة `Exit Sub`.

أما إذا أدخل المستخدم شيئاً، ولم يضغط على الزر `Cancel`، عندها يتأكد الإجراء بأن الرقم الذي أدخله المستخدم صحيح وذلك بواسطة التابع الوظيفي `IsNumeric()`:

```
If Not IsNumeric(UserInput) Then
    MsgBox "الرقم الذي كتبه غير صحيح"
Exit Sub
End If
```

فإذا كان المتحول `UserInput` قد ملئ برمز مثل `ABC`، يكون الشرط:

```
Not Is Numeric(UserInput)
```

محققاً ونتيجة لذلك ينفذ الجزء الواقع تحت عبارة الشرط `If`، فتظهر الرسالة "الرقم الذي كتبه غير صحيح". وينتهي الإجراء بواسطة العبارة `Exit Sub`. أما إذا كان المتحول `UserInput` مملوءاً برمز مثل `١٢٣`، فعندها يصبح الشرط السابق غير محقق وتنفذ آخر عبارة في الإجراء أي:

```
MsgBox userInput + " : لقد كتبت الرقم التالي"
```

بحيث تظهر هذه العبارة الرقم الذي أدخلته.

الإجراء `mnuGetdate_Click()`

ينفذ الإجراء `mnuGetdate_Click()` عند اختيار البند إدخال تاريخ من قائمة مربعات الحوار. يطلب منك الإجراء إدخال تاريخ ثم يتحقق من أن التاريخ الذي أدخلته صحيح. يبدأ الإجراء بتنفيذ التابع الوظيفي `InputBox()`:

```
UserInput = _
    InputBox("مثال على مربع الإدخال", "اكتب هنا أي تاريخ:")
```

تستخدم عبارة `If` كما تعودنا للتحقق من النقر على الزر `Cancel` أو من عدم إدخال شيء:

```
If userInput = "" Then
```

```

MsgBox "أنت لم تكتب أي شيء أو أنك ضغطت الزر إلغاء الأمر"
Exit Sub
End If

```

فعند عدم إدخال أي شيء أو عند نقر الزر Cancel يتحقق الشرط `userInput = ""`، وتظهر الرسالة "أنت لم تكتب أي شيء أو أنك ضغطت الزر إلغاء الأمر".

يلي خطوة التحقق السابقة خطوة يتأكد منها الإجراء من إدخال التاريخ بصيغة صحيحة وذلك بواسطة التابع الوظيفي `IsDate()`:

```

If Not IsDate(UserInput) Then
    MsgBox "التاريخ الذي كتبتة غير صحيح"
    Exit Sub
End If

```

يعتبر الشرط `Not IsDate(UserInput)` محققاً عندما يكون المتحول `UserInput` مملوءاً برموز لا تلي صيغة التاريخ الصحيحة أي قد يكون `UserInput` مملوءاً برموز مثل ١٢٣ أو أ ب ت.

ونتيجة ذلك، تظهر الرسالة "التاريخ الذي كتبتة غير صحيح"، وينتهي الإجراء بواسطة العبارة `Exit Sub`. وعلى العكس من ذلك لا يتحقق الشرط إذا كانت صيغة التاريخ قد أدخلت بشكل صحيح.

ينتقل التنفيذ نتيجة ذلك إلى آخر أربع عبارات في الإجراء:

```

DayOfWeek = Format(UserInput, "dddd")
Msg = "لقد أدخلت التاريخ: " & UserInput
Msg = Msg + vbCrLf + "ويوم التاريخ الذي أدخلته هو: " & DayOfWeek
MsgBox Msg

```

تظهر هذه العبارات التاريخ الذي أدخله المستخدم بواسطة المتحول `UserInput`، إضافة إلى اسم اليوم التابع لذلك التاريخ عبر المتحول `DayOfWeek`، يتم تحديث قيمة المتحول `DayOfWeek` بواسطة التابع `Format()`. فمثلاً إذا كان المتحول `UserInput` يساوي ١/١/٢٠٠٠، فعندها يعود التابع `Format(UserInput, "dddd")` بالقيمة "السبت".

استخدام الوسائط الأخرى للتابع InputBox

استخدم برنامج مربعات الحوار، التابع الوظيفي InputBox() مع وسيطين فقط: الرسالة التي سيتم إظهارها وعنوان مربع الحوار. يمكنك أيضاً استخدام وسائط اختيارية أخرى مع التابع المذكور، يحدد الوسيط الثالث قيمة افتراضية تظهر في الحقل النصي لمربع الحوار. أما الوسيطين الرابع والخامس فيحددان موقع مربع الحوار على الشاشة بوحدات مقدرة بوحدة Twips. فمثلاً استخدم العبارة التالية:

```
MyNumber=InputBox("أدخل رقماً: ", "مثال", "7", 100, 200)
```

لإظهار مربع حوار يحمل الخصائص التالية:

الرسالة هي: أدخل رقماً.

العنوان: مثال.

القيمة الافتراضية التي ستظهر في الحقل النصي هي: ٧.

تبعد الحافة اليسارية لمربع الحوار بمقدار 100 twips عن الحافة اليسارية للشاشة.

تبعد الحافة العلوية لمربع الحوار بمقدار 200 twips عن قمة الشاشة.

نتيجة ذلك، يظهر مربع الحوار عند إحداثيي الشاشة X=100 و Y=200، كما أن القيمة الافتراضية الموجودة في الحقل النصي تساوي ٧. يمكن للمستخدم النقر على OK في مربع الحوار، أو يمكنه إدخال رقم آخر بدلاً من الرقم ٧ ويتبعه بالنقر على OK.

مربعات الحوار المخصصة

يتم تفصيل مربعات الحوار المخصصة حسب ما يتضح من اسمها، من قبل المستخدم (بينما كانت مربعات الحوار مسبقة التعريف، تسمح لك بوضع أشياء محددة مثل الأزرار والرموز).

تتشابه عملية تصميم مربع حوار مخصص من تصميم النموذج Form.

بل إن مربع الحوار المخصص بالواقع، هو نموذج عادي يستخدم لإظهار والحصول على المعلومات من المستخدم، وبمجرد الانتهاء من تصميم مربع الحوار المخصص، يصبح بالوسع استخدامه من قبل أي برامج أخرى مكتوبة بلغة فيجول بيسك.

سنعمل على تحسين برنامج مربعات الحوار بغية توضيح كيفية تصميم واستخدام مربع حوار مخصص في البرنامج. وسنصمم مربع حوار مخصص يدعى GetMonth.frm ويسمح للمستخدم اختيار شهر ما. وكعادتنا بعد الانتهاء من طور التصميم سنضيف إلى برنامج مربعات الحوار النص اللازم لاستخدام مربع الحوار .GetMonth.frm

تصميم مربع حوار مخصص

استخدم الخطوات التالية لتصميم مربع الحوار :GetMonth.Frm:

□ تأكد أن المشروع Dialogs.VBp هو المشروع المفتوح حالياً في فيجول بيسك ثم

اختر البند **Add From** من قائمة **Project**.

يستجيب فيجول بيسك بإظهار الإطار **Add Form**. انظر الشكل ٦-٨.

□ اختر الرمز **Form** من الإطار **Add Form** ثم انقر على الزر **فتح**.

يستجيب فيجول بيسك بإضافة نموذج جديد يدعى **Form1** إلى المشروع **Dialogs.vbp**.

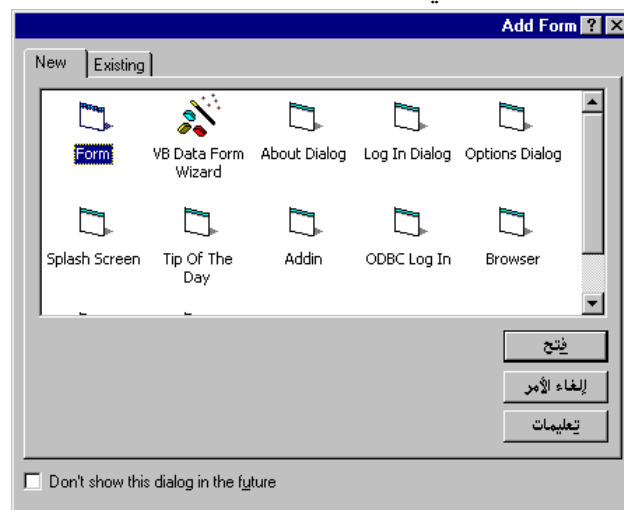
□ احفظ النموذج الجديد بالاسم **GetMonth.Frm** وذلك باختيار البند **Save Form1**

As من قائمة **File**، ثم استخدم مربع الحوار **Save File As** لحفظ النموذج بالاسم

GetMonth.frm في الدليل **C:\VB5Prg\Ch06**.

الشكل ٦-٨

الإطار **Add Form**.



قبل أن نذهب أبعد من ذلك، دعنا نتحقق بأن المشروع Dialogs.Vbp يحوي الآن نموذجين داخله.

□ اختر البند **Project Explorer** من قائمة **View** لإظهار الإطار Project.

يظهر الآن الإطار Project بشكل الشاشة المبينة في الشكل ٩-٦.

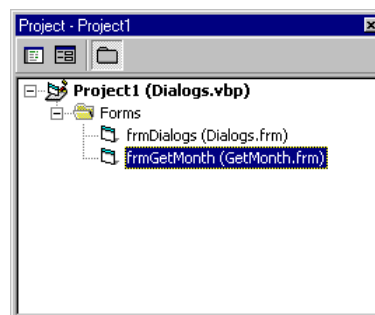
الشكل ٩-٦

يُظهر الإطار Project بأن المشروع

يملك نموذجين داخله:

Dialogs.Frm

و GetMonth.Frm



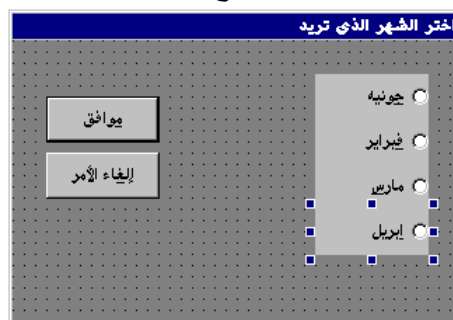
عليك الآن بناء مربع الحوار GetMonth.Frm طبقاً للجدول ٨-٦.

يفترض أن يظهر مربع الحوار المكتمل كما في الشكل ١٠-٦.

الشكل ١٠-٦

النموذج frmGetMonth

(طور التصميم).



□ احفظ العمل المنجز باختيار **Save Project** من القائمة **File** فإذا كنت قد أجريت

تعديلات على النموذج frmDialogs أو على النموذج GetMonth.Frm فسوف تُحفظ هذه التعديلات.

الجدول ٨-٦. جدول خصائص مربع الحوار GetMonth.Frm.

القيمة	الخاصية	الكائن
frmGetMonth	Name	Form
اختر الشهر الذي تريد	Caption	
Dark gray	BackColor	
1-Fixed Single	BorderStyle	
False	ControlBox	
False	MaxButton	
False	MinButton	
True	RightToLeft	
cmdOK	Name	CommandButton
&موافق	Caption	
False	Cancel	
True	Default	
True	RightToLeft	
cmdCancel	Name	CommandButton
إل&غاء الأمر	Caption	
True	Cancel	
False	Default	
True	RightToLeft	
optJan	Name	Option Button
جوانيه&	Caption	
optFeb	Name	Option Button
&فبراير	Caption	
optMar	Name	Option Button
مارس&	Caption	
optApr	Name	Option Button
&ابريل	Caption	

يطالبك الجدول ٦-٨ بوضع أربعة أزرار خيارات Option Buttons لأربع شهور فقط، ولهذا فالمستخدم سيكون قادراً على اختيار أحد الشهور التالية: جونه، فبراير، مارس، إبريل. والسبب وراء ذلك رغبتنا باختصار زمني التصميم وكتابة البرنامج. لكنك تستطيع تحسين نموذج frmGetMonth.frm لشمّل الشهور الاثني عشر كاملة.

خصائص مناسبة لمربعات الحوار

يصمم مربع الحوار النموذجي، بحيث لا يمكن تغيير شكل ظهوره أثناء زمن التنفيذ. وكما يتبين لنا من الجدول ٦-٨ فإن خصائص النموذج frmGetMonth قد أُعدت بحيث لا تسمح لك بتكبير أو تصغير مربع الحوار أثناء زمن التنفيذ. وُضعت الخاصية BorderStyle لمربع الحوار frmGetMonth على الحالة Fixed Single لكي لا يسمح بتغيير حجم مربع الحوار خلال زمن التنفيذ. وضعت الخاصية ControlBox لمربع الحوار frmGetMonth على الحالة False مما يعني أن مربع الحوار سيظهر بدون مربع قائمة التحكم Control Menu Box (وهو ذلك المربع الذي يظهر في الزاوية اليسرى العليا من النموذج). وضعت الخاصية MaxButton لمربع الحوار frmGetMonth على الحالة False مما يعني أن مربع الحوار سيظهر بدون زر التكبير في زاويته العليا اليمنى. مما يمنع المستخدم من تكبير مربع الحوار أثناء زمن التنفيذ. وضعت الخاصية MinButton لمربع الحوار على الحالة False مما يعني أن مربع الحوار سيظهر بدون زر التصغير في الزاوية اليمنى العليا. مما يمنع المستخدم من تصغير مربع الحوار أثناء زمن التنفيذ.

الخاصيتان Cancel و Default لأزرار الأوامر

كما تبين لنا من الجدول ٦-٨ فقد وضعت الخاصية Default للزر موافق على الحالة True كما وضعت الخاصية Cancel للزر إلغاء الأمر على الحالة True أيضاً.

يؤدي تعيين حالة الخاصية Default لزر الأمر على القيمة True إلى جعل زر الأمر هذا وكأنه زر افتراضي Default وهو ذلك الزر الذي يُعتبر أنه قد نقر عليه، إذا تم الضغط على المفتاح Enter.

ولذلك وبسبب وضع الخاصية Default للزر موافق على الحالة True فإن الضغط على المفتاح Enter عندما يكون مربع الحوار frmGetMonth ظاهراً أثناء زمن التنفيذ، يتسبب بنفس الأثر الذي ينجم عن نقر الزر موافق.

يؤدي تعيين حالة الخاصية Cancel لزر الأمر على القيمة True إلى جعل هذا الزر بمثابة مفتاح الهروب Escape، (هو الزر الذي يتم اختياره نتيجة الضغط على المفتاح Esc الموجود في لوحة المفاتيح). لذلك ونتيجةً لوضع الخاصية Cancel للزر إلغاء الأمر على الحالة True، فإن الضغط على المفتاح Esc أثناء ظهور مربع الحوار خلال زمن التنفيذ، يتسبب بنفس الأثر الذي ينجم عن نقر الزر إلغاء الأمر.

إظهار و إخفاء مربع حوار محدد

□ استخدم الطريقة Show لإظهار مربع حوار مخصص. المفروض في مربع الحوار أن يظهر كمربع حوار محدد Modal، (بمعنى أن المستخدم يجب أن يستجيب له، قبل أن يتمكن من استئناف تنفيذ البرنامج)، ولإظهار مربع الحوار كمربع حوار محدد، تستخدم الطريقة Show مع متحول وسيطي يساوي الواحد. فمثلاً لإظهار frmMyDialog كمربع حوار محدد استخدم العبارة التالية:

```
frmMyDialog.Show 1
```

□ استخدم الطريقة Hide لإخفاء مربع حوار. فمثلاً استخدم العبارة اللاحقة لإخفاء مربع الحوار frmMyDialog الظاهر حالياً:

```
frmMyDialog.Hide
```

سنضيف نص برنامج إلى برنامج مربعات الحوار، لتباين كيفية عمل الطريقتين Show و Hide على أرض الواقع.

سيُظهر نص البرنامج هذا، مربع الحوار frmGetMonth عندما ينتقي المستخدم عنصر القائمة المدعو الحصول على شهر، ثم يخفي ذلك المربع عندما يستجيب المستخدم له بالنقر على أحد الزرين: موافق أو إلغاء الأمر في ذلك المربع.

□ أضف عنصرين جديدين مع المواصفات التالية إلى قائمة النموذج frmDialogs (ضعها فوق البند خروج).

mnuGetMonth	...الحصول على الشهر (مربع حوار مخصص)
mnuSep3	...

و بعد إضافة هذين العنصرين يفترض أن يبدو جدول قائمة البرنامج Dialogs كذاك المبين في الجدول ٩-٦.

ملاحظة

استخدم الخطوتين التاليتين لإضافة عناصر القائمة إلى النموذج frmDialogs:

- اختر النموذج frmDialogs.
- اختر **Menu Editor** من القائمة **Tools**.

يستجيب فيجول بيسك بإظهار *Menu Editor* وإظهار قائمة النموذج frmDialogs. إذا كان النموذج frmGetMonth هو النموذج الحالي أي الإضاءة متوضعة عليه، ولم تختار النموذج frmDialogs قبل اختيار **Menu Editor** من القائمة **Tools**، فسوف يعتقد فيجول بيسك بأنك ترغب باستخدام محرر القوائم لتعديل قائمة النموذج frmGetMonth.

الجدول ٩-٦. قائمة برنامج مربعات الحوار بعد إضافة البند الحصول على شهر.

العنوان	الخاصية Name
---------	--------------

mnuDialogs	&مربعات الحوار
mnuOkCancel	...مربع الحوار موافق - إلغاء الأمر
mnuAbortRetryIgnore	...مربع الحوار إحباط - إعادة المحاولة - تجاهل
mnuYesNoCancel	...مربع الحوار نعم - لا - إلغاء الأمر
mnuYesNo	...مربع الحوار نعم - لا
mnuRetryCancel	...مربع الحوار إعادة المحاولة - إلغاء الأمر
mnuSep1	...
mnuGetString	...إدخال نص
mnuGetNumber	...إدخال رقم
mnuGetDate	...إدخال تاريخ
mnuSep2	...
mnuGetMonth	...الحصول على الشهر (مربع حوار مخصص)
mnuSep3	...
mnuExit	...&خروج

نص الإجراء mnuGetMonth_Click()

```
Private Sub mnuGetMonth_Click()
```

```
    'إظهار مربع الحوار المخصص frmGetMonth
    frmGetMonth.Show vbModal

    'إظهار الشهر الذي اختاره المستخدم
    'إذا قام بعملية الاختيار
    If frmGetMonth.Tag = "" Then
        MsgBox "لقد ألغيت عملية الاختيار"
    Else
```

```
MsgBox " : لقد اخترت الشهر" & frmGetMonth.Tag
End If
```

```
End Sub
```

يُنفذ الإجراء mnuGetMonth_Click() عند اختيار البند الحصول على شهر من قائمة مربعات الحوار، يستخدم نص برنامج هذا الإجراء الطريقة Show لإظهار مربع الحوار frmGetMonth كما يظهر الشهر الذي اختاره المستخدم، بإظهار الخاصية Tag لمربع الحوار frmGetMonth، وستلاحظ قريباً أن نص برنامج مربع الحوار frmGetMonth استخدم الخاصية Tag للنموذج frmGetMonth (أي frmGetMonth.Tag) لحفظ اسم الشهر الذي اختاره المستخدم.

□ تحقق أن العبارة Option Explicit تظهر في قسم التصاريح العامة للنموذج frmGetMonth بالشكل التالي:

```
يجب التصريح عن كل المتحولات'
Option Explicit
```

ملاحظة

لإظهار إطار برنامج النموذج frmGetMonth انقر نقرة مزدوجة بواسطة الفأرة فوق أي مكان من النموذج، ليظهر إطار برنامج النموذج. تأكدنا للتو من وجود العبارة Option Explicit في قسم التصاريح العامة للنموذج frmDialogs. يعني هذا أنه يتوجب عليك التصريح عن أي متحول قبل استخدامه في إجراءات النموذج frmDialogs. وبأسلوب مشابه يجب أن تكون العبارة Option Explicit موجودة في قسم التصاريح العامة للنموذج frmGetMonth. مما يعني أنه يتوجب التصريح عن أي متحول قبل استخدامه في إجراءات النموذج frmGetMonth. لهذا لا بد من استخدام العبارة Option Explicit مع كل نموذج من النماذج.

نص الإجراء cmdOK_Click()

```
Private Sub cmdOk_Click()
```

```

    'frmGetMonth.Tag تغيير قيمة الخاصية
    'إلى الشهر الذي اختاره المستخدم
    'تستخدم هذه الخاصية كخرج لهذا النموذج
    If optJan.Value = True Then frmGetMonth.Tag = "جونه"
    If optFeb.Value = True Then frmGetMonth.Tag = "فراير"
    If optMar.Value = True Then frmGetMonth.Tag = "مارس"
    If optApr.Value = True Then frmGetMonth.Tag = "ابريل"

    'إخفاء النموذج frmGetMonth
    frmGetMonth.Hide

```

```
End Sub
```

ينفذ الإجراء cmdOK_Click() عند نقر الزر موافق في مربع الحوار frmGetMonth يستخدم نص برنامج هذا الإجراء سلسلة من أربع عبارات للتعرف على زر الخيار المنتقى حالياً، ويعدل وفقاً لذلك، الخاصية Tag للنموذج frmGetMonth مكلفاً إياها بالشهر الذي تم انتقاؤه. يستخدم الإجراء بعد عبارات If الأربعة، طريقة Hide، لإخفاء مربع الحوار frmGetMonth من الشاشة.

تستخدم الخاصية Tag لمربع الحوار frmGetMonth كخرج لمربع الحوار. أي بمعنى أن باستطاعة من يعمل على مربع الحوار المذكور استخدام frmGetMonth.Tag لتحديد الشهر الذي تم اختياره.

إليك الآن عبارة If ذات الشكل:

```
If optJan.Value = True Then frmGetMonth.Tag = "جونه"
```

تعتبر العبارة هذه مطابقة للعبارة التالية:

```

If optJan.Value = True Then
    frmGetMonth.Tag = "جونه"
End If

```

أي لا تعود هناك حاجة لعبارة End If، إذا اتسعت عبارة If على سطر واحد. يفيد هذا الأمر بتقليل الجهد البرمجي، بل و يسهل عملية قراءة وضع البرنامج الذي يحمل مجموعة من عبارات If المتلاحقة.

نص الإجراء cmdCancel_Click()

```
Private Sub cmdCancel_Click()

    'تغيير قيمة الخاصية frmGetMonth
    'إلى لا شيء'
    frmGetMonth.Tag = ""

    'إخفاء النموذج frmGetMonth
    frmGetMonth.Hide

End Sub
```

ينفذ الإجراء cmdCancel_Click() عند نقر الزر **إلغاء الأمر**، في النموذج frmGetMonth. ويسند نص برنامج الإجراء هذا القيمة "" (أي Null) إلى الخاصية Tag للنموذج (أي frmGetMonth.Tag). ثم يستخدم الطريقة Hide لإخفاء مربع الحوار frmGetMonth من الشاشة. يعني تكليف frmGetMonth.Tag بالقيمة ""، أن المستخدم نقر الزر **إلغاء الأمر**.

□ احفظ العمل المنجز باختيار **Save Project** من القائمة **File** لفيجول بيسك.

□ نفذ برنامج مربعات الحوار، واختر الحصول على شهر من القائمة **مربعات**

الحوار، وتمرن على مربع الحوار المخصص.

لاحظ بشكل خاص، أن نقر أحد أزرار الخيارات، يؤدي إلى تمركز التحكم عنده. من وجهة نظر لوحة المفاتيح، نستطيع ضغط المفتاح Esc على لوحة المفاتيح، الأمر الذي يملك نفس التأثير الناتج عن نقر الزر **إلغاء الأمر** بواسطة الفأرة، كما أن ضغط المفتاح Enter عندما يكون التحكم متركزاً عند أزرار الخيار، يكافئ نقر الزر **موافق**،

والسبب كما ذكرنا قبلاً، أن الخاصية Default للزر موافق تساوي True، كما أن الخاصية Cancel للزر إلغاء الأمر تساوي True أيضاً.

□ اختر البند خروج من القائمة مربعات الحوار لإنهاء البرنامج.

مربعات الحوار الشائعة

يتضمن فيجول بيسك عنصر تحكم، يُطلق عليه الاسم CommonDialog (أي مربعات الحوار الشائعة)، يمكن استخدام عنصر التحكم هذا، لإظهار مربعات الحوار الشائعة أثناء زمن التنفيذ بمجرد تعيين خصائصها. يُستخدم مربع الحوار الشائع غالباً، كمربع حوار يسمح للمستخدم باختيار وحفظ الملفات.

برنامج مربعات الحوار الشائعة

سنكتب الآن برنامج مربعات الحوار الشائعة، لتوضيح كيفية استخدام عنصر التحكم CommonDialog في برنامج ما. يُظهر هذا البرنامج، مربعي حوار: مربع الحوار اختيار لون ومربع الحوار فتح.

التمثيل المرئي لبرنامج مربعات الحوار الشائعة

سنبدأ كعادتنا بتمثيل نموذج البرنامج:

□ ابدأ مشروعاً من النوع التنفيذي القياسي Standard EXE.

□ احفظ نموذج المشروع بالاسم Common.Frm في الدليل C:\VB5Prg\Ch06.

□ احفظ ملف المشروع بالاسم Common.Vbp في الدليل نفسه.

يعتبر زر الأمر Command Button مثالاً على عنصر تحكم، يظهر دائماً في إطار مربع الأدوات، كما أن مربع النص و زر الخيار كلها أمثلة على عناصر تحكم تظهر دائماً في إطار مربع الأدوات.

من جهة أخرى، يُعتبر عنصر التحكم Common Dialog مثالاً على عنصر تحكم لا يظهر دائماً في مربع الأدوات، و إنما يضاف أو يزال من قبل المستخدم.

يطلق على عنصر التحكم مربعات الحوار الشائعة، اسم عنصر التحكم OCX. وكما ذكرنا في الفصل الأول، تحت عنوان كتابة أول برنامج لك في فيجول بيسك، أنه يتوجب على المستخدم التحقق من وجود الملف Msvbvm50.DLL في الدليل System. وهو ملف يحوي المكتبات الضرورية لتشغيل برنامجك التنفيذي. فمثلاً، يمكن البرنامج من استخدام القوائم والفأرة وكل المظاهر الأخرى التي قد يحتويها ملفك التنفيذي. ومن بين الأشياء التي يحويها الملف Msvbvm50.DLL هي عناصر التحكم التي تشاهد دوماً في إطار مربع الأدوات، وهو السبب الذي يمكنك من توزيع برامجك المكتوبة بلغة فيجول بيسك، دون توزيع ملفات إضافية ترافق عناصر التحكم هذه.

أما عناصر التحكم من أمثال مربعات الحوار الشائعة، فليست محتواة في الملف Msvbvm50.DLL، مما يعني أنه يتوجب عليك عند توزيع ملفاتك التنفيذية EXE إرفاقها بالملفات المرافقة لعناصر OCX التي يستخدمها برنامجك.

يدعى الملف الذي يرافق عنصر تحكم مربعات الحوار الشائعة، بالاسم Comdlg32.ocx ويفترض أن تشاهده ضمن الدليل System. (يتم تنصيب هذا الملف أثناء عملية إعداد فيجول بيسك).

لننظر الآن إذا كان عنصر التحكم CommonDialog محتوياً في المشروع Common.Vbp.

□ اختر البند Custom Components من القائمة Project لفيجول بيسك.

يستجيب فيجول بيسك بإظهار مربع الحوار Components (انظر الشكل ٦-١١).

يمتلك الإطار Components ثلاث صفحات في الأعلى، كما يظهر من الشكل ٦-١١.

□ تحقق أن الصفحة Controls هي الظاهرة (كما في الشكل ٦-١١).

□ تدرّج عبر البنود، حتى تصل للبند Microsoft CommonDialog Control 5.0،

(انظر الشكل ٦-١١). إذا كانت علامة الاختيار ظاهرة في الخانة المجاورة لهذا

البند، فهذا يعني أنه موجود مسبقاً في إطار مربع الأدوات.

□ أما إذا لم تكن علامة الاختيار موجودة في الخانة المجاورة لهذا البند Microsoft

CommonDialog Control 5.0، فانقر فوق الخانة لوضع علامة اختيار.

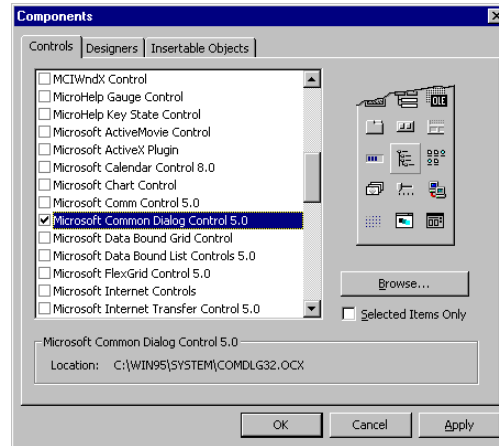
□ انقر الزر OK في الإطار Component.

يضاف نتيجة ذلك، عنصر التحكم *CommonDialog* إلى مربع الأدوات. ستضطر غالباً إلى سحب الحافة السفلية لمربع الأدوات Tool Box لتتمكن من مشاهدة عنصر التحكم *CommonDialog*، يوضح الشكل ١٢-٦ عنصر التحكم *CommonDialog* في إطار مربع الأدوات.

الشكل ١١-٦

مربع الحوار

Components



يظهر مستطيل أصفر عند وضع مؤشر الفأرة فوق عنصر التحكم هذا، لتظهر الرسالة *CommonDialog* داخل هذا المستطيل. وبهذه الطريقة تتحقق من وضع رمز عنصر التحكم *CommonDialog* في إطار مربع الأدوات.

❑ اختر النموذج *Common.Frm* بالنقر في أي مكان عليه.

❑ انقر نقرة مزدوجة على عنصر التحكم *CommonDialog*.

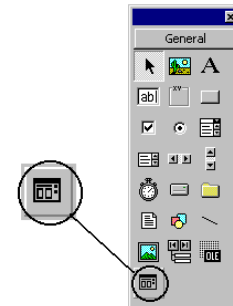
يستجيب فيجول بيسك بوضع عنصر التحكم *CommonDialog* في النموذج *Common.Frm*، والاسم الافتراضي الذي يطلقه فيجول بيسك على عنصر التحكم هذا هو *CommonDialog1*.

الشكل ١٢-٦

رمز عنصر تحكم

مربعات الحوار الشائعة

في نافذة الأدوات.



❑ أنشئ النموذج *Common.Frm* طبقاً للجدولين ١٠-٦ و ١١-٦ (الجدول ١١-٦)

هو جدول قائمة النموذج frmCommon.

الجدول ٦-١٠. جدول خصائص النموذج Common.Frm.

القيمة	الخاصية	الكائن
frmCommon	Name	Form
برنامج مربعات الحوار الشائعة	Caption	
True	RightToLeft	
CommonDialog1	Name	Common Dialog
True	CancelError	
(انظر الجدول ٦-١١)	(انظر الجدول ٦-١١)	Menu

الجدول ٦-١١. جدول قائمة النموذج frmCommon.

العنوان	الاسم
&ملف	mnuFile
...مربع حوار &اختيار الألوان	mnuColor
...مربع حوار &فتح ملف	mnuOpen
...	mnuSep1
...&خروج	mnuExit

الشكل ٦-١٣

النموذج frmCommon.



كتابة نص برنامج مربعات الحوار الشائعة

سنبدأ الآن بإدخال نص برنامج مربعات الحوار الشائعة:

□ تحقق من ظهور العبارة Option Explicit في قسم التصاريح العامة للنموذج frmCommon أي:

```
' يجب التصريح عن كل المتحولات '
Option Explicit
```

□ اكتب ما يلي في الإجراء :mnuColor_Click()

```
Private Sub mnuColor_Click()
    تشغيل متصيد الأخطاء لمعرفة أن المستخدم ضغط
    زر إلغاء الأمر في مربع حوار الألوان
    On Error GoTo ColorError
    إظهار مربع حوار الألوان
    CommonDialog1.ShowColor

    تغيير لون خلفية البرنامج إلى اللون الذي
    اختاره المستخدم
    frmCommon.BackColor = CommonDialog1.Color

    الخروج من الإجراء
    Exit Sub

ColorError:
    ينفذ هذا الجزء من الإجراء عندما يضغط المستخدم
    على زر إلغاء الأمر
    MsgBox "لقد ألغيت مربع حوار الألوان"
    Exit Sub
End Sub
```

□ اكتب ما يلي في الإجراء :mnuOpen_Click()

```
Private Sub mnuOpen_Click()

    Dim Filter As String
```

```

تشغيل متصيد الأخطاء لمعرفة أن المستخدم ضغط'
زر إلغاء الأمر في مربع حوار فتح ملف'
On Error GoTo OpenError

تحديد أنواع الملفات المراد فتحها'
Filter = "(.*)|*.txt|*"
Filter = Filter + "(.*)|*.txt|*"
Filter = Filter + "(.*)|*.bat|*"

CommonDialog1.Filter = Filter

تحديد نوع الملفات الافتراضي وهو الملفات النصية'
CommonDialog1.FilterIndex = 2

إظهار صندوق الحوار'
CommonDialog1.ShowOpen

إظهار اسم الملف الذي اختاره المستخدم'
MsgBox "لقد اخترت الملف " & CommonDialog1.filename

Exit Sub

OpenError:
ينفذ هذا الجزء من الإجراء عندما يضغط المستخدم'
على زر إلغاء الأمر'
MsgBox "لقد ألغيت مربع حوار فتح ملف"
Exit Sub

End Sub

```

□ اكتب ما يلي في الإجراء :mnuExit_Click()

```

Private Sub mnuExit_Click()
    End
End Sub

```

□ احفظ المشروع كاملاً.

تنفيذ برنامج مربعات الحوار الشائعة

ننّفذ برنامج مربعات الحوار الشائعة، ولاحظ أن عنصر التحكم CommonDialog لا يكون مرئياً أثناء مرحلة التنفيذ، وإنما يظهر حالما يتم اختيار مربع حوار اختيار الألوان أو مربع حوار فتح ملف من القائمة ملف.

□ اختر مربع حوار اختيار الألوان من القائمة ملف.

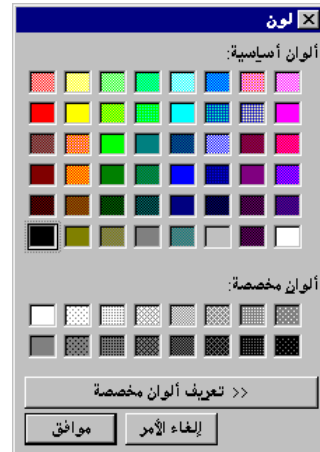
يستجيب برنامج مربعات الحوار الشائعة بإظهار مربع الحوار لون، انظر الشكل ٦-١٤.

□ انتق لوناً من مربع حوار لون، بنقر اللون المطلوب، ثم انقر الزر موافق.

يستجيب برنامج مربعات الحوار الشائعة، بإغلاق مربع الحوار لون، وتبديل لون خلفية البرنامج إلى اللون المنتقى.

الشكل ٦-١٤

مربع الحوار لون.



□ اختر مربع الحوار فتح ملف من القائمة ملف.

يستجيب البرنامج بإظهار مربع الحوار فتح، انظر الشكل ٦-١٥.

□ اختر دليلاً محدداً على قرصك الصلب، واختر من مربع الملفات، البند كل الملفات، ثم اكتب *.* في مربع اسم الملف، واختر ملفاً ثم انقر الزر فتح في مربع الحوار فتح (انظر الشكل ٦-١٥).

يستجيب برنامج مربعات الحوار الشائعة، بإظهار رسالة تخبرك اسم الملف الذي اخترته.

الشكل ٦-١٥

مربع الحوار فتح.



□ تمرن على برنامج مربعات الحوار الشائعة ثم اختر خروج من القائمة ملف لإنهاء عمل البرنامج.

كيف يعمل برنامج مربعات الحوار الشائعة

يستخدم البرنامج عنصر التحكم المخصص CommonDialog، لإظهار مربع الحوار لون ومربع الحوار فتح، يمكنك مربع حوار اللون من اختيار لون ما من لوح الألوان، ويمكنك مربع الحوار فتح من اختيار ملف.

مربع الحوار لون

يُنَفَّذُ الإجراء mnuColor_Click() آلياً، عند اختيار مربع حوار اختيار الألوان من القائمة ملف، يستخدم نص برنامج هذا الإجراء، عنصر التحكم المخصص CommonDialog لإظهار مربع الحوار لون:

```
Private Sub mnuColor_Click()
```

```
    'تشغيل متصيد الأخطاء لمعرفة أن المستخدم ضغط'  
    'زر إلغاء الأمر في مربع حوار الألوان'  
    On Error GoTo ColorError
```

```
    'إظهار مربع حوار الألوان'  
    CommonDialog1.ShowColor
```

```
    'تغيير لون خلفية البرنامج إلى اللون الذي'
```



```

'اختاره المستخدم'
frmCommon.BackColor = CommonDialog1.Color

'الخروج من الإجراء'
Exit Sub

ColorError:
'ينفذ هذا الجزء من الإجراء عندما يضغط المستخدم'
'على زر إلغاء الأمر'
MsgBox "لقد ألغيت مربع حوار الألوان"
Exit Sub

```

End Sub

يضع الإجراء mnuColor_Click() مصيدة للأخطاء، قبل إظهاره لمربع الحوار:

```
On Error GoTo ColorError
```

وظيفة هذه المصيدة، مراقبة (أو تصيد) الخطأ أثناء إظهاره مربع الحوار. لنفترض جدلاً، أنك وضعت الخاصية CancelError لعنصر التحكم المخصص CommonDialog1 أثناء طور التصميم على الحالة True، وهكذا، سيؤدي ضغط الزر إلغاء الأمر، أثناء ظهور مربع الحوار إلى توليد خطأ. ونتيجة لمرحلة تصيد الخطأ السابقة، سيتحول تنفيذ البرنامج إلى اللافتة ColorError.

يُظهر الإجراء بعد مرحلة تصيد الخطأ، مربع الحوار لون عن طريق العبارة التالية:

```
CommonDialog1.ShowColor
```

كما اتفقنا، يؤدي نقر الزر إلغاء الأمر أثناء ظهور مربع الحوار إلى توليد خطأ، ونتيجة لذلك ينفذ جزء البرنامج الواقع تحت اللافتة ColorError. يُظهر جزء البرنامج هذا رسالة، ثم ينهي تنفيذ الإجراء:

```

ColorError:
'ينفذ هذا الجزء من الإجراء عندما يضغط المستخدم'
'على زر إلغاء الأمر'
MsgBox "لقد ألغيت مربع حوار الألوان"
Exit Sub

```

إما إذا لم تضغط الزر **إلغاء الأمر**، وإنما اخترت بدلاً من ذلك لوناً من لوح الألوان، ثم أتبع ذلك بنقر الزر **موافق**. فسينفذ الإجراء عند ذلك، العبارتين التاليتين:

```
frmCommon.BackColor = CommonDialog1.Color
Exit Sub
```

تُبدّل العبارة الأولى لون النموذج إلى اللون الذي اخترته، والذي يتحدد بالخاصية `Color` لعنصر التحكم `CommonDialog1` (أي `CommonDialog1.Color`). وتنتهي العبارة الثانية تنفيذ الإجراء.

مربع الحوار فتح

ينفذ الإجراء `mnuOpen_Click()` عند اختيار **فتح ملف** من القائمة **ملف**. يستخدم نص هذا البرنامج عنصر التحكم `CommonDialog1` لإظهار مربع الحوار **فتح**:

```
Private Sub mnuOpen_Click()

    Dim Filter As String

    'تشغيل متصيد الأخطاء لمعرفة أن المستخدم ضغط'
    'زر إلغاء الأمر في مربع حوار فتح ملف'
    On Error GoTo OpenError

    'تحديد أنواع الملفات المراد فتحها'
    Filter = "(*.*)|*.*)"
    Filter = Filter + "الملفات النصية (*.txt)|*.txt|"
    Filter = Filter + "الملفات الدفعية (*.bat)|*.bat"

    CommonDialog1.Filter = Filter

    'تحديد نوع الملفات الافتراضي وهو الملفات النصية'
    CommonDialog1.FilterIndex = 2

    'إظهار صندوق الحوار'
    CommonDialog1.ShowOpen
```

```

إظهار اسم الملف الذي اختاره المستخدم'
MsgBox "لقد اخترت الملف" & CommonDialog1.filename

Exit Sub

OpenError:
ينفذ هذا الجزء من الإجراء عندما يضغط المستخدم'
على زر إلغاء الأمر'
MsgBox "لقد ألغيت مربع حوار فتح ملف"
Exit Sub

End Sub

```

تعد أول عبارة في الإجراء فحاً للخطأ الممكن وقوعه:

```
On Error GoTo ColorError
```

وبنتيجة مرحلة التصيد هذه، يتسبب نقر الزر **إلغاء الأمر** أثناء ظهور مربع الحوار فتح إلى تولد خطأ أثناء مرحلة التنفيذ، مما يؤدي إلى انتقال التنفيذ إلى الالفة OpenError: يستخدم الإجراء بعد مرحلة التصيد هذه، الخاصية Filter لعنصر التحكم CommonDialog1 لملء بنود مربع سرد أنواع الملفات الموجود في مربع فتح. يُملأ مربع سرد أنواع الملفات بثلاثة أنواع:

```

Filter = "كل الملفات (*.*)|*.*|"
Filter = Filter + "الملفات النصية (*.txt)|*.txt|"
Filter = Filter + "الملفات الدفعية (*.bat)|*.bat"
CommonDialog1.Filter = Filter

```

تملأ العبارات السابقة، الخاصية Filter لعنصر التحكم CommonDialog1 بسلسلة تُحدد نصوص البنود التي ستظهر في مربع سرد الملفات، إضافة إلى أنواع هذه الملفات: (*.* و *.txt و *.bat).

فمثلاً يتحدد أول بند يظهر في مربع سرد الملفات بالسلسلة التالية:

```
Filter = "كل الملفات (*.*)|*.*|"
```

السلسلة مؤلفة من جزئين، يفصلان بالحرف (|)، ويمكن الحصول عليه بالضغط على Shift+\ . يُشير الجزء الأول من السلسلة إلى النص الذي يظهر في مربع سرد الملفات:

```
(*.*) كل الملفات
```

أما الجزء الثاني، فيشير إلى هيكلية الملفات التي ستظهر عند اختيار هذا البند أي عند اختيار:

```
*.*
```

بمعنى أن اختيار البند الأول من مربع سرد الملفات، سيؤدي إلى ظهور الملفات ذات الهيكلية *.* (كل الملفات) في مربع الحوار.

لاحظ أن العبارات السابقة عمدت إلى ملء الخاصية Filter لعنصر التحكم CommonDialog1 على مراحل، وذلك بهدف التوضيح. فتم أولاً ملء المتحول Fitter:

```
Filter = " (*.*)|*.*|"
Filter = Filter + " الملفات النصية (*.txt)|*.txt|"
Filter = Filter + " الملفات الدفعية (*.bat)|*.bat"
```

عنصر التحكم CommonDialog1 بقيمة المتحول Filter:

```
CommonDialog1.Filter = Filter
```

بعد الانتهاء من ملء مربع سرد الملفات الموجود في مربع الحوار فتح، يتم تحديد البند الافتراضي الذي سيظهر في مربع سرد الملفات، وهو هنا البند ٢، وتتجز هذه الخطوات بإسناد القيمة ٢ إلى الخاصية FilterIndex للعنصر CommonDialog1.

```
CommonDialog1.FilterIndex = 2
```

وباعتبار البند الثاني في مربع سرد الملفات، هو الملفات النصية (*.txt) فالملفات الافتراضية التي ستدرج في مربع الحوار فتح، هي الملفات ذات الامتداد .TXT. يُظهر الإجراء، مربع الحوار فتح بواسطة العبارة التالية:

```
CommonDialog1.ShowOpen
```

يتولد خطأ في مرحلة التنفيذ، إذا تم نقر الزر إلغاء الأمر أثناء ظهور مربع الحوار، ونتيجة لذلك، يُنفذ جزء البرنامج الواقع تحت الالفة OpenError ، يُظهر هذا الجزء رسالة، وينتهي تنفيذ البرنامج OpenError:

```

OpenError:
ينفذ هذا الجزء من الإجراء عندما يضغط المستخدم '
على زر إلغاء الأمر'
MsgBox "لقد ألغيت مربع حوار فتح ملف"
Exit Sub

```

أما إذا لم يتم النقر على الزر **إلغاء الأمر**، وإنما اختار المستخدم ملفاً من مربع الحوار، فسوف يتم إظهار اسم الملف وينتهي الإجراء:

```

MsgBox "لقد اخترت الملف" & CommonDialog1.filename
Exit Sub

```

يقع اسم الملف الذي اخترته، في الخاصية `CommonDialog1.FileName`.

مربعات الحوار الشائعة الأخرى

يستخدم برنامج مربعات الحوار الشائعة، عنصر التحكم `CommanDialog`، لإظهار مربعي الحوار لون وفتح. يمكن في الواقع استخدام عنصر التحكم هذا، لإظهار مربعات الحوار الشائعة الأخرى، يُدرج الجدول ٦-١٢ مربعات الحوار الشائعة التي يمكن إظهارها، والقيم التي يجب إسنادها للخاصية `Action` حتى تظهر هذه المربعات. فمثلاً لإظهار مربع الحوار **حفظ باسم**، يتوجب عليك إسناد القيمة ٢ إلى الخاصية

:Action

```
CommonDialog1.Action = 2
```

أو بواسطة العبارة التالية الأكثر وضوحاً:

```
CommonDialog1.ShowSave
```

الجدول ٦-١٢. مربعات الحوار الشائعة، وكيفية إظهارها.

إظهار مربع الحوار	أسند للخاصية Action القيمة	الطريقة المباشرة
فتح	١	ShowOpen
حفظ باسم	٢	ShowSave

ShowColor	٣	الألوان
ShowFont	٤	الخطوط
ShowPrinter	٥	الطابعة

وكما كان عليه الحال مع مربعي الحوار **فتح و لون**، تستطيع استخدام خصائص عنصر التحكم **CommonDialog**، لتحديد استجابة المستخدم لمربع الحوار.

نماذج ومربعات حوار أخرى

تعلمت حتى الآن، كيفية إنشاء مربعات حوار باستخدام العبارة **MsgBox**، والتابع الوظيفي **MsgBox()**، وباستخدام عنصر التحكم **CommonDialog**.

لازالت هناك طريقة أخرى لإنشاء مربعات حوار خاصة، وذلك باستخدام النماذج مسبقة التحضير **Prepared Forms**، لنشاهد ذلك على أرض الواقع. اتبع الخطوات التالية:

❑ أنشئ مشروعاً جديداً من النوع التنفيذي القياسي **Standard EXE**.

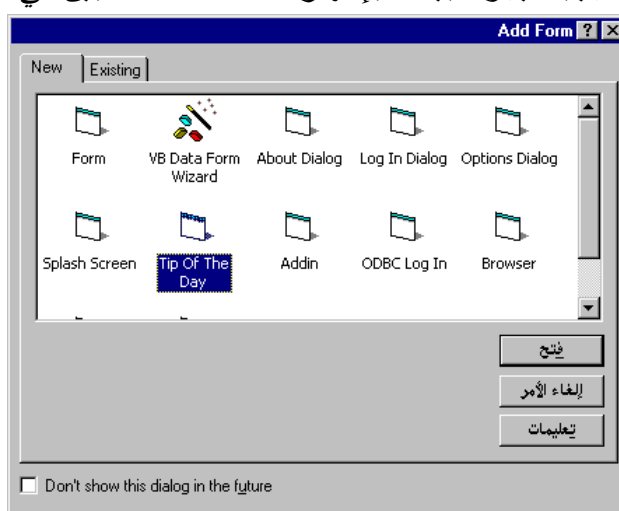
لن نطالبك بحفظ هذا المشروع، لأن غرضنا من هذه الخطوات، مجرد توضيح كيفية الوصول إلى النماذج مسبقة التحضير.

❑ اختر **Add Form** من القائمة **Project**.

يستجيب فيجول بيسك بإظهار **Add Form** المبين في الشكل التالي:

الشكل ٦-١٦

الإطار **Add Form**.



يوجد في الشكل ٦-١٦، كما تلاحظ أنواعاً مختلفة من النماذج مسبقة التحضير يمكن الاختيار من بينها.

إذا رجعت في هذا الفصل إلى الوراء قليلاً. ستجد أننا أضفنا النموذج frmGetMonth باختيار البند Add Form من القائمة Project ثم اخترنا الرمز Form من الإطار Add Form ، وبعد ذلك أنهينا تمثيل النموذج بوضع عناصر التحكم في النموذج، وربطها مع النص البرمجي اللازم لإظهار وإخفاء النموذج frmGetMonth.

□ اختر الرمز Tip of the Day من الإطار Add Form، ثم انقر الزر فتح في ذلك الإطار.

يستجيب فيجول بيسك بإظهار النموذج Tip of the Day في مرحلة التصميم. (انظر الشكل ٦-١٧):

الشكل ٦-١٧

نموذج تلميح اليوم

في مرحلة التصميم.



ملاحظة

تم تعريب النموذج السابق، حتى أصبح بهذا الشكل، فالنموذج Tip Of The Day الأصلي والذي يأتي مع فيجول بيسك غير معرب أساساً، تستطيع الحصول على النموذج المعرب، من القرص المدمج المرافق لهذا الكتاب في دليل أمثلة الكتاب، الفصل السادس، باسم Tip.frm.

يتضمن فيجول بيسك بعض النماذج مسبقة التحضير التي يسود استخدامها في التطبيقات (مثل النموذج Tip of the Day). تُستأنف عملية التصميم بنفس الطريقة التي اتبعناها عند تصميم النموذج frmGetMonth، والاختلاف الوحيد أن النموذج Tip of the

Day. أعدّ مسبقاً من أجلك، (وإذا رجعت إلى النص البرمجي له، فستجد جزءاً برمجياً كُتب أيضاً من أجلك).

باختصار لازال عليك الاستمرار في العمل على النموذج (فمثلاً يجب عليك أيضاً بناء النموذج الرئيسي للتطبيق)، ثم كتابة نص البرنامج اللازم لإظهار وإخفاء النموذج أيضاً Tip of the Day تماماً كما فعلنا مع النموذج frmGetMonth .

يعتبر استخدام النماذج مسبقة التحضير، مفيداً أحياناً، فمثلاً تستطيع الاستفادة من النموذج Tip Of the Day الموجود أصلاً عند الحاجة إلى بناء تطبيق ما، (فما الفائدة من إعادة اختراع العجلات) وطبعاً ستكون أقدر على الاستفادة من النماذج الجاهزة هذه عندما تلم بشكل صحيح بلغة فيجول بيسك، (لأنك ستكون أقدر على فهم النصوص البرمجية المكتوبة لهذه النماذج).

أما في الوقت الحاضر فلن نستخدم النماذج الجاهزة، ولكن بعد انتهاءك من هذا الكتاب والتمرس على بناء تطبيقاتك الخاصة، ارجع إلى الإطار Add Form وحاول بناء بعض التطبيقات التي تستخدم بعض النماذج الجاهزة.

□ أنه فيجول بيسك، ولا توجد حاجة لحفظ المشروع أو نماذج هذا المشروع.

الخلاصة

شرح هذا الفصل بإسهاب، مربعات الحوار على اختلاف أنواعها. وتعلمت كيفية إظهار رسالة ما للمستخدم، أو سؤال المستخدم عن أمر معين ومعرفة رده ومعالجته. باستخدام العبارة MsgBox أو التابع الوظيفي MsgBox().

كما تعلمت كيفية الحصول من المستخدم، على معلومات نصية أو رقمية أو زمنية. باستخدام التابع الوظيفي InputBox().

تعلمت أيضاً، كيفية إنشاء مربعات حوار خاصة بك، واستخدامها في برامجك. وعند إنشاء مربعات الحوار الخاصة، فإنه لا يوجد حدود لعدد الأزرار أو عناصر التحكم التي يمكنك وضعها على مربع حوارك الخاص.

غطى هذا الفصل أيضاً مربعات الحوار الشائعة، مثل مربع حوار لون أو مربع حوار فتح أو مربع حوار حفظ، وغيرها من مربعات الحوار الأخرى.