
ユーザーフォームの作り方

ここでは、Excel 2003 でのユーザーフォームを作成しマクロを貼り付ける方法について説明する。

1. ユーザーフォームの作成

Excel 2003 のメニューバーの「ツール」をクリックし、表示されるメニューで「マクロ／Visual Basic Editor」をクリックすると、図 1 のような「Microsoft Visual Basic」のウィンドウ（以下「VBA ウィンドウ」と呼ぶ。）が開く。

最初に留意すべきことは、このウィンドウを用いて作成するユーザーフォームやマクロはすべて元々の Excel ファイル（右図の例では“Book1.xls”）に格納されることである。ユーザーフォームやマクロを作成した後、このウィンドウを閉じたとしても、すべての情報が Excel ファイルに格納されているので、再表示すればさらに修正等が可能である。

図 1 の VBA ウィンドウで、左側上部に表示される「プロジェクト」ウィンドウは、「プロジェクトエクスプローラ」と呼ばれるもので当該 Excel ファイル（右図の例では“Book1.xls”）の構成を表示するものであり、左側下部に表示される「プロパティ」ウィンドウは、その時点でアクティブになっているユーザーフォームやコントロールの属性を表示するものである。

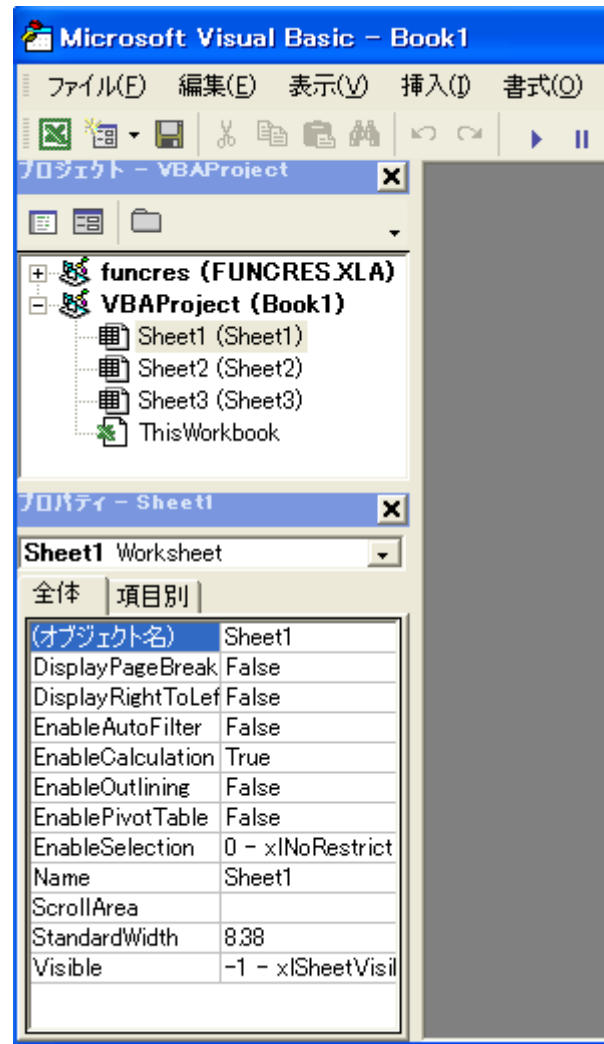


図 1 「Microsoft Visual Basic」ウィンドウ

図 2 のように左から 2 番目にある「ユーザーフォームの挿入」ボタンをクリックすると、“作業領域”に図 3 のユーザーフォームが表示され、あわせて図 4 の「ツールボックス」が表示される。

ツールボックスに表示されるアイコンは「コントロール」と呼ばれるものであり、コントロールをユーザーフォーム上に貼り付けることでユーザーフォームを完成させることになる。

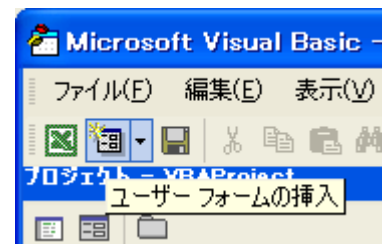


図 2

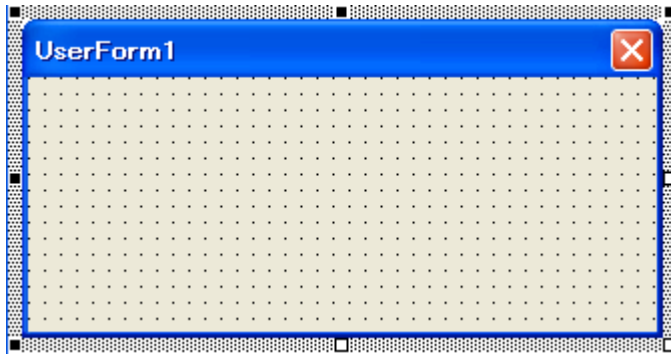


図 3



図 4

このとき、図 3 のようにユーザーフォームの周囲が網掛け線で囲まれているが、このことはユーザーフォームがアクティブになっていることを意味する。さらに、プロジェクトエクスプローラに「UserForm1」が追加され、プロパティはこの時点でアクティブな「UserForm1」の属性を表示していることが確認できる。

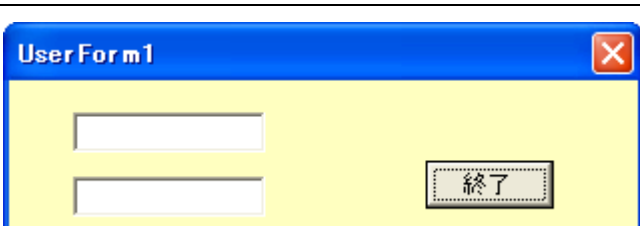
2. コントロール

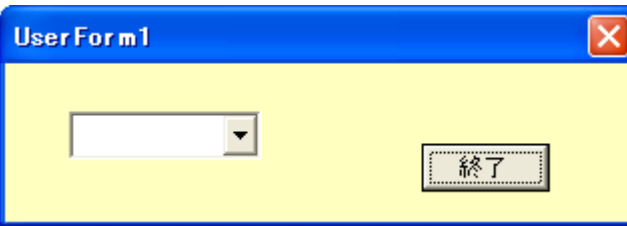
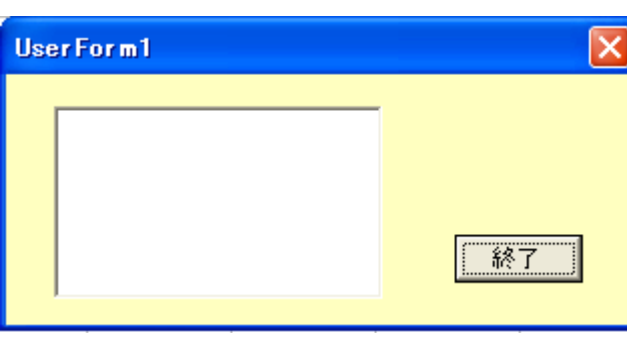
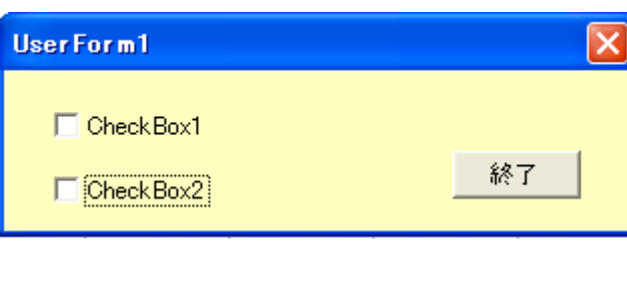
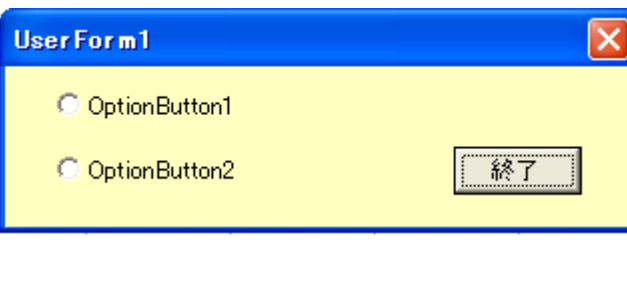
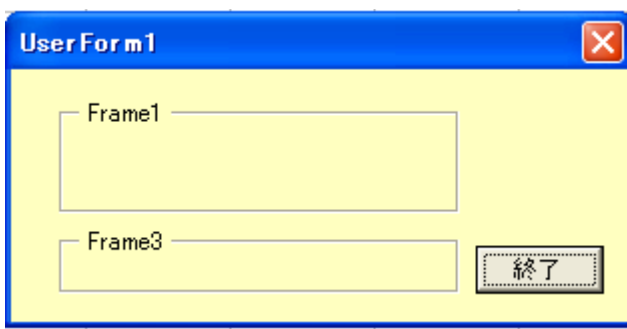
各コントロールの名称は次表（ツールボックスの位置にあわせてある。）のようなものである。

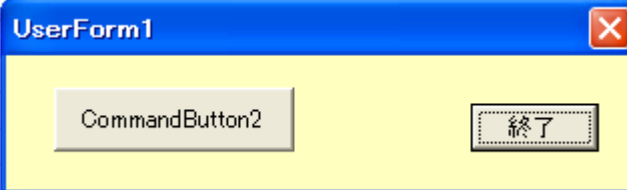
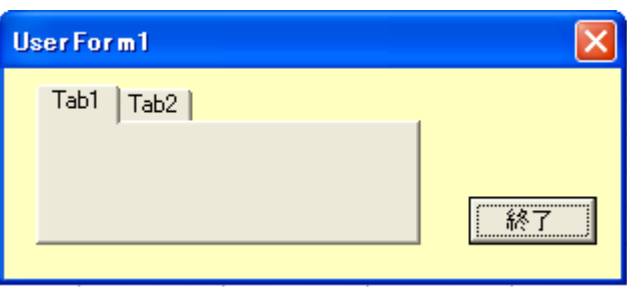
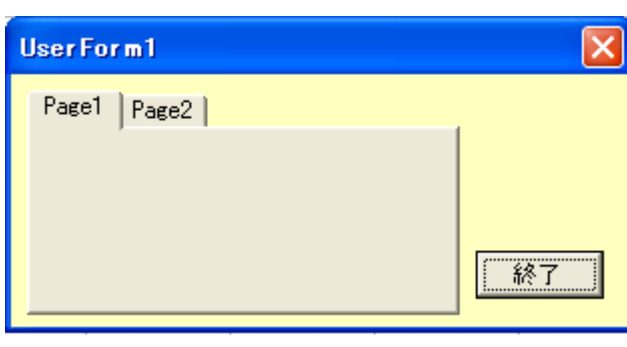
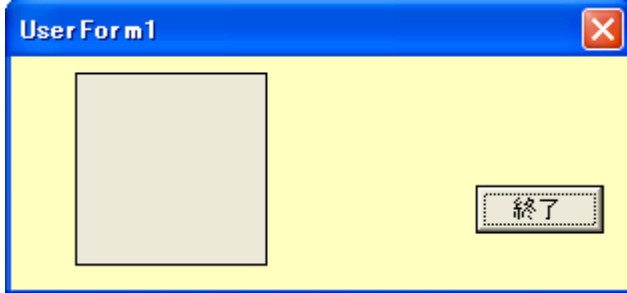
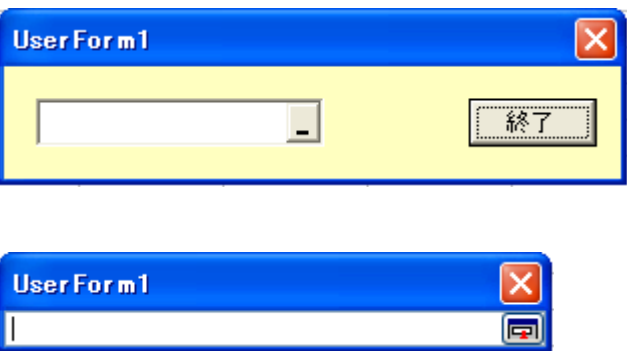
 オブジェクトの選択	 ラベル	 テキストボックス	 コンボボックス	 リストボックス
 チェックボックス	 オプションボタン	 トグルボタン	 フレーム	 コマンドボタン
 タブストリップ	 マルチページ	 スクロールバー	 スピンボタン	 イメージ
 RefEdit				

その他にもコントロールがあり、デフォルトコントロールが表示されているツールボックス上で右クリックすると多数のコントロールが表示されるが、ここでは説明を省略する。

それぞれのコントロールの用途は次のようになる。

1. ラベル 右図の「氏名」「住所」のようにユーザーフォーム上に文字（文章）を表示するための枠を作成するときに使用する。	
2. テキストボックス 右図のようにデータを書き込むためのボックスを作成するときに使用する。	

3. コンボボックス 右図のように、メニュー形式で複数の項目を表示し、1つの項目を選択するためのボックスを作成するときに使用する。ボックス内に選択リストにない項目を直接入力することも可。	
4. リストボックス 右図のように、データをリスト表示し、その中の一つのデータを選択するためのボックスを作成するときに使用する。 コンボボックスでは直接入力が可能であるが、リストボックスでは直接入力はいできない。	
5. チェックボックス 右図のように複数の項目からいくつかの項目を選択するためのボタンを作成するときに使用する。このボックスをクリックすればチェックが付き、再度クリックすればチェックが外れる。複数の項目にチェックが可能。	
6. オプションボタン 右図のように複数の項目から1つの項目を選択するためのボタンを作成するときに使用する。オプションボタンをクリックすればチェックが付き、別のボタンをクリックすればチェックが外れる。複数の項目のチェックは不可。	
7. トグルボタン 右図のようなオンかオフを指定するためのボタンを作成するときに使用する。このボタンをクリックすると押し込まれたよう（オン）になり、再度クリックすると元（オフ）に戻る。	
8. フレーム 右図のようにコントロールをグループ分けするための囲み線を作成するためのものであり、ユーザーフォームを見やすくするためのものである。	

9. コマンドボタン 右図のように、クリックすることで命令 (サブマクロ) を実行するためのボタンを作成するときに使用する。	
10. タブストリップ 右図のようにタブ付きのボックスを作成するためのものであり、各ページのコントロールの構成は同一のものとなる。テキストボックスをタブストリップ上に配置すれば、タブごとに異なるデータを表示することができる。	
11. マルチページ 右図のように、タブ付きのダイアログを作成するためのものであり、各ページに異なる構成のコントロールを配置することができる。 標準は2ページであるが、ユーザーフォーム作成時にタブの部分で右クリックすることでページの追加も可能です。	
12. スクロールバー 連動したテキストボックスの値を増減させるためのスクロールバーを作成するときに使用する。	
13. スピンボタン 連動したテキストボックスの値を増減させるためのボタンを作成するときに使用する。	
14. イメージ 右図のようにユーザーフォームに画像を表示するための枠を作成するときに使用する。	
15. RefEdit 右上図のように Excel ワークシートのセル範囲を読み取るためのボックスを作成するときに使用する。このボックスの右端のボタンをクリックすると、右下図のようにユーザーフォームが最小化される。この状態で Excel ワークシートをドラッグするとドラッグしたセル範囲が入力される。さらに、右下図のボックスの右端のボタンをクリックすれば元に戻る。	

3. コントロールの貼り付け

コントロールをユーザーフォームに貼り付けるためには、図4のツールボックスのコントロールアイコンをクリックし、図3のユーザーフォーム上でクリックすればよい。

いくつかのコントロールを連続して貼り付けるときは、ツールボックスのコントロールアイコンをダブルクリックすればよい。さらに、連続貼り付けを解除するためには、ツールボックスのコントロールアイコンをクリックすればよい。

貼り付けたコントロールをクリックすれば、そのコントロールがアクティブになる。アクティブな状態で網掛け枠線にマウスポインタをあわせドラッグすれば、コントロールを移動させることができる。さらに、網掛け枠線の8カ所にある四角なポインタをドラッグすれば、コントロールのサイズを変更できる。このサイズ変更はユーザーフォーム（ポインタは3カ所のみである。）についても同様に行うことができる。

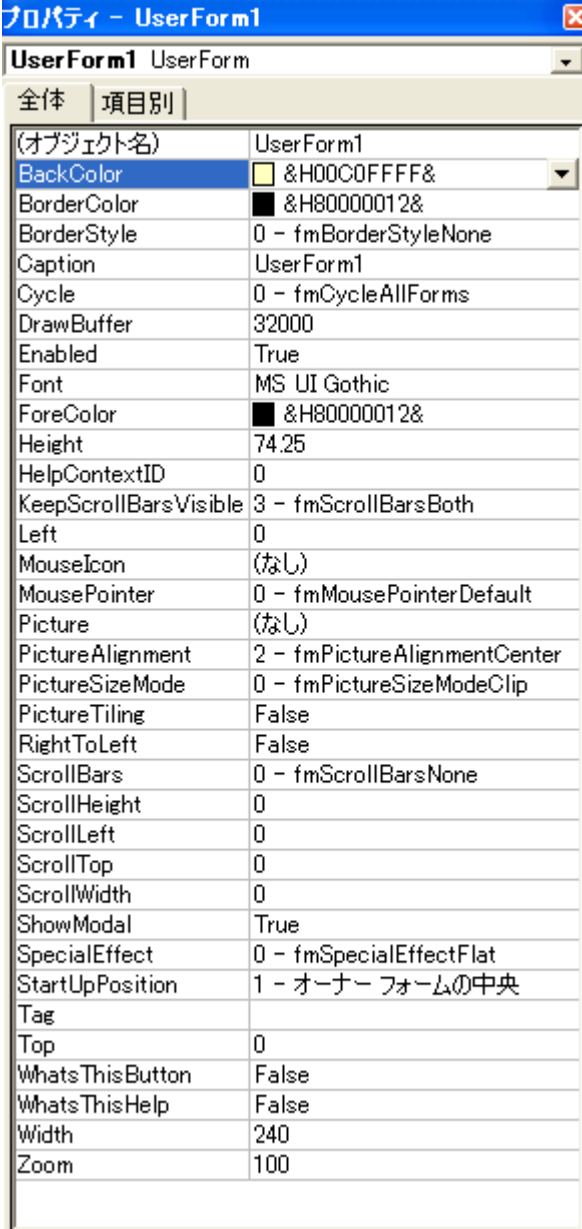
上述のようなコントロールの移動・サイズ変更はユーザーフォームに表示されているドット（6ピクセル間隔）単位で行われる。6ピクセルよりも細かい単位での移動やサイズ変更を行うためには「プロパティ」を使用しなければならない。

4. プロパティ

図5のように、プロパティには、各項目をアルファベット順に表示する「全体」と、グループ分けして表示する「項目別」があり、タブのクリックで切り替えられる。各項目は、項目名とその値からなっている。値の変更は、項目名をクリックして選択（図5では「BackColor」が選択されている。）してから、値欄を変更することで行うことができる。

この値の変更には2つのパターンがあり、直接値を入力すべき項目と、項目名をクリックしたとき値欄にボタンが表示（図5では「BackColor」を変更するためのボタンが表示されている。）され、ボタンをクリックしメニューの中から必要な値を選択する項目とがある。

以下ではよく使う項目についてのみ説明する。



(オブジェクト名)	UserForm1
BackColor	&H00C0FFFF&
BorderColor	&H80000012&
BorderStyle	0 - fmBorderStyleNone
Caption	UserForm1
Cycle	0 - fmCycleAllForms
DrawBuffer	32000
Enabled	True
Font	MS UI Gothic
ForeColor	&H80000012&
Height	74.25
HelpContextID	0
KeepScrollBarsVisible	3 - fmScrollBarsBoth
Left	0
MouseIcon	(なし)
MousePointer	0 - fmMousePointerDefault
Picture	(なし)
PictureAlignment	2 - fmPictureAlignmentCenter
PictureSizeMode	0 - fmPictureSizeModeClip
PictureTiling	False
RightToLeft	False
ScrollBars	0 - fmScrollBarsNone
ScrollHeight	0
ScrollLeft	0
ScrollTop	0
ScrollWidth	0
ShowModal	True
SpecialEffect	0 - fmSpecialEffectFlat
StartupPosition	1 - オーナー フォームの中央
Tag	
Top	0
WhatsThisButton	False
WhatsThisHelp	False
Width	240
Zoom	100

図5 ユーザーフォームのプロパティ

(1) (オブジェクト名) : 直接入力項目

「オブジェクト名」は、ユーザーフォームやコントロールを指定するための名称であり、サブマクロでこれらを指定するときに使用する。サブマクロ作成時にこの名称を入力する必要があるので、簡単な名称に変更することもできる。

(2) Caption : 直接入力項目

「Caption」は、デフォルト値はオブジェクト名になっている。

- ユーザーフォームの「Caption」は、ユーザーフォームのタイトルバーの名前
 - ラベルの「Caption」は、ラベルとして表示する文章
 - チェックボックスとオプションボックスの「Caption」は、チェックボックスの右側に表示する文章
 - トグルボタンとコマンドボタンの「Caption」は、ボタンの表面に表示する文章
 - フレームの「Caption」は、フレームの左上部に表示されるフレームの名称
 - マルチページの「Caption」は、マルチページの各ページを示すタブの名称
- その他のコントロールでは「Caption」が存在しない。

(3) Left、Top : 直接入力項目

「Left」と「Top」は、コントロールの位置を指定するもので、コントロールの左上隅の位置をユーザーフォームの左上隅からのピクセル数で指定する。

(4) Height、Width : 直接入力項目

「Height」と「Width」は、コントロールのサイズを指定するもので、コントロールの「高さ」と「幅」をピクセル数で指定する。

(5) BackColor : メニュー選択項目

「BackColor」は、背景色を指定するための項目である。値欄右側のボタンをクリックし、表示されるメニューから色を選択することになるが、「パレット」タブをクリックし、色見本の中から色を指定する。

(6) Font : メニュー選択項目

「Font」は、Caption の文字フォントを指定するためのものであり、テキストボックス、コンボボックス、リストボックス、RefEdit の場合は、ボックス内に入力ないし表示する文字のフォントを指定することになる。

(7) Value : 直接入力項目

- テキストボックス、コンボボックス及び RefEdit の「Value」は、初期にボックス内に表示する文字列

- チェックボックス及びオプションボックスの「Value」は、初期状態でのチェックの有無を意味し、デフォルト値は「False」で「True」に変更可能。「False」はチェックなし、「True」はチェックありとなる
- トグルボタンの「Value」は、ボタンが押された状態（オン）と押されていない状態（オフ）を意味し、デフォルト値が「False」であり、「True」に変更可能。「False」はオフ、「True」はオンなる
- タブストリップの「Value」は、初期に表示するタブを意味し、左から順に 0, 1, 2 となる。デフォルト値は「0」である

(8) Index : 直接入力項目

「Index」は、タブストリップに対する「Value」と同様に、マルチページにおいて初期に表示するページを意味し、左から順に 0, 1, 2 となる。デフォルト値は「0」である

(9) TextAlign : メニュー選択項目

「TextAlign」は、ラベル、テキストボックス、コンボボックス、リストボックス、チェックボックス、オプションボタン、トグルボタン、RefEdit の各コントロールにおいて、表示する文字列の横位置を指定する。「1」は左詰め、「2」は中央揃え「3」は右詰めに意味し、デフォルト値はラベル、テキストボックス、コンボボックス、リストボックス、チェックボックス、オプションボタン、RefEdit が「1」、トグルボタンが「2」である。

5. コントロールへのサブマクロの関係付け

Excel マクロは、サブマクロと呼ばれる個々の処理を行うプログラムの集合体として構成される。さらにこれらのサブマクロは 2 種類に分類され、ユーザーフォームでの処理に関連したサブマクロと一般のサブマクロに分けられる。一般のサブマクロは、「標準モジュール」として「Module1 (マクロ名)」等にまとめられ Excel ファイルに添付される。一方、ユーザーフォームでの処理に関連したサブマクロは、「フォーム」として「UserForm1 (ユーザーフォーム名)」等にまとめられて Excel ファイルに添付される。

ユーザーフォームは、各種のコントロールを組み込んだダイアログボックスと、ユーザーフォームでの処理に関連したサブマクロの集合体とで構成される。

ユーザーフォームはそれだけで実行できるものではなく、標準モジュールから呼び出して使用するものである。標準モジュールとユーザーフォームとは別のものとして保存されるので、異なる標準モジュールで 1 つのユーザーフォームを利用することもできる。

(1) ダブルクリックによる方法

ユーザーフォーム上に貼り付けたコントロールに、サブマクロを関係付けるためには、コントロールをダブルクリックし、作成される空のサブマクロに必要な事項を記述すればよい。なお、ダブルクリ

ックする前に、そのコントロールをクリックしアクティブにしておくのがよいようです。

プロジェクト名が“CommandButton1”であるコマンドボタンをダブルクリックすると、次の空のサブマクロが作成される。

```
Private Sub CommandButton1_Click()  
  
End Sub
```

このサブマクロの意味は「コマンドボタン“CommandButton1”がクリックされたとき、記述されたサブマクロを実行せよ。」というものである。

一般に、コマンドボタンとタブストリップ以外のコントロールに直接サブマクロを関係付けることはないように思われる。もちろん、コマンドボタン以外のラベル、リストボックス、チェックボックス、オプションボタン、トグルボタン、フレーム、イメージをクリックしたときに何らかの命令（サブマクロ）を実行させることも可能である。

（２）コードの表示による方法

コントロールをダブルクリックせずに空のサブマクロを作成する方法は、

①作成中のユーザーフォーム上で右クリックし、開くメニューの中の「コードの表示」をクリックしないし

②プロジェクトエクスプローラの一番左側のボタン「コードの表示」をクリック

することで、コードを表示させることも可能である。この場合は、図6のようにコードウィンドウを表示するだけで、ダブルクリックの場合と異なり空のサブマクロが作成されない。

コードは、ユーザーフォームに関連して作成されたサブマクロを表示するためのものであり、1つのユーザーフォームに対して1つしかなく順次追加される。

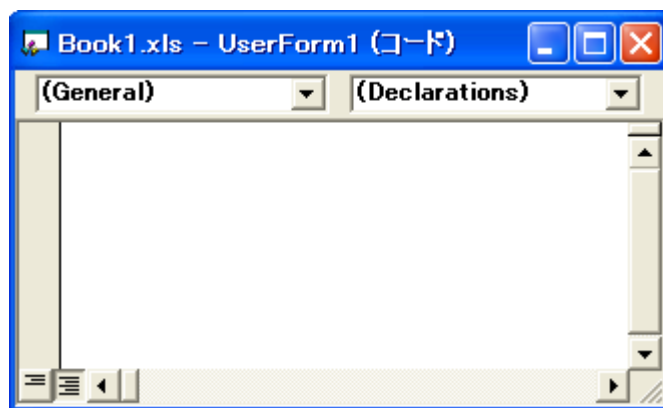


図6 コードウィンドウ

空のサブマクロを作成するためには、コードウィンドウの一番上にある2つのボックスを使用することができる。まず、左側のボックス（図6では「(General)」となっている。）右端のボタンをクリックし、リスト表示される「オブジェクトリスト」から目的のコントロールを選択する。次に、右側のボックス（図6では「(Declarations)」となっている。）右端のボタンをクリックし、リスト表示される「プロシジャリスト」から目的のプロシジャを選択すれば空のサブマクロが作成される。

6. コントロールの順番

コントロールに順番をつけておくと便利であり、一般的には左上にあるコントロールから順に順番がつけられる。デフォルトの順番は、コントロールの作成順であるため、移動を繰り返す内に滅茶苦茶な順番になってしまうことが多い。

ユーザーフォーム作成後に、コントロールの順番を整理するために

- ① VBA ウィンドウのメニューの「表示」をクリック

ないし

- ② 作成中のユーザーフォーム上で右クリック

し、開かれるメニューの中の「タブオーダー」をクリックする。開かれるダイアログボックスに、その段階での順番でオブジェクト名が表示される。移動したいオブジェクト名をクリックし、「上に移動」ないし「下に移動」のボタンをクリックすることで順番を整理し、「OK」ボタンをクリックすればよい。

実際にユーザーフォームを使用する際、「Tab」キーを押せば、指定された順にコントロールがアクティブになる。

7. ユーザーフォームの表示と終了

Excel で、作成してあるユーザーフォームは、サブマクロによって“開く”ことになり、「オブジェクト名」が「UserForm1」であるユーザーフォームを開くためのサブマクロは次のとおりである。

```
Sub Macro1()  
,  
' Keyboard Shortcut: Ctrl+e  
,  
    UserForm1.Show  
End Sub
```

オブジェクト名が「UserForm1」以外のユーザーフォーム（例としてオブジェクト名を「abc」とする。）を開くときは、上記サブマクロで「UserForm1.Show」の部分で「abc.Show」とすればよい。

ユーザーフォームを“閉じる”ためには、Caption を“終了”等としたコマンドボタンを作成し、そのコマンドボタンに次のサブマクロを関係付ければよい。

```
Private Sub CommandButton2_Click()  
    Unload Me  
End Sub
```

ただし、ここではオブジェクト名が「CommandButton2」であるコマンドボタンが“終了”ボタンであるとしている。

8. UserForm_Initialize サブマクロの作成

UserForm_Initialize は、当該ユーザーフォームが Excel から呼び出されたとき、最初に自動的に実行されるサブマクロであり、プロパティでは指定できない色々な設定を記述するためのものである。

「5. コントロールへのサブマクロの関係付け」の後半で説明した「コードの表示」で、「オブジェクトリスト」から「UserForm」を選択し、「プロシジャリスト」から「Initialize」を選択する。すると次の空のサブマクロ（UserForm_Initialize）が作成される。

```
Private Sub UserForm_Initialize()  
  
End Sub
```

9. タブストリップのタブの名称を変更する

マルチページの各ページの名称（デフォルト名は Page1、Page2 等）はプロパティの Caption で変更できるが、タブストリップのタブの名称（デフォルト名は Tab1、Tab2 等）はサブマクロを使って定義しなければ変更できない。このような定義を行うためのサブマクロが UserForm_Initialize である。

このサブマクロで

オブジェクト名.Tabs(0).Caption = "タブの名称"

を記述すればタブ名称を変更することができる。なお、Tabs(0)は最も左側のタブを意味し、2つめ、3つ目のタブは Tabs(1)、Tabs(2)となる。

例えば、オブジェクト名が TabStrip1 であり、4つのタブがあるとき、タブ名称を左から順に春、夏、秋、冬にするためには次のようにすればよい。

```
Private Sub UserForm_Initialize()  
    TabStrip1.Tabs(0).Caption = "春"  
    TabStrip1.Tabs(1).Caption = "夏"  
    TabStrip1.Tabs(2).Caption = "秋"  
    TabStrip1.Tabs(3).Caption = "冬"  
End Sub
```

10. コンボボックスないしリストボックスに項目リストを組み込む

コンボボックスないしリストボックスに項目リストを組み込むためには、UserForm_Initialize サブマクロに必要事項を書き込まなければならず、UserForm_Initialize サブマクロに追加すべき項目を

オブジェクト名.AddItem “項目”

として記述すれば、項目リストを組み込むことができる。

オブジェクト名が「ComboBox1」であるコンボボックスに次にリストを組み込み

日本
アメリカ
中国

オブジェクト名が「ListBox1」であるリストボックスに次にリストを組み込むこととする。

軽自動車
小型乗用車 排気量 2000cc 以下
普通乗用車 排気量 2000cc 以下
普通乗用車 排気量 2000cc 超

このためのサブマクロは次のようになる。

```
Private Sub UserForm_Initialize()  
    ComboBox1.AddItem "日本"  
    ComboBox1.AddItem "アメリカ"  
    ComboBox1.AddItem "中国"  
    ListBox1.AddItem "軽自動車"  
    ListBox1.AddItem "小型乗用車 排気量 2000cc 以下"  
    ListBox1.AddItem "普通乗用車 排気量 2000cc 以下"  
    ListBox1.AddItem "普通乗用車 排気量 2000cc 超"  
End Sub
```

ここで、同一のオブジェクトで、表示される項目の順番は UserForm_Initialize に書き込まれた順番になる

1 1. リストボックスに複数列のデータを表示する

リストボックスには複数列のデータを表示することができる。当該リストボックスのプロパティにおいて

ColumnCount を 3

ColumnHeads を True

ColumnWidths を 40;40;70

Widths を 150

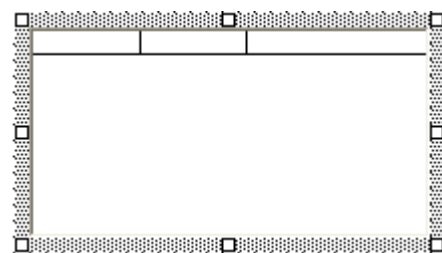


図 7

とすれば、図 7 のようになり、複数列を表示することが可能となる。

「ColumnCount」で列数、「ColumnWidths」で各列の幅を指定することになる。なお、最後の列の幅はリストボックス全体の幅に依存し、全体の幅が優先されるので指定しなくてもかまわない。すなわち、上の例で「ColumnWidths」を「40;40」としても同じ結果になる。

「ColumnHeads」は、見出し欄を使用するかどうかを指定するためのもので、「False」にすれば見出し用の枠が表示されない。

リストボックスに表示する項目を指定する方法はいくつかあるが、一般的な方法は Excel シートから読み込む方法であろう。

Excel の「Sheet2」に図 8 のようにデータが入力されているときにプロパティで

RowSource を Sheet2!A2:C3

とすれば、図 9 のようになる。

ここで注意すべきことは、見出しが入力されているセル（図 8 の例では A1:C1）を指定せず、データの領域のみを指定することである。もし、この例で Sheet2!A1:C3 とすると、見出しもデータとして扱われてしまい、見出し欄には

「列 A」、「列 B」、「列 C」と表示される。これは、指定行が Excel シートの最上位行であったため、さらにその上の列名が見出しとして表示されたわけで、最上位行でないときはすぐ上の行のセル内容が見出しとして表示される。

	A	B	C	D
1	A	B	C	
2	1	2	3	
3	4	5	6	
4				

図 8 Excel シート

A	B	C
1	2	3
4	5	6

図 9

1 2. コントロールに入力されたデータの読み込み

入力されたデータとしては

- ① テキストボックス、コンボボックス、RefEdit のボックス内に入力された文字列
- ② リストボックスは選択された項目（文字列）
- ③ チェックボックス、オプションボックス、トグルボタンは、チェックあるいは押し込まれているかどうかのデータで、チェックボックスとオプションボックスは、チェックされているとき「True」、チェックされていないとき「False」、トグルボタンは、オンの（押し込まれている）とき「True」、オフの（押し込まれていない）とき「False」の値となる。
- ④ タブストリップは、選択されているタブ（番号）

等があり、①と②の文字列は、

オブジェクト名.Text

で読み込むことができる。また、③と④のデータは

オブジェクト名.Value

で読み込むことができる。

サブマクロにおいてこれらのデータを読み込みいろいろな処理に使用することができる。