

実習 9 使用ファイル：「Sample2」

「f_顧客データ変更」フォームを下記の手順で完成します。

【1】 「検索」ボタンをクリックすると、対応する「会員 ID」の情報を表示します。

Step 1 ▶ Do…Loop ステートメントで「顧客一覧」シートの A 列の値と会員 ID テキストボックスの値が一致するかを調べ、一致したらループを抜けます。

Step 2 ▶ シートの会員 ID とフォームの会員 ID が一致しない（ループを抜けた時点の行の会員 ID とフォームに表示されている会員 ID が異なる）場合は「会員 ID が存在しません。」と表示し、会員 ID へフォーカスを移動後、Sub プロシージャを終了します。

Step 3 ▶ 「顧客一覧」シートの内容をフォームに表示します。

【2】 データを検索、変更後、「更新」ボタンをクリックすると、情報を上書きします。

Step 1 ▶ 【1】の Step1 と Step2 の処理を記述します。

Step 2 ▶ 空白や未設定のコントロールがある場合は「〇〇は必須入力です。」と表示し、空白のコントロールにフォーカスを移動して、終了します。

Step 3 ▶ 同じ会員 ID の各コントロールの値をセルに代入し、更新日には本日の日付を入れます。最後に「更新完了」と表示します。

【3】 データを検索し、「削除」ボタンをクリックすると、該当する行を削除します。

Step 1 ▶ 【1】の Step1 と Step2 の処理を記述します。

Step 2 ▶ 「削除してもよろしいですか？」と表示し、一致した会員 ID の行を削除します。最後に「削除完了」と表示します。

【4】 「閉じる」ボタンをクリックすると、フォームを閉じます。

【5】 「f_メニュー」フォームを開きます。「顧客データ変更」ボタンをクリックし、会員 ID「4」の会員を削除します。また、会員 ID「1」の住所を「新橋5丁目1-2」に変更します。

【6】 「f_メニュー」フォームを開き、「利用状況登録」ボタンをクリックし、会員 ID「1」で検索を実行して、検索結果を確認します。

！ HINT1-Step1

変数「i」の初期値は「2」とします。一致した時点で「顧客一覧」シートの会員 ID = フォームの会員 ID です。

```
Do While .Cells(i, 1) <> ""
    If .Cells(i, 1) = Me.会員ID. _
        Text Then
        Exit Do
    End If
    i = i + 1
Loop
```

！ HINT1-Step1

テキストボックスの値は Text プロパティで取得します。

！ HINT1-Step3

実習8とは逆の考え方でシートの内容をフォームのコントロールに表示します。

```
If .Cells(i, 5) = "男" Then
    Me.opt男.Value = True
Else
    ...
```

！ HINT2-Step2

データの検索後にコントロールの値が削除されていないかを確認します。すべてのコントロールに IF ステートメントを記述します。

！ HINT3-Step2

削除するのは変数 (i) に代入された数値の行です。

◎行の削除

```
Rows(行番号).Delete
```

MEMO**▶「利用状況登録」ボタン**

会員 ID を入力すると、対応する個人情報と、過去の履歴が表示されます。また新規利用状況を入力できます。

▶ RowSource プロパティでリスト項目を登録

表示するデータがワークシートに入力されている場合は、RowSource プロパティにセル範囲を指定して、リスト項目を登録することもできます。

下記のいずれかの方法が利用できます。

課題

使用ファイル：「Sample2」 使用シート：「部署一覧」

「UserForm1」ユーザーフォームの右上と右下にリストボックスを作成します。上のリストボックスにはセル範囲「A2:A8」の値を表示し、下のリストボックスにはセル範囲「B2:B4」の値が表示されるようにします。

▶ プロパティリストに指定して、登録します。

右上のリストボックスを選択して、[RowSource] プロパティに表示する項目のセル範囲を入力します。

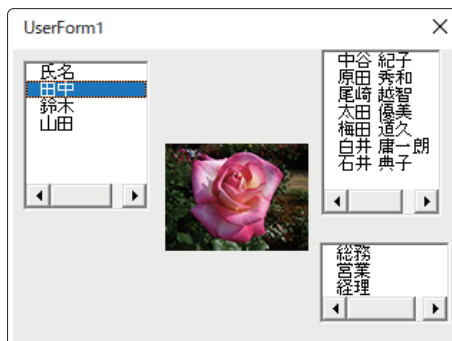
MultiSelect	0 - fmMultiSelectSingle
RowSource	部署一覧!a2:a8
SpecialEffect	2 - fmSpecialEffectSunl

▶ コードに RowSource プロパティを記述して、登録します。

構文：フォーム名.コントロール名.RowSource = "セル範囲"

ユーザーフォームの Initialize イベントに下記のコードを追加で記述します。

```
Private Sub UserForm_Initialize()
    Me.ListBox3.RowSource = "部署一覧!B2:B4"
End Sub
```



実習 10 使用ファイル：「Sample2」 使用シート：「部署一覧」

[1] 「f_list1」フォームの「表示」ボタンをクリックした際のイベントを作成します。

指示

コードに RowSource プロパティを記述して、リストボックスに「部署一覧」シートの「A2:A8」の値が表示されるようにします。一番上の候補が初期値になるように指定します。

[2] 「リストの追加」ボタンをクリックして、インプットボックスに「谷村」と入力し、「書き込みできません」というエラーが出ることを確認します。

[3] 「表示」ボタンのイベントを AddItem メソッドに変更します。
For...Next ステートメントの繰り返し処理で、A 列の 2 行目から最終行までを順番に表示します。

[4] 「リストの追加」ボタンをクリックして「小山礼」を追加します。

MEMO

▶ RowSource プロパティの特徴

- セル範囲はダブルクォーテーションで囲み、文字列として指定します。
- RowSource プロパティを設定したリストボックスへは、AddItem メソッドで新しい項目を追加できません。
次のように AddItem メソッドを実行するとエラーになります。

```
Me.ListBox1.RowSource = "Sheet1!A1:A3"
Me.ListBox1.AddItem "谷部"
```

- RemoveItem メソッドでの項目の削除はできません。
- RowSource プロパティにセル範囲を指定すると、セルとリストボックスが「リンクされた状態」になります。ユーザーフォームを表示している状態でも、セルの値が変化すれば、リストボックスのデータも同じように変化します。

MEMO

同じリストボックスに RowSource プロパティと AddItem メソッドは追加できず、エラーとなります。リストボックスの値を変更する場合は、どちらかに統一します。

! HINT3

AddItem はメソッドですので「=」は入力しません。

```
Me.l_data.AddItem Cells(i, 1)
```