

コンボボックス自由自在

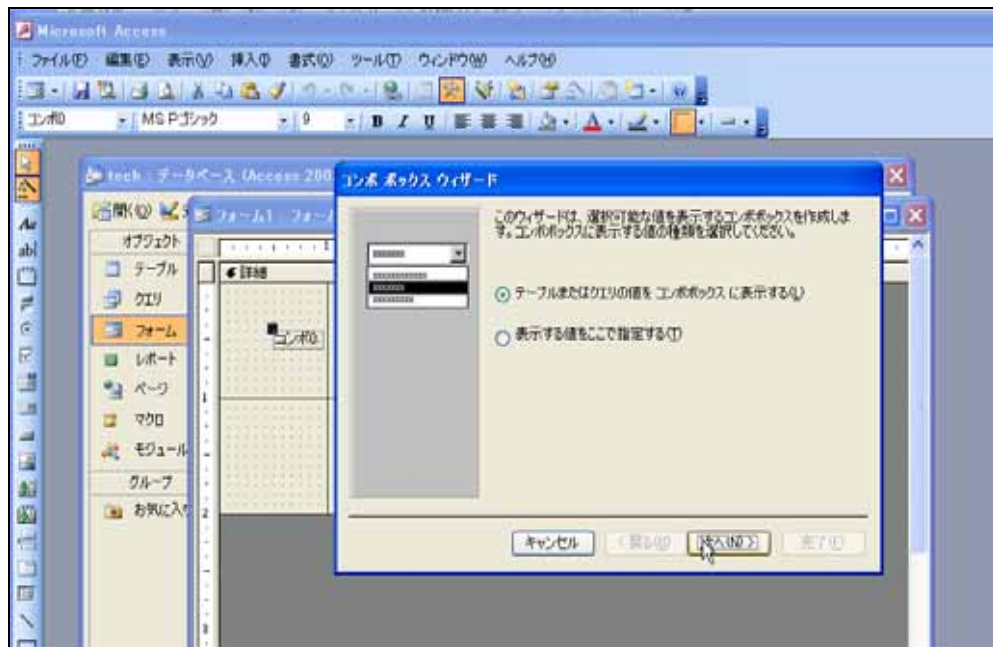
平成 19 年 1 月 13 日
有限会社ビット
浦 秀樹

Access のコンボボックスは実に素晴らしい機能を持っている。コンボボックスの活用こそがアプリケーションの操作性を左右するといっても過言ではない。

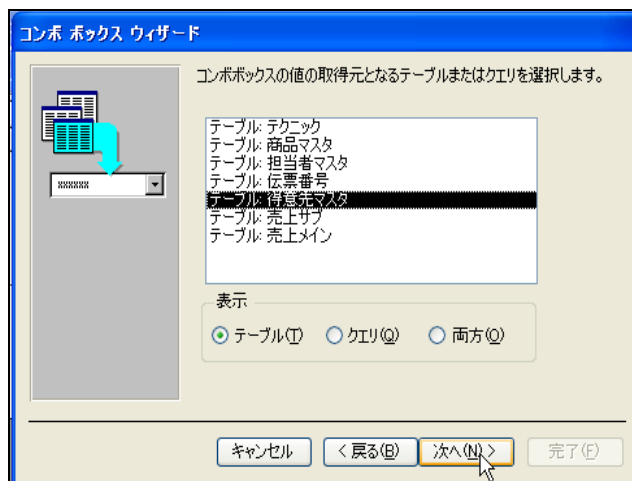
以下の解説は無料ダウンロードコーナーにアップされている、kantan uriage.mdb をもとに行われている。

まずはコンボボックスをウィザードで作ってみよう。
コンボボックス作成の手順は次の通り。

1. フォームのデザインビューで、左側の[コンボボックス]アイコンをクリックすると、ウィザード画面が表示される。ここで[テーブルまたはクエリの値を・・・]を選択して[次へ]ボタンをクリックする。



2. ここでは[得意先マスタ]を指定して[次へ]ボタンをクリックする。



3. [選択可能なフィールド]欄からすべてのフィールドを選択して、[次へ]ボタンをクリックする。

コンボボックスウィザード

コンボボックスの値の取得元となるフィールドを選択します。選択したフィールドは、コンボボックスの列として表示されます。

選択可能なフィールド:

選択したフィールド:

得意先コード
得意先名
よみ

キャンセル < 戻る(B) 次へ(N) > 完了(F)

4. 並べ替えフィールドに[よみ]を指定して、[次へ]ボタンをクリックする。

コンボボックスウィザード

表示する値を並べ替える方法を指定してください。

並べ替えを行うフィールドを4つまで選択できます。それぞれのフィールドごとに昇順または降順を指定できます。

1 よみ 昇順
2 昇順
3 昇順
4 昇順

キャンセル < 戻る(B) 次へ(N) > 完了(F)

5. ここでは何もせずに[次へ]ボタンをクリックする。

コンボボックスウィザード

コンボボックスに表示する列の幅を指定してください。

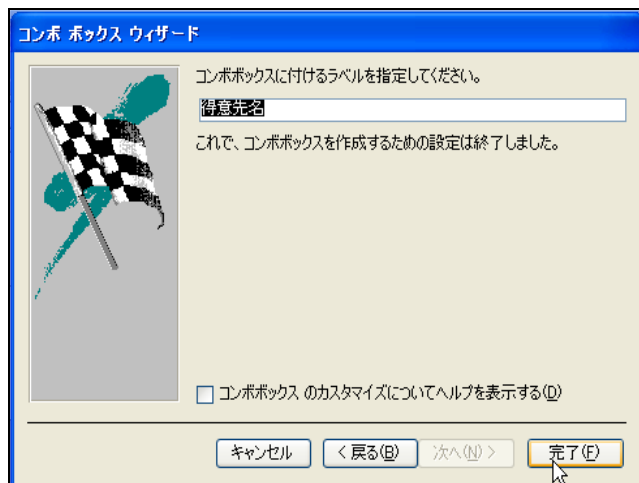
列幅を調整するには、列の右端をドラッグします。また、右端をダブルクリックすると、入力した値の長さに合わせて列幅が自動的に調整されます。

☒ キー列を表示しない(推奨)(H)

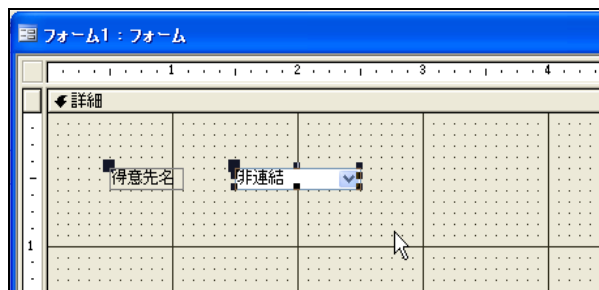
	得意先名	よみ
▶	伊藤花子	いとうはなこ
	上島二郎	うえしまじろう
	浦秀樹	うらひでき
	鈴木一郎	すずきいちろう
	長田真一	ながたしんいち
	山田太郎	やまだたろう
	吉田幸子	よしたゆきこ

キャンセル < 戻る(B) 次へ(N) > 完了(F)

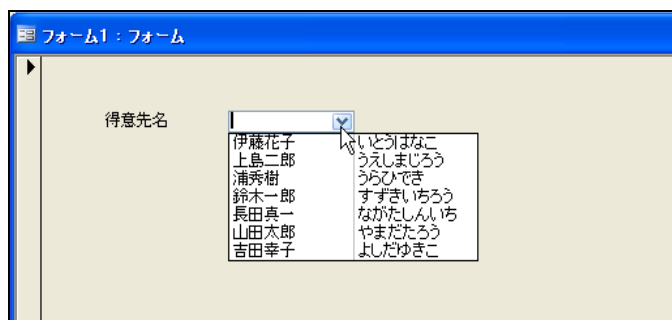
6. コンボボックスの左横に表示されるラベルの文字列を指定して[完了]ボタンをクリックする。



7. コンボボックスが登録されたのを確認する。



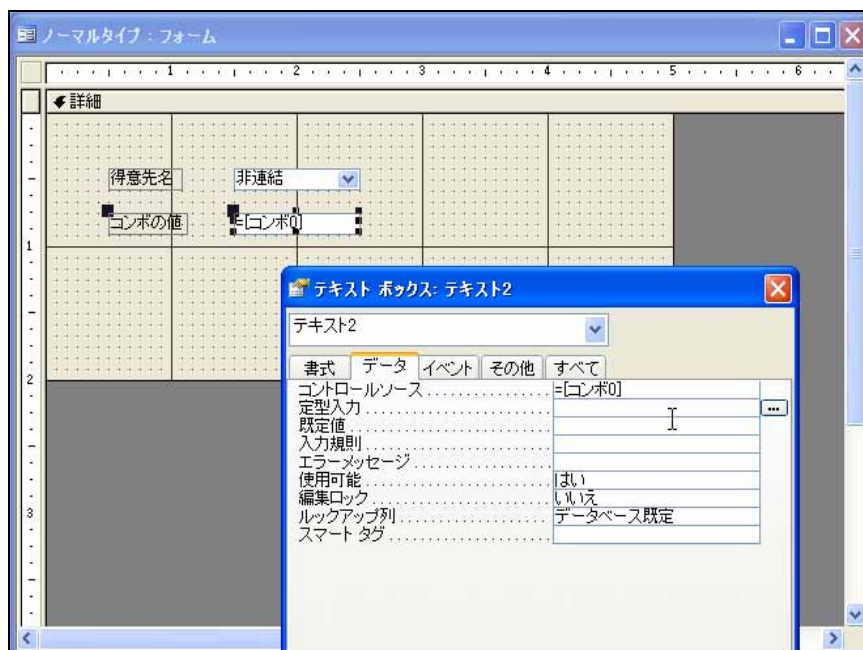
8. フォームを表示して動作を確認する。



9. 先に作成した出来上がったコンボボックスを操作すると得意先名が表示されるが、実はコンボボックスの値は得意先コードとなっている。このことを検証するために、テキストボックスを一つ用意して、次のような式を設定する。

式：=[コンボ 0]

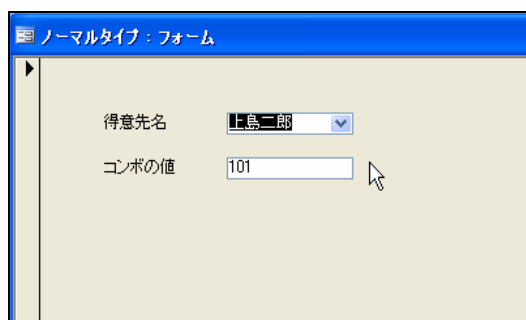
コンボの名前が[コンボ 0]という前提での話。



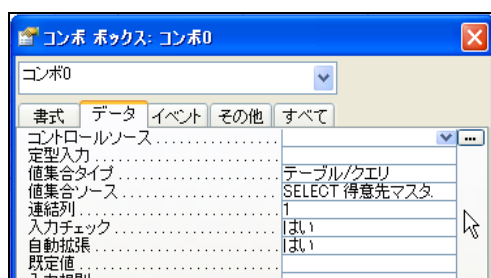
どのフィールドの値をとるかは、コンボボックスのプロパティ、[連結列]で決まる。

通常は連結列が1となっていて、クエリ左端のフィールドの値となる。

10. フォームを開いて動作を確認してみると、次のようにテキストボックスには数値が表示される。このことから上のコンボボックスの値は[得意先名]ではなく[得意先コード]であるということになる。このように、コンボボックスでは見かけ上の値と実際の値とは別に設定することができる。

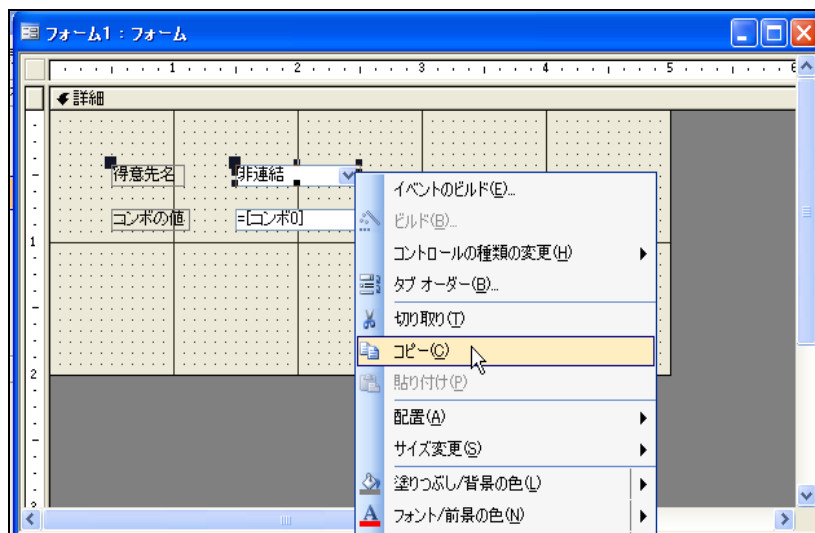


どの列の値をとるかは、コンボボックス、プロパティの[連結列]で決まる。

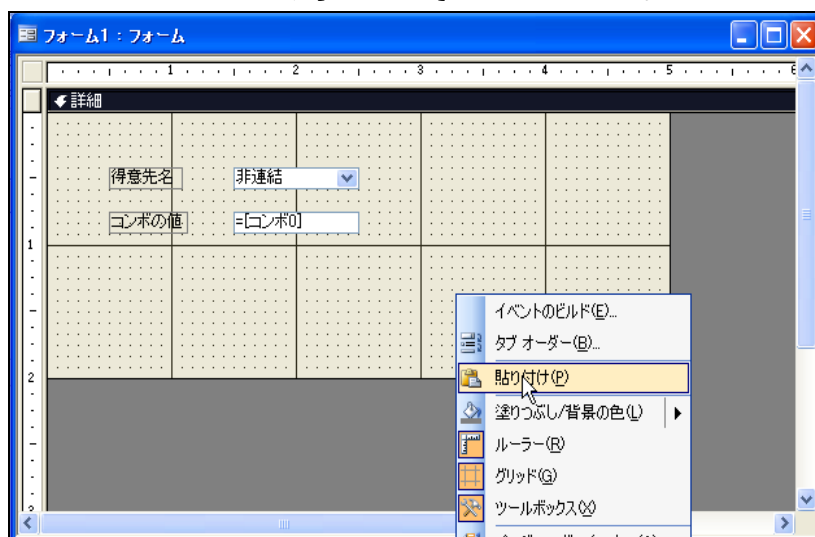


11. 先のコンボボックスを複写してみよう。

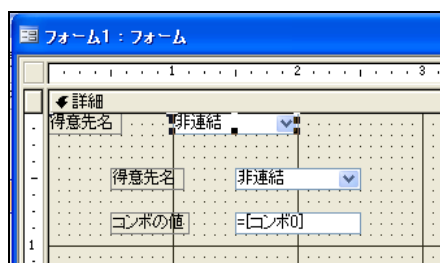
複写するにはマウスでコンボボックスを選択して、マウスの右ボタンをクリックしてショートカットメニューを表示し、[コピー]をクリックする。



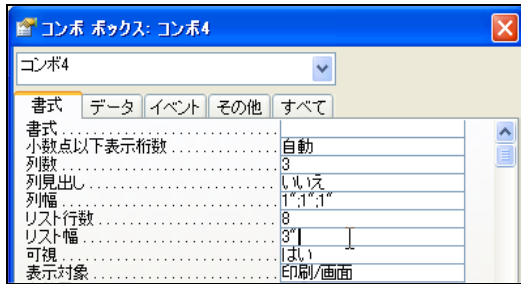
12. フォームの[詳細]セクションをクリックしてから、マウスの右ボタンをクリックしてショートカットメニューを表示させ、[貼り付け]をクリックする。



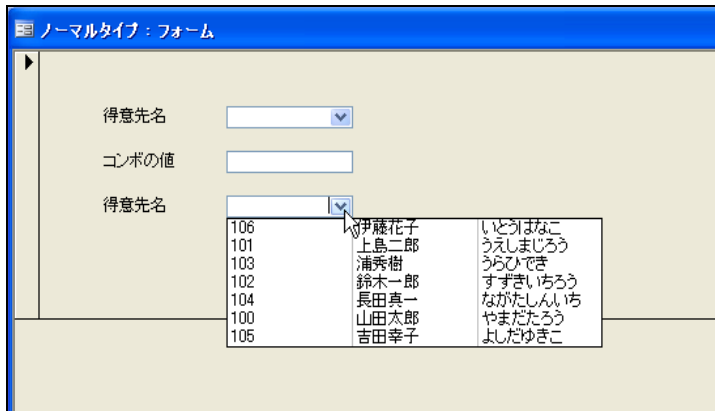
13. コンボボックスが複写された。



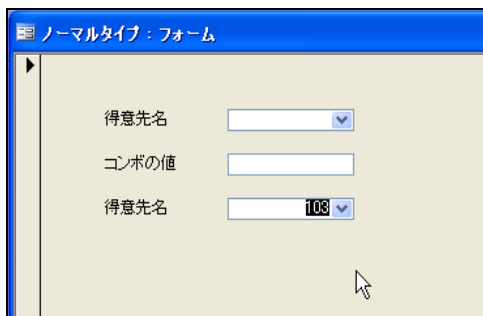
14. 複写されたコンボボックスのプロパティを開き、列幅とリスト幅を変更してみよう。
列幅：1",1",1"
リスト幅：3"



15. フォームを表示して動作を確認してみる。



16. コンボボックス内のいずれかのレコードを選択すると、今度は得意先コードが表示される。



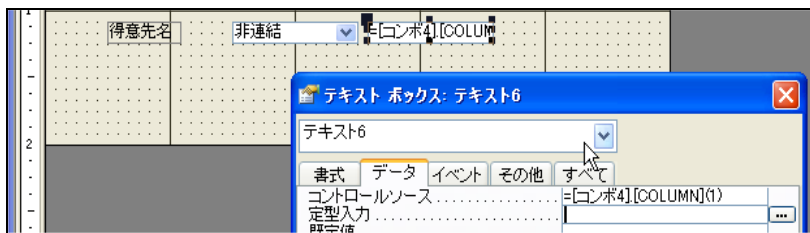
17. 次はコンボボックスの右横に得意先名を表示するテキストボックスを置きたい。
そこで、デザインビューに戻してテキストボックスをコンボボックスの右横に置く。



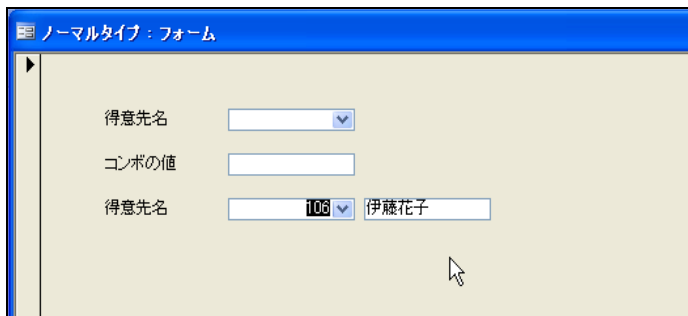
18. テキストボックスの[コントロールソース]に次の式を設定する。

式：=[コンボ 4].[COLUMN].(1)

エラーが表示されたり動作しない場合はコンボボックスの名前を要確認。

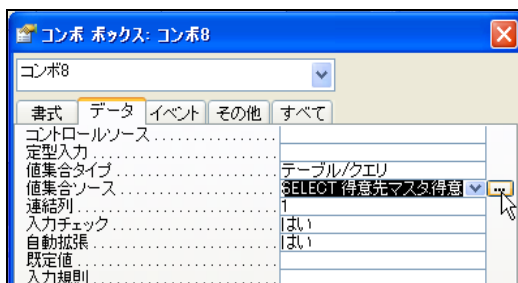


19. フォームを表示して動作を確認する。

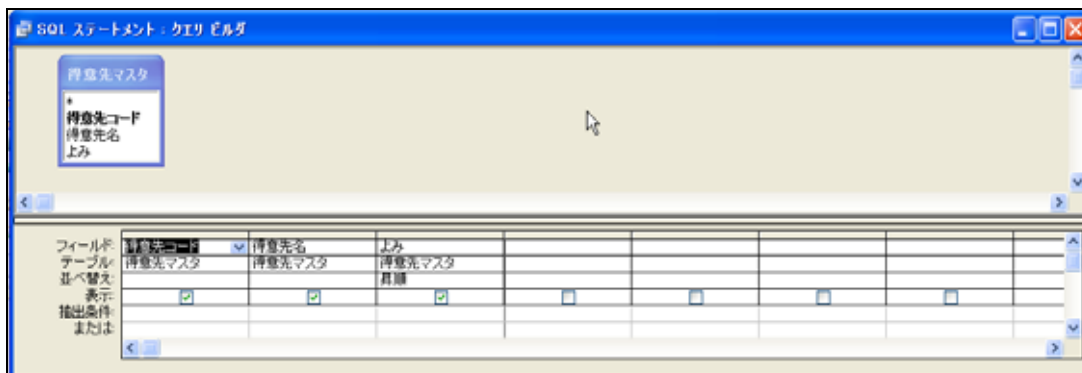


20. 2 番目に作成したコンボボックスをコピーして、今度はコンボボックス内のデータの並び順を変えてみよう。コピーの手順は前述の通り。

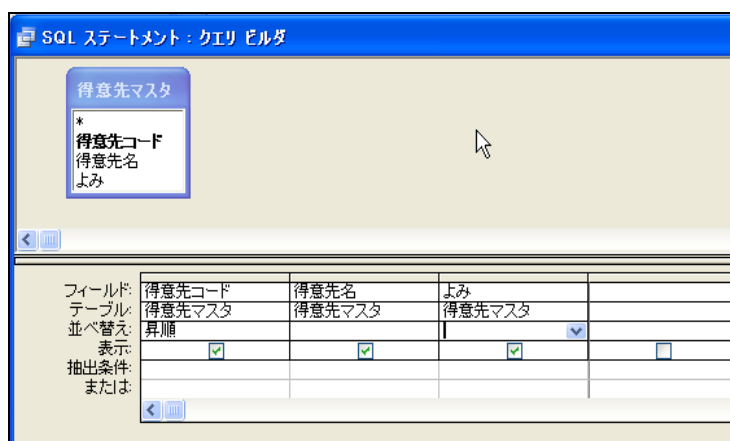
21. コピーされたコンボボックスをマウスで選択して、プロパティシートを表示し、[値集合ソース]の右端のアイコンをクリックする。



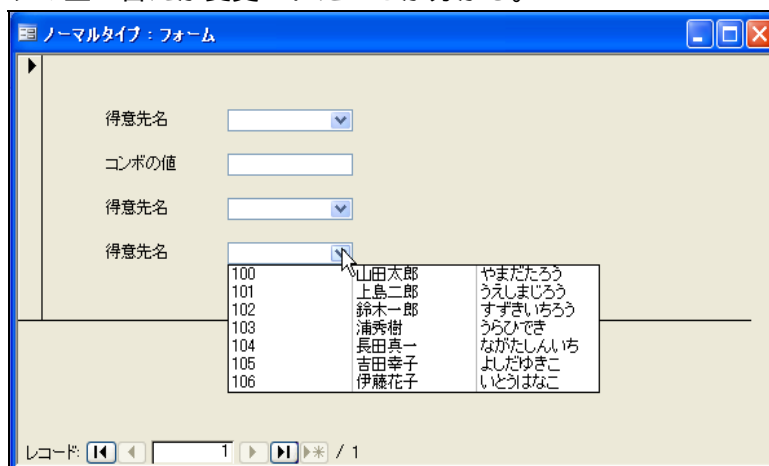
22. クエリビルダが表示される。



23. [よみ]の並べ替え設定を削除して、[得意先コード]の並べ替えを『昇順』に設定する。
設定が終了したらクエリビルダを閉じる。



24. フォームを表示して動作を確認する。
データの並べ替えが変更されたことが分かる。



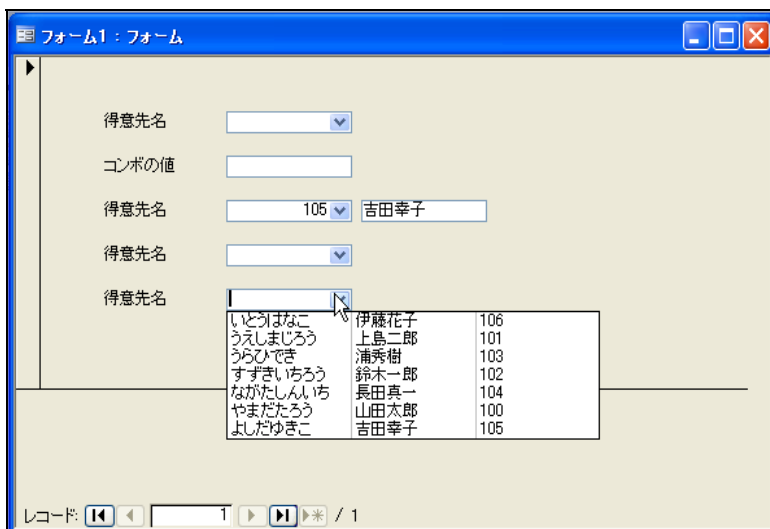
25. 今度はさらにコンボボックスの動作をより深く理解していこう。
3 番目のコンボボックスをコピーして 4 個目を作成する。



26. コンボボックスのプロパティを開き、[値集合ソース]のクエリを開く。
次に、[得意先コード]と[よみ]の位置を図のように変更する。
並べ替えの設定も忘れずに行うこと。
マウスのドラッグ&ドロップ操作で行うと良い。



27. フォームを表示し、コンボボックスを開いてフィールドの位置を確認する。



28. ここで、『う』と一文字入力（文字を確定する必要あり）すると、コンボボックスの選択レコードは該当する場所に即座に移動することがわかる。これなどは膨大なデータから該当するデータを探す際に非常に便利な機能だ。

得意先名	グループコード	得意先コード
いとうはなこ	伊藤花子	106
うえしまじろう	上島二郎	101
うらひでき	浦秀樹	103
すずきいちろう	鈴木一郎	102
ながたしんいち	長田真一	104
やまだたろう	山田太郎	100
よしだゆきこ	吉田幸子	105

29. 次に、コンボボックス内のデータ絞込みを行ってみよう。
先の[得意先マスタ]テーブルのデザイン画面を開き、[グループコード]フィールドを追加する。
データの型は長整数型

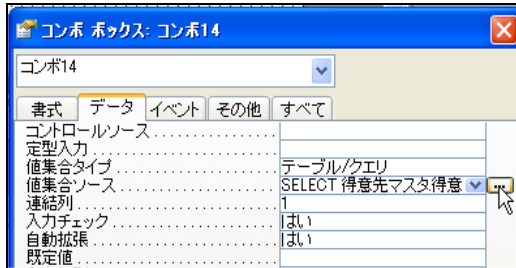
30. テーブルを開いて図のように任意にグループコードを入力しておく。

31. フォームのデザインビューを開いてテキストボックスを配置し、ラベルを適当に変更する。
このときのテキストボックス名は[テキスト 12]とする。

32. 3 番目に作成したコンボボックスをコピーしてテキストボックスの右横に貼り付ける。



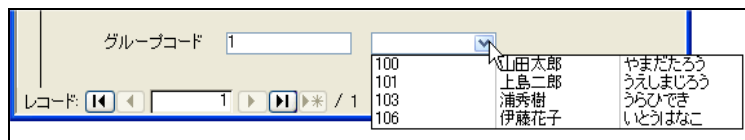
33. コピーしたコンボボックスのプロパティを開いて、[値集合ソース]右端のアイコンをクリックしてクエリビルダを表示する。



34. フィールド欄に先ほど追加した[グループコード]を追加し、[抽出条件]に次の式を設定する。
式: [FORMS]![フォーム 1]![テキスト 12]
これは、[フォーム 1]というフォームの[テキスト 12]というオブジェクトの値を参照する、という意味になる。



35. フォームを表示して、動作を確認する。テキストボックスに『1』と入力して隣のコンボボックスを開くと次のようになる。該当するグループで絞り込まれているようだ。



36. 動作をさらに確認すべく、テキストボックスに『2』という値を入れて再度コンボボックスを開いてみると、先ほどと同じ結果となることがわかる。どうやら絞込みがうまく行われていないようだ。

レコード	氏名	住所
100	山田太郎	やまだたろう
101	上島二郎	うえしまじろう
103	浦秀樹	うらひでき
106	伊藤花子	いとうはなこ

37. そこで、キーボードのファンクションキー[F 9]を押して再度コンボボックスを開いてみると、次のようになった。
[F 9]はフォーム上のすべてのオブジェクトでリクエリ(データを取り込みなおす)を強制的に行うのだ。

レコード	氏名	住所
102	鈴木一郎	すずきいちろう
104	長田真一	ながたしんいち
105	吉田幸子	よしだゆきこ

38. いちいちファンクションキーを操作するのは面倒である。
そこで、これらを自動化するために簡単なプロシージャ(プログラム)を作成する。
テキストボックスの[更新後処理]イベントを[イベントプロシージャ]としてから右横のアイコンをクリックする。
こうすることで、テキストボックスの文字列が更新された直後(厳密に言うと文字列が確定した直後)に右のコンボボックスが自動的にリクエリされるということになる。

39. イベントを次のように入力してフォームを閉じる。

```

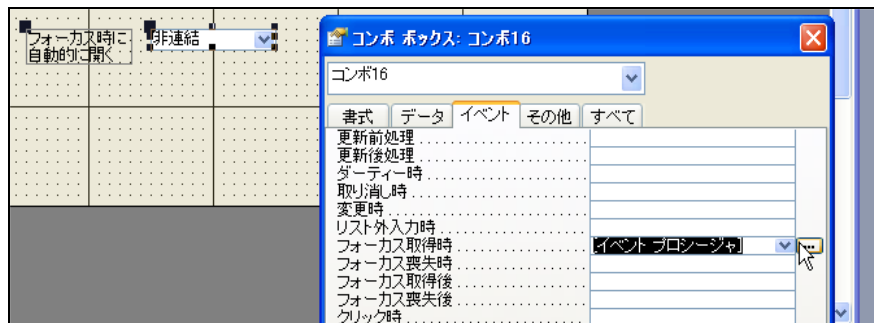
Option Compare Database

Private Sub テキスト12_AfterUpdate()
    Me.コンボ14.Requery
End Sub
    
```

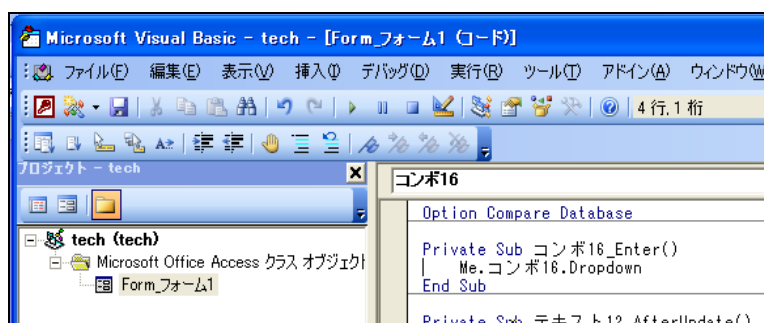
40. フォームを表示して動作を確認する。
期待したとおり、コンボボックスが絞込みを行っていることがわかる。
●コンボボックスを自動的に開くには

これまでのコンボボックスは、わざわざコンボボックスのプルダウンアイコンをクリックして表示していたが、フォーカスが来たとき（カーソルが来たとき）に自動的にコンボボックスを開くこともできる。自動的に開くには、コンボボックスの[フォーカス取得時]イベントに次のプロシー

ジャを記述する。
コンボボックスのプロパティを開いて,[フォーカス取得時]イベントに[イベントプロシージャ]を指定し、右端のアイコンをクリックする。



次のようにプロシージャを記述する。



```
Private Sub コンボ16_Enter()
    Me.コンボ16.DropDown
End Sub
```

フォームを表示してカーソルを上から順に移動させて、コンボボックスの位置に来たときにコンボボックスが自動的に開くことを確認する。

得意先名																							
グループコード																							
フォーカス時に自動的に開く	<table border="1"> <tr><td>いとうはなこ</td><td>伊藤花子</td><td>106</td></tr> <tr><td>うえしまじろう</td><td>上島二郎</td><td>101</td></tr> <tr><td>うらひでき</td><td>浦秀樹</td><td>103</td></tr> <tr><td>すずきいちろう</td><td>鈴木一郎</td><td>102</td></tr> <tr><td>ながけしんいち</td><td>長田真一</td><td>104</td></tr> <tr><td>やまだたろう</td><td>山田太郎</td><td>100</td></tr> <tr><td>よしだゆきこ</td><td>吉田幸子</td><td>105</td></tr> </table>		いとうはなこ	伊藤花子	106	うえしまじろう	上島二郎	101	うらひでき	浦秀樹	103	すずきいちろう	鈴木一郎	102	ながけしんいち	長田真一	104	やまだたろう	山田太郎	100	よしだゆきこ	吉田幸子	105
いとうはなこ	伊藤花子	106																					
うえしまじろう	上島二郎	101																					
うらひでき	浦秀樹	103																					
すずきいちろう	鈴木一郎	102																					
ながけしんいち	長田真一	104																					
やまだたろう	山田太郎	100																					
よしだゆきこ	吉田幸子	105																					

- テキストボックスの値によってコンボボックス内のデータを先頭一致で絞り込むには

たとえば、次のように検索値を入力するテキストボックスとコンボボックスがあるとしよう。
ここではテキストボックスに『い』と入力した後下のコンボボックスを開くと次のような状態と
なって欲しいわけだ。

このようにコンボボックス内のデータが絞り込まれるとデータの選択がしやすくなる。

検索値

上の値から
絞り込まれたもの

いとうはなこ	伊藤花子	106
いづきごろう	伊吹五郎	107

このようなコンボボックスを作成するには、テキストボックスの名前を[検索値]とし、コンボボッ
クスの[値集合タイプ]を『テーブル/クエリ』として、次のような[値集合ソース]とする。

フィールド欄の[よみ]の[抽出条件]に次の式を登録する。

式：Like [forms][フォーム 1][検索値] & "*"

『*』アスタリスクがワイルドカードを意味し、結局、[検索値]の値によってデータを先頭一致で
探し出すという意味になる。

SQL ステートメント: クエリビルダ

得意先マスタ

*
得意先コード
得意先名
よみ
グループコード

フィールド: よみ
テーブル: 得意先マスタ
並べ替え: 昇順
表示: ☒
抽出条件: Like [forms][フォーム 1][検索値] & "*" または

得意先コード	得意先名	よみ
	得意先マスタ	得意先マスタ

[検索値]テキストボックスの[更新後処理]イベントに、次のコードを忘れずに入れる。

これによって、テキストボックスの値が書き換えられた直後に、コンボボックスのリクエリが行
なわれる。

```
Private Sub 検索値_AfterUpdate()  
    Me.コンボ 59.Requery  
End Sub
```

そのほかの設定はこれまでのコンボボックスと変わりはない。

その他の検索方法としては、

完全一致
値と検索値を完全一致させて検索する。
例: [Forms][フォーム名][テキストボックス名]

部分一致
検索値の前後をワイルドカードで囲む
例: Like "*" & [Forms][フォーム名][テキストボックス名] & "*"

後方一致
ワイルドカードを検索値の前に持ってくる

例: Like "*" & [Forms]![フォーム名]![テキストボックス名]

などがある。

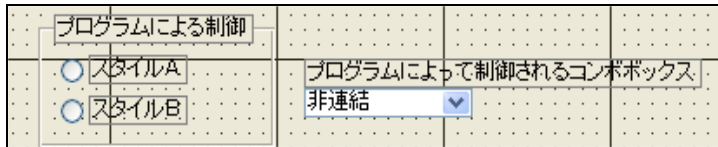
式の記述方法については、Access のヘルプで『クエリの抽出条件の例』を参照されたい。

- コンボボックスをプログラムで書き換えるには

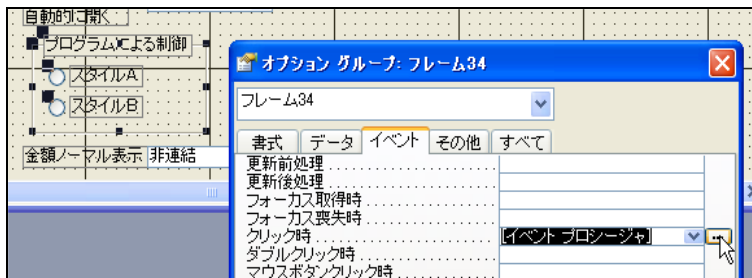
これまではコンボボックスの設定をフォームデザインビューで手動で行ってきたが、プログラムによってコンボボックスの設定を変更することもできる。

たとえば、次のようなオプショングループを配置して、オプショングループの[クリック時]に次のようなプロシーダを設定する。

オプションボタン横のコンボボックスは、3 番目に作成したコンボボックスを複写したものである。



オプショングループをマウスで選択し、[クリック時]イベントに[イベントプロシーダ]を指定して、右端のアイコンをクリックする。



オプショングループの[クリック時]イベントのプロシーダは次の通り。

この処理ではオプショングループがクリックされるたびに、コンボボックスの[値集合ソース]の文字列を入れ替えているのである。

```
Private Sub フレーム 34_Click()  
    Select Case Me.フレーム 34  
        Case 1  
            Me.コンボ 41.RowSource = "SELECT 得意先マスタ.得意先コード, 得意先マスタ.得意先名,  
得意先マスタ.よみ FROM 得意先マスタ ORDER BY 得意先マスタ.得意先コード;"  
        Case 2  
            Me.コンボ 41.RowSource = "SELECT 得意先マスタ.よみ, 得意先マスタ.得意先名, 得意先マ  
スタ.得意先コード FROM 得意先マスタ ORDER BY 得意先マスタ.よみ;"  
    End Select  
End Sub  
青文字が記述されたステートメント。ステートメントが折り返しているが本来は 1 行で書かれたもの。
```

フォームを表示して動作を確認する。

スタイルAを選択した後、コンボボックスを開く。

プログラムによる制御

☒ スタイルA
☐ スタイルB

プログラムによって制御されるコンボボックス

100	山田太郎	やまだたろう
101	上島二郎	うえしまじろう
102	鈴木一郎	すずきいちろう
103	浦秀樹	うらひでき
104	長田真一	ながたしんいち
105	吉田幸子	よしだゆきこ
106	伊藤花子	いとうはなこ

スタイルBを選択した後、コンボボックスを開く。

あきらかにプログラムで制御されていることがわかる。

プログラムによる制御

☐ スタイルA
☒ スタイルB

プログラムによって制御されるコンボボックス

いとうはなこ	伊藤花子	106
うえしまじろう	上島二郎	101
うらひでき	浦秀樹	103
すずきいちろう	鈴木一郎	102
ながたしんいち	長田真一	104
やまだたろう	山田太郎	100
よしだゆきこ	吉田幸子	105

●コンボボックス内で数値を右寄せするには

たとえば、数値フィールドを表示する次のようなコンボボックスがある。右端の数値を見ればわかるように、普通に数値フィールドを表示させると文字列は左寄せに表示される。

金額ノーマル表示

金額右揃え表示

105	吉田幸子	よしだゆきこ	¥1,000
100	山田太郎	やまだたろう	¥3,000
104	長田真一	ながたしんいち	
102	鈴木一郎	すずきいちろう	
103	浦秀樹	うらひでき	¥1,000,000
101	上島二郎	うえしまじろう	¥400
106	伊藤花子	いとうはなこ	

これを次のように右寄せ表示したいわけだ。

金額右揃え表示

105	吉田幸子	よしだゆきこ	1,000
100	山田太郎	やまだたろう	3,000
104	長田真一	ながたしんいち	
102	鈴木一郎	すずきいちろう	
103	浦秀樹	うらひでき	1,000,000
101	上島二郎	うえしまじろう	400
106	伊藤花子	いとうはなこ	

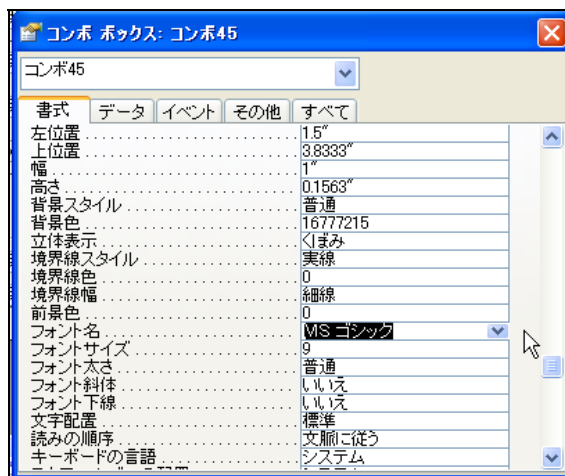
右寄せにするには、コンボボックスの[値集合ソース]のクエリで数値フィールド(たとえば[金額])を次のように変更する。

式 1: Format(Format([金額],"#,##0"),"@@@@@@@@@@")

よみ	式1: Format(Format([金額],"##0"),"@@@@@@@@@")
得意先マスタ	
降順	☒
	☒

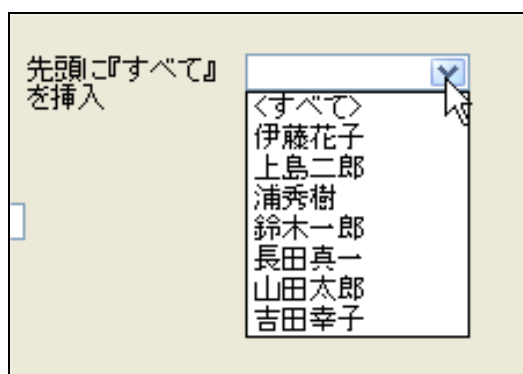
Format 関数は文字列や数値をさまざまな書式で表示する関数である。

さらに、コンボボックスのプロパティで、[フォント名]を『MSゴシック』または『MS明朝』などの固定文字幅フォントを指定すること。



- コンボボックスの先頭に例えば『<すべて>』などの文字列を入れるには

たとえば、次のようなコンボボックスを作成したいと思われたことはないだろうか？
普通ならテーブルの値をそのまま表示するしか方法はないのだが、工夫することでこのように任意の文字列を先頭行に表示することができる。

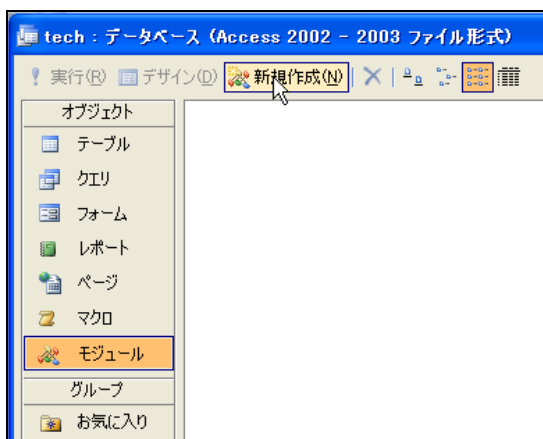


このようなコンボボックスを実現する手順は次の通り。

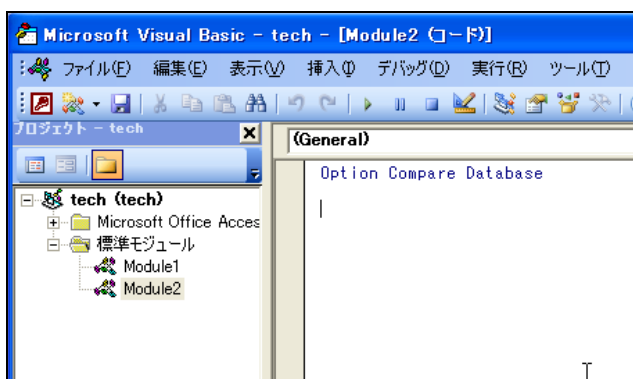
なお、この方法はマイクロソフトのサイトで公開されている方法をそのままご紹介するものである。

1. 次に示すモジュールを登録する。

モジュールを登録するには、データベースウィンドウの[モジュール]を選択して、[新規作成]をクリックする。



2. 次の図のようにモジュール作成ウィンドウが表示される。



3. 次に示すファンクションプロシージャをすべてコピーして張り付ける。

このプログラムはマイクロソフトのサポートより発表されているもの。関数名は AddAllToList
Access2000 ~ Access2003 で動作することが確認されている。

```
Function AddAllToList(C As Control, ID As Long, Row As Long, _  
Col As Long, Code As Integer) As Variant  
*****  
' FUNCTION: AddAllToList()  
' PURPOSE:  
'   Adds "(all)" as the first row of a combo box or list box.  
' USAGE:  
'   1. Create a combo box or list box that displays the data you  
'       want.  
'   2. Change the RowSourceType property from "Table/Query" to  
'       "AddAllToList."  
'   3. Set the value of the combo box or list box's Tag property to  
'       the column number in which you want "(all)" to appear.  
' NOTE: Following the column number in the Tag property, you can  
'       enter a semicolon (;) and then any text you want to appear  
'       other than the default "all."  
'       For example  
'       Tag: 2;<None>
```

```
'           displays "<None>" in the second column of the list.
'*****

Static DB As Database, RS As Recordset
Static DISPLAYID As Long
Static DISPLAYCOL As Integer
Static DISPLAYTEXT As String
Dim Semicolon As Integer

On Error GoTo Err_AddAllToList

Select Case Code
Case LB_INITIALIZE
    ' See if the function is already in use.
    If DISPLAYID <> 0 Then
        MsgBox "AddAllToList is already in use by another Control!"
        AddAllToList = False
        Exit Function
    End If

    ' Parse the display column and display text from the Tag
    ' property.
    DISPLAYCOL = 1
    DISPLAYTEXT = "(All)"
    If Not IsNull(C.Tag) Then
        Semicolon = InStr(C.Tag, ";")
        If Semicolon = 0 Then
            DISPLAYCOL = Val(C.Tag)
        Else
            DISPLAYCOL = Val(Left(C.Tag, Semicolon - 1))
            DISPLAYTEXT = Mid(C.Tag, Semicolon + 1)
        End If
    End If

    ' Open the recordset defined in the RowSource property.
    Set DB = DBEngine.Workspaces(0).Databases(0)
    Set RS = DB.OpenRecordset(C.RowSource, DB_OPEN_SNAPSHOT)

    ' Record and return the ID for this function.
    DISPLAYID = Timer
    AddAllToList = DISPLAYID

Case LB_OPEN
    AddAllToList = DISPLAYID

Case LB_GETROWCOUNT
    ' Return the number of rows in the recordset.
    RS.MoveLast
    AddAllToList = RS.RecordCount + 1
```

```
Case LB_GETCOLUMNCOUNT
    ' Return the number of fields (columns) in the recordset.
    AddAllToList = RS.Fields.Count

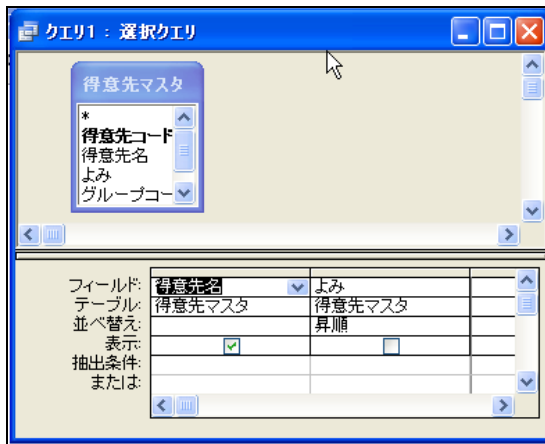
Case LB_GETCOLUMNWIDTH
    AddAllToList = -1

Case LB_GETVALUE
    ' Are you requesting the first row?
    If Row = 0 Then
        ' Should the column display "(All)"?
        If Col = DISPLAYCOL - 1 Then
            ' If so, return "(All)."
            AddAllToList = DISPLAYTEXT
        Else
            ' Otherwise, return NULL.
            AddAllToList = Null
        End If
    Else
        ' Grab the record and field for the specified row/column.
        RS.MoveFirst
        RS.Move Row - 1
        AddAllToList = RS(Col)
    End If
Case LB_END
    DISPLAYID = 0
    RS.Close
End Select

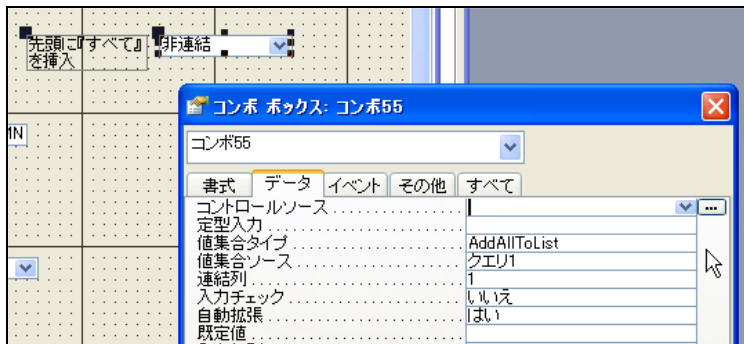
Bye_AddAllToList:
    Exit Function

Err_AddAllToList:
    Beep:    MsgBox Error$, 16, "AddAllToList"
    AddAllToList = False
    Resume Bye_AddAllToList
End Function
```

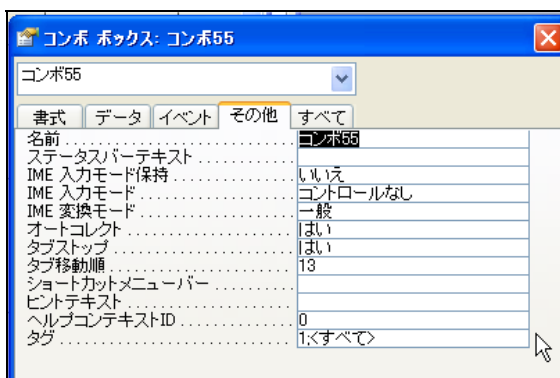
4. コンボボックスで表示するテーブルまたはクエリを用意する。
今回は次のクエリを準備した。



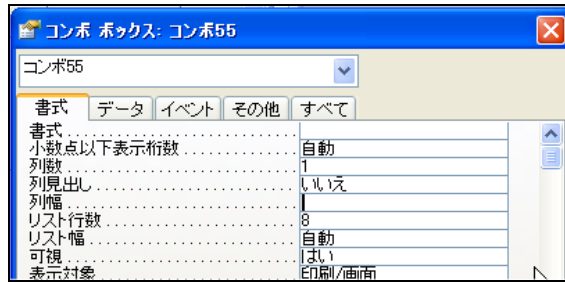
5. 該当するコンボボックスのプロパティで、[値集合タイプ]に先ほど登録した関数名『AddAllToList』を登録し、[値集合ソース]にテーブル名またはクエリを指定する。



6. コンボボックスのプロパティで、[その他]タブをクリックし、[タグ]に次の文字列を設定する。
文字列: 1;<すべて>
左の数値は文字列を表示する列を意味し、右側の文字列はコンボボックスに表示する文字列を意味する。



7. 今回の書式設定は次の通り。



8. フォームを表示して、動作を確認する。

これは、[値集合タイプ]で指定された関数(小さなプログラムと考えるとよい)を、コンボボックスのレコードを表示するたびに実行していると思えばよい。

ほかの方法としてはユニオンクエリーを用いるという方法もあるが、手間がかからないという点ではこの方法がよいのかもしれない。

以上