

徹底的にサブフォーム活用

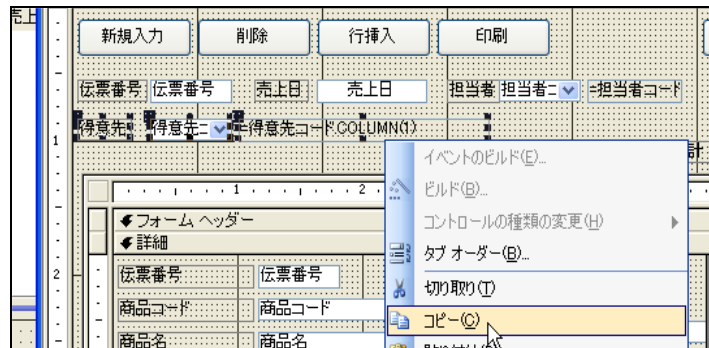
平成 19 年 1 月 18 日
有限会社ビット
浦 秀樹

● コンボボックスとサブフォームの連携

コンボボックスとサブフォームの連携は非常によくあるパターンだ。まずは、このスタンダードな手法を修得して後の応用に備えよう。

フォームを新規作成し、任意の場所にコンボボックスとサブフォームを配置すればよいのだが、ここでは以前作成した別フォームのコンボボックスをコピーして再利用することにする。このように、コンボボックスはコピーして使われることが非常に多い。

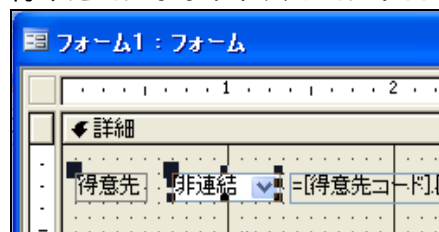
1. 参考にしたいコンボボックスを選んでコピーする。



2. 新規作成したフォームで貼り付けする。



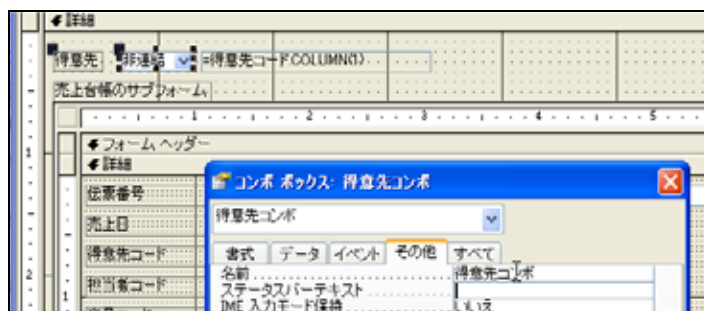
3. 貼り付けたコンボボックスのコントロールソース名を削除する。



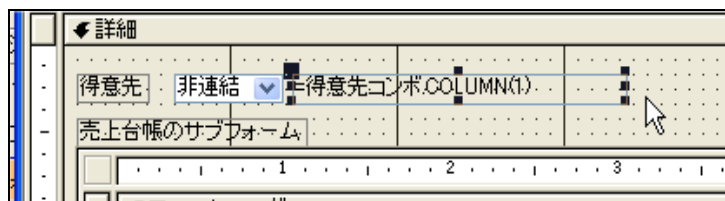
コントロールソース名が削除されると『非連結』という表示となる。

4. 貼り付けたコンボボックスの名前を変更する。

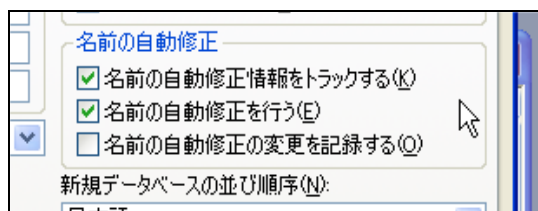
このままの使用でもかまわないが今回は変更する。



5. コンボボックスの名前を変更すると、右のテキストボックスの計算式が自動的に変更される。



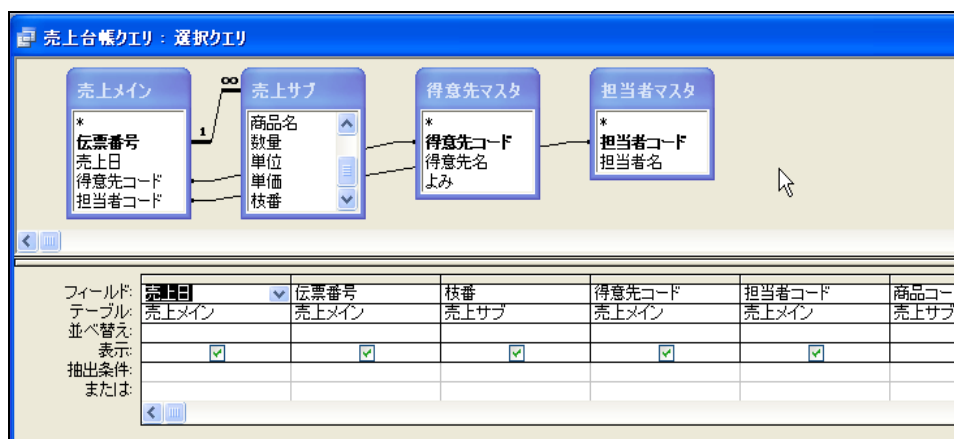
自動的に変更したくない場合は、メニュー[ツール]、[オプション]をクリックしてオプション画面を開き、[全般]タブをクリックして[名前の自動修正]をすべてオフにする。



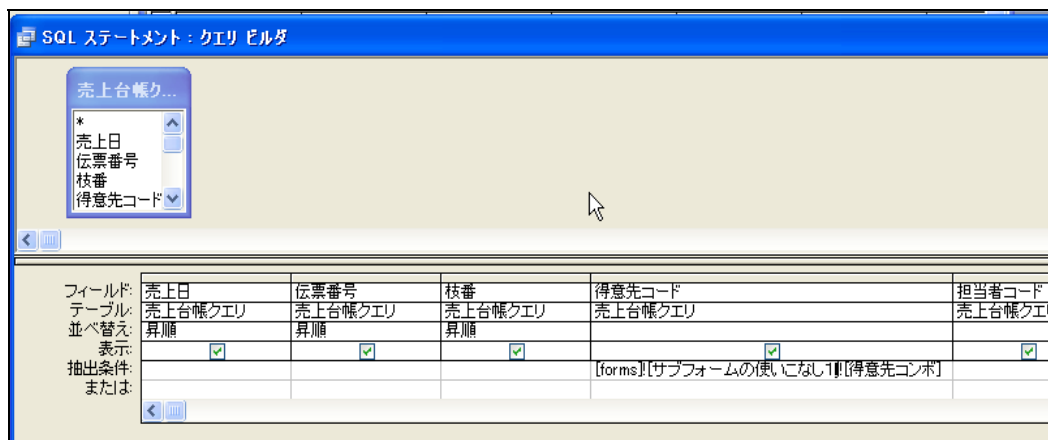
6. フォームにサブフォームを作成する。

サブフォームの作成は、ツールバーから[サブフォーム/サブレポート]アイコンをクリックしてウィザードに従って作成する。

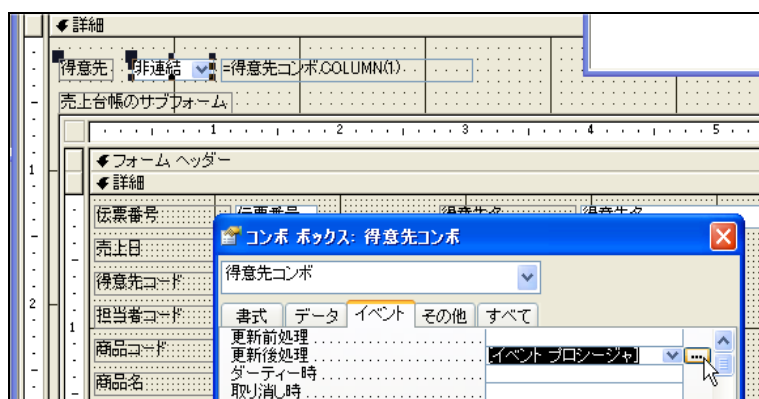
ここでは、あらかじめサブフォーム用のクエリを次のように作成しておく。



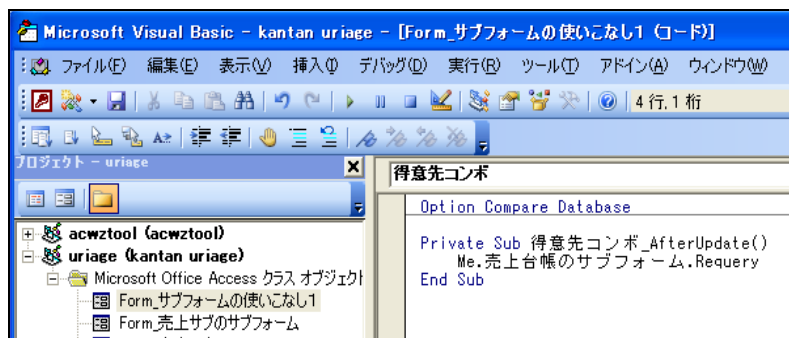
7. いったんウィザードで作成した後、サブフォームのプロパティから[レコードソース]を開いて、[並べ替え]の設定とサブフォームのデータがコンボボックスの値によって絞り込まれるよう[得意先コード]の抽出条件を次のように設定する。
[得意先コード]の抽出式：forms!サブフォームの使いこなし1!得意先コンボ



8. このままではコンボボックスの操作にサブフォームが反応してくれないので、コンボボックスのプロパティを表示し、[イベント]タブをクリックして[更新後処理]を『イベントプロシージャ』として右端のアイコンをクリックしてイベントプロシージャを登録する。



9. 次のようなプロシージャを登録する。



```
Private Sub 得意先コンボ_AfterUpdate()
    Me.売上台帳のサブフォーム.Requery
End Sub
```

10. フォームを表示して、動作を確認する。

● サブフォームの並び順を変更するには

たとえば、次の図のようにオプショングループを配置して、サブフォームデータの並び順を指定したい場合がある。

サブフォームの並び順をコントロールするためには次のようなオプショングループを配置し、[クリック時]イベントに次のようなプロシージャを登録する。

プロシージャは次の通り。

```
Private Sub 並び替え指定_Click()
    Select Case Me.並び替え指定
        Case 1
            '並び替えの条件を指定する
            Me![売上台帳のサブフォーム].Form.OrderBy = "[売上日],[伝票番号],[枝番]"
        Case 2
            'DESC は降順指定を意味する
            Me![売上台帳のサブフォーム].Form.OrderBy = "[売上日] DESC,[伝票番号],[枝番]"
        Case 3
            Me![売上台帳のサブフォーム].Form.OrderBy = "[商品コード],[売上日],[伝票番号],[枝番]"
    End Select
End Sub
```

```
End Select
'次のステートメントで並べ替えが実行される
Me![売上台帳のサブフォーム].Form.OrderByOn = True
End Sub
```

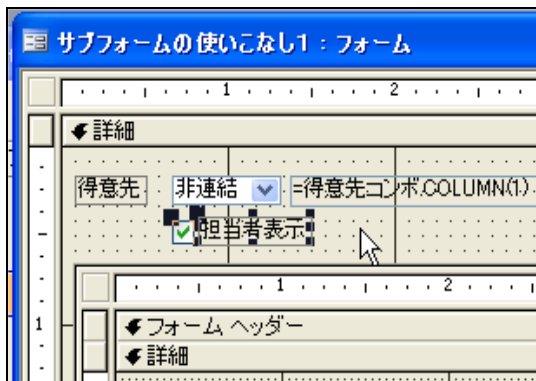
メインフォームの[開く時]イベントに、忘れずに次のプロシージャを登録する。

```
Private Sub Form_Open(Cancel As Integer)
'フォームを開いたときに並べ替えが行われないようにセット
Me![売上台帳のサブフォーム].Form.OrderByOn = False
'並べ替え条件をリセット
Me![売上台帳のサブフォーム].Form.OrderBy = ""
End Sub
```

- サブフォームの列を非表示にするには

サブフォームの列をプログラムによって表示非表示を切り替える。

たとえば、次のようなチェックボックスをフォームに配置する。このチェックボックスがオンのときは担当者の列が表示となり、オフのときは非表示となる。



このチェックボックスの[クリック時]イベントに次のプロシージャを登録する。

```
Private Sub チェック 13_Click()
If Me.チェック 13 Then
'カラム表示
Me.売上台帳のサブフォーム!担当者コード.ColumnHidden = 0
Me.売上台帳のサブフォーム!担当者名.ColumnHidden = 0

Else
'カラム非表示
Me.売上台帳のサブフォーム!担当者コード.ColumnHidden = -1
Me.売上台帳のサブフォーム!担当者名.ColumnHidden = -1
End If
End Sub
```

わずかこれだけで表示非表示はコントロールできる。

● 列の並び順を指定するには

たとえば、次に示すフォームのように列の並び順を指定するチェックボックスを登録し、操作によって列の並び順を変更したい場合がある。[得意先名先頭表示]チェックボックスに注目して欲しい。

[得意先名先頭表示]チェックボックスをクリックすると次の図となる。

[得意先コード]と[得意先名]が左端先頭に表示されていることが分かる。

このような制御を実現するためには、[得意先名先頭表示]チェックボックスの[更新後処理]イベントに次のようなプロシージャを設定する。

```
Private Sub 得意先名先頭表示_AfterUpdate()
    If Me.得意先名優先 Then
        Me.売上台帳のサブフォーム!得意先コード.ColumnOrder = 1
        Me.売上台帳のサブフォーム!得意先名.ColumnOrder = 2
    Else
        Me.売上台帳のサブフォーム!伝票番号.ColumnOrder = 1
        Me.売上台帳のサブフォーム!売上日.ColumnOrder = 2
        Me.売上台帳のサブフォーム!得意先コード.ColumnOrder = 3
        Me.売上台帳のサブフォーム!得意先名.ColumnOrder = 4
        Me.売上台帳のサブフォーム!担当者コード.ColumnOrder = 5
        Me.売上台帳のサブフォーム!担当者名.ColumnOrder = 6
        Me.売上台帳のサブフォーム!商品コード.ColumnOrder = 7
        Me.売上台帳のサブフォーム!商品名.ColumnOrder = 8
    End If
End Sub
```

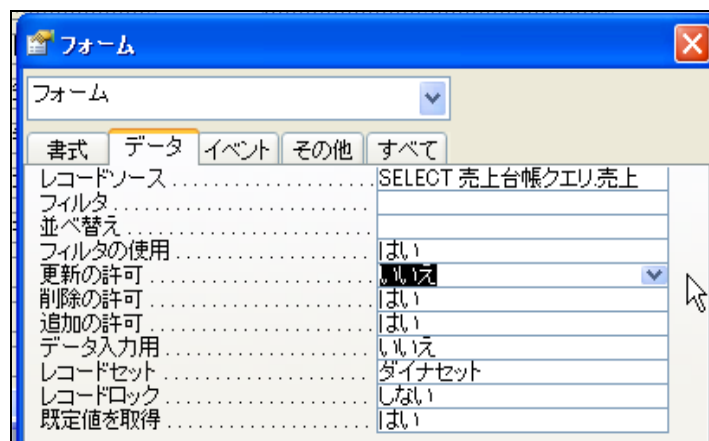
さらに、メインフォームの[開く時]イベントに列の並び順を初期化する次のようなプロシーダを登録する。

```
'列の並び替えをリセット
Me.売上台帳のサブフォーム!伝票番号.ColumnOrder = 1
Me.売上台帳のサブフォーム!売上日.ColumnOrder = 2
Me.売上台帳のサブフォーム!得意先コード.ColumnOrder = 3
Me.売上台帳のサブフォーム!得意先名.ColumnOrder = 4
Me.売上台帳のサブフォーム!担当者コード.ColumnOrder = 5
Me.売上台帳のサブフォーム!担当者名.ColumnOrder = 6
Me.売上台帳のサブフォーム!商品コード.ColumnOrder = 7
Me.売上台帳のサブフォーム!商品名.ColumnOrder = 8
```

同じ処理がほかにも多数あるとすれば、上プロシーダ部分を別プロシーダとして登録して呼び出す形で利用されてもよい。

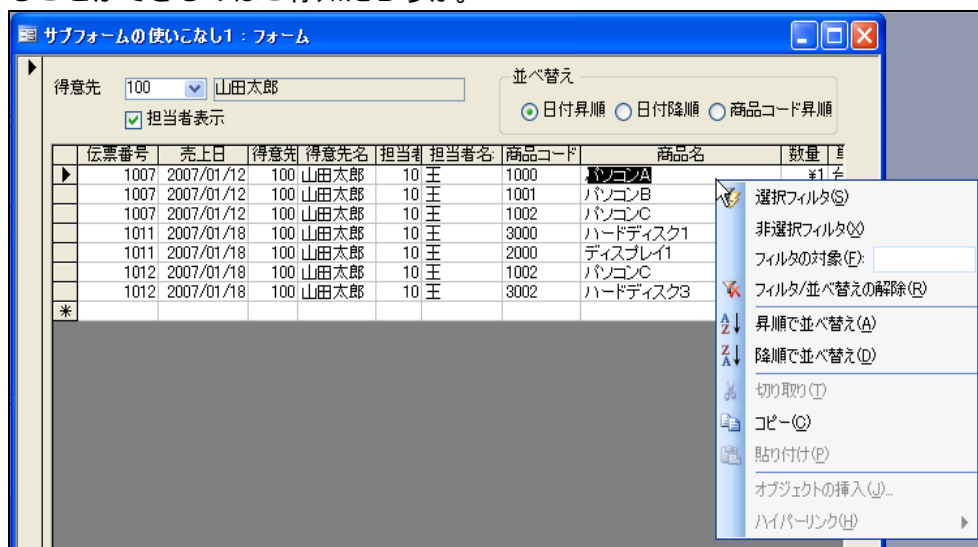
- サブフォームでデータの変更を許可しないには

サブフォーム上のデータを変更されたくない場合は、サブフォームのプロパティで[更新の許可]を『いいえ』に設定する。ほかに、[削除の許可]、[追加の許可]なども必要に応じて設定する。



- サブフォーム上でショートカットを利用する

たとえば、フォームを表示して次のようにショートカットメニューを表示し、表示される各処理を実行することができるのはご存知だろうか。

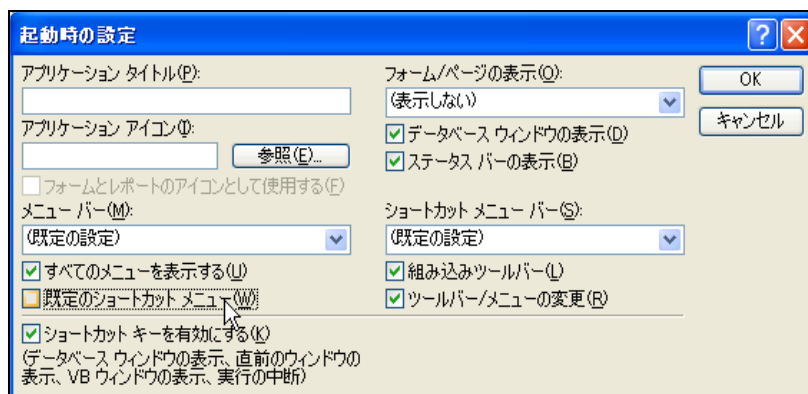


この機能は Access ランタイムでは利用不可能

ただし、この機能は逆に使わせたくない場合もある
この機能を制限するには次に示す 2 通りがある。

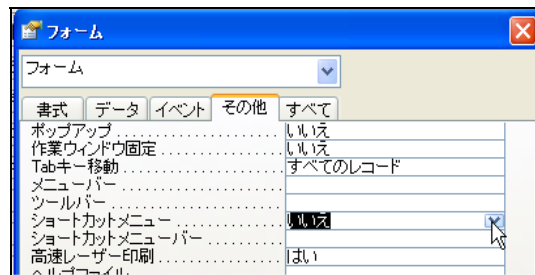
- ・データベース全体で使用禁止にする

データベースの[起動時の設定]でデータベース全体での制御を行うことができる。
この設定を変更したときは、データベース再起動が必要



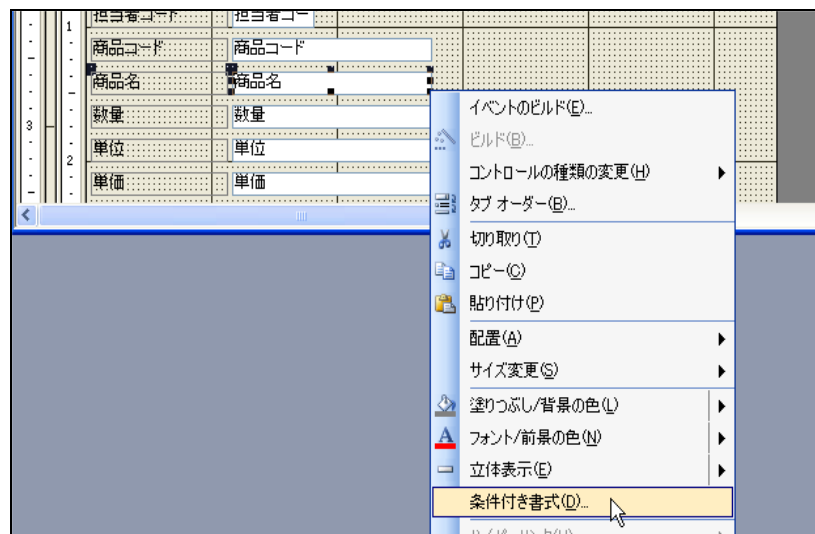
- ・フォームごとに制限する

フォームごとに制限する場合は、データベース全体の設定を許可した後、フォームのプロパティを開いて[ショートカットメニュー]を『いいえ』に設定する。

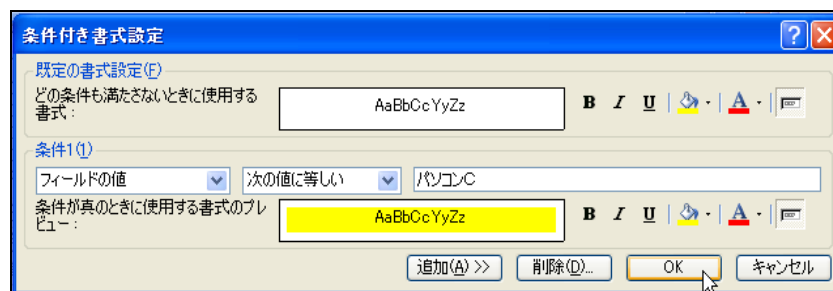


- サブフォームに色を付けるには

サブフォーム上ではデータの値によってレコードを色分け表示することができる。
色分け表示を設定するには、目的のテキストボックスを選択してショートカットメニューを表示し、[条件付き書式]をクリックする。



[条件付き書式設定]画面で、たとえばテキストボックスの値が『パソコンC』という値である場合に黄色表示するには次の設定を行う。



フォームを表示して動作を確認する。

その他、

- ・ 式の結果によって
- ・ フォカスのあるレコード

などのケースにも色を指定することができる。

● サブフォームの yes/no 型データとオプションボタンの連携

たとえば、フォーム上に次のようなオプショングループを配置して、データを表示したい場合がある。

右側の[表示制御]オプショングループに注目してほしい。

この表示制御で、[未完了]をクリックするとサブフォームの[完了]チェックがオフのレコードが表示され、[完了]をクリックするとオンのレコードが表示され、[すべて]がクリックされるとすべてのレコードが表示される。

このようなサブフォームの抽出を実現するためには、サブフォームのレコードソースの[完了]フィールドに次の抽出条件を設定する。

抽出条件： <>[FORMS][サブフォームの使いこなし1][表示制御]

フォームに図のようなオプショングループを設定し、オプション値を次のように設定する。

オプションボタン名	オプション値
未完了	-1
完了	0
すべて	2

[すべて]のオプション値は-1、0 以外ならなんでもよい。

次に、[表示制御]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ジャを設定する。

```
Private Sub 表示制御_Click()
    Me.売上台帳のサブフォーム.Requery
End Sub
```

フォームを表示して動作を確認する。

サブフォームの使いこなし1: フォーム

得意先: 100 山田太郎

☒ 担当者表示

並べ替え: ☒ 日付昇順 ☐ 日付降順 ☐ 商品コード昇順

表示制御: ☐ 未完了 ☒ 完了 ☐ すべて

売上台帳のサブフォーム

	完了	伝票番号	売上日	得意先	得意先名	担当者	担当者名	商品コード	商品名
▶	☒	1007	2007/01/12	100	山田太郎	10	主	1002	パソコンC
▶	☒	1012	2007/01/18	100	山田太郎	10	主	1002	パソコンC

サブフォームの使いこなし1: フォーム

得意先: 100 山田太郎

☒ 担当者表示

並べ替え: ☒ 日付昇順 ☐ 日付降順 ☐ 商品コード昇順

表示制御: ☐ 未完了 ☐ 完了 ☒ すべて

売上台帳のサブフォーム

	完了	伝票番号	売上日	得意先	得意先名	担当者	担当者名	商品コード	商品名
▶	☐	1007	2007/01/12	100	山田太郎	10	主	1000	パソコンA
▶	☐	1007	2007/01/12	100	山田太郎	10	主	1001	パソコンB
	☒	1007	2007/01/12	100	山田太郎	10	主	1002	パソコンC
	☐	1011	2007/01/18	100	山田太郎	10	主	3000	ハードディスク1
	☐	1011	2007/01/18	100	山田太郎	10	主	2000	ディスプレイ1
	☒	1012	2007/01/18	100	山田太郎	10	主	1002	パソコンC
	☐	1012	2007/01/18	100	山田太郎	10	主	3002	ハードディスク3

●サブフォームとサブフォームの連携

一つのフォーム上で二つのサブフォームを表示して連携をとりたい場合がある。たとえば、次のようなフォームだ。

このフォームは、得意先マスタのいずれかをマウスでクリックすると、その得意先コードによって絞り込まれた売上データが下のサブフォームで表示されるというものだ。

サブフォーム同士の連携: フォーム

得意先サブフォーム

得意先コード	得意先名	よみ
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	よしたゆきこ
106	伊藤花子	いとうはなこ
*	0	

レコード: 1 / 7

売上台帳サブフォーム

売上日	伝票番号	得意先	得意先名	担当者	担当者名	商品コード	商品名
2007/01/12	1007	100	山田太郎	10	王	1000	パソコンA
2007/01/12	1007	100	山田太郎	10	王	1001	パソコンB
2007/01/12	1007	100	山田太郎	10	王	1002	パソコンC
2007/01/18	1011	100	山田太郎	10	王	3000	ハードディスク1
2007/01/18	1011	100	山田太郎	10	王	2000	ディスプレイ1
2007/01/18	1012	100	山田太郎	10	王	1002	パソコンC
2007/01/18	1012	100	山田太郎	10	王	3002	ハードディスク3
*							

レコード: 1 / 7

レコード: 1 / 1

上は[得意先名]の「山田太郎」を選択した場合

サブフォーム同士の連携: フォーム

得意先サブフォーム

得意先コード	得意先名	よみ
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	よしたゆきこ
106	伊藤花子	いとうはなこ
*	0	

レコード: 3 / 7

売上台帳サブフォーム

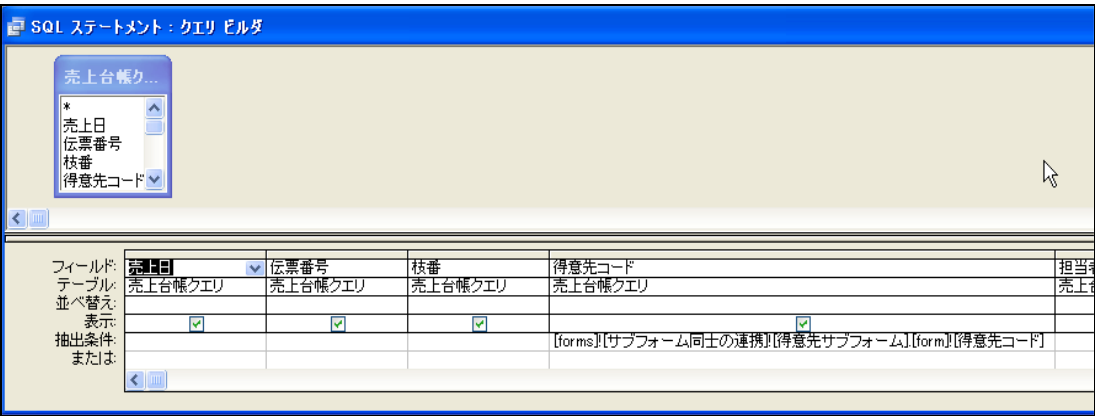
売上日	伝票番号	得意先	得意先名	担当者	担当者名	商品コード	商品名
2007/01/12	1008	102	鈴木一郎	12	金田	2000	ディスプレイ1
2007/01/12	1008	102	鈴木一郎	12	金田	3000	ハードディスク1
2007/01/17	1009	102	鈴木一郎	12	金田	1000	パソコンA
2007/01/17	1009	102	鈴木一郎	12	金田	2000	ディスプレイ1
2007/01/17	1009	102	鈴木一郎	12	金田	3000	ハードディスク1
2007/01/17	1009	102	鈴木一郎	12	金田	3002	ハードディスク3
*							

上は[得意先名]の「鈴木一郎」を選択した場合

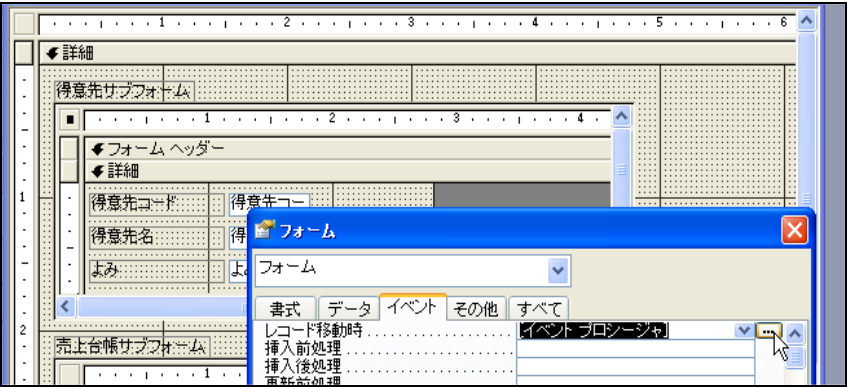
このようなフォームは実に簡単に作成できる。

まず、フォーム上にそれぞれ2つのサブフォームを配置する。次に、下サブフォームのレコードソースをクエリビルダで開いて、[得意先コード]の抽出条件に次のような設定を行う。

抽出条件: [forms]![サブフォーム同士の連携]![得意先サブフォーム].[form]![得意先コード]



さらに、上サブフォームの[レコード移動時]イベントに次のプロシージャを設定する。



```
Private Sub Form_Current()  
    Forms!サブフォーム同士の連携!売上台帳サブフォーム.Requery  
End Sub
```

メインフォームが開かれたときにサブフォームの[レコード移動時]イベントが発生する。

わずかこれだけでダイナミックな連携が可能となるのである。

●サブフォームとを印刷を連動させるには

たとえば次のようなフォームがあり、[得意先サブフォーム]で任意のレコードをマウスで選択して、その後に[印刷]ボタンがクリックされたら、印刷データは当然ながら指定された得意先のデータでなければならない。

サブフォーム同士の連携: フォーム

得意先サブフォーム

得意先コード	得意先名	よみ
▶ 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
※	0	

レコード: 1 / 7

売上台帳サブフォーム

売上日	伝票番号	得意先	得意先名	担当者	担当者名	商品コード	商品名
▶ 2007/01/12	1007	100	山田太郎	10	主	1000	パソコンA
2007/01/12	1007	100	山田太郎	10	主	1001	パソコンB
2007/01/12	1007	100	山田太郎	10	主	1002	パソコンC
2007/01/18	1011	100	山田太郎	10	主	3000	ハードディスク1
2007/01/18	1011	100	山田太郎	10	主	2000	ディスプレイ1
2007/01/18	1012	100	山田太郎	10	主	1002	パソコンC
2007/01/18	1012	100	山田太郎	10	主	3002	ハードディスク3
※							

レコード: 1 / 7

レコード: 1 / 1

[印刷]ボタンがクリックされたら次のようなレポートが印刷されることとする。このレポートはあらかじめ作成してあるものとする。

売上台帳

得意先	売上日	伝票番号	商品コード	商品名	数量	単位	単価	金額
100 山田太郎								
	07/01/12	1007	1000	パソコンA	¥1	台	¥10,000	10000
	07/01/12	1007	1001	パソコンB	¥10	台	¥12,000	120000
	07/01/12	1007	1002	パソコンC	¥2	台	¥9,800	19600
	07/01/18	1011	3000	ハードディスク1	¥3	台	¥5,000	15000
	07/01/18	1011	2000	ディスプレイ1	¥3	台	¥3,000	9000
	07/01/18	1012	1002	パソコンC	¥5	台	¥9,800	49000
	07/01/18	1012	3002	ハードディスク3	¥5	台	¥7,000	35000
102 鈴木一郎								
	07/01/12	1008	2000	ディスプレイ1	¥2	台	¥3,000	6000

ページ: 1

レポートのレコードソースに抽出条件などはなにも設定されていない。

レポートの[並べ替え/グループ化]設定は次の通り。

レポートが準備できたら、[印刷]ボタンの[クリック時]イベントに次のプロシージャを登録する。

```
Private Sub 印刷_Click()
'印刷中に不意なエラーメッセージを表示させないための制御
On Error Resume Next
    If MsgBox("印刷しますか？", vbYesNo + vbInformation + vbDefaultButton2, "印刷") = vbYes Then
        '次のステートメントで印刷が実行される
        DoCmd.OpenReport "売上台帳", acViewPreview, "[得意先コード]=[forms]![サブフォーム同士の連携]![得意先サブフォーム].[form]![得意先コード]"
    End If
End Sub
```

青文字部分に注目。レポートを呼び出す際に、絞込み条件を文字列で与えているのである。

- 同時利用に関すること

以上