

一歩進んだ

Visual Basic

2001年3月21日
北陸工業専門学校

目次

1. はじめに	・・・	3
2. 英和辞典を作ろう		
2-1. 辞書データを作成しよう	・・・	4
2-2. Form レイアウトを作成しよう	・・・	5
2-3. プログラムを作成しよう	・・・	6
3. 和英辞典を作ろう		
3-1. Form レイアウトを作成しよう	・・・	10
3-2. プログラムを考えよう	・・・	11
3-3. 部分一致でも検索可能にしよう	・・・	12
3-4. いくつも検索できるようにしよう	・・・	14
4. 辞書データ編集画面を作ろう		
4-1. Form レイアウトを作成しよう	・・・	15
4-2. 辞書データを読み込んで表示させよう	・・・	16
4-3. 辞書データを変更して保存させよう	・・・	17
4-4. 辞書データを印刷しよう	・・・	18
4-5. 辞書編集画面を終了しよう	・・・	19
4-6. ファイル名を変更して編集できるようにしよう	・・・	20
5. 3つのプログラムをメニューから操作しよう		
5-1. メニュー画面を作成しよう	・・・	26
5-2. 3つのFormを追加しよう	・・・	27
5-3. それぞれを呼び出すプログラムを作成しよう	・・・	28
6. システムをグレードアップさせよう		
6-1. 各Formが終了してもメニューは残るようにしよう	・・・	29
6-2. 辞書ファイルを指定できるようにしよう	・・・	30

1. はじめに

本テキストは、既に Visual Basic でプログラムを作成した経験のある方を対象として編集しました。

Visual Basic の基本的な操作方法やプログラミング方法は知っているが、更に実用的なプログラミング技術を身につけたいという方にお薦めします。

ファイル操作、テキストデータの編集・保存・印刷、文字列検索、複数フォームのコントロールなどの技術が含まれています。

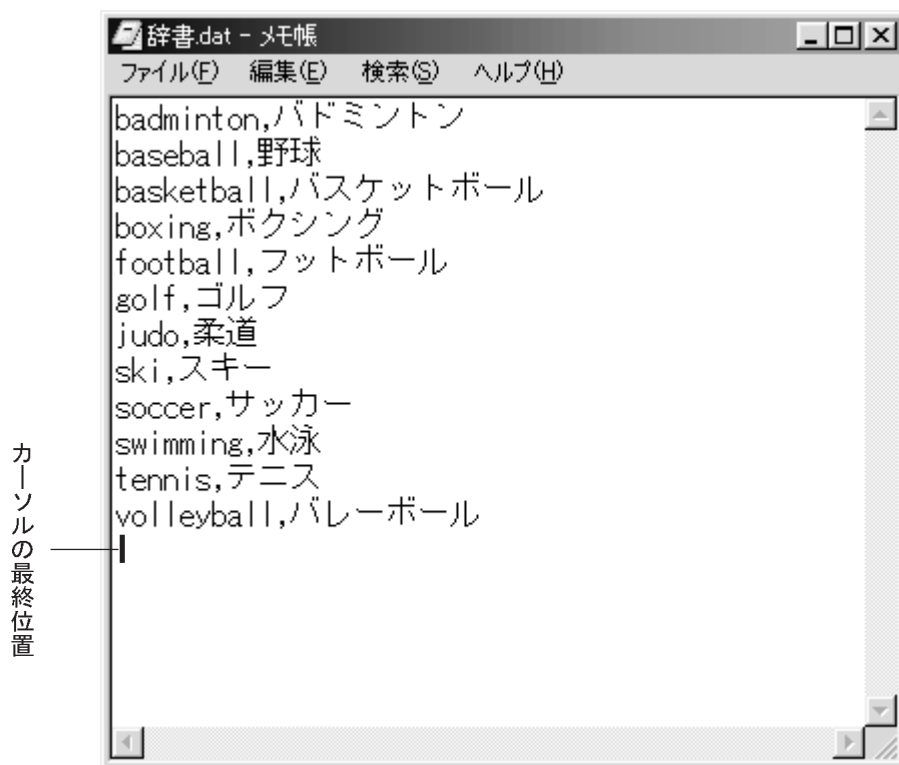
また、Visual Basic はまだよく分からないという方にも、できるだけ理解できるように操作方法を中心に説明しています。あきらめずに読み進んでください。

2. 英和辞典を作ろう

2-1. 辞書データを作成しよう

まず、辞書データを作成しましょう。

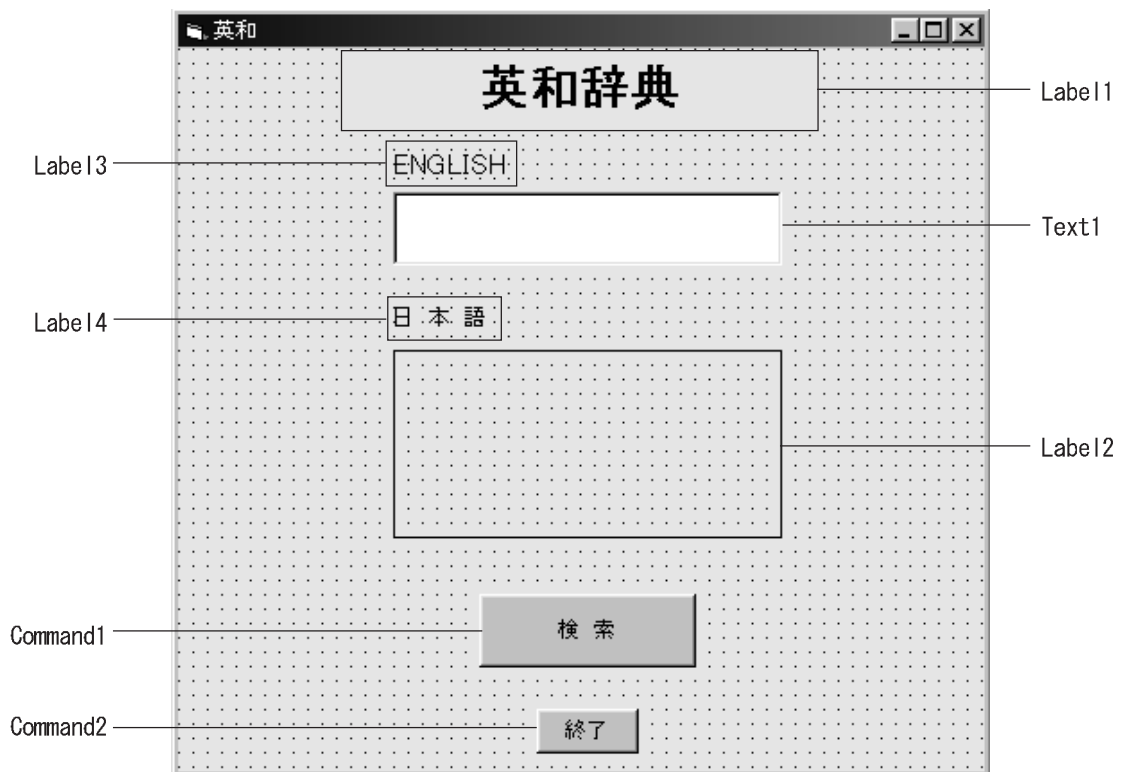
- ① メモ帳を開いて、下記データを作成します。
 - ・「英単語, 日本語」の形式で入力していきます。
 - ・間の「,」(カンマ)は必ず半角にします。
 - ・最後は「バレーボール」の次の行の先頭にカーソルが行くようにします。
 - ・カーソルの後ろにはデータが何も無い状態にします。
- ② フロッピーディスクに「辞書.dat」という名前で保存します。



2-2. Formレイアウトを作成しよう

英和辞典の画面を作成しましょう。

- ① Visual Basic を立ち上げます。
- ② 下記のようなFormを作成します。



- ③ Form のオブジェクト名、Caption を「英和」に変更します。

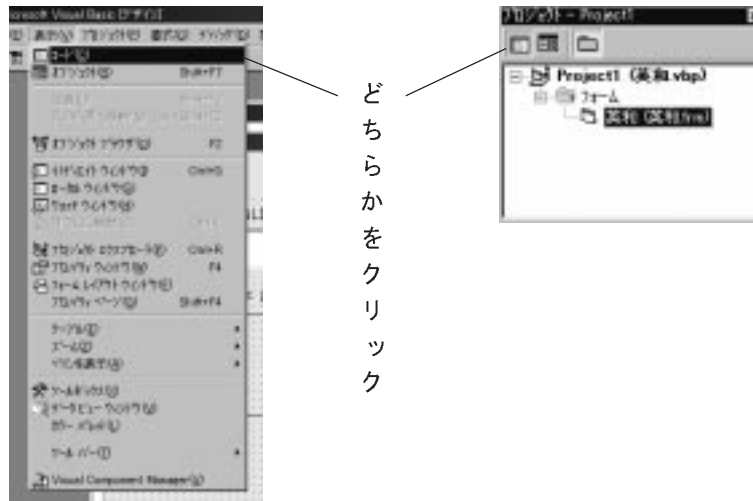


- ④ フロッピーディスクにプロジェクトとフォームを「英和」というファイル名で保存します。

2-3. プログラムを作成しよう

英和辞典のプログラム部分を作成しましょう。

- ① メニューバーの **表示** - **コード** を選択するか、プロジェクトエクスプローラからコードの表示を選択してコードエディタウィンドウを表示します。



- ② オブジェクトボックスを「Form」に設定し、イベントプロシジャーボックスを「Activate」に設定します。



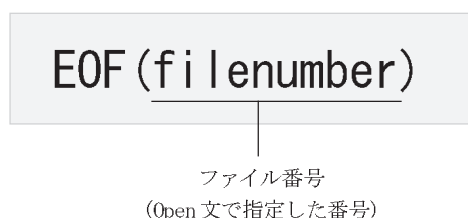
- ③ 辞書データをファイルからプログラム上に読み込む部分をプログラムします。
このプログラムは、Form が実行可能状態になった時 (Activate 時)、まず最初に実行されます。

<pre>Dim eng(50), jpn(50) As String Dim fname As String Dim n As Integer Private Sub Form_Activate() n = 0 fname = "A:辞書.dat" Open fname For Input As #1 While EOF(1) <> True n = n + 1 Input #1, eng(n), jpn(n) Wend Close #1 End Sub</pre>	<ul style="list-style-type: none"> ・ ・ ・ 英単語 50 件、日本語 50 件の記憶エリア eng, jpn を準備する (文字列) ・ ・ ・ 辞書データファイル名を記憶する変数 fname を準備する (文字列) ・ ・ ・ 辞書データ件数を記憶する変数 n を準備する (整数) ・ ・ ・ 辞書データ件数 n を 0 にする ・ ・ ・ 辞書データファイル名をセット ・ ・ ・ 辞書データファイルを読み用として開く ・ ・ ・ データが終わりでない間繰り返す ・ ・ ・ 辞書データ件数 n に 1 プラスする ・ ・ ・ n 件目のデータを読み込む ・ ・ ・ 繰り返し終了 ・ ・ ・ 辞書データファイルを閉じる
---	---

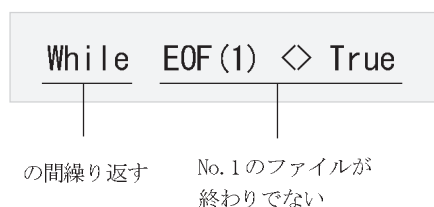
辞書データ読み込みプログラム


EOF 関数について

EOF 関数は、ファイルの現在位置がファイルの末尾に達している場合、真 (True) の値を返し、達していない場合は偽 (False) の値を返します。



上記の繰り返し処理のプログラムは次のような意味を持っています。





④ ツールバーの開始ボタン  を押してみます。

このプログラムはデータを読込んだだけなので、画面上では何も変化が起こりません。エラーメッセージが出ないことだけを確認してください。

このプログラム実行後、各変数には次のデータが入ります。

	eng	jpn		n	
eng(1)	1	badminton	バドミントン	jpn(1)	辞書データの 件数
	2	baseball	野球		
	3	basketball	バスケットボール		
	4	boxing	ボクシング		
	5	football	フットボール		
	6	golf	ゴルフ		
	7	judo	柔道		
	8	ski	スキー		
	9	soccer	サッカー		
	10	swimming	水泳		
	11	tennis	テニス		
eng(12)	12	volleyball	バレーボール	jpn(12)	

⑤ 英単語を入力後、 ボタンを押した時の検索処理をプログラムします。

 ボタンをダブルクリックして、コードエディタウィンドウを開き、以下のプログラムを入力します。

前ページで使用

```
Dim eng(50), jpn(50) As String
Dim fname As String
Dim n As Integer
Dim word As String
Dim ban As Integer
```

繰り返し処理

```
Private Sub Command1_Click()
    word = Text1.Text
    ban = 1

    While (word <> eng(ban)) And (ban <= n)
        ban = ban + 1
    Wend
```

```
    If ban <= n Then
        Label2.Caption = jpn(ban)
    Else
        Label2.Caption = "見つかりません。"
    End If
End Sub
```

- ▪ ▪ キー入力された文字列を記憶する変数 word を準備(文字列)
- ▪ ▪ 検索されたデータ番号を記憶する変数 ban を準備(整数)

- ▪ ▪ Text1に入力された文字列を word に入れる
- ▪ ▪ ban を 1 にセット

- ▪ ▪ 入力された文字列と英単語が一致せず かつ 辞書データの最後まで調べ終わっていない間繰り返す
- ▪ ▪ ban を 1 増やす

- ▪ ▪ n 番以内に見つかったのなら
- ▪ ▪ 検索された番号の日本語を表示する
- ▪ ▪ n 番以内に見つからなかったら
- ▪ ▪ 「見つかりません」と表示

⑥ **終了** ボタンを入力すると終了するようプログラムします。

終了 ボタンをダブルクリックして、コードエディタウィンドウを開き、以下のプログラムを入力します。

```
Private Sub Command2_Click()  
    End      . . .      プログラムの終了  
End Sub
```

⑦ ツールバーの開始ボタン  を押して、プログラムを実行します。


- 英単語を入力して、**検索** ボタンをクリックすると日本語が表示されます。
- 辞書データに無い英単語を入力すると、「見つかりません。」と表示されます。
- **終了** ボタンをクリックするとプログラムが終了します。



検索が成功したとき



検索が失敗したとき

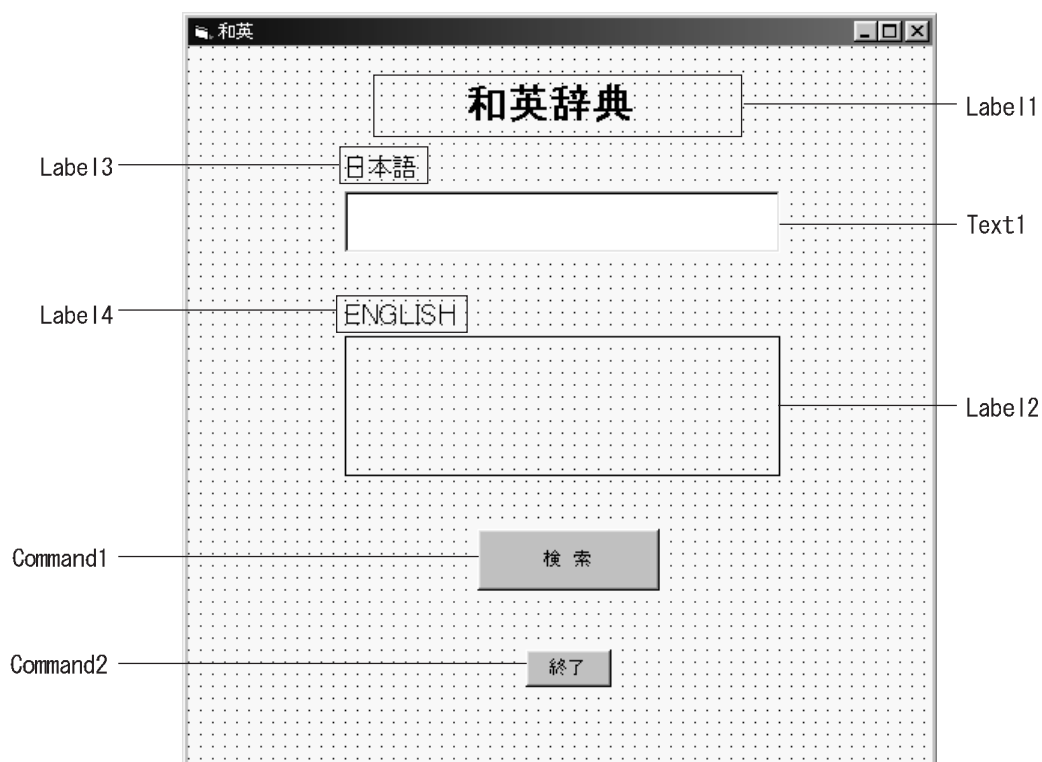
⑧  ボタンをクリックして、フロッピーディスクに上書き保存します。

3. 和英辞典を作ろう

3-1. Formレイアウトを作成しよう

和英辞典の画面を作成しましょう。

- ① 新たに Visual Basic を立ち上げます。
(または **ファイル** - **新しいプロジェクト** を選択)
- ② 英和辞典と同様に、下記のような Form を作成します。



- ③ Form のオブジェクト名、Caption を「和英」に変更します。
- ④ フロッピーディスクにプロジェクトとフォームを「和英」というファイル名で保存します。

3-2. プログラムを考えよう

問題 1

英和辞典のプログラムを参考にして、和英辞典のプログラムを作成しましょう。

ヒント

英和辞典は英単語を入力して、日本語を検索しました。

和英辞典は日本語を入力して、英単語を検索します。

それ以外の、辞書データの読み込み、プログラムの終了などは、全く同じ処理を行います。



検索が成功したとき



検索が失敗したとき

3-3. 部分一致でも検索可能にしよう

文字の一部だけが一致しても検索できるようにしましょう。

例えば、「バレーボール」と入力しても「バレー」と入力しても、「volleyball」と検索されるようにするには下記のようにプログラムを変更します。

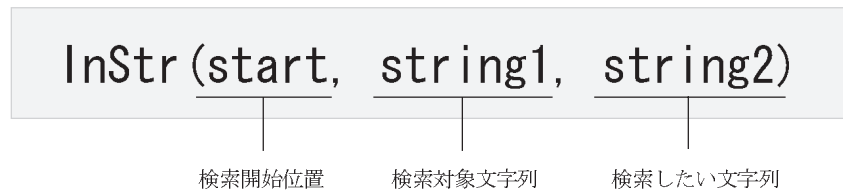
```
Private Sub Command1_Click()  
    word = Text1.Text  
    ban = 1  
    While (word <> jpn(ban)) And (ban <= n)  
        ban = ban + 1  
    Wend  
    If ban <= n Then  
        Label2.Caption = eng(ban)  
    Else  
        Label2.Caption = "見つかりません。"  
    End If  
End Sub
```

↓ 変更

```
Private Sub Command1_Click()  
    word = Text1.Text  
    ban = 1  
    While (InStr(1, jpn(ban), word) = 0) And (ban <= n)  
        ban = ban + 1  
    Wend  
    If ban <= n Then  
        Label2.Caption = eng(ban)  
    Else  
        Label2.Caption = "見つかりません。"  
    End If  
End Sub
```

InStr 関数について

InStr 関数は、ある文字列の中から指定した文字列を検索し、最初に見つかった文字位置を返す文字列処理関数です。

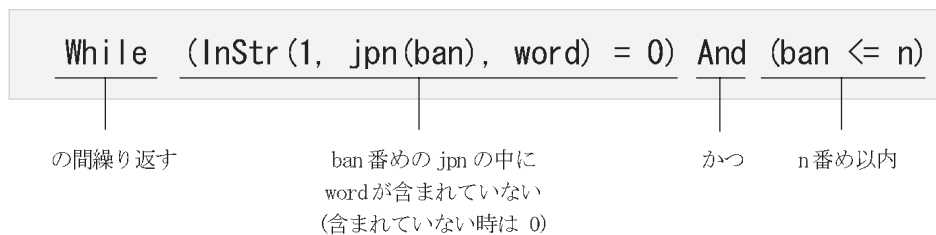


例えば

```
moji1 = " abcdefg"  
moji2 = " cde"  
a = InStr(1, moji1, moji2)      . . .      moji1 の 1 文字めから moji2 を検索する
```

というプログラムを実行すると、**a** には **3** が入ります。
検索できなかった場合は、**a** には **0** が入ります。

前ページの修正部分のプログラムは次のような意味を持っています。



3-4. いくつも検索できるようにしよう

文字列が一致する全ての英単語を検索できるようにしましょう。

例えば、「ボール」で検索すると、日本語の「バスケットボール」「フットボール」「バレーボール」と「ボール」を含む日本語に対応する英単語の「basketball」「football」「volleyball」を検索できるようにするには、下記のようにプログラムを変更します。

```
Private Sub Command1_Click()  
    word = Text1.Text  
    ban = 1  
    Label2.Caption = ""  
    While ban <= n  
        If InStr(1, jpn(ban), word) <> 0 Then  
            Label2.Caption = Label2.Caption & eng(ban) & vbCrLf  
        End If  
        ban = ban + 1  
    Wend  
    If Label2.Caption = "" Then  
        Label2.Caption = "見つかりません。"  
    End If  
End Sub
```

・ ・ ・ 英単語表示部分をクリア
・ ・ ・ n番め以内の間繰り返す
・ ・ ・ ban 番めの jpn の間に word が含まれていたら

既に表示されている英単語 今回検索一致した英単語 改行

・ ・ ・ 結果が何も表示されていないとき



「ボール」で検索したとき

演算子「&」について

演算子「&」は、文字列と文字列をつなぎ合わせる時に使います。例えば

```
moji = " abc" & " efg"
```

というプログラムを実行すると、

moji には " abcdefg" が入ります。

定数「vbCrLf」について

定数「vbCrLf」は、文字列型の変数のようなものですが、何もしなくても最初から値が入っています。

この値は「改行コード」と呼ばれるもので、この定数を表示または印刷すると改行されます。

4. 辞書データ編集画面を作ろう

4-1. Formレイアウトを作成しよう

辞書データの内容を編集する(読込・変更・保存・印刷)画面を作成しましょう。

- ① Visual Basic を立ち上げます。
(または **ファイル** - **新しいプロジェクト** を選択)
- ② 下記のようなFormを作成します。
- ③ Text1のプロパティの **MultiLine** を **True** に、**ScrollBars** を **垂直** に設定します。



- ④ Formのオブジェクト名、Captionを「辞書データ編集」に変更します。
- ⑤ フロッピーディスクにプロジェクトとフォームを「辞書データ編集」というファイル名で保存します。

4-2. 辞書データを読み込んで表示させよう

読込 ボタンを押すと、辞書データを画面に表示するようにしましょう。

- ① まず、辞書データファイルからデータを読み込みます。

問題2

読込 ボタンを押すと、辞書データファイルからデータを読み込むプログラムを作成しましょう。

ヒント

英和辞典や和英辞典にも同じ処理があります。

- ② Text1 に、読込んだデータを表示します。
 - ① のプログラムのデータを読み込んだ直後に下記プログラムを挿入します。

```
Text1.Text = Text1.Text & eng(n) & ", " & jpn(n) & vbCrLf
```

既に表示されている
辞書データ

” 英単語, 日本語” の
形で表示

改行

- ③ プログラムを実行すると、次のように「辞書.dat」のデータが表示されます。



実行結果

- ④ 表示データの再表示処理を追加しましょう。

問題3

読込 ボタンを押すたびに、表示データが1度消えるように下記プログラムを適当な個所に挿入しましょう。

```
Text1.Text = ""
```


4-3. 辞書データを変更して保存しよう

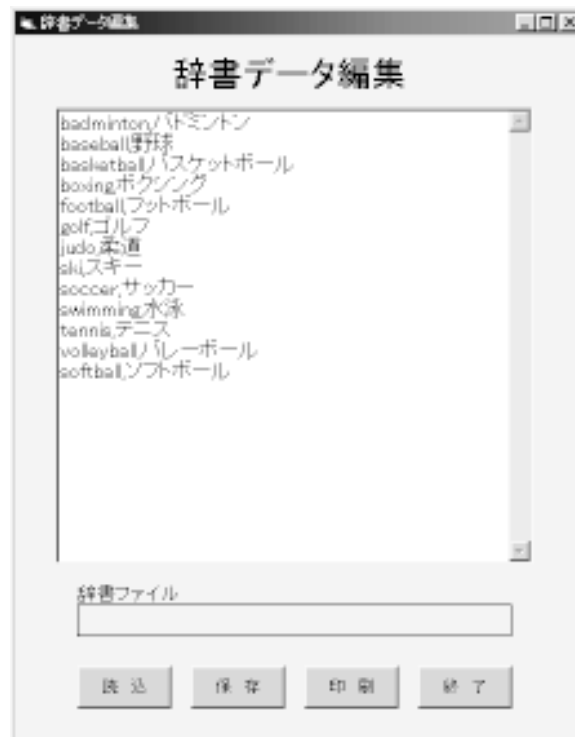
保存 ボタンを押すと、変更した辞書データを保存するようにしましょう。

① **保存** ボタンを押したときに、次のようなプログラムを実行させます。

```
Private Sub Command2_Click()  
    fname = " A:¥辞書.dat"  
    Open fname For Output As #1  
        Print #1, Text1.Text;  
    Close #1  
End Sub
```

- ・ ・ ・ 辞書データファイルを書込み用として開く
- ・ ・ ・ Text1の内容を書込む

② プログラムを実行して実際にデータを追加して保存します。



実行結果

③ プログラムを一度停止し、再度実行して **検索** ボタンを押すと、変更したデータが表示されことを確認してください。


4-4. 辞書データを印刷しよう

 ボタンを押すと、辞書データが印刷されるようにしましょう。

①  ボタンを押したときに、次のようなプログラムを実行させます。

印刷する内容

```
Private Sub Command3_Click()  
    Printer.FontSize = 13          . . .   フォントサイズを13ポイントに  
    Printer.Print "辞書データ"  
    Printer.Print  
    Printer.Print "-----"  
    Printer.Print Text1.Text;  
    Printer.Print "-----"  
    Printer.Print "                written by YAMADA"  
    Printer.EndDoc                . . .   印刷データ終了  
                                     印刷開始  
End Sub
```

② プログラムを実行して、辞書データを読込んだ後、 ボタンを押すと以下のように印刷されます。

```
辞書データ  
  
-----  
badminton, バドミントン  
baseball, 野球  
basketball, バスケットボール  
boxing, ボクシング  
football, フットボール  
golf, ゴルフ  
judo, 柔道  
ski, スキー  
soccer, サッカー  
swimming, 水泳  
tennis, テニス  
volleyball, バレーボール  
softball, ソフトボール  
  
-----  
                written by YAMADA
```

印刷結果

4-5. 辞書データ編集画面を終了しよう

問題4

 ボタンを押すと、プログラムが終了するようにしましょう。

ここまでのプログラム

```
Dim eng(50), jpn(50) As String
Dim fname As String
Dim n As Integer

Private Sub Command1_Click()
    n = 0
    fname = " A:¥辞書.dat"
    Text1.Text = ""
    Open fname For Input As #1
    While EOF(1) <> True
        n = n + 1
        Input #1, eng(n), jpn(n)
        Text1.Text = Text1.Text & eng(n) & "," & jpn(n) & vbCrLf
    Wend
    Close #1
End Sub

Private Sub Command2_Click()
    fname = " A:¥辞書.dat"
    Open fname For Output As #1
    Print #1, Text1.Text;
    Close #1
End Sub

Private Sub Command3_Click()
    Printer.FontSize = 13
    Printer.Print " 辞書データ "
    Printer.Print
    Printer.Print "-----"
    Printer.Print Text1.Text;
    Printer.Print "-----"
    Printer.Print "                               written by YAMADA"
    Printer.EndDoc
End Sub

Private Sub Command4_Click()
    End
End Sub
```

4-6. ファイル名を変更して編集できるようにしよう

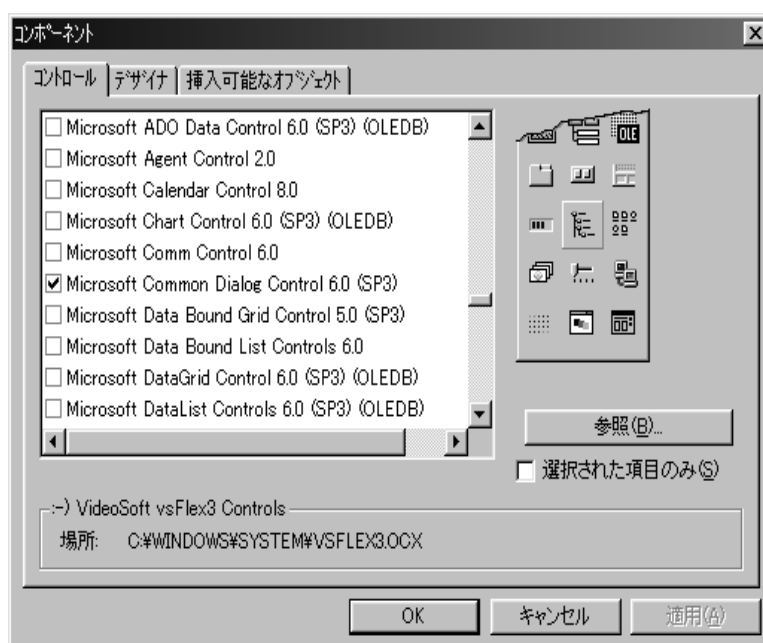
編集するファイル名を自由に指定できるようにしましょう。

- ① メニューバーの **プロジェクト** - **コンポーネント** を選択します。



- ② コントロールの「Microsoft Common Dialog Control6.0」を選択します。

- ③ **OK** をクリックするとツールボックスにCommonDialogボタンが追加されます。

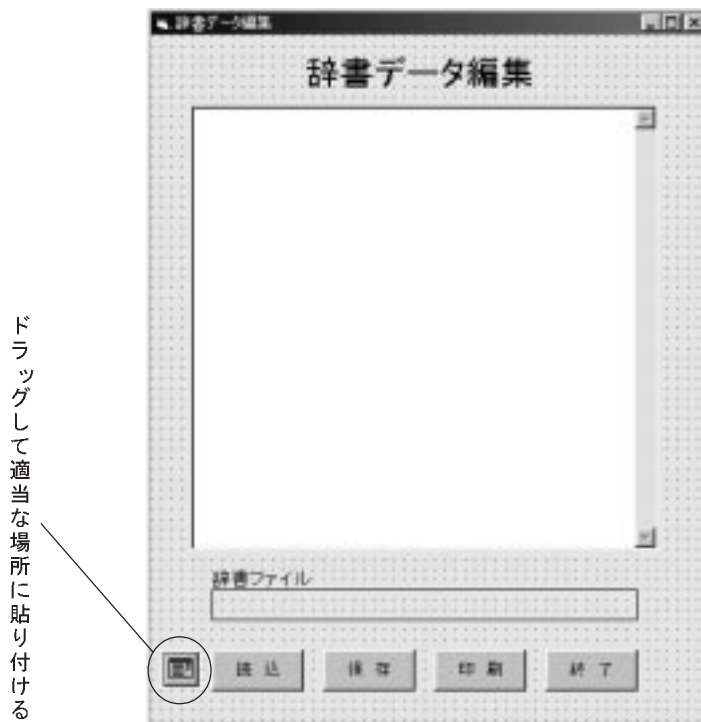



OK をクリック
→



追加
され
ます

- ④ ツールボックスの CommonDialog ボタン  をクリック後、Form 内の適当な場所でドラッグし、CommonDialog1 を作成します。



- ⑤  ボタンをクリックしたときのプログラムを次のように変更します。

```
Private Sub Command1_Click()
```

```
    n = 0
```

```
    CommonDialog1.ShowOpen
```

- ・ ・ ・ 読み込み用ダイアログボックスを表示する

```
    fname = CommonDialog1.FileName
```

- ・ ・ ・ ダイアログボックスで指定したファイル名を fname とする

```
    Label2.Caption = fname
```

- ・ ・ ・ fname を Label2 に表示する

```
    Text1.Text = ""
```

```
    Open fname For Input As #1
```

```
    While EOF(1) <> True
```

```
        n = n + 1
```

```
        Input #1, eng(n), jpn(n)
```

```
        Text1.Text = Text1.Text & eng(n) & ", " & jpn(n) & vbCrLf
```

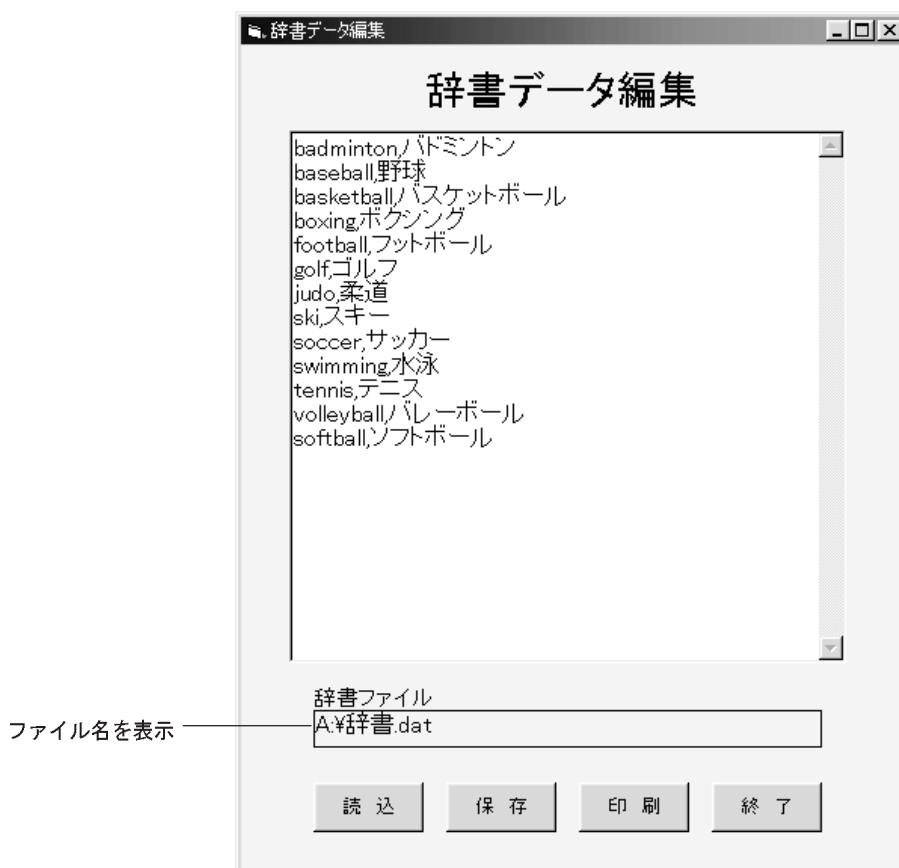
```
    Wend
```

```
    Close #1
```

```
End Sub
```

⑥ プログラムを実行します。

読込 ボタンをクリックすると読込み用のダイアログボックスが表示され、ファイル名を指定すると、そのファイルに保存されているデータが Text1 に表示されます。また、指定したファイル名も表示されます。



- ⑦ **保存** ボタンをクリックしたときに、保存用のダイアログボックスが表示されるようにするにはプログラムを次のように変更します。

```
Private Sub Command2_Click()
    CommonDialog1.ShowSave          . . . 保存用ダイアログボックスを
                                     表示する
    fname = CommonDialog1.FileName  . . . ダイアログボックスで指定した
                                     ファイル名を fname とする
    Label2.Caption = fname          . . . fname を Label2 に表示する

    Open fname For Output As #1
        Print #1, Text1.Text;
    Close #1

End Sub
```

- ⑧ プログラムを実行します。

読み込 ボタンをクリックしてファイルを読み込み、データを変更して **保存** ボタンをクリックすると、保存用のダイアログボックスが表示され、指定したファイル名で保存されます。また、保存したファイル名が表示されます。



ファイル名を表示

⑨ ファイル名を印刷します。

問題5

印刷内容にファイル名を追加しましょう。

辞書データ
ファイル名 : A:¥辞書.dat

badminton, バドミントン
baseball, 野球
basketball, バスケットボール
boxing, ボクシング
football, フットボール
golf, ゴルフ
judo, 柔道
ski, スキー
soccer, サッカー
swimming, 水泳
tennis, テニス
volleyball, バレーボール
softball, ソフトボール

written by ○○○

追加

印刷結果

完成プログラム

```
Dim eng(50), jpn(50) As String
Dim fname As String
Dim n As Integer

Private Sub Command1_Click()
    n = 0
    CommonDialog1.ShowOpen
    fname = CommonDialog1.FileName
    Label2.Caption = fname
    Text1.Text = ""
    Open fname For Input As #1
    While EOF(1) <> True
        n = n + 1
        Input #1, eng(n), jpn(n)
        Text1.Text = Text1.Text & eng(n) & ", " & jpn(n) & vbCrLf
    Wend
    Close #1
End Sub

Private Sub Command2_Click()
    CommonDialog1.ShowSave
    fname = CommonDialog1.FileName
    Label2.Caption = fname
    Open fname For Output As #1
        Print #1, Text1.Text;
    Close #1
End Sub

Private Sub Command3_Click()
    Printer.FontSize = 13
    Printer.Print "辞書データ"
    Printer.Print
    Printer.Print "ファイル名 : " & Label2.Caption
    Printer.Print "-----"
    Printer.Print Text1.Text;
    Printer.Print "-----"
    Printer.Print "                               written by ○○○"
    Printer.EndDoc
End Sub

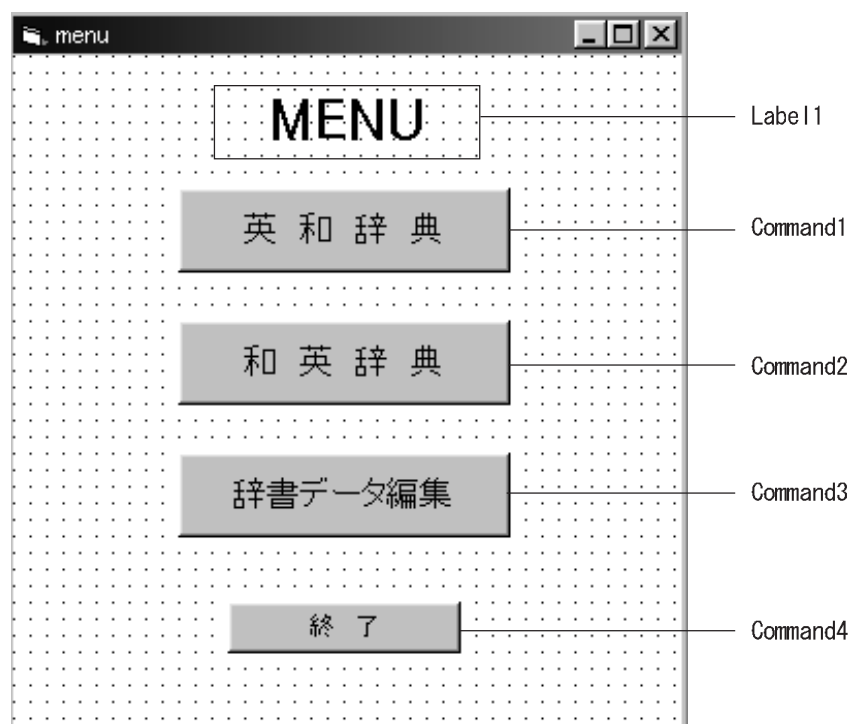
Private Sub Command4_Click()
    End
End Sub
```

5. 3つのプログラムをメニューから操作しよう

5-1. メニュー画面を作成しよう

英和辞典・和英辞典・辞書データ編集を呼び出すメニュー画面を作成しましょう。

- ① Visual Basic を立ち上げます。
(または `ファイル` - `新しいプロジェクト` を選択)
- ② 下記のようなFormを作成します。



- ③ Form のオブジェクト名、Caption を「menu」に変更します。
- ④ フロッピーディスクにプロジェクトとフォームを「menu」というファイル名で保存します。

5-2. 3つのFormを追加しよう

menuに「英和」「和英」「辞書データ編集」の3つのFormを読み込みましょう。

- ① プロジェクト - フォームモジュールの追加 - 既存のファイル を選択して、「英和.frm」「和英.frm」「辞書データ編集.frm」を追加します。



プロジェクトエクスプローラの表示

- ② プロジェクト - Project1のプロパティ - スタートアップの設定 で、最初に実行されるFormを「menu」に設定します。



5-3. それぞれを呼び出すプログラムを作成しよう

各ボタンを押すとそれぞれのFormが実行されるようにプログラムしましょう。


① それぞれの Command ボタンの処理として次のようなプログラムを作成します。

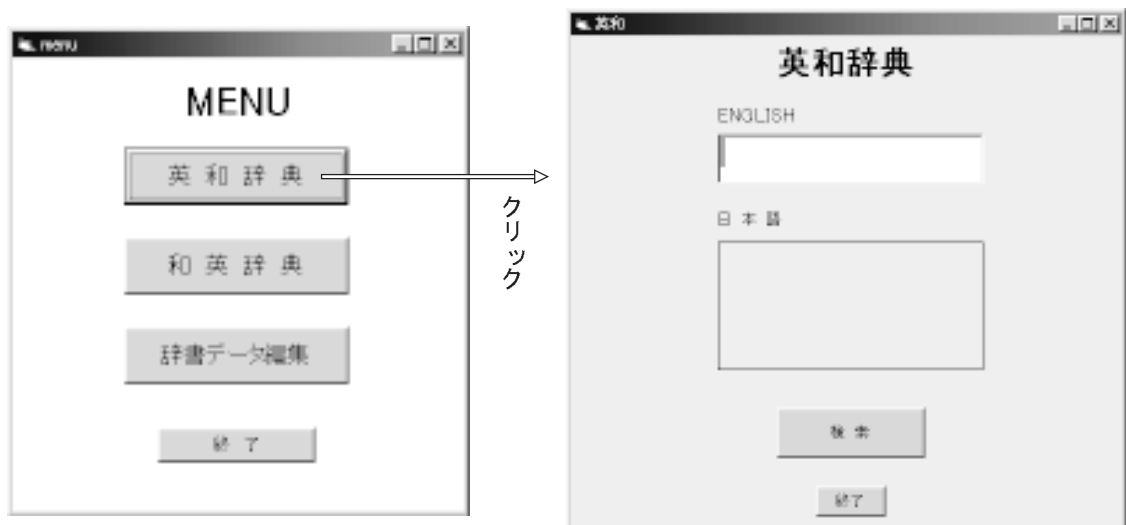
```
Private Sub Command1_Click()  
    Load 英和  
    英和.Show  
End Sub  
  
Private Sub Command2_Click()  
    Load 和英  
    和英.Show  
End Sub  
  
Private Sub Command3_Click()  
    Load 辞書データ編集  
    辞書データ編集.Show  
End Sub  
  
Private Sub Command4_Click()  
    End  
End Sub
```

・・・ Form「英和」を呼出す
・・・ Form「英和」を表示する
・・・ プログラムの終了

② プログラムを実行します。

menu で実行したい処理のボタンをクリックするとその画面が表示されます。

 を押すとプログラムが終了します。



6. システムをグレードアップさせよう

6-1. 各Formが終了してもメニューは残るようにしよう

「End」命令はプロジェクト全体を終了します。

そのFormだけを終了したい場合は「Unload」命令を使用します。

Form「英和」を終了させる場合

```
Private Sub Command2_Click()  
    Unload 英和  
End Sub
```

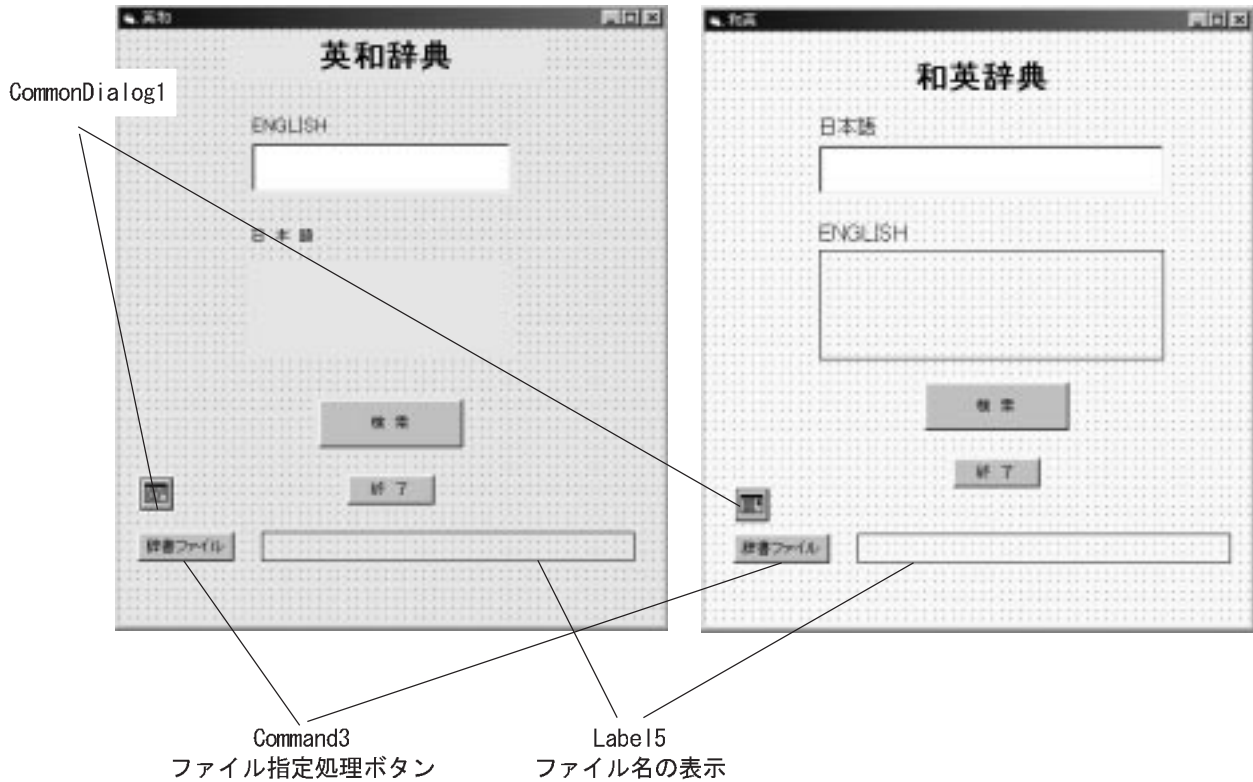
問題6

英和辞典・和英辞典・辞書データ編集のそれぞれを終了してもメニュー画面が残るようにしましょう。

6-2. 辞書ファイルを指定できるようにしよう

英和辞典、和英辞典の辞書ファイルを指定できるようにしましょう。

- ① CommonDialog、Command、Label を両Formに追加します。



- ② Form_Activate のプログラムを全て削除します。
③ Command3_Click() に以下のプログラムを作成します。

```
Private Sub Command3_Click()  
    n = 0  
    CommonDialog1.ShowOpen  
    fname = CommonDialog1.FileName  
    Label15.Caption = fname  
    Open fname For Input As #1  
    While EOF(1) <> True  
        n = n + 1  
        Input #1, eng(n), jpn(n)  
    Wend  
    Close #1  
End Sub
```

④ プログラムを実行します。

辞書ファイル ボタンをクリックすると読み込みのダイアログボックスが表示されます。

辞書 ボタンをクリックするとダイアログボックスで指定したファイルからデータを検索します。

