

## JRA-VAN 競馬ソフト作成体験教室

### Lesson-5：コード変換を行う

当コーナーでは、Microsoft Visual Basic 2005 Express Edition(以下 VB 2005 と省略)で「JRA-VAN Data Lab.」サービス対応の競馬ソフトを作成していく過程をステップアップ形式で解説していきます。

前回までで、実際に JV-Data をダウンロードして内容を表示する簡単な仕組みを実装しました。今回は、JRA-VAN で提供されているプログラミングパーツを利用してコード変換を行い、取得した JV-Data を見て分かる情報として表示してみましょう。

#### 【 今回の目標 】

JRA-VAN で提供されているプログラミングパーツ「コード変換クラス」を利用して、取得した JV-Data をコード変換して表示する。

（ 具体的には、取得した情報の中の競馬場コードを競馬場名に変換して表示エリアに表示します。 ）

JV-Link に実装されている以下のメソッドの使い方を理解する。

- ・JVSkip メソッドで不要なデータを読み飛ばす。

#### 【 やってみよう 】

JRA-VAN Data Lab.サイトにアクセスし、「ソフト開発コーナー」「プログラミングパーツ提供コーナー」から、コード変換クラスの中の「Microsoft Visual Basic .NET」版をダウンロードし、解凍しておきます。

#### [ ワンポイントメモ ]

ダウンロードファイルを解凍した際に生成される「README.TXT」は、必ずご一読ください。

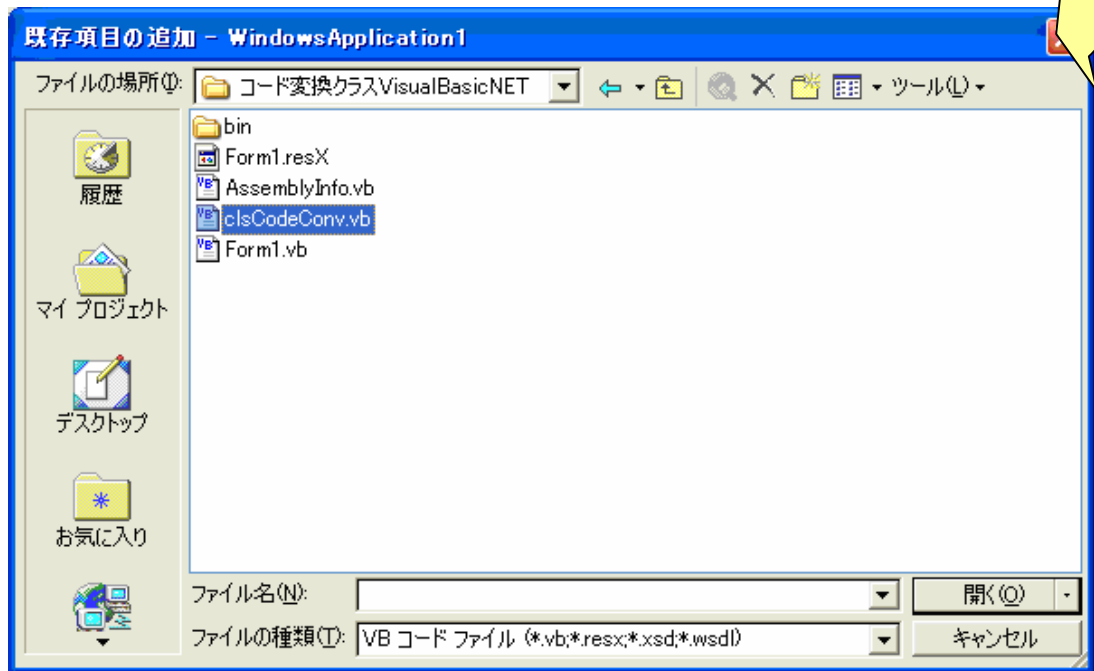
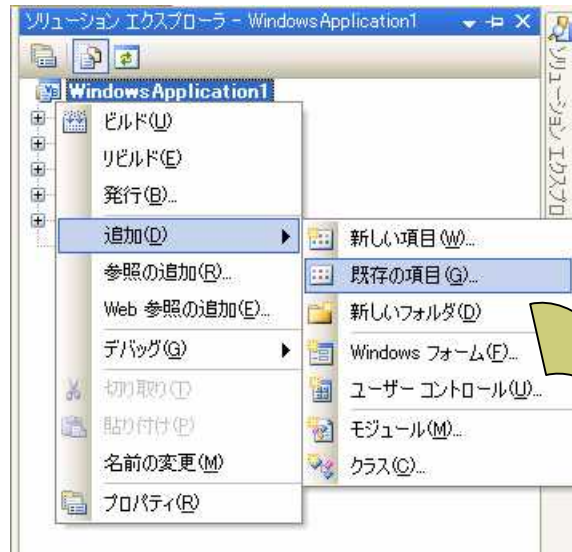
Lesson-4 までを実装したフォーム(frmMain)を含むプロジェクトを開きます。

( 前回のレッスンの続きから開始するのであれば、この作業は必要ありません。 )

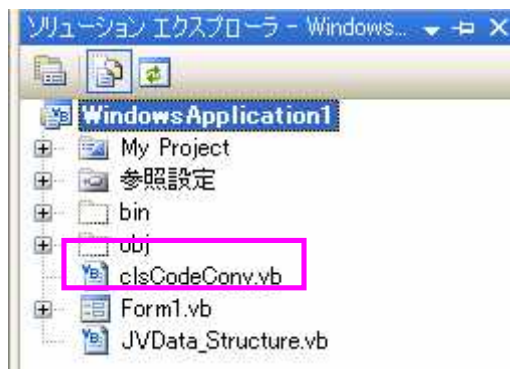
で解凍したフォルダーの中から「CodeTable.csv」を実行モジュールと同じフォルダーにコピーします。

( 例：¥Visual Studio Projects¥WindowsApplication1¥bin¥Release )

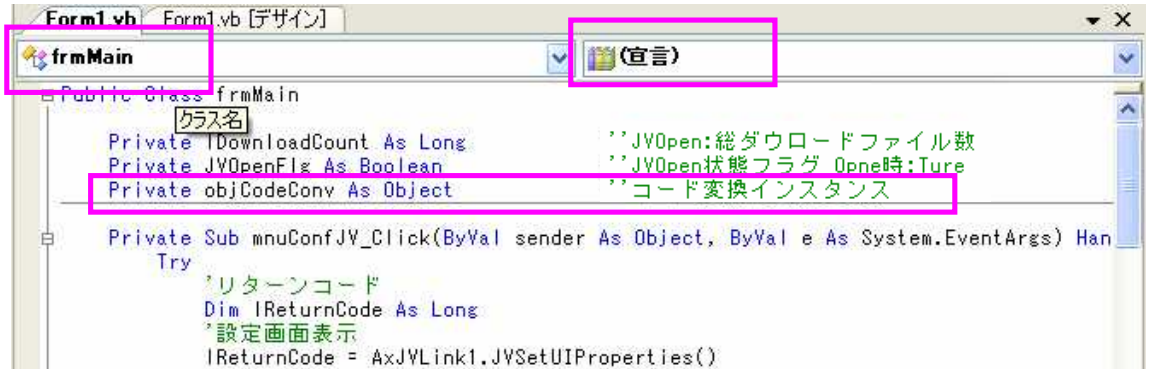
ソリューションエクスプローラーより  
右クリックにて「追加(D)」 「既存の  
項目(G)」を選択してダイアログを  
表示します。このダイアログで「  
コード変換クラス」の解凍後ディレ  
クトリの中の「コード変換クラス  
VisualBasicNET」にある「clsCode  
Conv.vb」を選択して「開く」  
をクリックします。



ソリューションエクスプローラーに「clsCodeConv.vb」が追加されます。



コードエディタウィンドウを開き、(全般)-(宣言)でコード変換インスタンスを宣言します。



```
Private objCodeConv As Object          ''コード変換インスタンス
```

[ソースコード 007-01]

コードエディタウィンドウで、frmMain\_Load メソッドに以下のコードを追加します。(灰色部分は前回コーディング済み)

```
Private Sub frmMain_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Load
    Dim sid As String
    Dim IReturnCode As Long

    '引数設定
    sid = "Test"

    'JVLink初期化
    IReturnCode = Me.AxJVLink1.JVInit(sid)

    'エラー判定
    If IReturnCode <> 0 Then
        MsgBox("JVInitエラー コード:" & IReturnCode & ":", MessageBoxButtons.Error)
        Me.Cursor = System.Windows.Forms.Cursors.Default
        Exit Sub
    End If

    'コード変換インスタンス生成
    objCodeConv = New clsCodeConv
    'パスを指定し、コードファイルを読み込む
    objCodeConv.FileName = Application.StartupPath & "%CodeTable.csv"

End Sub
```

[ソースコード 007-02]

このコードは、プログラム起動時にコード変換クラスのインスタンスを生成し、コード表を読み込む処理を行います。

JV-Data 読込時の処理に修正を加えます。

```

(前省略)
Select Case IReturnCode
  Case 0 ' 全ファイル読み込み終了
    prgJVRead.Value = prgJVRead.Maximum ' 進捗表示
    Exit Do
  Case -1 ' ファイル切り替わり
    prgJVRead.Value = prgJVRead.Value + 1
  Case -3 ' ダウンロード中
  Case -201 ' Initされてない
    MsgBox("JVInitが行われていません。")
    Exit Do
  Case -203 ' Openされてない
    MsgBox("JVOpenが行われていません。")
    Exit Do
  Case -503 ' ファイルがない
    MsgBox(strFileName & "が存在しません。")
    Exit Do
  Case Is > 0 ' 正常読み込み
    ' レコード種別IDの識別
    If Mid(strBuff, 1, 2) = "RA" Then
      ' レース詳細のみ処理
      ' レース詳細構造体への展開
      RaceInfo.SetData(strBuff)

      ' データ表示
      rtbData.AppendText( _
        "年:" & RaceInfo.id.Year & _
        " 月日:" & RaceInfo.id.MonthDay & _
        " 場:" & objCodeConv.GetCodeName("2001", RaceInfo.id.JyoCD, "3") & _
        " 回次:" & RaceInfo.id.Kaiji & _
        " 日次:" & RaceInfo.id.Nichiji & _
        " R:" & RaceInfo.id.RaceNum & _
        " レース名:" & RaceInfo.RaceInfo.Ryakusyo10 & vbCrLf)
    Else
      ' レース詳細以外は読み飛ばす
      Call AxJVLink1.JVSkip()
    End If
End Select
(後省略)

```

[ソースコード 007-03]

#### ・ Point1

取得した場コードに対応する競馬場名（2文字略称）を表示するために、メソッド `GetCodeName` を使用します。引数には、競馬場コードに対応するコード番号(2001)とコード、および表示する列(「CodeTable.csv」の何列目に対応させるか)を指定します。「CodeTable.csv」では、行ごとに「コード番号,コード,列 1,列 2,...」となっており、このファイルを修正することにより変換後の文字列を設定することができます。

なお、コード番号については、JV-Data 仕様書をご覧ください。

・ Point2

JVSkip メソッドを使用し、レース詳細(RA)以外のレコードはまとめて読み飛ばします。詳しくは、JV-Link インターフェース仕様書をご覧ください。

【 確認しよう 】

それでは、実際に動かしてみましょ。

メニューから「実行」「開始」を選択し、プログラムを実行します。

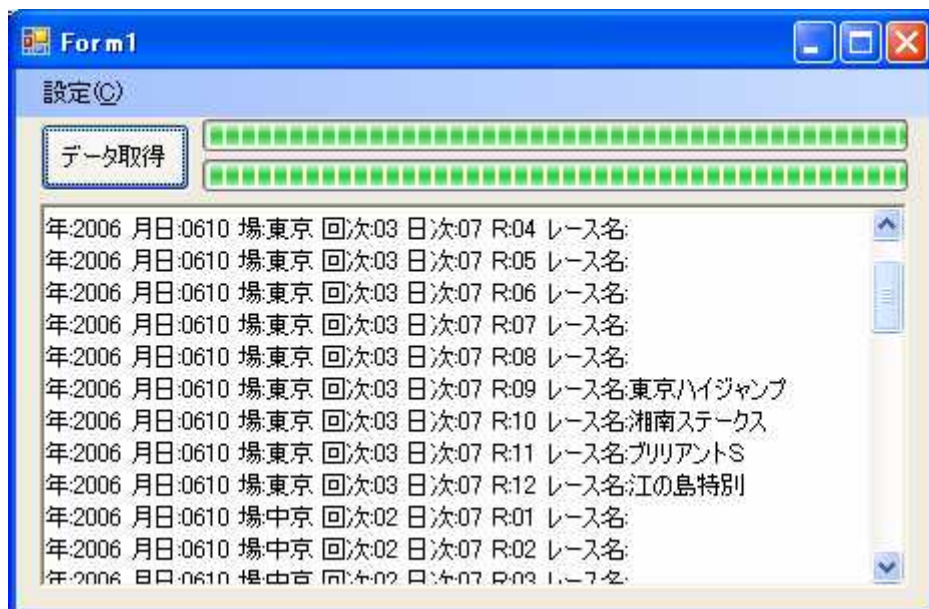
JV-Link を使用するための「サービスキー」を設定していない場合は、メインフォーム上のメニューから「設定」「JV-Link の設定」を選択し、「サービスキー」を設定します（詳細は Lesson-1 を参照）。

フォームの「データ取得」ボタンをクリックすると、メッセージボックスがポップアップします。メッセージボックスには、リターンコードや読み込みファイル数、ダウンロードファイル数、最新読み込みファイルのタイムスタンプが表示されます。

これらの数値は、実行するタイミングによって異なります。詳しくは、Lesson-2 をご覧下さい。

メッセージボックスを「OK」で閉じると、レース詳細情報のうち「開催年、開催月日、競馬場名、開催回、開催日目、レース番号、競走名略称 10 文字」が表示されます。

JV-Data のダウンロード進捗状況や読み込み進捗状況は上下のプログレスバーで確認することができます。



[ ワンポイントメモ ]

メソッド GetCodeName の第 3 引数を変化させて、競馬場名を色々な形式（1 文字略

称や正式名称など)で表示してみましょう。