



第四回 「Windows ライクなプログラムを作ってみよう」

皆さんこんにちは、今日はいつまでも黒い窓ではつまらない背景のが白く、ボタン操作などで制御するWindowsらしいGUI(Graphical User Interface)にしよう！ ということでリソースと拡張ライブラリを使ってプログラムを作ってみましょう。皆さんの今お持ちのプログラムもリソースを使えばちょっと手を加えるだけでWindowsらしいGUIのプログラムになります。さらに今回ご紹介するプログラムは、F-BASIC97から新しく追加された関数・拡張ライブラリを主に使って作っていきたいと思います。それでは、まずF-BASIC97で新しく追加された関数・拡張ライブラリを下の表にまとめてみましょう。

新関数	機能
DSKF 関数	指定したディスクの未使用の領域の大きさを返します。単位はキロバイト(KB)です。
TODAY\$関数	内蔵タイマーのしめす現在の日付を yyyy/mm/dd の形式で返します。(yyyy:年、mm:月、dd:日)
MILLITIME 関数	00:00:00 から現在までの 1/1000 秒刻みでの時間を返します。

新拡張ライブラリ	機能	オブジェクト
GETMODIFY	エディットのモディフィケーションフラグの状態を返します。	エディットコントロール
GETTOPINDEX	リストボックスの表示領域の先頭に表示されている項目の番号を返します。	リストボックス、コンボボックス
LINESCROLL	エディットコントロールの表示内容をスクロールします。スクロール量を桁数、行数で指定します。	エディットコントロール
REPLACESEL	エディットコントロール上で選択されている文字列を別の文字列で置き換えます。文字列が選択	エディットコントロール

	されていない場合はカーソルの位置に挿入します。	
SETFOREGROUNDWINDOW	指定されたフォームを前面に移動します。	フォーム
SETMODIFY	エディットコントロールのモディフィケーションフラグを設定します。	エディットコントロール
SETTOPINDEX	指定した項目がリストボックスの表示領域の先頭に表示されるように、リストボックスの表示内容をスクロールします。	リストボックス、コンボボックス
SETTOPMOSTWINDOW	フォームを常に一番手前に表示するようにします。	フォーム



第四回 「Windows ライクなプログラムを作ってみよう」

さて、新しく追加された関数・拡張ライブラリがわかりましたので、それを使ってプログラムを作ってみましょう。このプログラムは、現在のディスクの空き容量を表示するプログラムです。最近は 1GB, 2GB なんて当たり前の時代になってしまったのであまり空き容量を気にしないかもしれません、自分でファイルを作るときには一応空き容量をチェックする必要がありますよね。このように、自分の使っているディスクの空き容量を調べてみましょう。

おっと、忘れてはいけないのが、今回の目的は「Windows ライクなプログラムを作る」ですから、ただ新しい関数を使うだけではありません、リソースファイルを使用します。 といっても、身構える必要はありません。現に第一回の OnePointLesson で、すでにリソースファイルを使っているのですから。

では、始めるとしましょう。

【手順 1】

F-BASIC を起動しましょう。

メニューの[プロジェクト] - [ウィザード(W)...]を選びます。 そうすると図 1-1 のようなダイアログが現れます。それぞれの項目で好きな名前をつけてください。 このとき、ファイルの拡張子は変えないようにしてください。

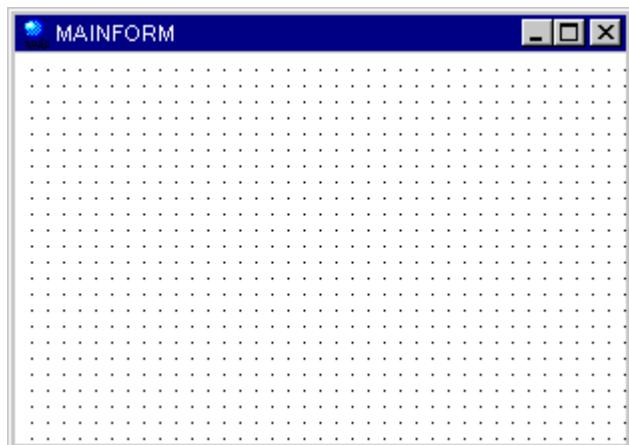
そうすると、必要なファイルが作られます。

その中から、*.RC という拡張子のタイトルが書いてあるウィンドウから MAINFORM というファイルをダブルクリックして開いて下さい。図 1-2 のような画面が出てきましたよね。

これが、フォームです。 このフォームを使うことによって、Windows らしいプログラムを簡単に作ることが可能なのです。



(図 1-1)



(図 1-2) Mainform

One Point Lesson

第四回 「Windows ライクなプログラムを作ってみよう」

【手順 2】

さて、外観を作ることにしましょう。Mainform(フォーム)にいろいろと『コントロール』を貼っていきます。今回貼るコントロールとそれらの変更点は次のとおりです。赤の枠の部分をプロパティウィンドウをつかって変更して下さい。

エディットコントロールは貼りつけるだけです。メニューについては[手順 3]で説明します。

コントロール ID	MAINFORM
表示内容	フレーム
テキスト	DiskFreeSpace
サイズ(幅)	250
サイズ(高さ)	150
コントロール ID	COMBO1
表示内容	スタイル
種類	ドロップダウンリスト
ソート	あり
コントロール ID	TEXT1
表示内容	スタイル
テキスト	ドライブ
3D 表示	あり
コントロール ID	TEXT2
表示内容	スタイル
テキスト	MB
コントロール ID	TEXT3
表示内容	スタイル
テキスト	日付

①フォーム

メニュー

③テキスト

②コンボボックス

エディットコントロール

⑤テキスト

⑥テキスト

⑦テキスト

④テキスト

?	コントロール ID	TEXT4
?	コントロール ID	TEXT5
	表示内容	スタイル
	テキスト	現在



第四回 「Windows ライクなプログラムを作ってみよう」

【手順 3】

さて、必要なコントロールは全て貼りました。次はメニューを作りましょう。このメニューをつけることでより一層 Windows プログラムらしくなりますね。

といっても特に難しい事はありません。F-BASIC97 のメニューから「リソース」 - 「新規作成 (N)...」を選ぶとリソース新規作成ダイアログが出てきます。メニューを選んで OK ボタンを押してください。そうすると、図 3-1 のダイアログが出てきますので、これを使ってメニューを作ります。

まずは、いつも見えているメニュー項目から作ります。「項目名」のところに「メニュー」と書きます。「コントロール ID」も書きます(ここでは MENU とします)。最後に「アクセスキー」です。これは、Alt キー + 何らかのキーでメニューが開くようにするために設定するものです。ここでは、「N」としておきます。

さて、今度は「メニュー」項目の中に入れるものを作ります。「***」をクリックして入力できる状態にします。新しいメニュー項目を作る時は「***」をクリックすると入力できる状態になります。そこに、同じような要領で「項目名」 - 「終了」、「コントロール ID」 - 「EXIT」、「アクセスキー」 - 「E」と入力します。

この項目で忘れてはいけないのが、「? (R)」ボタンです。これを押すことによって「メニュー」という項目の中に「終了」という項目が入ります。これで、「メニュー」項目をクリックするとプルダウンメニューが出てきて「終了」が表示されます。

これで、メニューが完成しました。OK ボタンを押せばリソースファイルに自動的に登録されます。



(図 3-1)メニュー編集ダイアログ

One Point Lesson

第四回 「Windows ライクなプログラムを作ってみよう」

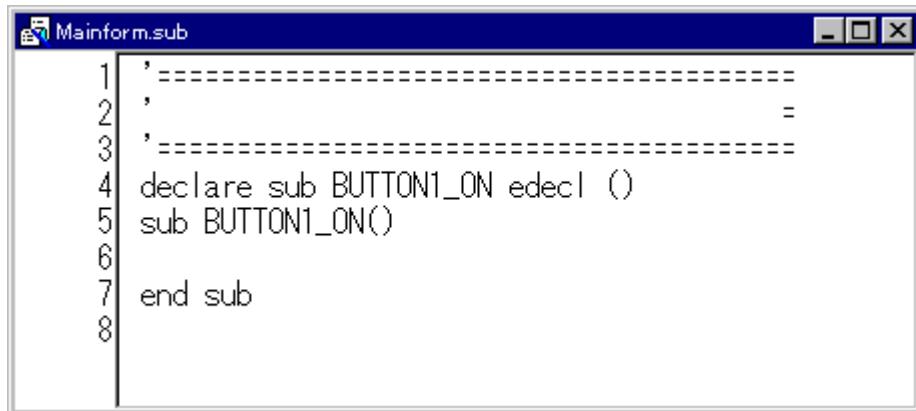
【手順 4】

さていよいよプログラム本体の方に取り掛かりましょう。

プログラムは下のようなものになります(リスト 1)。これ全部打つの大変ですよね。

そこで、F-BASIC ではプログラムの決まりごとはリーソースファイルをダブルクリックするだけだそのコントロールに関係する部分の雛形を自動生成します(図 4-1)。この雛形の中にコントロールがクリックされた時の処理を書けば良い訳です。このままサブプログラムとして使ってもいいですし、今回の様に本体のプログラムにきりは利せしてくれつけでも構いません。

初期化処理の部分はリソースを使う時には必要な拡張ライブラリです。コントロールとプログラムを結びつけるものとなります。



```
1 '=====
2 ' =====
3 '=====
4 declare sub BUTTON1_ON edecl ()
5 sub BUTTON1_ON()
6
7 end sub
8
```

(図 4-1)BUTTON1 をダブルクリックした時の雛形

(リスト 1)ドライブ空き容量測定・プログラム

#include "windows.bi" '拡張プログラムをする時には必ず必要です。

```
'初期化処理
var shared COMB01 as object      'コントロールを扱えるようにします。
var shared EDIT1 as object
var shared TEXT3 as object
var shared TEXT4 as object
```

```

var FORM1 as object

COMBO1.ATTACH GETDLGITEM("COMBO1")
EDIT1.ATTACH GETDLGITEM("EDIT1")
COMBO1.SETDRIVE -1
TEXT3.ATTACH GETDLGITEM("TEXT3")
TEXT4.ATTACH GETDLGITEM("TEXT4")
FORM1.ATTACH GETHWND
SETTOPMOSTWINDOW (-1)           '必ず一番上に出ているようにする。

while 1
    WAITEVENT                  '操作(イベント)を待つ
wend

'=====
' COMBO1 が変更された時の処理
'=====

declare sub COMBO1_CHANGE edecl ()
sub COMBO1_CHANGE()
    DRV$_ = mid$(COMBO1.GETWINDOWTEXT,3,1)      'コンボボックスの文字列を獲得
    DRV$ = DRV$_+":$"                         '獲得した文字列からドライブ名を獲得
    F = dskf(DRV$)/1024                      'ドライブの空き容量を求める
    D$ = today$                                '日付を獲得
    T$ = time$                                  '時間を獲得
    EDIT1.SETWINDOWTEXT str$(F)                 'エディットに空き容量を書き込む
    TEXT3.SETWINDOWTEXT D$
    TEXT4.SETWINDOWTEXT T$
end sub

'=====
' メニューの「終了」が選ばれた時の処理(終了処理)
'=====

declare sub EXIT_ON edecl()
sub EXIT_ON()
    end
end sub

```

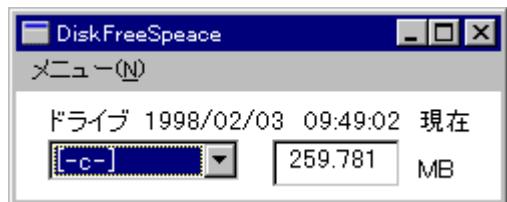


第四回 「Windows ライクなプログラムを作ってみよう」

【手順 5】

これで、後は翻訳をかけるだけです。これで立派な Windows プログラミングができあがりました。また今回紹介した、プログラムをダウンロードできるようにしてありますのでご利用ください。今回は、新しいことがたくさん出てきたので少しばかり難しかったかもしれませんがあくまで少しづつ理解していってください。

Windows プログラムは今までの BASIC プログラムと違いイベント駆動型のプログラムです。すぐには理解できないかもしれません。しかし、皆さんなら必ず理解できます。F-BASIC を使ってどんどん Windows プログラムに挑戦してみてください。



ソフト完成図