

OpenOffice.org Basic について (第1回) - ゼミ資料

張 梁

2007年9月28日

1 基本統計量計算フォームの改良 (VBA)

1.1 Sub プロシージャと Function プロシージャ

プロシージャ Excel でいうマクロのことを、VBA では「プロシージャ」と呼ぶ。

Sub プロシージャは主に処理を実行する、Function プロシージャは処理した結果の値を返す機能があり、Excel のワークシートで使う関数のように扱うことができる。

1.2 プログラムリスト

1.2.1 モジュール

```
Sub CreateUserform()  
    '*****  
    ' 選択フォームを表示  
    '*****  
    On Error Resume Next  
    基本統計量選択フォーム.Show  
  
End Sub
```

1.2.2 フォーム

```
Private Sub CommandButton1_Click()  
    '*****  
    ' 実行ボタン  
    '*****  
  
    If RefEdit1.Value = "" Then ' 選択した範囲は空白だったら  
        MsgBox "データを選択してください" ,  
        RefEdit1.SetFocus  
    Else  
  
        data = RefEdit1.Value  
  
    End If  
    '*****
```

```

',      結果表示シートの作成
'*****

Dim cnt As Integer
cnt = Worksheets.Count

MsgBox data
' 一番右に作業用新しいシートを作成する
Worksheets.Add After:=Worksheets(cnt), Count:=1
ActiveSheet.Name = "計算結果"
Range("計算結果!A1").Value = "基本統計量"
Call Rinterface.StartRServer
Call Rinterface.PutDataframe("inputdata", Range(data),
WithRowNames:=False, RespectHidden:=False)
Call Rinterface.RRun("nrowname<-names(inputdata[, -1])")
Call Rinterface.RRun("mnrowname<-matrix(nrowname, nrow=1)")
Call Rinterface.GetArray("mnrowname", Range("計算結果!B1"))
Call Rinterface.StopRServer

If CheckBox1.Value = True Then
    Call 平均値 (data)
End If

If CheckBox2.Value = True Then
    Call 中央値 (data)
End If

If CheckBox3.Value = True Then
    Call 最小値最大値 (data)
End If

If CheckBox4.Value = True Then
    Call 分散標準偏差 (data)
End If

Unload 基本統計量選択フォーム

End Sub

Private Sub CommandButton2_Click()
'*****
'      キャンセルボタン
'*****

On Error Resume Next
Unload 基本統計量選択フォーム

End Sub

Private Sub UserForm_Click()

End Sub

Function 平均値 (data)

```

```

    Call Rinterface.StartRServer
    Call Rinterface.PutDataframe("inputdata", Range(data),
WithRowNames:=False, RespectHidden:=False)
    Call Rinterface.RRun("inputdataMean<-apply(inputdata[, -1], 2, mean)")
    Call Rinterface.RRun("vinputdataMean<-as.vector(inputdataMean)")
    Call Rinterface.RRun("mvinputdataMean<-matrix(vinputdataMean, nrow=1)")
    Range("計算結果!A2").Value = "平均値："
    Call Rinterface.GetArray("mvinputdataMean", Range("計算結果!B2"))
    Call Rinterface.StopRServer

End Function

Function 中央値 (data)
    Call Rinterface.StartRServer
    Call Rinterface.PutDataframe("inputdata", Range(data),
WithRowNames:=False, RespectHidden:=False)
    Call Rinterface.RRun("inputdataMedian<-apply(inputdata[, -1], 2, median)")
    Call Rinterface.RRun("vinputdataMedian<-as.vector(inputdataMedian)")
    Range("計算結果!A3").Value = "中央値："
    Call Rinterface.RRun("mvinputdataMedian<-matrix(vinputdataMedian, nrow=1)")
    Call Rinterface.GetArray("mvinputdataMedian", Range("計算結果!B3"))
    Call Rinterface.StopRServer

End Function

Function 最小値最大値 (data)
    Call Rinterface.StartRServer
    Call Rinterface.PutDataframe("inputdata", Range(data),
WithRowNames:=False, RespectHidden:=False)
    Call Rinterface.RRun("inputdataMax<-apply(inputdata[, -1], 2, max)")
    Call Rinterface.RRun("vinputdataMax<-as.vector(inputdataMax)")
    Call Rinterface.RRun("mvinputdataMax<-matrix(vinputdataMax, nrow=1)")
    Call Rinterface.RRun("inputdataMin<-apply(inputdata[, -1], 2, min)")
    Call Rinterface.RRun("vinputdataMin<-as.vector(inputdataMin)")
    Call Rinterface.RRun("mvinputdataMin<-matrix(vinputdataMin, nrow=1)")
    Range("計算結果!A4").Value = "最小値："
    Call Rinterface.GetArray("mvinputdataMax", Range("計算結果!B4"))
    Range("計算結果!A5").Value = "最大値："
    Call Rinterface.GetArray("mvinputdataMin", Range("計算結果!B5"))
    Call Rinterface.StopRServer

End Function

Function 分散標準偏差 (data)
    Call Rinterface.StartRServer
    Call Rinterface.PutDataframe("inputdata", Range(data),
WithRowNames:=False, RespectHidden:=False)
    Call Rinterface.RRun("inputdataVar<-apply(inputdata[, -1], 2, var)")
    Call Rinterface.RRun("vinputdataVar<-as.vector(inputdataVar)")
    Call Rinterface.RRun("mvinputdataVar<-matrix(vinputdataVar, nrow=1)")
    Call Rinterface.RRun("inputdataSd<-apply(inputdata[, -1], 2, sd)")
    Call Rinterface.RRun("vinputdataSd<-as.vector(inputdataSd)")
    Call Rinterface.RRun("mvinputdataSd<-matrix(vinputdataSd, nrow=1)")

```

```
Range("計算結果!A6").Value = "分散："
Call Rinterface.GetArray("mvinputdataVar", Range("計算結果!B6"))
Range("計算結果!A7").Value = "標準偏差："
Call Rinterface.GetArray("mvinputdataSd", Range("計算結果!B7"))
Call Rinterface.StopRServer
```

End Function

2 OpenOffice.orgのためのマクロ入門

2.1 マクロで使った関数について

パブリックドメイン変数 Sub ~ End Sub の外で変数を宣言しているものはパブリックドメイン変数と呼び、同じライブラリ内にあるプロシージャまたは関数で変数を共有することができる。

LoadLibrary LoadLibrary メソッドの引数には、表示させたいダイアログが所属するライブラリのナを指定する。

CreateUnoDialog CreateUnoDialog 関数には DialogLibraries. ライブラリ名. ダイアログボックス名という引数を与える。そのダイアログを新しく作成しそのダイアログのオブジェクト参照を返す。

Execute Execute メソッドを実行するとダイアログが表示される。

EndExecute EndExecute メソッドを実行するとダイアログが消される。

GetControl ダイアログオブジェクトの GetControl メソッドにダイアログエディタで設定した名前を引数として渡すと、そのコントロールのオブジェクト参照が返ってくる。

Value Value プロパティは、入力されている数値に対応する。

Text Text プロパティは、入力されている文字列に対応する。

ThisComponent ThisComponent はアクティブなドキュメントを指す。

GetCellByPosition シートオブジェクトの GetCellByPosition メソッドは A1 セルを原点 (0,0) とする XY 座標を指定することでセルの参照を取得することができるメソッドである。

Formula Formula プロパティに代入された文字列は、数式として扱われ、その数式を計算した結果がセルに表示される。String, Value, Formula はそれぞれがプロパティである。

GetType GetType メソッドは、セルに入力されたデータの種別を定数で返す。
com.sun.star.table.CellContentType.Empty(空白), VALUE(数値), TEXT(文字列), FORMULA(数式) のいずれかである。

参考文献

- [1] プロジェクト A & できるシリーズ編集部「できる大辞典 Excel VBA」インプレスジャパン (2004)
- [2] 飯島 弘文「誰でもわかる VBA 完全マスター」メディア・テック出版 (2005)
- [3] オープンガイドブック OpenOffice.org2.0 公開版 ver0.0.1 (第三, 七, 九章)
(リンク先は Wiki のページ「OOo Basic」参照)