



【CN流】

ExcelVBA基礎 第2單元(2of5)

2022年11月
株式会社セルネッツ

【学習ポイント】制御構造「1.処理」

- 1.マクロ[開発タブ]の表示
- 2.コマンドボタン
- 3.プログラム制御構造は3つ「1.処理」「2.繰り返し」「3.条件分岐」
- 4.Errorメッセージ表示
- 5.最終行、最終列の取得2つの方法
- 6.データ転記（Fr:自シート To：自シート）
- 7.変数の型
- 8.文字列操作
- 9.並べ替えSORT（マクロ記録そのままは駄目）



学ぶこと

【第1～5单元】構成

【第1单元】オリエンテーション

- ・【CN流】All-In-One VBA製品デモ

【第2单元】3つのプログラム制御構造「1.処理」

【第3单元】3つのプログラム制御構造「2.繰り返し」

【第4单元】3つのプログラム制御構造「3.条件分岐」

【第5单元】まとめ

マクロとは？ VBAとは？

実際に見ていただきながら解説します。だいたいこんな感じです。

- 【1】マクロとは？

- 特定の操作手順を記録し、再現する機能
- Excel、WORD、PowerPoint、OutLookにも標準装備
- 例) Excelの操作手順を自動化することができる
- 例) アプリケーションの操作を自動化する機能

- 【2】VBAとは？

- 正式名称は「Visual Basic for Applications」
- Microsoft社が開発したプログラミング言語

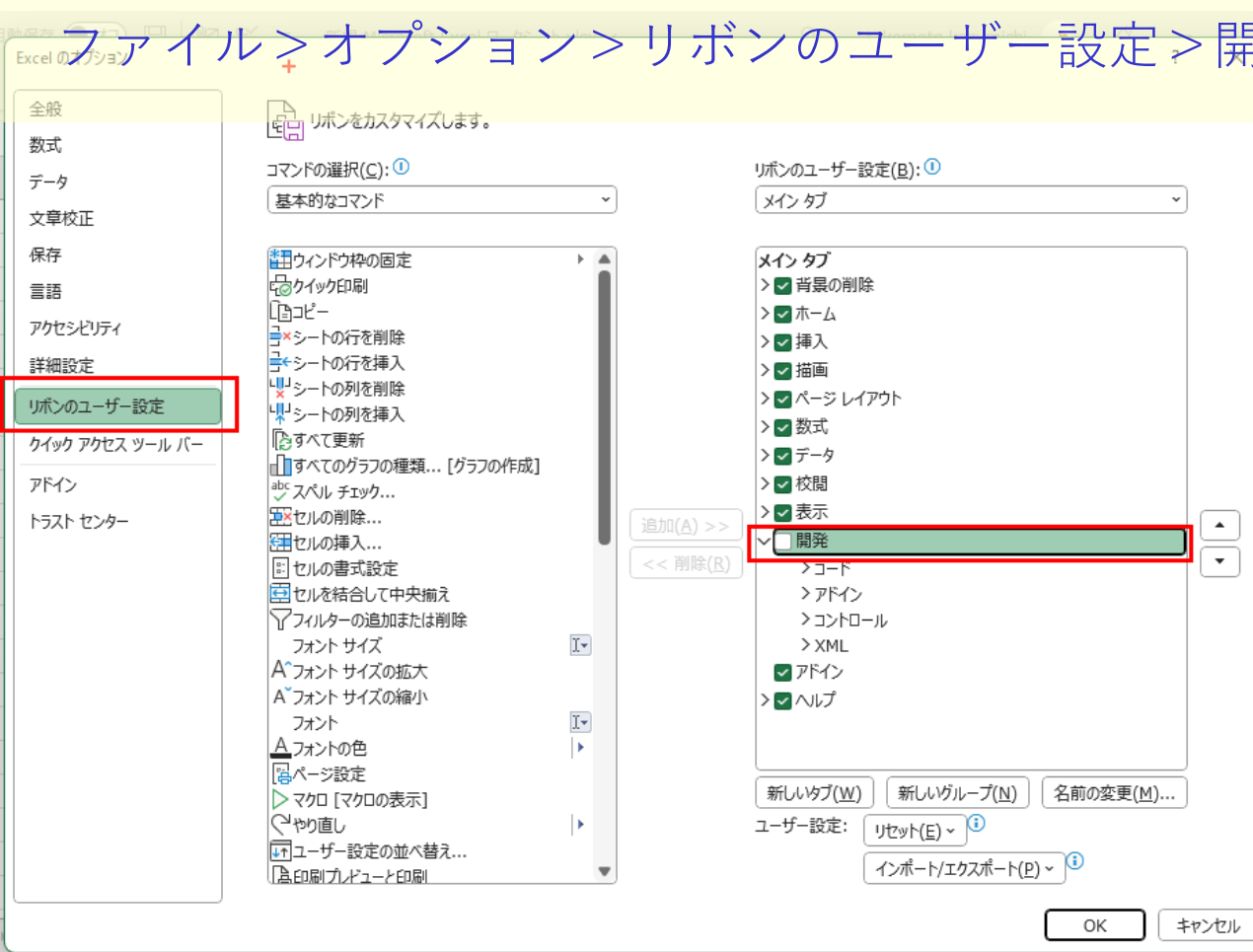
マクロとVBAは**別モノ**です

- 【3】VBEとは？

- 正式名称は「VBE(Visual Basic Editor)」
- VBAプログラムを作成・修正するための編集用ソフトのことです。

1. マクロ[開発タブ]の表示

ファイル > オプション > リボンのユーザー設定 > 開発



Excel のオプション

リボンをカスタマイズします。

コマンドの選択(C): ①
基本的なコマンド

リボンのユーザー設定(B): ①
メイン タブ

メイン タブ

- > 背景の削除
- > ホーム
- > 挿入
- > 描画
- > ページ レイアウト
- > 数式
- > データ
- > 校閲
- > 表示
- > 開発
- > コード
- > アドイン
- > コントロール
- > XML
- アドイン
- > ヘルプ

追加(A) >>
<< 削除(B)

新しいタブ(W) 新しいグループ(N) 名前の変更(M)...

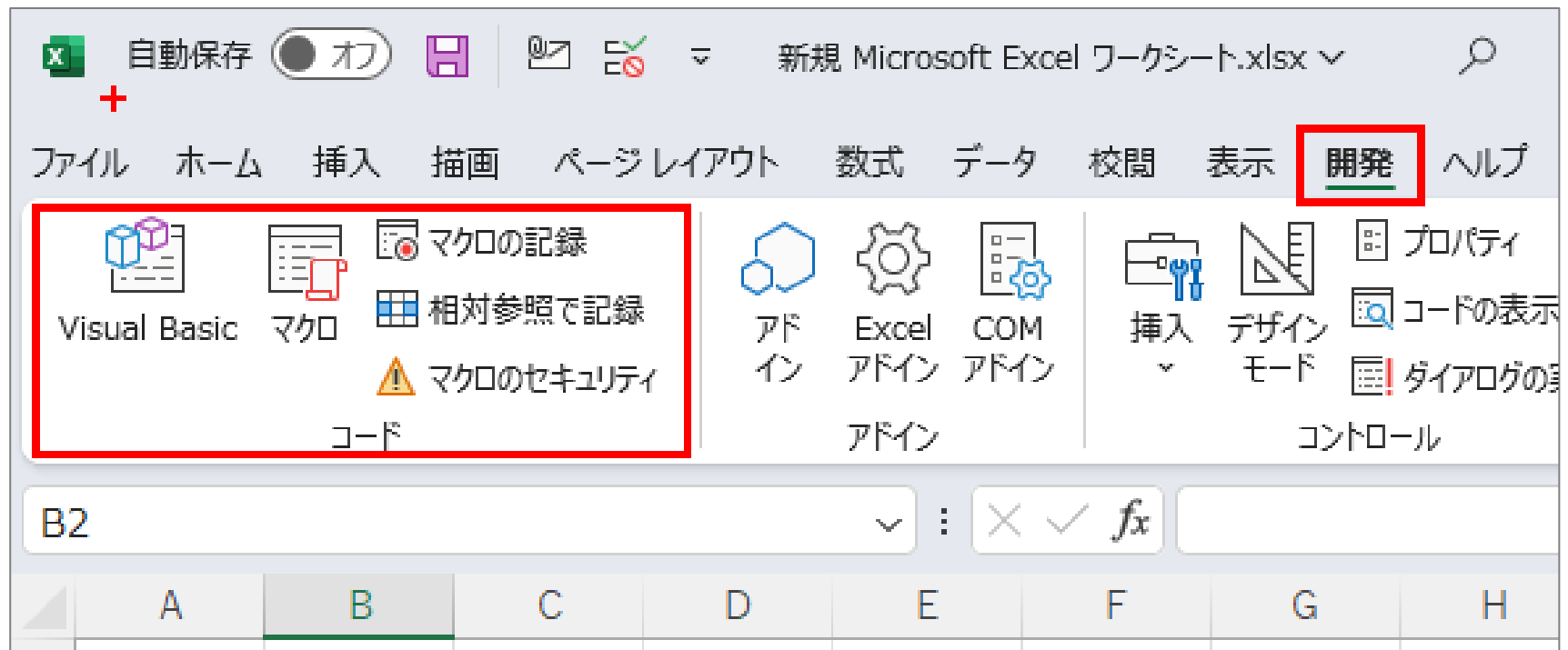
ユーザー設定: リセット(E) ①
インポート/エクスポート(P) ①

OK キャンセル

準備完了 アクセシビリティ: 問題ありません 100%

1.マクロ[開発タブ]の表示

ファイル>オプション>リボンのユーザー設定>開発



1. マクロ[開発タブ]の表示

Visual Basic>ツール>オプション>編集>変数の宣言を強制する(R)

Option Explicit

オプション

編集 エディターの設定 全般 ドッキング

コードの設定

- 自動構文チェック(K)
- 変数の宣言を強制する(R)
- 自動メンバー表示(L)
- 自動クイックヒント(Q)
- 自動データヒント(S)

自動インデント(O)
タブ間隔(T): 4

ウインドウの設定

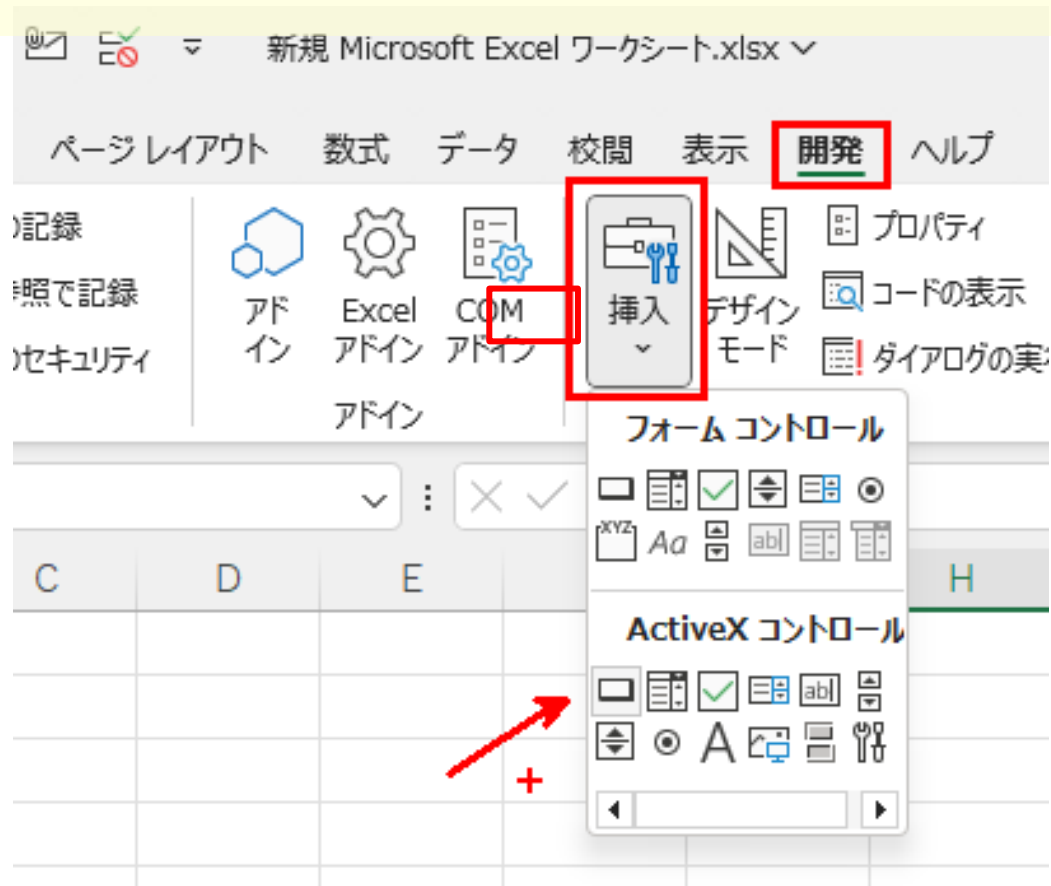
- テキストエディターでのドラッグアンドドロップ(D)
- モジュール全体を連続表示(M)
- プロシージャの区分線(P)

OK キャンセル ヘルプ

他は
規定値

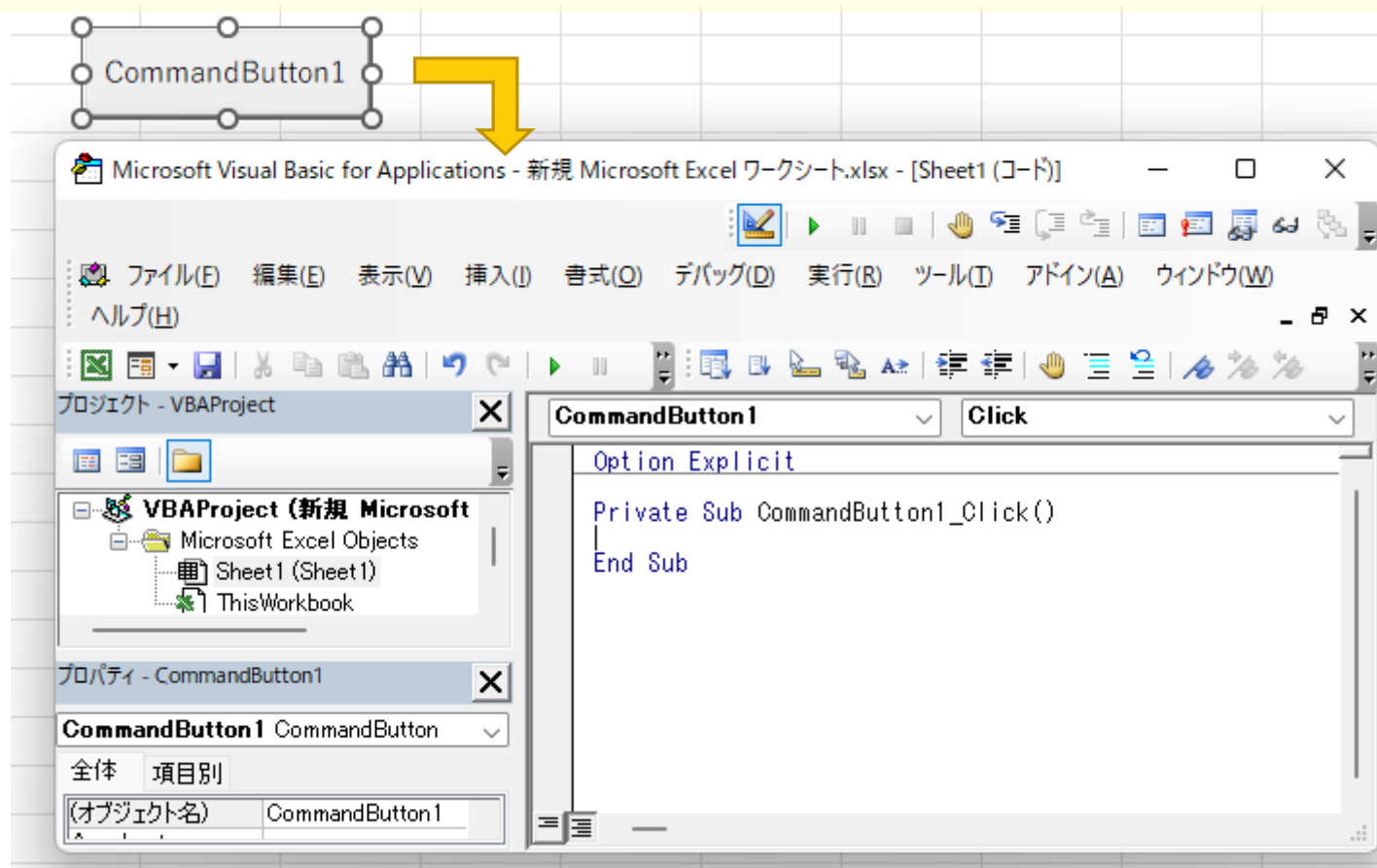
2.コマンドボタン

開発 > 挿入 > ActiveXコントロール > コマンドボタン



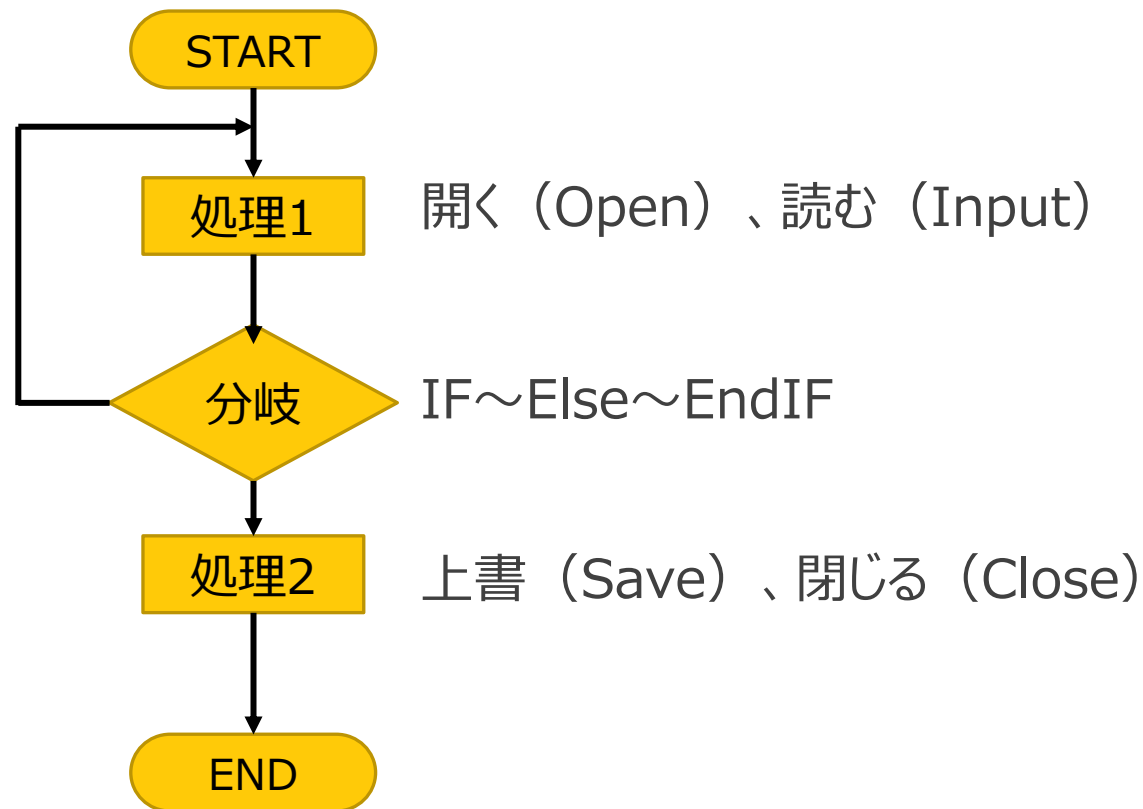
2. コマンドボタン

開発 > 挿入 > ActiveXコントロール > コマンドボタン



3.プログラム制御構造は3つ

「1.処理」 「2.繰り返し」 「3.条件分岐」

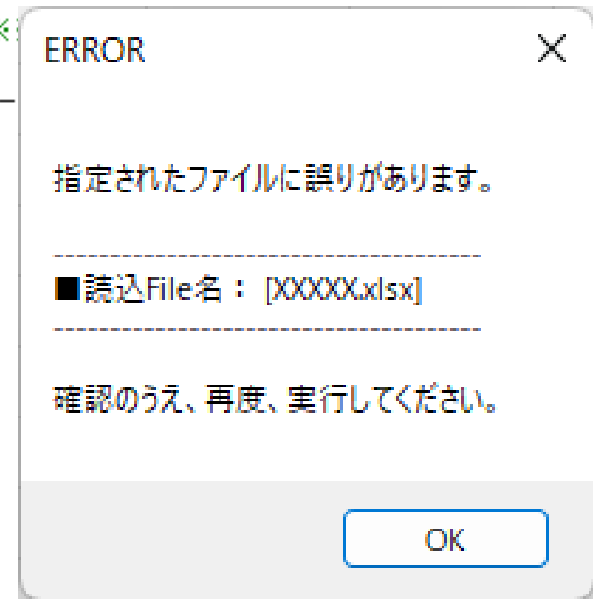


4.Errorメッセージ表示

「1.処理」 「2.繰り返し」 「3.条件分岐」

```
CommandButton1 Click
Option Explicit
'*****
Private Sub CommandButton1_Click()
'*****
MsgBox "指定されたファイルに誤りがあります。" & vbCrLf & vbCrLf & _
"-----" & vbCrLf & _
"■読込File名： [XXXXX.xlsx]" & vbCrLf & _
"-----" & vbCrLf & vbCrLf & _
"確認のうえ、再度、実行してください。", , "ERROR"
End Sub
```

お決まり
構文



5.最終行、最終列の取得2つの方法

取得方法による「有効データ範囲」認識の違い

```
Dim S_Fin行 As Long
Dim S_Fin列 As Long

'*****
Private Sub B_Fin行Fin列を取得A_Click()
'*****

'★ (A)特定列の末尾から取得
S_Fin行 = O_11.Cells(O_11.Rows.Count, 1).End(xlUp).Row
S_Fin列 = O_11.Cells(1, O_11.Columns.Count).End(xlToLeft).Column 'Fin列基点で左へ

MsgBox "(A)特定列の末尾から取得" & vbCrLf & _
"S_Fin行=" & S_Fin行 & vbCrLf & _
"S_Fin列=" & S_Fin列

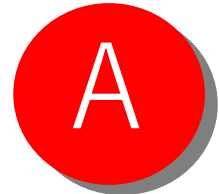
End Sub

'*****
Private Sub B_Fin行Fin列を取得B_Click()
'*****

'★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★
'★ カレント範囲から、最終行・最終列を取得
'★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★
S_Fin行 = Range("A1").CurrentRegion.Item(Range("A1").CurrentRegion.Count).Row
S_Fin列 = Range("A1").CurrentRegion.Item(Range("A1").CurrentRegion.Count).Column

MsgBox "(B)カレント範囲から取得" & vbCrLf & _
"S_Fin行=" & S_Fin行 & vbCrLf & _
"S_Fin列=" & S_Fin列

End Sub
```



5.最終行、最終列の取得2つの方法

注意！シート内のデータ状態

```
Dim S_Fin行 As Long
Dim S_Fin列 As Long

'*****
Private Sub B_Fin行Fin列を取得A_Click()
'*****

'★ (A)特定列の末尾から取得
S_Fin行 = O_11.Cells(O_11.Rows.Count, 1).End(xlUp).Row
S_Fin列 = O_11.Cells(1, O_11.Columns.Count).End(xlToLeft).Column 'Fin列基点で左へ

MsgBox "(A)特定列の末尾から取得" & vbCrLf & _
"S_Fin行=" & S_Fin行 & vbCrLf & _
"S_Fin列=" & S_Fin列

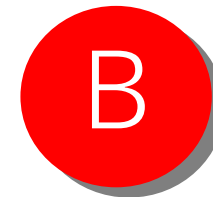
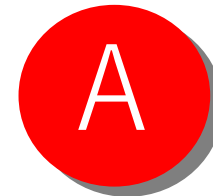
End Sub

'*****
Private Sub B_Fin行Fin列を取得B_Click()
'*****

'★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★
'★ カレント範囲から、最終行・最終列を取得
'★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★
S_Fin行 = Range("A1").CurrentRegion.Item(Range("A1").CurrentRegion.Count).Row
S_Fin列 = Range("A1").CurrentRegion.Item(Range("A1").CurrentRegion.Count).Column

MsgBox "(B)カレント範囲から取得" & vbCrLf & _
"S_Fin行=" & S_Fin行 & vbCrLf & _
"S_Fin列=" & S_Fin列

End Sub
```



6.データ転記 (Fr:自シート To : 自シート)

注意！シート内のデータ状態

```
Dim S_Fin行 As Long
Dim S_Fin列 As Long

'*****
Private Sub B_Fin行Fin列を取得A_Click()
'*****

  '★ (A)特定列の末尾から取得
  S_Fin行 = O_11.Cells(O_11.Rows.Count, 1).End(xlUp).Row
  S_Fin列 = O_11.Cells(1, O_11.Columns.Count).End(xlToLeft).Column 'Fin列基点で左へ

  MsgBox "(A)特定列の末尾から取得" & vbCrLf & _
    "S_Fin行=" & S_Fin行 & vbCrLf & _
    "S_Fin列=" & S_Fin列

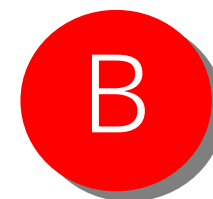
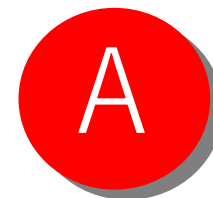
End Sub

'*****
Private Sub B_Fin行Fin列を取得B_Click()
'*****

'★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★
'★ カレント範囲から、最終行・最終列を取得
'★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★
  S_Fin行 = Range("A1").CurrentRegion.Item(Range("A1").CurrentRegion.Count).Row
  S_Fin列 = Range("A1").CurrentRegion.Item(Range("A1").CurrentRegion.Count).Column

  MsgBox "(B)カレント範囲から取得" & vbCrLf & _
    "S_Fin行=" & S_Fin行 & vbCrLf & _
    "S_Fin列=" & S_Fin列

End Sub
```



7.変数の型

頻度の高い変数は限定される

【変数名の命名ルール】 変数名 "xxxx"は、基本的に日本語を使用する

■変数のスコープ

(A) パブリック変数.....「P_xxxx」 パブリックの **P**

(B) その他のローカル変数....「S_xxxx」 その他の **S**

■変数の型

◎ Variant(バリエーション型)	◎ String(文字列型)	◎ Long(長整数型)
◎ Date(日付型)	◎ Double(度浮動小数点実数型)	
×Integer(整数型)	×Single(浮動小数点実数型)	×Boolean(ブール型)
×Byte(バイト型)	×Currency(通貨型)	

8.文字列操作

```
B_文字列操作 Click
'*****
Private Sub B_文字列操作_Click()
'*****
' (1)取得値 [都道府県名]
S_セル値 = Range("I1")
● Debug.Print S_セル値
' (2)先頭1桁
S_セル値 = Mid(Range("I1"), 1, 1)
● Debug.Print S_セル値
' (3)右から2桁
S_セル値 = Right(Range("I1"), 2)
● Debug.Print S_セル値
' (4)部分変換
S_セル値 = Replace(Range("I1"), "県", "KEN")
● Debug.Print S_セル値
End Sub
'*****
Private Sub B_並べ替え_Click()
'*****
```

イミディエイト

```
都道府県名
都
県名
都道府KEN名
```


9.並べ替えSORT

マクロ記録そのままは駄目

```
'*****  
Private Sub B_並べ替え_Click()  
'*****  
  
'★[昇順：xlAscending] [降順：xlDescending]  
Range("A1").Sort Key1:=Range("H1"), Order1:=xlDescending, Key2:=Range("A1"), Order2:=xlAscending, Header:=xlYes  
  
'以下、ダメなやつ！  
' Range("A1:I21").Select  
' ActiveWorkbook.Worksheets("sample").Sort.SortFields.Clear  
' ActiveWorkbook.Worksheets("sample").Sort.SortFields.Add2 Key:=Range("H2:H21") _  
' , SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal _  
' ActiveWorkbook.Worksheets("sample").Sort.SortFields.Add2 Key:=Range("A2:A21") _  
' , SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal _  
' With ActiveWorkbook.Worksheets("sample").Sort  
' .SetRange Range("A1:I21")  
' .Header = xlYes  
' .MatchCase = False  
' .Orientation = xlTopToBottom  
' .SortMethod = xlPinYin  
' .Apply  
' End With  
  
End Sub
```

第1キー
H列：金額（降順）

第2キー
A列：顧客CD（昇順）

まとめ

- 1.マクロ[開発タブ]の表示
- 2.コマンドボタン
- 3.プログラム制御構造は3つ「1.処理」「2.繰り返し」「3.条件分岐」
- 4.Errorメッセージ表示
- ★ 5.最終行、最終列の取得2つの方法 Excel開発ではシート内の**データ状態チェック**が必須！
- 6.データ転記（Fr:自シート To：自シート）
- 7.変数の型
- 8.文字列操作
- 9.並べ替えSORT（マクロ記録そのままは駄目）

【第3单元】制御構造「2.繰り返し」

- 繰り返しLoop処理
- For ~ Next
- Do whileはLoop回数が不明な時だけ

フォルダ内のファイル処理とか

画像やスタイルの個数とか

