

ゼミC 第3回

VBA ビジュアルベーシックフォーアプリケーション

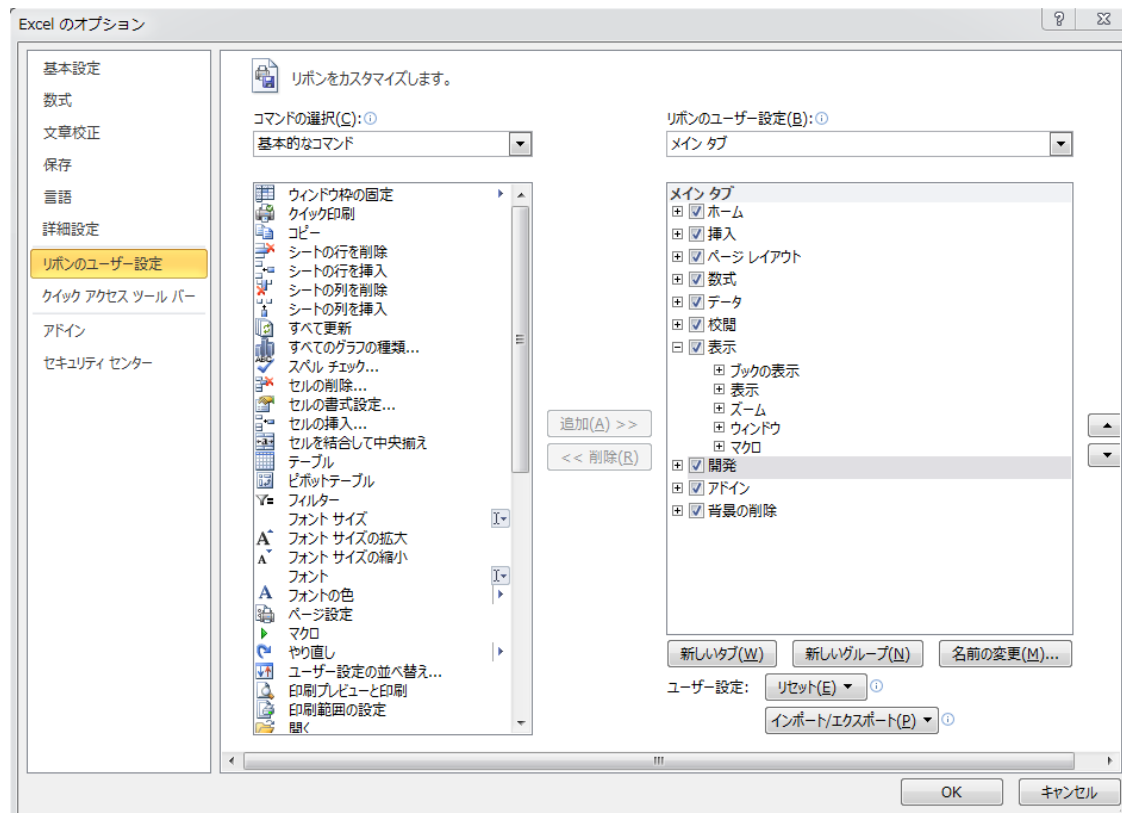
2014年8月5日 エレクトロニクス研究部

Visual Basic for Applicationsとは

- **マイクロソフト社のMicrosoft Officeシリーズに搭載されているプログラミング言語。オブジェクト指向型の言語である。**
- **ExcelとかWordとかそのあたりのある程度の業務を自動化することができる。**
- **拡張子は.bas**

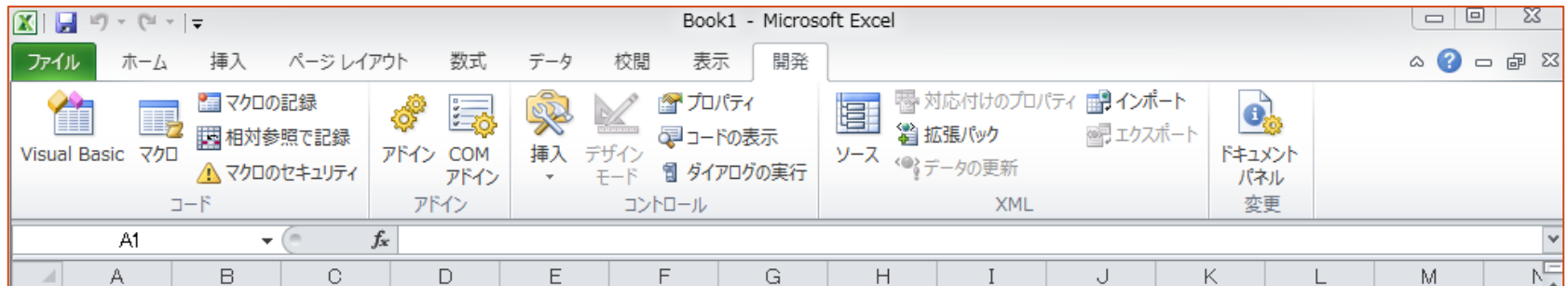
Visual Basic for Applicationsの起動

- **まずExcelを開いてください。そしてファイル→オプション→リボンのユーザー設定から「開発」の部分にチェックを入れてください。**



そうするとExcelに「開発」というのが現れたと思います。

そこをクリック→Visual Basic

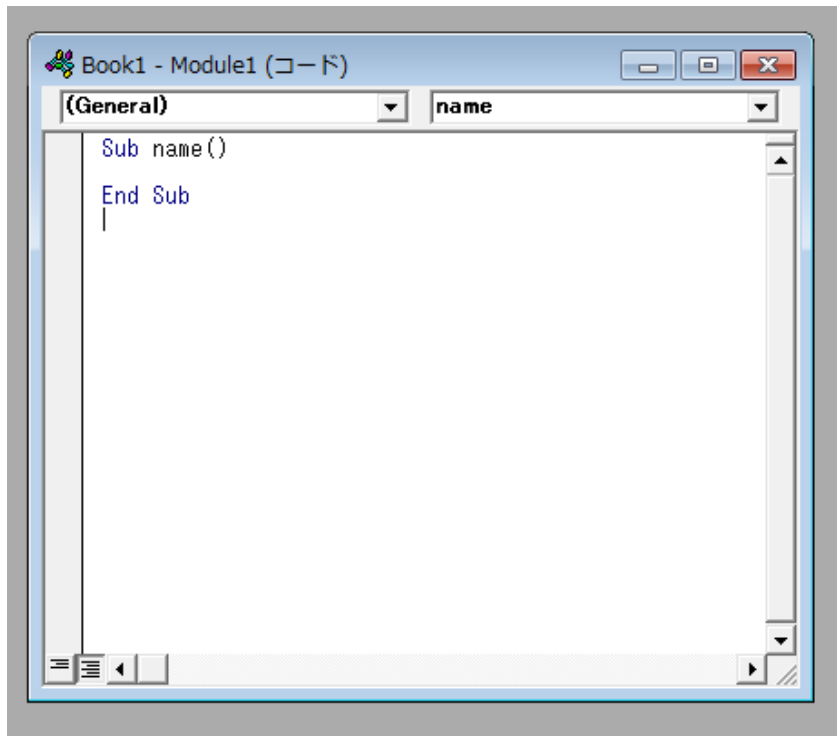


これでVBAが起動します。

次に開いたVBAの挿入→標準モジュールを押してください。コードを入力するところが表示されるはずです。これをモジュールの挿入といいます。

マクロ（プログラム）の作成

- 試に現れたウィンドウに「Sub name」と入れてえんたーキーを押してみてください。



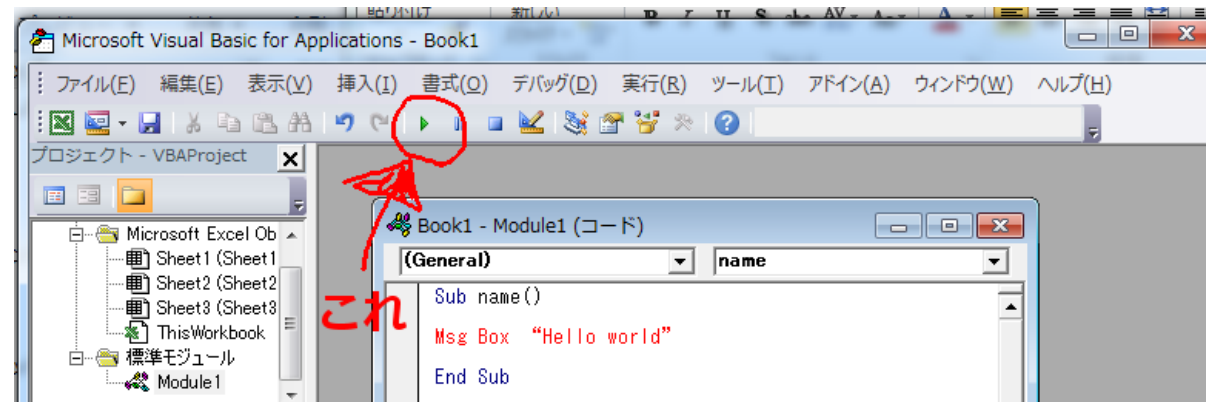
なんか「End Sub」ってのが現れたと思います。

基本的にマクロ（プログラム）での処理はこの「Sub マクロ名」で始まり「End Sub」で終わります。今回はマクロ名「name」になっています。

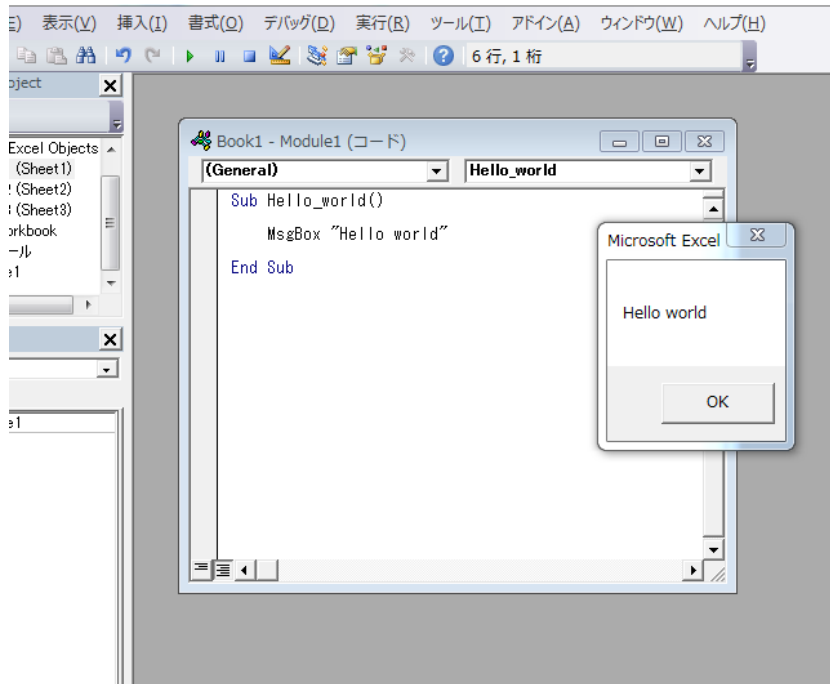
- 今回は試に「Hello world」と文字列が表示されたメッセージボックスを表示したいと思います。

Msg Box “Hello world”

- 上の文章をさっきの「Sub name()」と「End Sub」の間に試に入力してみてください。
- 次に実行を行ってしてみてください。実行の行い方としては



- 三角のボタンを押すとまた何かウインドウが出るので実行したいマクロ名（ここではname）を選択し実行してください。
そうすると



実行されます。
今回はMsgBox関数という
ものを使用し、メッセー
ジボックスを出していま
す。詳しくはのちでやり
ます。

セルへの値の代入

- 実行するとExcelが表に出てきたと思います。基本的にこのVBAで実行したマクロはこのExcelを動かすものです。
つぎにこのExcelのセルの操作をしていきたいと思っています。セルはあのばーってなってるちっちゃい四角のことです。

- **まずセルに何か記入してみたいと思います。セルに記入する方法としては二つの形式があります。**

```
Range("B2").value = 10
```

```
Cells(2,1).value = 30
```

```
Cells(2,"B").value = 30
```

上の二つを記して実行するとExcelのほうに変化があると思います。それぞれ指定した場所に数字が入っています。

二つの違いは主にセルの場所の指定の仕方の違いです。RangeはA~から一つ、1~から一つ指定するのに対しCellsはその座標を指定するような形です。

出力する値は数字だけでなく文字列も可能です。その場合は出力したい文字列を””で囲むようにしてください。

- **記入する際、セルの指定をしましたが一気に複数のセルに記入することも可能です。**

```
Range("2:5").value = 10
```

```
Range("F:H").value = 30
```

これを実行してみてください。多くのセルに値が記入されました。

上のプログラムに沿って説明すると行2～行5とした範囲全てに10が記入されました。

- **縦、横に無制限に値を入れることも可能です。**

```
Range("A2","E5").EntireColumn.value = 10
```

前のプログラムとは違い「EntireColumn」が含まれています。これによってA2~E5を持つ列すべてに値が代入、つまり列A,B,C,D,Eにすべて10が入れられました。

```
Range("A2","E5").EntireRow.value = 10
```

これはA2~E5を含む行にすべて10が記入されました。つまり行2,3,4,5に記入された、ということです。

セルの値の削除

- セルに値を記入しましたがこの中の値を削除しようと思ったらプログラム側でRange~ってのを削除して実行する方法では消えません。セルの中のを削除するにはこちらからセルを指定してその中を削除する、としなければいけません。

```
Range("A2"). Delete
```

このようにすることでA2の中の値を削除できます。記入の時に使ったRange("A2","F5")やEntireColumn、EntireRowも同じように使えます。指定される範囲は記入と同じです。

- 削除の際にも注意点があります。
例えば

	A	B	C	D	E	F	G
1							
2		1	2	3	4	5	
3		6	7	8	9	0	
4		11	12	13	14	15	
5		16	17	18	19	20	
6		21	22	23	24	25	
7		26	27	28	29	30	
8							
9							
10							
11							

このように記入されており、D5(18)を削除しよう
とします。そのため下のように記し実行しました。

Range("D5").Delete

するとこのようになりました。

	A	B	C	D	E	F	G
1							
2		1	2	3	4	5	
3		6	7	8	9	0	
4		11	12	13	14	15	
5		16	17	23	19	20	
6		21	22	28	24	25	
7		26	27	29	29	30	
8							
9							
10							

- 削除されたところはこのように詰められてしまいます。

指定したところのセルだけ空白にしたいなって時は `Range("D5").Value = ""` こんな感じで新たに値を書き換えるようにしてください。

- 削除するセルの行数 = 列数だと上方向にシフト、
行数 < 列数だと左方向へシフト
行数 > 列数だと上方向にシフトします。

セルの挿入

- セルの中の値を書き換えるわけではなく、そのセルをその場所に差し込み、他の場所はそれに合わせてずらすような形にすることもできます。

```
Range("A2").Insert
```

A2の場所に挿入します。記入の時に使ったRange("A2","F5")やEntireColumn、EntireRowも同じように使えます。指定される範囲は記入と同じです。

- 削除するセルの行数 = 列数だと下方方向にシフト、
行数 < 列数だと右方向へシフト
行数 > 列数だと下方方向にシフトします。

変数

- 次に変数についてやっていきます。勿論VBAにも存在します。

Dim 変数名 As データ型

- 例)

Dim X As Integer

→変数名Xの整数型変数を宣言

変数名は略できますが基本していくようにしましょう。

データ型	意味	Cでは
Integer	整数型	int
Double	浮動小数点数型	double
String	文字列型	char
Boolean	ブール型	bool

などなど。

- 値の代入も今までと同じくすることができる。四則計算もまたしかり。

例)

```
Dim a As Integer
```

```
Dim b As Integer
```

```
Dim c As String
```

```
a = 10
```

```
b = 20
```

```
c = "get"
```

```
a = a + 5
```

```
b = b * b
```

セルからの値の読み込み

```
Range("A2").value = 10
```

これは記入です。これをした後

```
a = Range("A2").value
```

とするとaにA2に入っている値が代入されます。ここだと10が代入されます。

練習問題1

- セルの2か所、B2とC2から値を読み込み、その足し算をE2、引き算をF2、掛け算をG2、割り算をH2の結果を出力しなさい。

解答1

Sub Keisan()

Dim a As Integer

Dim b As Integer

a = Range("B2").Value

b = Range("C2").Value

Range("E2").Value = a + b

Range("F2").Value = a - b

Range("G2").Value = a * b

Range("H2").Value = a / b

End Sub

if文

- if文はCとは多少違います。

If **条件式1** Then

条件式1を満たした場合の処理

ElseIf **条件式2** Then

条件式2を満たした場合の処理

Else

いままでの条件を満たさなかった時の処理

End If

演算子

- ここで条件式に使う演算子の説明をしていきたいともいます。

まず比較演算子

演算子	意味	Cだと
=	等しい	==
<=	以下	<=
<	より小さい	<
>=	以上	>=
>	より大きい	>
<>	等しくない	!=

- **次に論理演算子**

演算子	意味	Cでは
条件式 And 条件式	かつ	&&
条件式 Or 条件式	または	
Not 条件式	否定	!

if文などの条件式にはこの演算子を作るようにしてください。

Select case文

- Cでいう switch文。 ifelse文で置き換えることもできる。

Select Case **変数**

Case **値1**

変数=値1の場合の処理

Case **値2**

変数=値2の場合の処理

Case Else

**変数=値1、変数=値2を満たさなかった場合
の処理**

End Select

For Next文

- 条件により処理を繰り返したいときに使う。Cで言うfor文。

```
For カウンタ変数 = 初期値 To 回数 の指定 Step 加算値  
    繰り返し回数までの処理
```

```
Next
```

10回処理を繰り返したいときは

```
For i= 1 To 10 Step 1  
    処理
```

```
Next
```

処理を一回行くとiの数はStepの横の値1ずつ増えていく。そのiの値がToの横の値10になるまで処理を繰り返し行う。

Cだとfor(i=1;i<=10;i++){~になる。

Do While文

- Cで言う while文。条件式を満たしている限り処理を繰り返し続ける。

Do While 条件式

条件式を満たすまで繰り返す処理

Loop

条件式を後に置くこともできる

Do

条件式を満たすまで繰り返す処理

Loop While 条件式

練習問題2

- Bの列に値が書いてあった場合、その値を3倍した値をそのCの列に書くようにせよ。行は1から15までの間とする。

実行するとこんな感じ

	コード		アドイン	
	F13		fx	
	A	B	C	D
1				
2		30	90	
3		3	9	
4		15	45	
5		3	9	
6				
7				
8		6	18	
9				
10				
11				
12		22	66	
13		2	6	
14		2	6	
15		2	6	
16		2		
17		2		
18				
19				
20				
21				
22				

解答2

```
Sub Keisan()
```

```
    Dim a As Integer
```

```
    Dim i As Integer
```

```
    For i = 1 To 15
```

```
        a = Cells(i, "B").Value
```

```
        If a <> "" Then
```

```
            Cells(i, "C").Value = a * 3
```

```
        End If
```

```
    Next
```

```
End Sub
```

配列

- 配列についてやっていきます。宣言の方法など今までと多少違います。

Dim 変数名(上限値) As データ型

例えば Dim a(3) As Integer とすると

上限値3の配列aが宣言されたことになります。このときa(0),a(1),a(2),a(3)が宣言されました。

今までと違って()が使われていること、上限値で設定した値の配列分も宣言されていることに注意してください。

動的配列

- **動的配列とは。**

いままでは配列の大きさを宣言時に決定していましたが動的配列だとその配列の大きさを変数宣言後も変えることができます。

```
Dim 変数名() As データ型
```

で上限値を設定せずに宣言、その後

```
ReDim 変数名(上限値)
```

で上限値を設定することができます。

2次元配列

- **2次元配列を宣言してみましょう。**

Dim 変数名(行方向の上限値,列方向の上限値) As データ型

これで2次元配列の宣言ができます。

Dim a(1,2) As Integer

これでa(0,0),a(0,1)~~~~~a(1,1),a(1,2)の6つの変数が宣言されることになります。

練習問題3

- 右の図のように Excelに値が入力されていたとする。
それぞれの列、それぞれの行に入力された値の平均を表示するようにせよ。

	A	B	C	D
1	120	65	2	
2	134	46	4	
3	156	73	7	
4	98	54	3	
5	111	59	9	
6				

解答3

Sub ave()

Dim i, j As Integer

Dim a(5, 3) As Integer

Dim sum As Integer

sum = 0

For i = 1 To 5 Step 1

For j = 1 To 3 Step 1

a(i, j) = Cells(i, j)

Next

Next

For i = 1 To 5 Step 1

For j = 1 To 3 Step 1

sum = sum + a(i, j)

Next

Cells(i, "E") = sum / 3

sum = 0

Next

For i = 1 To 3 Step 1

For j = 1 To 5 Step 1

sum = sum + a(j, i)

Next

Cells(7, i) = sum / 5

sum = 0

Next

End Sub

ワークシートの参照

- 下のところに



Sheet1,Sheet2ってのがあるとおもいます。これをワークシートといい、この参照もVBAで行うことができます。

Worksheets(インデックス)

これをつけることによりワークシートの選択を行うことができます。

- **例えば**

```
Worksheets(2).Range("H2").Value
```

こうすることによりワークシート1にいてもワークシート2のセルを参照することができます。

- **シートを選択するには**

```
Worksheets(2).Select
```

とします。

練習問題4

- ワークシート1に入力されている値とワークシート2に入力されている値の和差積をワークシート3に出力せよ。

解答4

```
Sub work()
```

```
    Dim i, j As Integer
```

```
    For i = 1 To 10 Step 1
```

```
        For j = 1 To 10 Step 1
```

```
            Worksheets(3).Cells(i, j).Value = Worksheets(1).Cells(i, j).Value + Worksheets(2).Cells(i, j).Value
```

```
        Next
```

```
    Next
```

```
    For i = 12 To 22 Step 1
```

```
        For j = 1 To 10 Step 1
```

```
            Worksheets(3).Cells(i, j).Value = Worksheets(1).Cells(i - 11, j).Value - Worksheets(2).Cells(i - 11, j).Value
```

```
        Next
```

```
    Next
```

```
End Sub
```

セルの色を塗る

今までセルに値を入れるだけでしたが次は色を塗ってみましょう。

```
Range("E2:F8").Interior.Color = RGB(255, 0, 0)
```

```
Range("E2:F8").Interior.ColorIndex = 35
```

Interior.Colorおよび**Interior.ColorIndex**を使うと指定した部分のセルの色を設定することができます。

Interior.Colorは色の指定にRGBを使います。

Interior.ColorIndexは色の指定にあらかじめ用意されたインデックス番号により指定します。

- RGBについてすこし

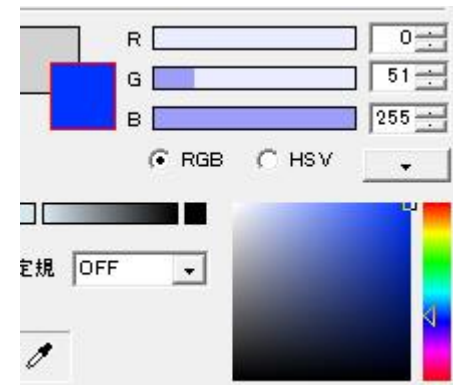
RGBはRはRed、GはGreen、BはBlueを指しており3つの数字から表されます。

各数字は最大値255からなります。

RGB=(0,0,0)で黒、(255,255,255)で白を表しています。

ペイントツールとかを使うと色のRGBデータを見たりできるのでやってみてください。

- こんなかんじ→



練習問題5

- 左上から 5×5 のセルの中に書かれている値を読み取り、その値によってそのセルの色を塗り分けよ。書かれている値が100以上だったら赤、100より小さく50以上だったら緑、それ以外は青で塗りつぶせ。

解答5

```
Attribute VB_Name = "Module2"
```

```
Sub color()
```

```
    Dim i, j As Integer
```

```
    For j = 1 To 5 Step 1
```

```
        For i = 1 To 5 Step 1
```

```
            If Cells(i, j) >= 100 Then
```

```
                Cells(i, j).Interior.ColorIndex = 3
```

```
            ElseIf Cells(i, j) >= 50 Then
```

```
                Cells(i, j).Interior.ColorIndex = 10
```

```
            Else
```

```
                Cells(i, j).Interior.ColorIndex = 5
```

```
            End If
```

```
        Next
```

```
    Next
```

```
End Sub
```

ゼミC

- これで以上です！お疲れ様でした。
- 夏合宿が近いです。ソフトゼミ員は夏合宿までにどんなゲームを作りたいかおおよそを決めておいてください。