

EXCEL 実習Ⅳ『**VBA** の基礎』

情報技術科 2 年 実習

1 目的

これまで表計算ソフトの定番「Excel」の基本的な使用方法を実習してきたが、さらに作業効率の向上を図るためには『マクロ』は欠かせない機能となる。ここでは、マクロの概念を理解するとともに、記録機能を使用して簡単にマクロを作成する方法を学習する。しかし、マクロは煩雑な処理を効率的にこなす手段として有用であるが、「マクロの記録」機能はマウスやキーボード操作の範囲内に止まるため、繰り返し処理や条件分岐、データ入力は記述できない。そこで、実用性を高めるためにはどうしても VBA の知識が必要となる。

今回はキーボードマクロのフォームボタンの配置とマクロの登録さらに、記述言語である VBA をとりあげ、実用性の向上を図ることとする。

2 予備知識

2-1 マクロ・キーボードマクロ とは

マクロとはマイクロソフトの Excel に標準装備されている、複数の手順を記憶して、自動的に実行させる機能のことをいう。プログラミング言語の一種である Excel マクロと VBA であるが、上手に使いこなすことで効率よく作業を進めることができる。別ファイル“マクロの操作”を参照し課題を進める。

2-2 VBA とは

VBA とは『 Visual Basic for Applications 』を省略したもので、プログラミング言語のひとつである。

「マクロの記録」で作成したマクロは VBA で記録されているため、マクロ = VBA と考えて差し支えない。

従って、マクロの記録と VBA によるプログラミングの2つを理解しておけば、マクロで記録したものを VBA で編集することができるし、またその逆も可能となる。

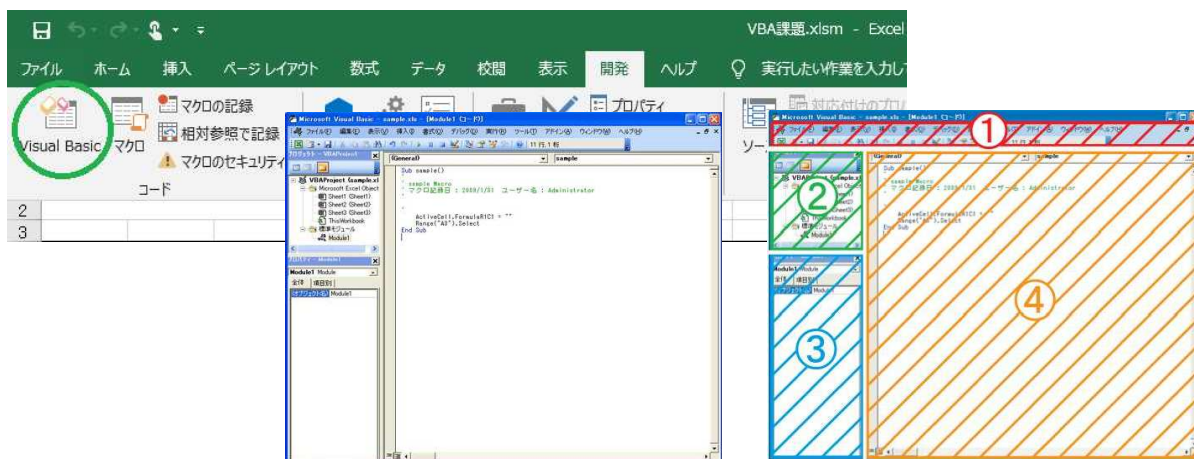
2-3 VBE

VBA でのプログラミングを行うためには、EXCEL に標準で搭載されている VBE(Visual Basic Editor)という機能を使っておこなう。まず、VBE を起動すると、以下のような画面が表示される。この、VBE の編集画面は大きく分けると4つの部分に分かれている。

注・[開発] タブがユーザリボンにない場合

2007 以降では、[ファイル]・[Excel のオプション] をクリックし、[基本設定] または、リボンのユーザ設定をクリックし、[開発] タブをリボンに表示する] チェック ボックスをオンにすると、リボンメニューに開発タブが表示されます。別ファイル“マクロの操作”を参照。

・起動方法



①メニューバー

VBE がプログラミングをするためのツールであるため、「実行」や「デバック」メニューなどがある。

②プロジェクトエクスプローラ

現在、開いているファイルがどれなのか、どのファイルがどれと連携しているのかなどを表しているのが プロジェクトエクスプローラの役割である。編集するファイルを選んだり、ファイル同士の連携を変更する場合はこのウィンドウの中から編集する。

③プロパティウィンドウ

個別のファイルについている細かな属性を編集する場合に使うウィンドウで、文字の大きさや、セルの中の色などの属性情報が個別に設定可能である。

④コードライティングエリア

現在、選択しているファイルのプログラムコードが表示されている部分。プログラムの新たに組む場合や内容を編集する場合は、この部分を修正する。

2-4 VBA の構成要素

どれも聞き慣れない言葉だが、VBA の理解のために知っておきたい。

(1) プロジェクト

プロジェクトは細かな実行単位(データファイルやマクロなど)をまとめたもので、ひとつのシートデータ(ワークブック)がひとつのプロジェクトと定義されている。マクロは単体では存在するのではなく、シート上で存在しており、プロジェクトは最も大きな単位のことになる。

(2) モジュール

モジュールとは、プロジェクトを構成している小さなプログラムの構成単位をいう。但し、次に紹介する「プロシージャ」よりは大きな構成単位となるので、中程度の規模ともいえる。「条件分岐」や「繰り返し処理」などが、このモジュールとよばれる「標準モジュール」を形成している。

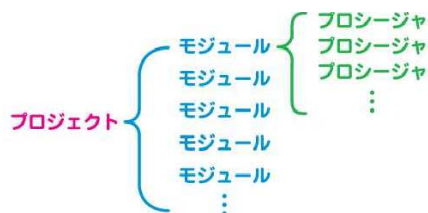
(3) プロシージャ

VBA マクロは「プロシージャ」と呼ばれるひとまとまりのプログラムを単位として構成されている。プロシージャは「Sub マクロ名()」ステートメントで始まり「End Sub」ステートメントで終わる。

Sub マクロ名()

マクロ本体

これらの関係を簡単にまとめると下図のようになる。



命令文をステートメントと呼ぶ。先述のプロパティの値を設定するのもステートメントである。

ステートメントの記述ルール

- 行の先頭から改行するまでの1行がひとつのステートメントになる。
- ひとつのステートメントを複数行に記述する場合は、行継続分「_」(半角スペースとアンダーバー)を使う。
- 「'」で始まる行はコメント行。
- 大文字と小文字の区別はしない。

2-5 オブジェクト・プロパティ・メソッド

VBA では操作対象をオブジェクト(Object)・オブジェクトの属性をプロパティ(Property)・オブジェクトに加える動的な操作をメソッド(Method)と呼んでいて、プログラム中ではピリオド(.)で区切って表す。

(1) オブジェクト

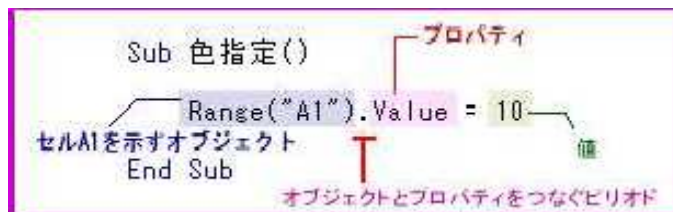
オブジェクトは操作の対象となるものを指す。具体的にはセル・セル範囲・ブック・シートなどがある。



Range("A1")	Range は「範囲」のことで、セルやセル範囲指定に用いる。 Range("A1")とは、A 1 のセルという意味で、Range("A1:C3")とすると A1 から C 3のセルまでのセル範囲を指す。
Worksheets("Sheet1")	Worksheets はワークシートことで、Worksheets("Sheet1")はシート 1 を指す。シート 1 のセル A1 を指すには、 Worksheets("Sheet1"). Range("A1")と記述する。
Workbooks("Book1.xls")	Workbooks はブックのことで、Workbooks("Book1.xls")で Book 1 というファイル名のブックを指すことになる。 Workbooks や Worksheets が省略された場合は、選択ファイルの選択シートが自動的にオブジェクトとなる。

(2) プロパティ

オブジェクトの持つ特性や属性のことをいう。オブジェクトの後ろに Color(色)・Size(サイズ)・Count(数)・Value(値)などを付けることで、その特徴や性質を示す。



他のプロパティの例

プロパティ		
背景 Interior	色	ColorIndex
	塗りつぶしのパターン	Pattern
文字 Font	色	ColorIndex
	文字の大きさ	FontSize
値 Value		

プロパティはふたつのコードを組み合わせる場合もある。

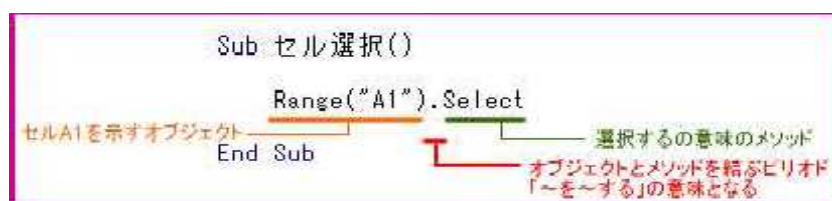
例えば、【背景の色を赤色に変更する】という指示を出す場合は、「Interior.ColorIndex = 1」となる。

対象	命令内容		VBA コード
A1 セル Range("A1")	背景 Interior	色 ColorIndex	Range("A1").Interior.ColorIndex = ○
		塗りつぶしパターン Pattern	Range("A1").Interior.Pattern = ○
	文字 Font	色 ColorIndex	Range("A1").Font.ColorIndex = ○
		文字の大きさ Size	Range("A1").Font.Size = ○
	値 Value		Range("A1").Value = ○

○には数値や数式・変数などが入る

(3) メソッド

メソッドはオブジェクトの動作を示す。メソッドを使うと、選択する・移動する・削除するなどのようにオブジェクトを直接操作できる。また、操作の具体的な指定のための引数を持つメソッドもある。



メソッドの例

選択する	Select
コピーする	Copy
削除する	ClearContents
印刷する	PrintOut

使用例

対象	命令内容	VBA コード
A1 セル Range("A1")	選択する Select	Range("A1").Select
	コピーする Copy	Range("A1").Copy
	削除する ClearContents	Range("A1").ClearContents
	印刷する PrintOut	Range("A1").PrintOut

2-6 マクロの記述場所

EXCEL にはマクロを記述できる対象(場所)が 5 種類ある。イベントプロシージャ以外は使い分けの明確な決まりはないが、機能や目的に応じて選択することで、プログラム全体を最適化することができる。

(1) ブックに書き込む

ブック全体の設定に関わるマクロはここに記述する。

(2) ワークシートに書き込む

対象のワークシートでしか使わないマクロはここに記述する。例えば、Sheet1 でしか使用しないマクロは Sheet1 に記述すべき。

(3) 標準モジュールに書き込む

汎用的な機能のマクロ(サブルーチン)を作成するときは、「標準モジュール」を作成してここに記述する。例えば、サウンドに関する処理・単位変換など、複数のワークシートから呼び出される処理を記述する。

(4) ユーザーフォームに書き込む

考え方はワークシートと同じ。

(5) クラスモジュールに書き込む

オブジェクト指向のアプリケーションを作成するとき、クラスモジュールを作成してここに記述すると良い。標準モジュールとの違いは「クラスモジュールで作成したモジュールは変数のデータ型として使用できる」点である。

2-7 セキュリティレベル

EXCEL ではウイルス対策として、セキュリティレベルの設定によりマクロの動作を制限している。

セキュリティレベル	動 作
高	マクロは一切動かない。
中	マクロを含むファイルを開く際には「マクロを有効にする」かを毎回確認する。
低	マクロを含むファイルを無条件で開く。

セキュリティレベルは「ツール」→「マクロ」→「セキュリティ」で設定できる。

3 実習

メソッドやプロパティは数多く存在し、そのすべてを短時間で習得するのは困難である。そこで基本的な記述方法を実習した後はアニメーションを教材にし、VBA に親しんでもらうことにする。

3-1 コードの入力と実行

ここでは、コーディングから実行までの流れを実習する。

- 課題 1 「ツール」→「マクロ」→「Visual Basic Editor」を選択して VBE を起動する。続いて次のコードを入力し、実行しなさい。

```
Sub test()  
    MsgBox ("Hey Hey Hey")  
End Sub
```

実行するには「実行」→「Sub/ユーザフォームの実行」(またはアイコン)とする。

- 要点
- ・ソースは「Sub マクロ名()」から「End Sub」までの間に記述する。
 - ・MsgBox は「メッセージボックス」を表示させる関数。
 - ・End Sub の後に、Sub ***() と次のマクロを続けて記述することもできる。この場合、カーソルがあるマクロが動作対象となる。
 - ・マクロを参照すると、いま記述したマクロが登録されており、VBA=マクロであることが理解できる。

Sub test() をマクロ同様フォームボタンに登録してみよう。

3-2 セルの指定方法

- ・以下課題 2 以降のプログラムはアクティブ X ボタンに登録すること。
- 課題 2 セル A1 に 100 を入力し、文字を黄色に色づけするマクロを作成しなさい。

記述の一部(例)

```
Range("A1").Select  
Selection.Font.Colorindex=6
```

- 要点
- ・オブジェクトやメソッド、プロパティの構成を理解する。
 - ・セルの指定方法にはこの他に「Cells」プロパティを使用する方法もあるが、ここでは省略する。

セル A1 に数値データを代入する。	ActiveSheet.Range("A1") = 123456
セル A2 に文字データを代入する。	ActiveSheet.Range("A2") = "Excel VBA"

(ActiveSheet. は省略可)

3-3 セルへの入力方法

- 課題 3 上記の方法に従い、数値・文字列・セル内データの代入を記述し、実行しなさい。

(ActiveSheet. は省略可)

3-4 セルの範囲指定

セル範囲を指定する方法も Range と Cells プロパティが使用される。

セル範囲に値をセットするには"A1","A5"という具合にセルを指定後、カンマ(,)で区切る方法と、

"A1:E7"という具合にセルを指定後、セミコロン(:)で区切る方法とがある。

次の例は、Range プロパティを使用し、Value プロパティに値をセットしている。

```
Range("A1","A5").Value="Excel VBA"  
Range("A1:A5").Value="Excel VBA"
```

- 課題 4 任意の縦横 10×10 のセルに「Hey Hey Hey」と記述させなさい。

3-5 変数宣言・繰り返し処理ほか

- 課題 5 下記マクロ「SmileyFace」は、オートシェイプを画面上のランダムの位置に表示するマクロである。

```
Sub SmileyFace()  
    Dim X As Integer  
    Dim Y As Integer  
  
    X = Int(Rnd * 600) + 1  
    Y = Int(Rnd * 300) + 1  
    ActiveSheet.Shapes.AddShape(msoShapeSmileyFace, X, Y, 50, 50).Select
```

要点 ・変数の宣言方法を理解する。

・オートシェイプのサイズ、出現範囲の設定方法を理解する。

- ・任意の座標を得る方法の例 $X = \text{Int}(\text{Rnd} * 600) + 1$

Rad 関数は 1 未満の乱数を返すが、画面の座標に対応させるには 1 以上の整数の方が都合が良い。

そこで 1 ~ 600 の範囲の乱数を整数値で取得する場合はこのように記述する。

- ・AddShape：オートシェイプ名、位置座標、縦横サイズを入力するためのメソッド
- ・msoShapeSmileyFace：オートシェイプ名

- 課題 6 出現するオートシェイプのサイズを、縦横 2 倍にせなさい。

- 課題 7 オートシェイプの出現範囲を画面一杯に拡大せなさい。

- 課題 8 オートシェイプに色を着けなさい。

参考："="の後は色番号を入力。ここではランダムに着色されるようにしてある。

```
Selection.ShapeRange.Fill.ForeColor.SchemeColor = Int(Rnd * 50) + 1
```

■ 課題 9 複数のオートシェイプが表示されるように改良しなさい。

参考：For ... To ～ Next で指定した回数分の処理を繰り返す方法

```
For カウンタ変数=初期値 To 終了値 (Step きざみ値)
  繰り返す処理
Next カウンタ変数
```

- ※注意
- ・ 終了値は 10 程度に留めておくこと。
 - ・ カウンタ変数の形宣言を追加すること。

■ 課題 10 表示されたオートシェイプが順番に消えていく機能を付加しなさい。

参考：現在出現しているのオートシェイプ個々に名前を付ける。
例えば、For...To ～ Next のループ内でオートシェイプを描いた後に…

```
Selection.Name = "NIKONIKO" & i
```

とすれば、各々のオートシェイプに NIKONIKO1・NIKONIKO2... と名前を付けることができる。ここで「&」は変数との結合を意味する。

続いてこれを消去するには新たにループを設け、ひとつずつ消していけばよい。参

考：名前を付けたオートシェイプを選択する方法…

```
ActiveSheet.Shapes("NIKONIKO" & i).Select
```

```
Selection.Delete
```

参考：選択したオートシェイプを消す方法…

■ 課題 11 ユーザが入力した数のオートシェイプを表示させるように改良しなさい。

参考：InputBox と組み合わせて…

```
N = InputBox("オートシェイプの数を入力してください")
```

■ 課題 12 表示されたオートシェイプに動きをつけたい。

参考：下記ループを解説せよ。

```
For i = 1 To 50
  For j = 1 To N
    With ActiveSheet.Shapes("SP" & j)
      .IncrementTop 2 * (Int(Rnd * 3) - 1)
      .IncrementLeft 2 * (Int(Rnd * 3) - 1)
    End With
  Next j
Next i
```

- 要点
- IncrementTop：指定した図形を垂直方向にポイント単位で移動する。
 - IncrementLeft：指定した図形を水平方向にポイント単位で移動する。
 - DoEvents：時間のかかる処理を実行している場合、ユーザーが行った操作は処理が終了するまで OS に渡らない。そこで処理を中断したい場合に備えて DoEvents 関数で一時的に OS に制御を移すことで、イベントの処理が可能となる。

■ 課題 13 選択・応用問題

VBA の機能は web 上でたくさんのサンプルデータが提供されています。各自で調べ、新たな機能を追加またはオリジナルのソフトを作成しなさい。

4 まとめ

- 各課題の実行結果と感想を、添付ファイルにて提出する。
- メールの件名は「EXCEL 実習__苗字」、添付ファイル名は「半角出席番号苗字(ブランクなし)」とする。
- 課題 13 は余力のある人、高得点を狙い人等個人の状況で選択問題とします。
添付するときは、動作説明を添えること。また参考にした URL 等も併記すること。

<提出課題>

8人の営業スタッフによる、6ヶ月間の契約件数集計表をもとに、次の機能のマクロを作成するとともに、それぞれフォームボタンに登録しなさい。使用するファイルは、VBA課題.xlsx

事前準備：空欄を埋めなさい。

- (1) 50音順による並べ替え
- (2) 営業成績順による並べ替え
- (3) セルB14に入力した人物のデータを、適切なグラフで表示
- (4) データの復元
- (5) データクリア

※表の書式・スタッフ名・数値データは任意とする。

レポートは次の通り。

- ・課題は添付ファイルにして提出。
- ・マクロの制作手順をまとめる

<レポート>

1. 目的 このファイルの目的を参照して記入のこと
2. 予備知識
使用した関数や機能について VBA の勘所をまとめる。
3. マクロ手順まとめ
4. 考察
VBA フォームコントロールとアクティブ X コントロールの違いについて
5. 感想