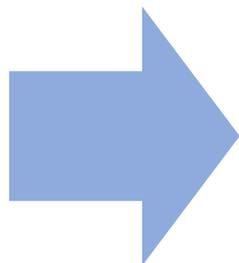
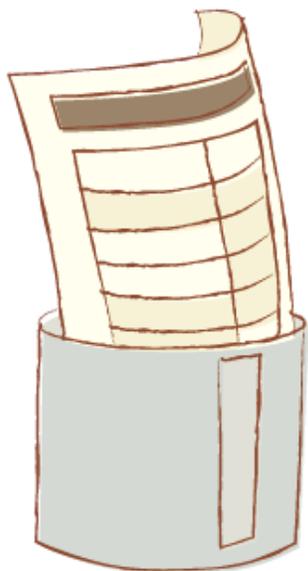


【はじめに】 このコースの進め方





ユーザーフォームを作り、高速入力！



受注フォーム(新規追加)

受注ID

日付 / /

受注元

商品ID

商品名

単価

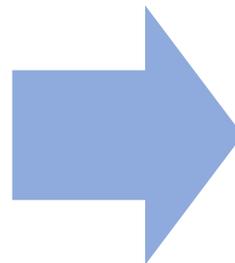
数量

金額

配送済み

請求済み

入金済み



受注一覧表

受注ID	日付	受注元	商品ID	商品名	単価	数量	金額	配送済み	請求済み	入金済み
1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000	2	4,000	済		
2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000	3	9,000	済		
3	2018/7/4	バウボ建設	3	コーヒー豆	10,000	2	20,000	済		
4	2018/7/6	エクセル運輸	3	コーヒー豆	10,000	3	30,000	済		
5	2018/7/11	VBA商事	7	業務用ストロー	900	7	6,300	済		
8	2018/7/10	表計算コーポ	3	紅茶葉	2,000	4	8,000			
9	2018/7/9	エクセル運輸	3	紅茶葉	2,000	5	10,000	済	済	

伝票など

フォーム

Excelシート

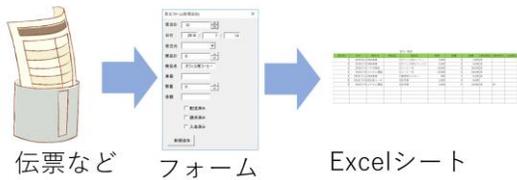


カリキュラムの全体像

- | | |
|-----|----------------------------|
| 第0章 | 全体像の把握、マクロの流れを整理 |
| 第1章 | ユーザーフォームの作成 |
| 第2章 | フォームからデータ入力 |
| 第3章 | 日付の自動入力、ComboBox（フォームの初期化） |
| 第4章 | フォームで数式やVLOOKUP関数を用いる |
| 第5章 | スピンボタンの利用！マウスで操作できるようにする |
| 第6章 | 発展編（データ更新フォームを作る） |



コース構成



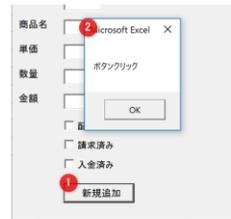
①ストーリー（導入）

「伝票の形式をそのままフォームに入力すれば
 どんどんExcelにデータ入力できる。」
 「ほほう？」

ボタンがクリックされた際の処理②
 とりあえずメッセージボックスを表示させてみよう

- (1) 「MsgBox "ボタンクリック"」と入力してみる
- (2) メッセージボックスが表示されるようになる

```
Private Sub btnAdd_Click()
    MsgBox "ボタンクリック"
End Sub
```



②講義（座学）

③実習（手を動かす）

受注ID	日付	受注元	商品ID	商品名	単価	数量
1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000	2
2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000	3
3	2018/7/4	バウボ建設	3	コーヒー豆	10,000	2
4	2018/7/6	エク			10,000	3
5	2018/7/11	VBA			900	7
6	2018/7/10	表計			2,000	4
7	2018/7/9	関数			2,000	5

次の章へ



Udemyでの受講

- ベネッセと提携
- オンラインで何度も受講
- PC版,アプリ版で倍速再生
- アプリで映像ダウンロード
→通勤通学中オフライン受講
- Q&Aにて講師に質問
- 返金保証



速習したい方へ



ダウンロード

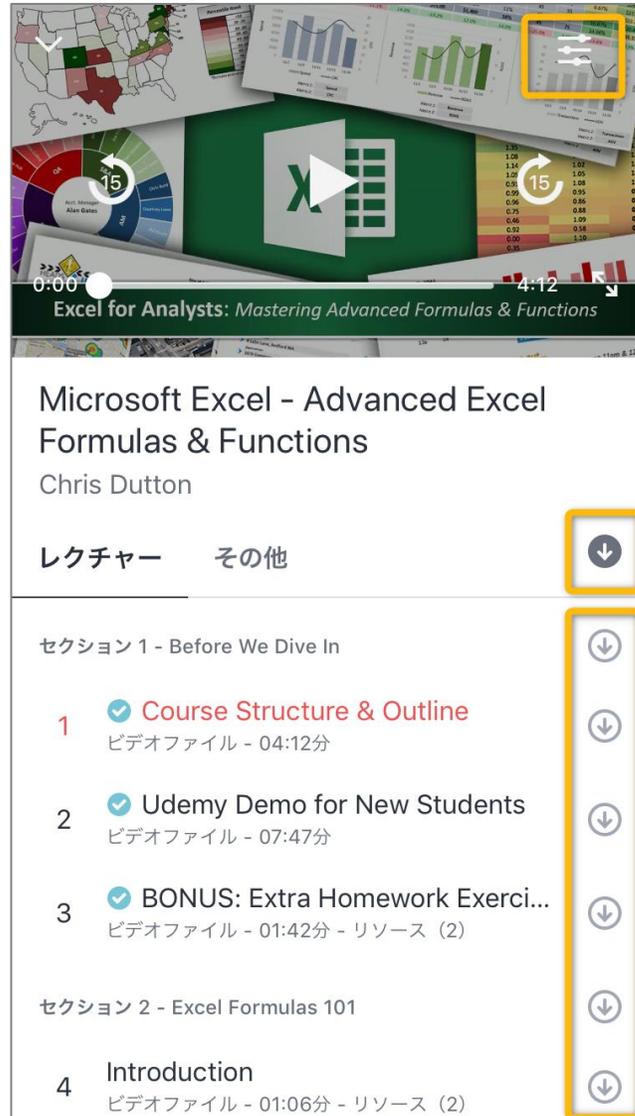
- テキスト (PDF)
- **アプリ**で動画を持ち歩ける



倍速再生

- PCか**アプリ**
- 1.2倍速 1.5倍速
2倍速など

Udemyアプリ (iPhone,Android)



全編
ダウンロード

個々に
ダウンロード





[実習]受注元～金額を各列に代入させよう

それぞれのオブジェクトから値を取得して、各列の最終行(MaxRow行)に代入しよう

入力フォーム

受注ID: _____

日付: _____ / _____ / _____

受注元: _____ (5)

商品ID: _____ (6)

商品名: _____ (7)

単価: _____ (8)

数量: _____ (9)

金額: _____ (10)

配送済み
 請求済み
 入金済み

新規追加

受注ID	日付	受注元	商品ID	商品名	単価	数量	金額	配送済み	請求済み	入金済み
1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000	2	4,000	済		
2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000	3	9,000	済		
3	2018/7/4	バクテラ建設	3	コーヒー豆	10,000	2	20,000	済		
4	2018/7/6	エタセル運輸	3	コーヒー豆	10,000	3	30,000	済		
5	2018/7/11	VBA商事	7	業務用ストロー	900	7	6,300	済		
6	2018/7/10	表計算コーポ	3	紅茶葉	2,000	4	8,000			
7	2018/7/9	エタセル運輸	3	紅茶葉	2,000	5	10,000	済		

概要	オブジェクト名	代入するセルの列
⑤ 受注元Combobox	cmbComp	C列
⑥ 商品IDのTextbox	txtProdId	D列
⑦ 商品名のTextbox	txtProdName	E列
⑧ 単価のTextbox	txtPrice	F列
⑨ 数量のTextbox	txtNum	G列
⑩ 金額のTextbox	txtTotal	H列

読者特典

- 講義テキストPDF (100ページ+)
- 印刷用の縮小版テキスト (ページ数を1/6に削減)



Checkboxの結果をセルに代入するには？

チェックが入っているかどうかにより、条件分岐させる

入力フォーム

受注ID: 123

日付: 2018 / 10 / 30

受注元: 会社A

商品ID: 5

商品名: コーヒー

単価: 10000

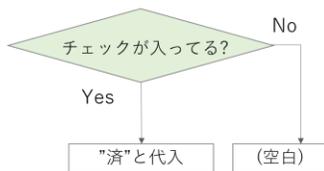
数量: 3

金額: 30000

配送済み
 請求済み
 入金済み

新規追加

- 配送済み (chkDeliv)
- 請求済み (chkInv)
- 入金済み (chkPay)



受注ID	日付	受注元	商品ID	商品名	単価	数量	金額	配送済み	請求済み	入金済み
1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000	2	4,000	済		
2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000	3	9,000	済		
3	2018/7/4	バクテラ建設	3	コーヒー豆	10,000	2	20,000	済		
4	2018/7/6	エタセル運輸	3	コーヒー豆	10,000	3	30,000	済		
5	2018/7/11	VBA商事	7	業務用ストロー	900	7	6,300	済		
6	2018/7/10	表計算コーポ	3	紅茶葉	2,000	4	8,000			
7	2018/7/9	エタセル運輸	3	紅茶葉	2,000	5	10,000	済		
123	2018/10/20	会社A	5	コーヒー	10,000	3	30,000	済		済



サンプルマクロを使ってみよう

- ✓ サンプルマクロをダウンロードして実行
- ✓ 「form_sample.xlsx」

(※ダウンロード配布ファイルを開いてください。)

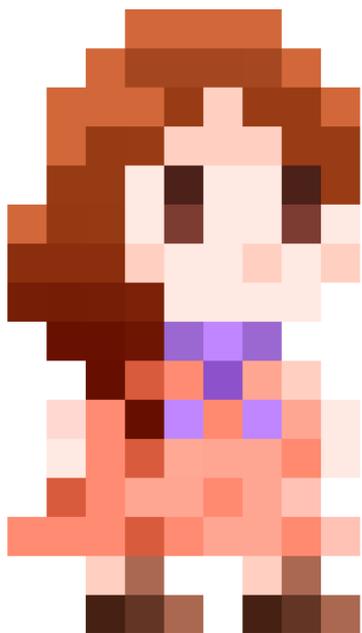
【第0章】

マクロを作る、その前に！
大まかな流れを把握しよう





フォームを便利に作り込んでいく…
でも、作り込めば**キリが無い!**



ユーザー

- ・こうして欲しい
- ・ああして欲しい
- ・こうなると便利

受注フォーム(新規追加) ×

受注ID	10
日付	2018 / 7 / 14
受注元	
商品ID	0
商品名	オフィス用コーヒー
単価	
数量	0
金額	

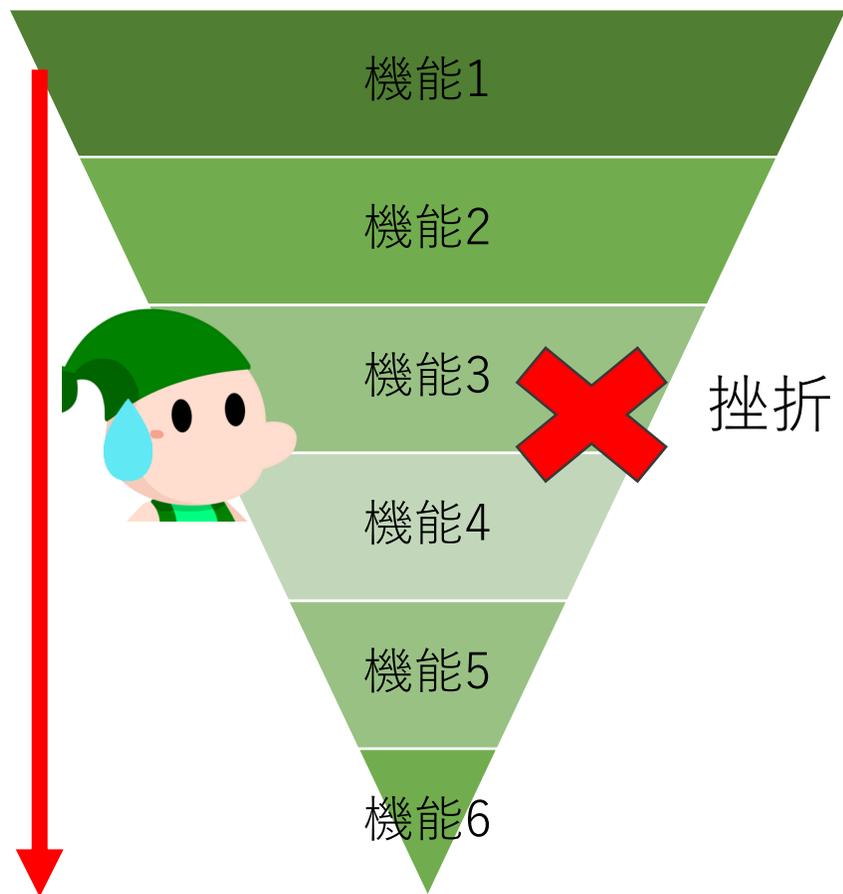
配送済み
 請求済み
 入金済み

新規追加

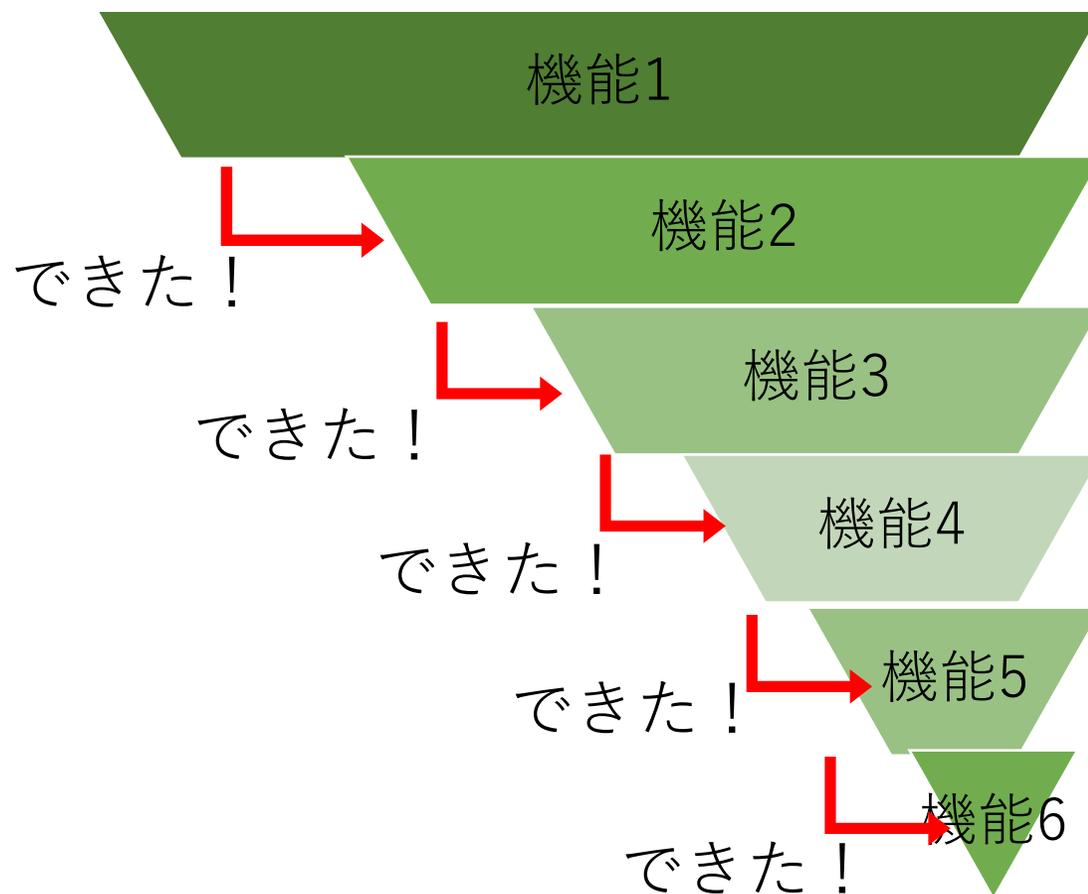


学習の順番を間違えると、挫折しやすい

①はじめから全部作り込もうとする



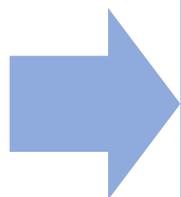
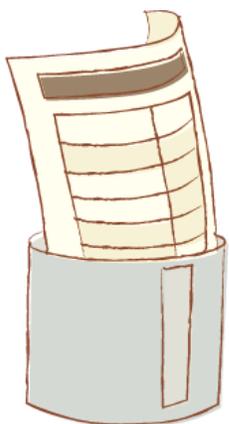
②1つずつ作り、「できた」を実感する





そこで、本コースの流れは…

(前半) シンプルに完成!



受注フォーム(新規追加)

受注ID 10

日付 2018 / 7 / 14

受注元

商品ID 0

商品名 オフィス用コーヒー

単価

数量 0

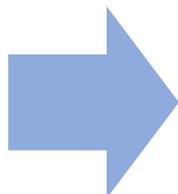
金額

配送済み

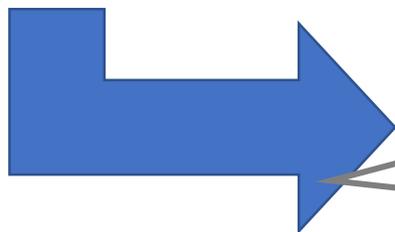
請求済み

入金済み

新規追加



受注一覧表										
受注ID	日付	受注元	商品ID	商品名	単価	数量	金額	配送済み	請求済み	入金済み
1	2018/6/1	VBA協事	1	オフィス用コーヒー	2,000	2	4,000	済		
2	2018/7/3	VBA協事	2	オフィス用カフェラテ	3,000	3	9,000	済		
3	2018/7/4	エクスセル運輸	2	コーヒー豆	10,000	2	20,000	済		
4	2018/7/6	エクスセル運輸	3	コーヒー豆	10,000	3	30,000	済		
5	2018/7/11	VBA協事	7	業務用ストロー	900	7	6,300	済		
8	2018/7/10	表計算コーポ	3	紅茶葉	2,000	4	8,000		済	
9	2018/7/9	エクスセル運輸	3	紅茶葉	2,000	5	10,000	済	済	



(後半)
便利な機能を盛り込んでいく

【第1章】

ユーザーフォームの作成とその基礎





課題ファイルを開いて下さい

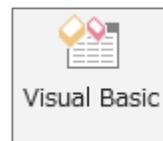
- ✓ サンプルマクロをダウンロードして実行
- ✓ 「vba3rd_1-1.xlsx」

(※ダウンロード配布ファイルを開いてください。)

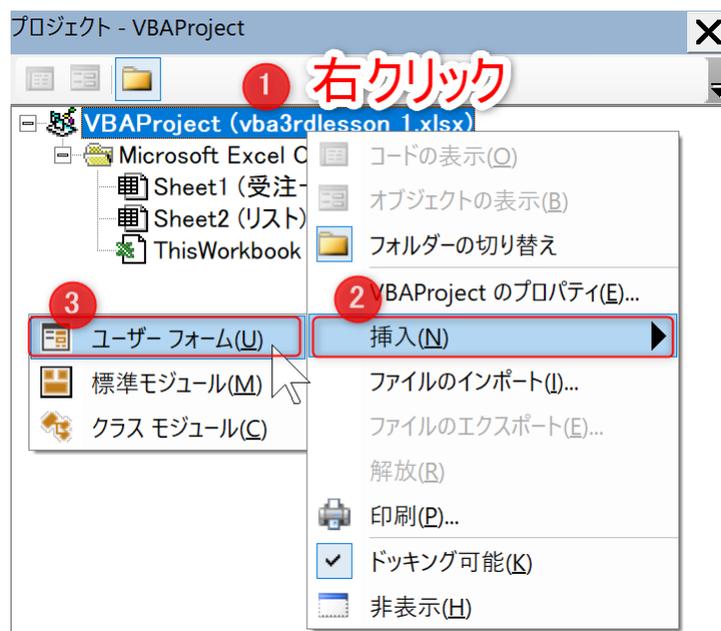


ユーザーフォームを作成しよう

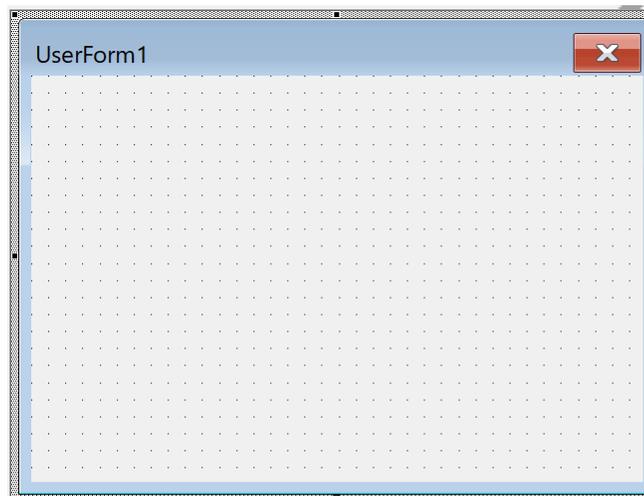
VBEを開いておく（[開発]



か、[Alt] + F11)



ユーザーフォームが作成される





プロパティの見かた

プロパティ - UserForm1	
UserForm1 UserForm	
全体 項目別	
(オブジェクト名)	UserForm1
BackColor	<input type="text" value=" &H8000000F&"/>
BorderColor	<input type="text" value=" &H80000012&"/>
BorderStyle	0 - fmBorderStyleNone
Caption	UserForm1
Cycle	0 - fmCycleAllForms
DrawBuffer	32000
Enabled	True
Font	MS UI Gothic
ForeColor	<input type="text" value=" &H80000012&"/>
Height	288.05
HelpContextID	0
KeepScrollBarsVisible	3 - fmScrollBarsBoth
Left	0
MouseIcon	(なし)
MousePointer	0 - fmMousePointerDefault
Picture	(なし)
PictureAlignment	2 - fmPictureAlignmentCenter
PictureSizeMode	0 - fmPictureSizeModeClip
PictureTiling	False
RightToLeft	False
ScrollBars	0 - fmScrollBarsNone
ScrollHeight	0
ScrollLeft	0
ScrollTop	0
ScrollWidth	0
ShowModal	True
SpecialEffect	0 - fmSpecialEffectFlat
StartPosition	1 - オーナー フォームの中央
Tag	
Top	0
WhatsThisButton	False
WhatsThisHelp	False
Width	224
Zoom	100

(オブジェクト名)	UserForm1
BackColor	<input type="text" value=" &H8000000F&"/>
BorderColor	<input type="text" value=" &H80000012&"/>
BorderStyle	0 - fmBorderStyleNone
Caption	UserForm1
Cycle	0 - fmCycleAllForms

プログラム上の呼び名

フォームの上部に
表示される文言

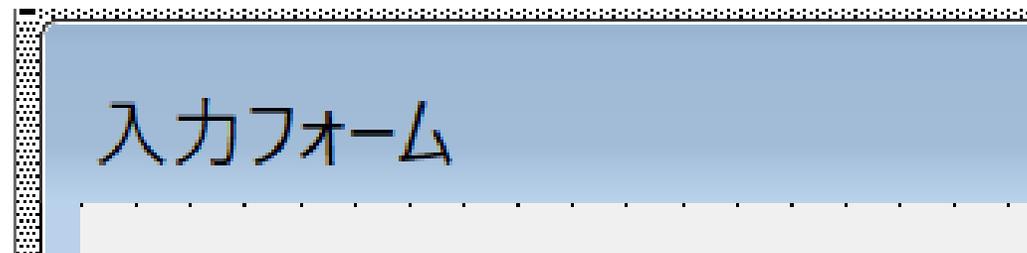


[実習] フォームのCaptionを変更しよう

☑ Captionを「入力フォーム」に変更

UserForm1 UserForm	
全体	項目別
(オブジェクト名)	UserForm1
BackColor	&H8000000F&
BorderColor	&H80000012&
BorderStyle	0 - fmBorderStyleNone
Caption	入力フォーム
Cycle	0 - fmCycleAllForms
DrawBuffer	32000
Enabled	True

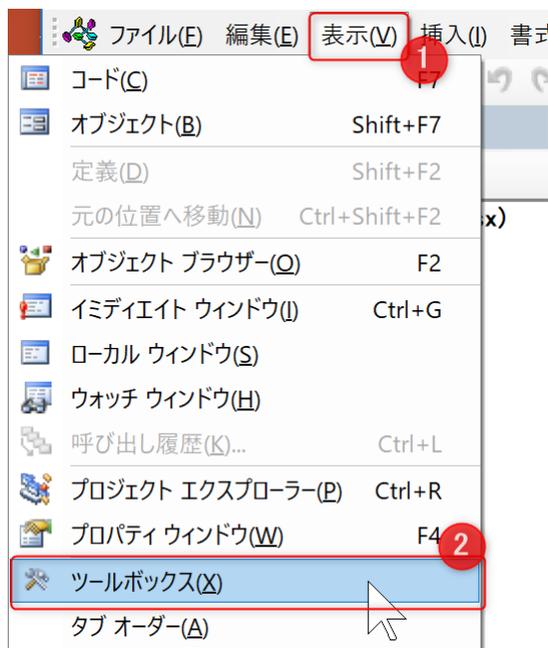
変更された



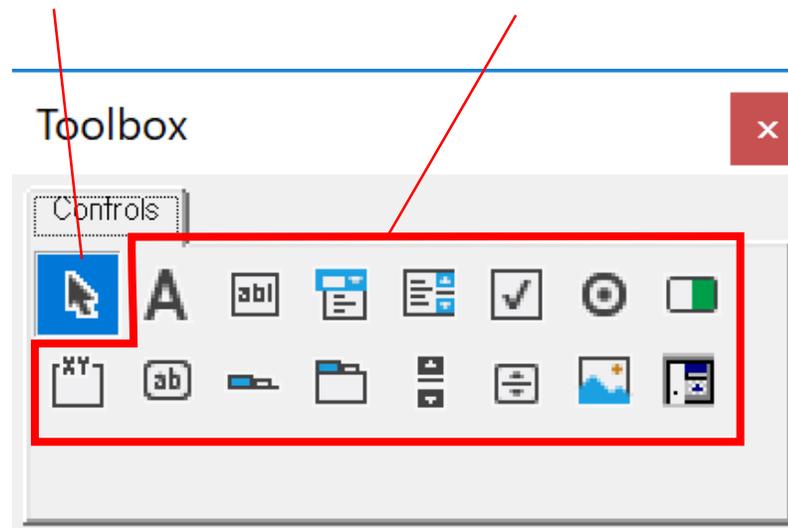


ツールボックスを表示させる

☑ 表示 > ツールボックス



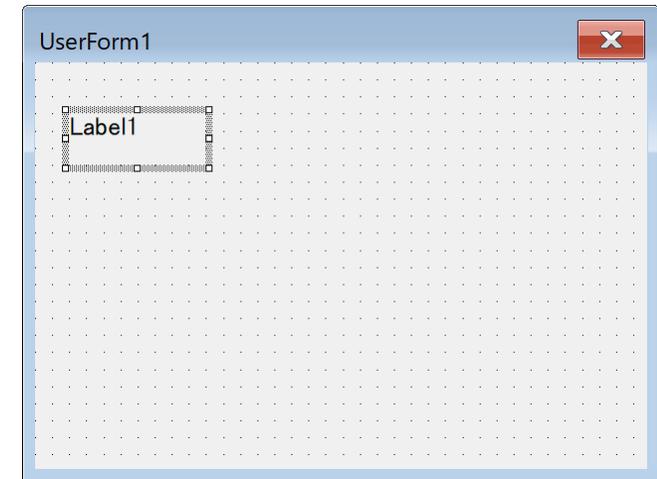
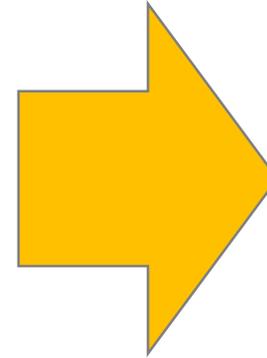
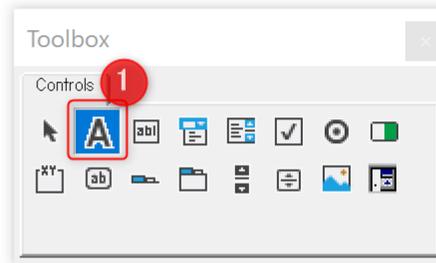
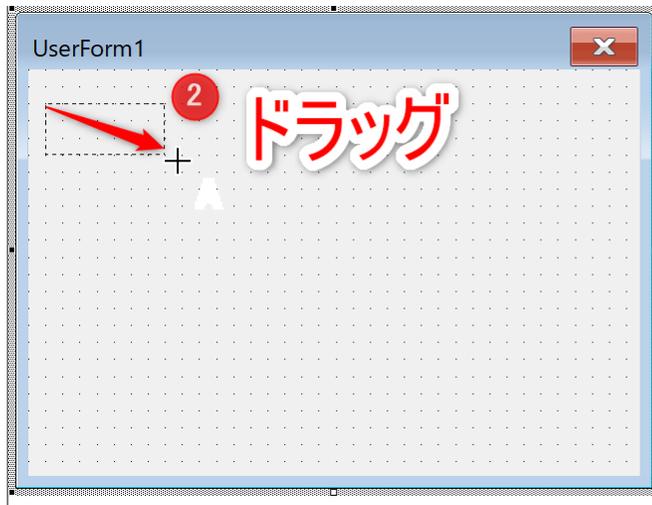
オブジェクトの選択 オブジェクトを配置





[実習] ラベルを設置してみよう

- ✓ ラベルボタンを押してドラッグ





ラベルのプロパティの見かた

プロパティ - Label1	
Label1 Label	
全体 項目別	
(オブジェクト名)	Label1
Accelerator	
AutoSize	False
BackColor	<input type="checkbox"/> &H8000000F&
BackStyle	1 - fmBackStyleOpaque
BorderColor	<input checked="" type="checkbox"/> &H80000006&
BorderStyle	0 - fmBorderStyleNone
Caption	Label1
ControlTipText	
Enabled	True
Font	MS UI Gothic
ForeColor	<input checked="" type="checkbox"/> &H80000012&
Height	18
HelpContextID	0
Left	12
MouseIcon	(なし)
MousePointer	0 - fmMousePointerDefault
Picture	(なし)
PicturePosition	7 - fmPicturePositionAboveCenter
SpecialEffect	0 - fmSpecialEffectFlat
TabIndex	0
TabStop	False
Tag	
TextAlign	1 - fmTextAlignLeft
Top	18
Visible	True
Width	48
WordWrap	True

(オブジェクト名)	Label1
Accelerator	
AutoSize	False
BackColor	<input type="checkbox"/> &H8000000F&
BackStyle	1 - fmBackStyleOpaque
BorderColor	<input checked="" type="checkbox"/> &H80000006&
BorderStyle	0 - fmBorderStyleNone
Caption	Label1
ControlTipText	
Enabled	True
Font	MS UI Gothic
ForeColor	<input checked="" type="checkbox"/> &H80000012&
Height	18

プログラム上の呼び名
(変更しなくても良い)

ラベルに表示させる文言
(フォームに表示される)

フォントの種類
(指定したい場合)



[実習] ラベルのCaptionを「受注ID」に変更

A screenshot of a Windows-style application window titled "入力フォーム" (Input Form). The window has a light blue title bar with a close button (X) on the right. The main area of the window is a white grid. In the top-left corner of the grid, there is a label with the text "受注ID" (Order ID). The label is enclosed in a red rectangular box, indicating it is the focus of the current task.



[実習] その他のラベルを設置してみよう

入力フォーム

受注ID
日付
受注元
商品ID
商品名
単価
数量
金額

ラベルを作る

下に拡大

✓ ※速習したい方へ...

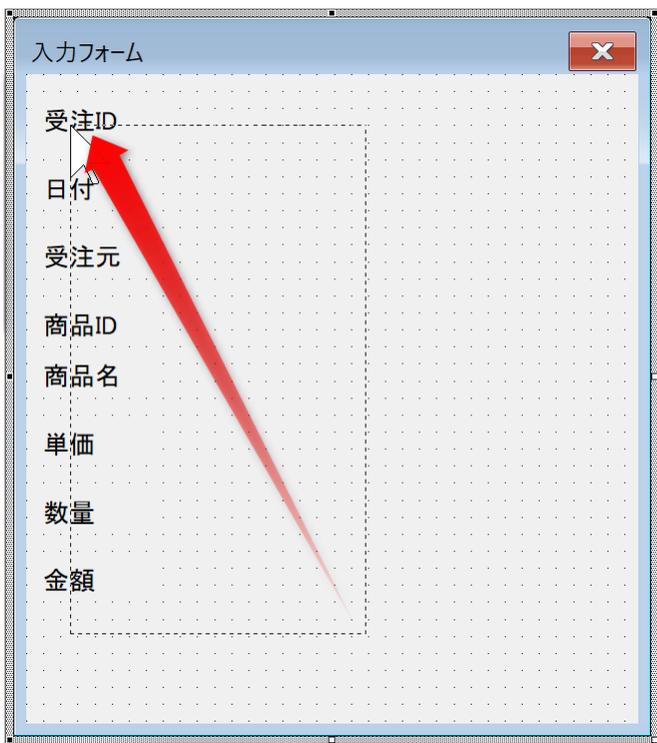
(ラベルを配置する作業は
少々手間がかかります。)

ラベルを設置済みのファイルを
ダウンロード配布します。

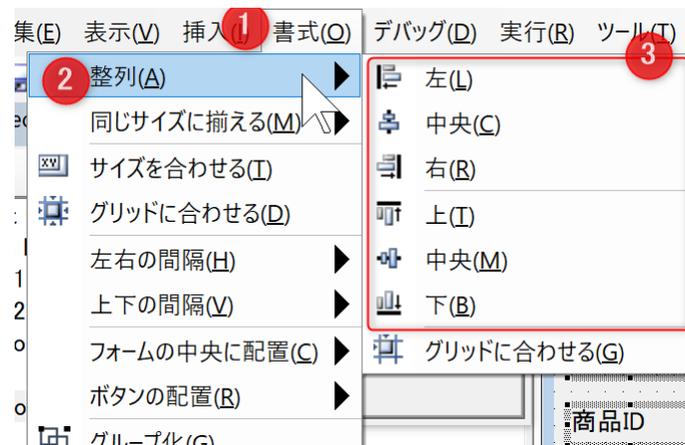


ラベルのレイアウトを整えるには

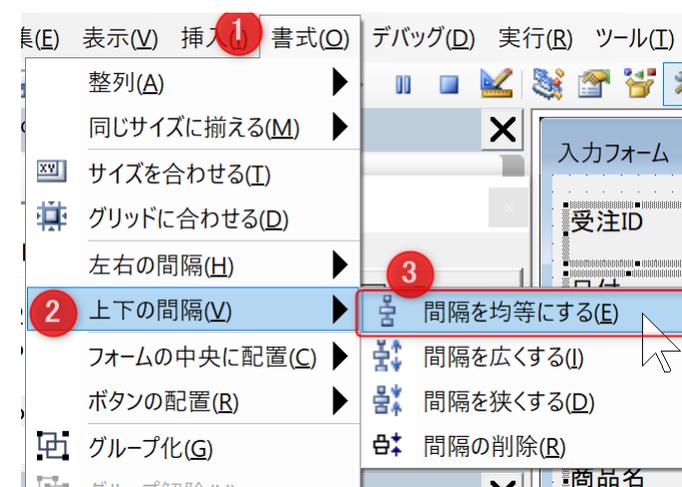
[1] まとめて選択



[2] 整列させる



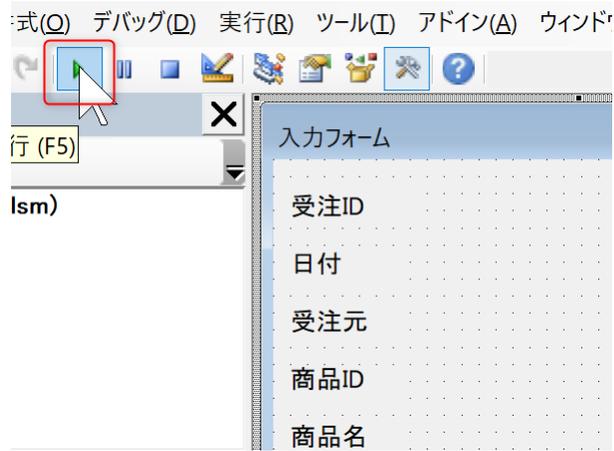
[3] 上下の間隔を均等に





フォームを起動してみる

[1]実行ボタン(またはF5)



[2]フォームが起動([×]ボタンで閉じる)

受注一覧表		入力フォーム		
	商品名	受注ID	配送済み	請
1	オフィス用コーヒ		00	済
2	オフィス用カフェ	日付	00	済
3	コーヒー豆	受注元	00	済
3	コーヒー豆	商品ID	00	済
7	業務用ストロー	商品名	00	
3	紅茶葉	単価	00	済
3	紅茶葉	数量	00	済
		金額		



[実習] ボタンを押せばフォームを起動 できるようにしておこう

✔ ボタンで起動できるように！

受注ID	日付	受注元	商品ID	商品名	配送
1	2018/6/1	VBA商事	1	オフィス用コーヒ	0 済
2	2018/7/3	VBA商事	2	オフィス用カフェ	0 済
3	2018/7/4	パワポ建設	3	コーヒー豆	0 済
4	2018/7/6	エクセル運輸	3	コーヒー豆	0 済
5	2018/7/11	VBA商事	7	業務用ストロー	0 済
8	2018/7/10	表計算コーポ	3	紅茶葉	0
9	2018/7/9	エクセル運輸	3	紅茶葉	0 済

[1]開発>挿入>「ボタン」

開発

挿入

ボタン (フォームコントロール)



[実習] その他のオブジェクトも配置しよう

入力フォーム

受注ID	<input type="text"/>
日付	<input type="text"/> / <input type="text"/> / <input type="text"/>
受注元	<input type="text"/>
商品ID	<input type="text"/>
商品名	<input type="text"/>
単価	<input type="text"/>
数量	<input type="text"/>
金額	<input type="text"/>

配送済み
 請求済み
 入金済み

新規追加

✔ ※速習したい方へ...

(オブジェクトをきれいに配置する作業は、少々手間がかかります。)

オブジェクトを配置済みのファイルをダウンロード配布します。



その他のオブジェクトの種類 (一部)

入力フォーム

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

新規追加



TextBox(テキストボックス)



ComboBox(コンボボックス)



CheckBox(チェックボックス)



CommandButton (コマンドボタン)

A

Label (ラベル)

受注ID

日付 / /

受注元

プロパティ「BackColor」を
「0 - fmBackColorTransparent」
に変更すると背景が透過します



[実習]Tabキーで順番に移動できる設定

TabIndexを変更しよう

Tabキー
で移動

入力フォーム

受注ID	<input type="text"/>
日付	<input type="text"/> / <input type="text"/> / <input type="text"/>
受注元	<input type="text"/>
商品ID	<input type="text"/>
商品名	<input type="text"/>
単価	<input type="text"/>
数量	<input type="text"/>
金額	<input type="text"/>
<input type="checkbox"/> 配送済み	
<input type="checkbox"/> 請求済み	
<input type="checkbox"/> 入金済み	
<input type="button" value="新規追加"/>	

TabIndexの数値を順番にオブジェクトに振る

プロパティ - TextBox1

TextBox1 TextBox	
全体	項目別
HideSelection	True
IMEMode	0 - fm
IntegralHeight	True
Left	48
Locked	False
MaxLength	0
MouseIcon	(なし)
MousePointer	0 - fm
MultiLine	False
PasswordChar	
ScrollBars	0 - fm
SelectionMargin	True
SpecialEffect	2 - fm
TabIndex	1
TabKeyBehavior	False
TabStop	True
Tag	
Text	
TextAlign	1 - fmTextAlignL
Top	12
Value	
Visible	True
Width	66
WordWrap	True



[実習]オブジェクト名を変更しよう

(人間にも、プログラミング的にも)わかりやすい名前をつけよう

(例) 受注IDを入力するTextboxなら 「txtOrdId」 など

入力フォーム

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

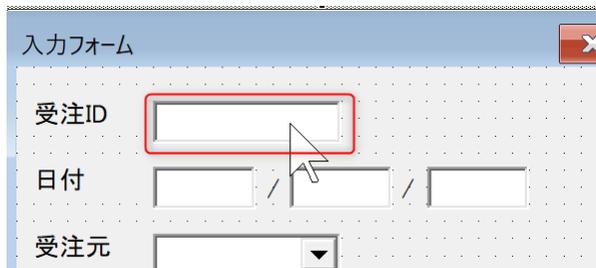
新規追加

	概要	オブジェクト名
①	受注IDのTextbox	txtOrdId
②	年のTextbox	txtYear
③	月のTextbox	txtMonth
④	日のTextbox	txtDay
⑤	受注元Combobox	cmbComp
⑥	商品IDのTextbox	txtProdId
⑦	商品名のTextbox	txtProdName
⑧	単価のTextbox	txtPrice
⑨	数量のTextbox	txtNum
⑩	金額のTextbox	txtTotal
⑪	配送済みのCheckbox	chkDeliv
⑫	請求済みのCheckbox	chkInv
⑬	入金済みのCheckbox	chkPay
⑭	新規追加のButton	btnAdd



入力モードを決めるIME Modeプロパティ

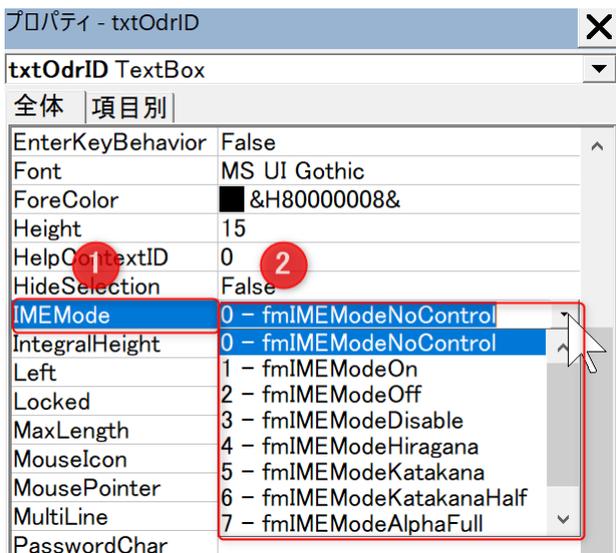
入力システムを決めて、快適な入力を手助けしよう



商品名



例えば、商品名のテキストボックスに移行したとき自動的に日本語モードに (IME… “Input Method Editor”文字入力において漢字カナ変換などを制御)



	選択肢	説明
0	fmIMEModeNoControl	IME のモードを変更しない
1	fmIMEModeOn	IME をオンにする
2	fmIMEModeOff	IME をオフにし、英語モードに
3	fmIMEModeDisable	IME をオフにする。(手動で変更不能に)
4	fmIMEModeHiragana	全角ひらがなモードで IME をオンにする
5	fmIMEModeKatakana	全角カタカナ モードで IME をオンにする
6	fmIMEModeKatakanaHalf	半角カタカナ モードで IME をオンにする
7	fmIMEModeAlphaFull	全角英数モードで IME をオンにする
8	fmIMEModeAlpha	半角英数モードで IME をオンにする



[実習]IME Modeを変更しよう

適切なIME Modeに変更しておく

入力フォーム

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み
 請求済み
 入金済み

新規追加

	概要	種類		選択肢	
①	受注IDのTextbox	txtOrdId	3	fmIMEModeDisable	IMEをオフ (手動変更不能)
②	年のTextbox	txtYear	//	//	
③	月のTextbox	txtMonth	//	//	
④	日のTextbox	txtDay	//	//	
⑤	受注元Combobox	cmbComp	4	fmIMEModeHiragana	全角ひらがな
⑥	商品IDのTextbox	txtProdId	3	fmIMEModeDisable	IMEをオフ
⑦	商品名のTextbox	txtProdName	4	fmIMEModeHiragana	全角ひらがな
⑧	単価のTextbox	txtPrice	3	fmIMEModeDisable	IMEをオフ
⑨	数量のTextbox	txtNum	3	fmIMEModeDisable	//
⑩	金額のTextbox	txtTotal	3	fmIMEModeDisable	//



入力文字数を制限するMaxLengthプロパティ

テキストボックスに文字数制限を設定し、後のエラー要因を防いでおこう

入力フォーム

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

新規追加

(日付)

「2018 / 09 / 30」のように、【4桁 / 2桁 / 2桁】のみ入力させたい
(後々に、プログラムのエラーの元になることを未然に防ぐため)



[実習] 日付のMaxLengthを変更しよう

事前に制限文字数を設定しておき、後のエラー要因を防いでおこう

MaxLengthプロパティを変更しておく

入力フォーム

受注ID

日付 / /

年 : 4 桁
月 : 2 桁
日 : 2 桁

プロパティ - txtYear

txtYear TextBox

全体 項目別

Left	48
Locked	False
MaxLength	4
MouseIcon	(なし)
MousePointer	0 - fmMousePointerDefault
MultiLine	False
PasswordChar	
ScrollBars	0 - fmScrollBarsNone



ふ〜。フォームの設置ができたぞう！



あれ？でも…
これ、何も起こらないっすよ？



あ… そうかあ、
まだ本番はこれからってわけだね！



うむ。これでフォームを
起動できるようになったな。



そうだ。今のところは何も起こらない。
なぜなら、**ボタンを押したら何をさせるかを
まだマクロに記述していない**からな。



そういうわけだ。**ここからが
大事**ということだな。では行くぞ！

【第2章】

フォームに入力されたデータを
ワークシートに転記する！





うん。そこで、
ボタンを押したら何をするかを
マクロに記述するわけだね？



え〜と、
フォームに入力されたデータを
Excelシートに転記する
って事っスね？



…というわけで、先ほど話したが
このままでは**フォームは起動**できるが
ボタンを押しても何も起こらない。



その通りだ。
では、ボタンを押したら何をさせれば
いいか、わかっているな？



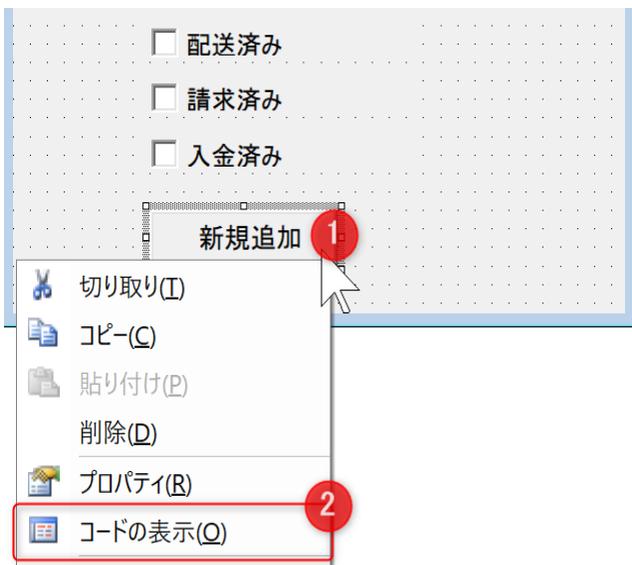
そういうわけだ！
では、早速やってみよう。



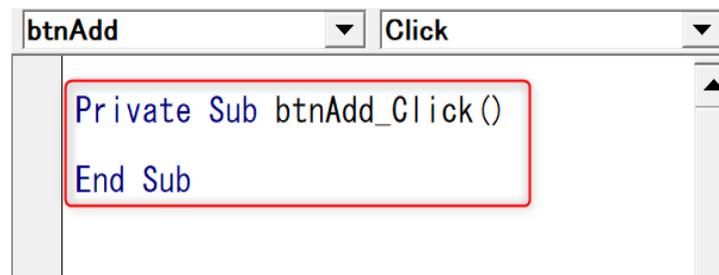
[実習] ボタンがクリックされた際の処理①

Clickイベントに対するプロシージャを挿入しよう

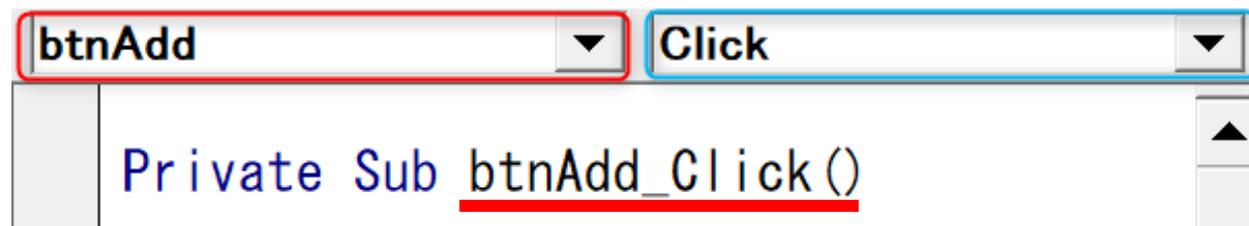
(1) 右クリックして「コードの表示」



(2) コードが自動挿入される



btnAddオブジェクトがClickされたときのイベントプロシージャ(処理)ということ





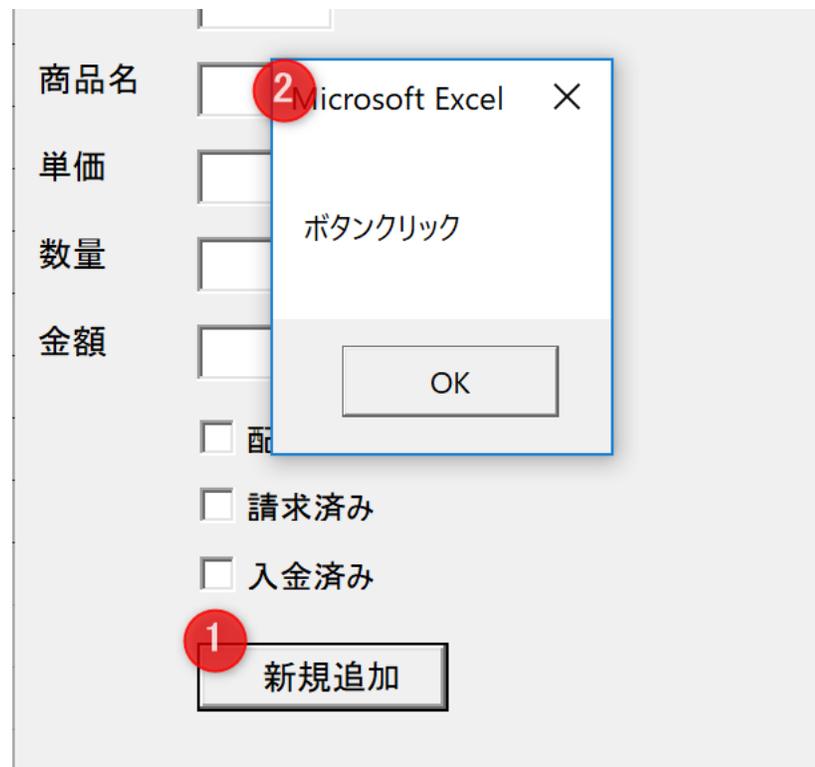
ボタンがクリックされた際の処理②

とりあえずメッセージボックスを表示させてみよう

(1) 「MsgBox “ボタンクリック”」 と入力してみる

```
Private Sub btnAdd_Click()  
    MsgBox “ボタンクリック”  
End Sub
```

(2) メッセージボックスが表示されるようになる





実際に実行させたい処理

フォームのデータを1つずつセルに転記する

入力フォーム

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

新規追加

(以下、順々にデータを転記する)

	A	B	C	D	E	F	G	H	I	J	K
1											
2			新規追加								
3					受注一覧表						
4	受注ID	日付	受注元	商品ID	商品名	単価	数量	金額	配送済み	請求済み	入金済み
5	1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000	2	4,000	済		
6	2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000	3	9,000	済		
7	3	2018/7/4	パワポ建設	3	コーヒー豆	10,000	2	20,000	済		
8	4	2018/7/6	エクセル運輸	3	コーヒー豆	10,000	3	30,000	済		
9	5	2018/7/11	VBA商事	7	業務用ストロー	900	7	6,300	済		
10	6	2018/7/10	表計算コーポ	3	紅茶葉	2,000	4	8,000			
11	7	2018/7/9	エクセル運輸	3	紅茶葉	2,000	5	10,000	済	済	
12											
13											
14											
15											

ワークシート「受注一覧」



まずは受注ID！値をセルに代入するには？

.Valueプロパティで値を取得し、セル範囲に代入する

入力フォーム

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

新規追加

```
Range("A12").Value = txtOrdId.Value
```

セル範囲A12の値

txtOrdIdの値

	A	B	C	D	E	F	G	H	I	J	K
1											
2			新規追加								
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											

ワークシート「受注一覧」



最終行を取得する(復習)

VBA講座第2弾コースで解説した方法の復習です。

ファイルの終端からたどり、最終行を見つける

```
Cells(Rows.Count, 1).End(xlUp).Row+1
```

Cells(行番号,列番号)

セル範囲を番地で指定する。

例)

Cells(2,4)なら2行4列 …セルD2

Cells(5,2)なら5行2列 …セルB5

Rows.Count

行の最大数を返す
(旧Excelなら65,536
最近のExcelは1,048,576)

	A	B	C	D	E
1					
2			新規追加		
3					受注一覧表
4	受注ID	日付	受注元	商品ID	商品名
5	1	2018/6/1	VBA商事	1	オフィス用コーヒ
6	2	2018/7/3	VBA商事	2	オフィス用カフ
7	3	2018/7/4	パワポ建設	3	コーヒー豆
8	4	2018/7/6	エクセル運輸	3	コーヒー豆
9	5	2018/7/11	VBA商事	7	業務用ストロー
10	6	2018/7/10	表計算コーポ	3	紅茶葉
11	7	2018/7/9	エクセル運輸	3	紅茶葉
12					
13					
14					
15					

End(xlUp)

1048573
1048574
1048575
1048576

Rows.Count



[実習]受注IDを最終行に代入しよう

最終行を変数MaxRowに取得して、A列の最終行(MaxRow行)に値を代入する

```
Private Sub btnAdd_Click()
```

```
MsgBox "ボタンクリック"
```

この行を削除

```
Dim MaxRow As Integer 変数MaxRowを宣言(整数型)
```

```
MaxRow = Cells(Rows.Count, 1).End(xlUp).Row + 1
```

MaxRowに最終行を代入

```
Range("A" & MaxRow).Value = txtOdrID.Value
```

A列のMaxRow行にtxtOdrIDの値を代入

```
End Sub
```



[実習の結果]

受注IDに入力したデータが、A列の最終行に代入された

	A	B	C	D	E	F	G
1							
2			新規追加 1				
3							
4	受注ID	日付	受注元	商品ID	受		
5	1	2018/6/1	VBA商事		1		
6	2	2018/7/3	VBA商事		2		
7	3	2018/7/4	パワポ建設		3		
8	4	2018/7/6	エクセル運輸		3		
9	5	2018/7/11	VBA商事		7		
10	6	2018/7/10	表計算コーポ		3		
11	7	2018/7/9	エクセル運輸		3		
12	123						
13							
14							
15							
16							
17							
18							
19							
20							
21							

入力フォーム

受注ID 2

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

新規追加 3

4



データを結合するには？(復習)

「&」を用いて文字列を結合する

日付

2018

/

10

/

21



2018

&

"/"

&

10

&

"/"

&

21



2018 / 10 / 21

「&」で文字列を結合



[実習]日付をB列に代入しよう

“/”区切りでデータを結合して、B列の最終行(MaxRow行)に値を代入する

```
Private Sub btnAdd_Click()
```

```
Dim MaxRow As Integer
```

```
MaxRow = Cells(Rows.Count, 1).End(xlUp).Row + 1
```

コメントをつけておく

```
' 受注ID
```

```
Range("A" & MaxRow).Value = UserForm1.txtOdrID.Value
```

```
' 日付
```

```
Range("B" & MaxRow).Value = txtYear.Value & "/" & _
```

```
txtMonth.Value & "/" & _
```

```
txtDay.Value
```

途中で改行する際は
「_」をつけてEnter



[実習の結果]

日付に入力したデータが、B列の最終行に代入された

	A	B	C	D	E	F	G	H
1								
2			新規追加 1					
3					受注一覧表			
4	受注ID	日付	受注元	商品ID	商品名			
5	1	2018/6/1	VBA商事	1	オフィス用コ			
6	2	2018/7/3	VBA商事	2	オフィス用カ			
7	3	2018/7/4	パワポ建設	3	コーヒー豆			
8	4	2018/7/6	エクセル運輸	3	コーヒー豆			
9	5	2018/7/11	VBA商事	7	業務用ストロ			
10	6	2018/7/10	表計算コーポ	3	紅茶葉			
11	7	2018/7/9	エクセル運輸	3	紅茶葉			
12	123	2018/10/21						
13			4					
14								
15								
16								
17								
18								
19								
20								
21								

入力フォーム

受注ID

日付 / / 2

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

3



[実習]受注元～金額を各列に代入させよう

それぞれのオブジェクトから値を取得して、各列の最終行(MaxRow行)に代入しよう

入力フォーム

受注ID

日付 / /

受注元 5

商品ID 6

商品名 7

単価 8

数量 9

金額 10

配送済み
 請求済み
 入金済み

新規追加

	A	B	C	D	E	F	G	H	I	J	K
1											
2			新規追加								
3			受注一覧表								
4	受注ID	日付	受注元	商品ID	商品名	単価	数量	金額	配送済み	請求済み	入金済み
5	1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000	2	4,000	済		
6	2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000	3	9,000	済		
7	3	2018/7/4	パワポ建設	3	コーヒー豆	10,000	2	20,000	済		
8	4	2018/7/6	エクセル運輸	3	コーヒー豆	10,000	3	30,000	済		
9	5	2018/7/11	VBA商事	7	業務用ストロー	900	7	6,300	済		
10	6	2018/7/10	表計算コーポ	3	紅茶葉	2,000	4	8,000			
11	7	2018/7/9	エクセル運輸	3	紅茶葉	2,000	5	10,000	済	済	
12											
13											

	概要	オブジェクト名	代入するセルの列
⑤	受注元Combobox	cmbComp	C列
⑥	商品IDのTextbox	txtProdId	D列
⑦	商品名のTextbox	txtProdName	E列
⑧	単価のTextbox	txtPrice	F列
⑨	数量のTextbox	txtNum	G列
⑩	金額のTextbox	txtTotal	H列



[実習] 答え

各列の最終行 (MaxRow行) のセルに、各オブジェクトの値を代入する

```
Private Sub btnAdd_Click()
```

```
Dim MaxRow As Integer
```

```
MaxRow = Cells(Rows.Count, 1).End(xlUp).Row + 1
```

```
' 受注ID
```

```
Range("A" & MaxRow).Value = UserForm1.txtOdrID.Value
```

```
' 日付
```

```
Range("B" & MaxRow).Value = txtYear.Value & "/" & _  
txtMonth.Value & "/" & _  
txtDay.Value
```

```
Range("C" & MaxRow).Value = cmbComp.Value ' 受注元  
Range("D" & MaxRow).Value = txtProdId.Value ' 商品ID  
Range("E" & MaxRow).Value = txtProdName.Value ' 商品名  
Range("F" & MaxRow).Value = txtPrice.Value ' 単価  
Range("G" & MaxRow).Value = txtNum.Value ' 数量  
Range("H" & MaxRow).Value = txtTotal.Value ' 金額
```

各セルに値を代入



[実習の結果]

入力データがそれぞれの列の最終行に代入された

	A	B	C	D	E	F	G	H
1								
2			新規追加 1					
3					受注一覧表			
4	受注ID	日付	受注元	商品ID	商品名	単価	数量	金額
5	1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000	2	4,000
6	2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000	3	9,000
7	3	2018/7/4	パワポ建設	3	コーヒー豆	10,000	2	20,000
8	4	2018/7/6	エクセル運輸	3	コーヒー豆	10,000	3	30,000
9	5	2018/7/11	VBA商事	7	業務用ストロー	900	7	6,300
10	6	2018/7/10	表計算コーポ	3	紅茶葉	2,000	4	8,000
11	7	2018/7/9	エクセル運輸	3	紅茶葉	2,000	5	10,000
12	123	2018/10/20	会社A	5	コーヒー	10,000	3	30,000
13				4				
14								
15								

入力フォーム

受注ID: 123

日付: 2018 / 10 / 20

受注元: 会社A

商品ID: 5 2

商品名: コーヒー

単価: 10000

数量: 3

金額: 30000

配送済み

請求済み

入金済み

新規追加 3



Checkboxの結果をセルに代入するには？

チェックが入っているかどうかにより、条件分岐させる

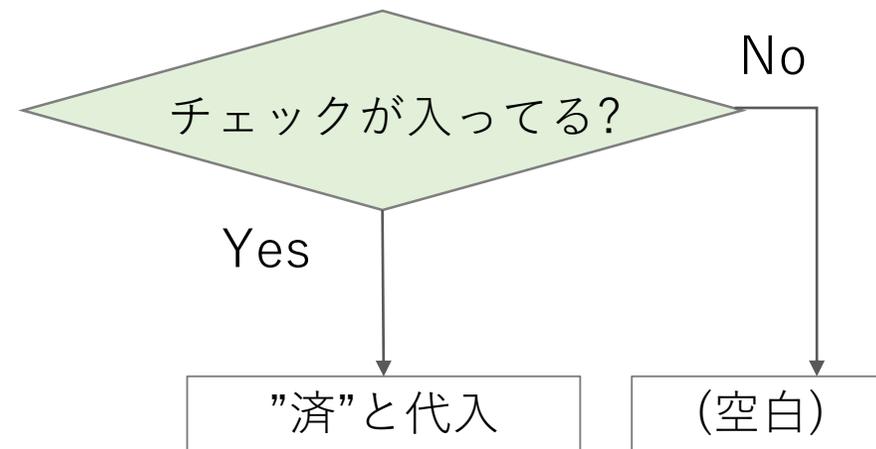
入力フォーム

受注ID: 123
日付: 2018 / 10 / 30
受注元: 会社A
商品ID: 5
商品名: コーヒー
単価: 10000
数量: 3
金額: 30000

配送済み
 請求済み
 入金済み

新規追加

- 配送済み (chkDeliv)
- 請求済み (chkInv)
- 入金済み (chkPay)



受注ID	日付	受注元	商品ID	商品名	単価	数量	金額	配送済み	請求済み	入金済み
1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000	2	4,000	済		
2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000	3	9,000	済		
3	2018/7/4	パワポ建設	3	コーヒー豆	10,000	2	20,000	済		
4	2018/7/6	エクセル運輸	3	コーヒー豆	10,000	3	30,000	済		
5	2018/7/11	VBA商事	7	業務用ストロー	900	7	6,300	済		
6	2018/7/10	表計算コーポ	3	紅茶葉	2,000	4	8,000			
7	2018/7/9	エクセル運輸	3	紅茶葉	2,000	5	10,000	済	済	
123	2018/10/20	会社A	5	コーヒー	10,000	3	30,000	済		済

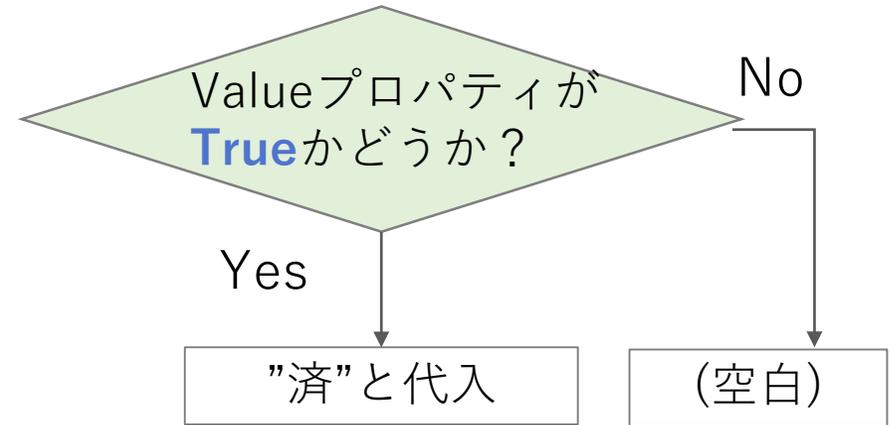


チェックが入っているか状態を取得

CheckboxのValueプロパティにより「True」か「False」を取得できる

- 配送済み
- 請求済み
- 入金済み

Valueプロパティで取得される値	
<input checked="" type="checkbox"/> チェック入	True
<input type="checkbox"/> チェック無し	False



(例)If文で条件分岐して、Trueの場合とそうでない場合で処理を分ける

If chkDeliv.Value = **True** **Then**

(Trueの場合の処理)

Else

(そうでない場合の処理)

End If



[実習]Checkboxに対応する値をセルに代入

それぞれのオブジェクトから値を取得して、各列の最終行(MaxRow列)に代入しよう

```
Private Sub btnAdd_Click()  
    MsgBox "ボタンクリック"  
  
    Dim MaxRow As Integer  
    MaxRow = Cells(Rows.Count, 1).End(xlUp).Row + 1  
  
    ' 受注ID  
    Range("A" & MaxRow).Value = txtOdrID.Value  
  
    ' 日付  
    Range("B" & MaxRow).Value = txtYear.Value & "/" & _  
        txtMonth.Value & "/" & _  
        txtDay.Value  
  
    Range("C" & MaxRow).Value = cmbComp.Value  
    Range("D" & MaxRow).Value = txtProdId.Value  
    Range("E" & MaxRow).Value = txtProdName.Value  
    Range("F" & MaxRow).Value = txtPrice.Value  
    Range("G" & MaxRow).Value = txtNum.Value  
    Range("H" & MaxRow).Value = txtTotal.Value  
    ' 受注元  
    ' 商品ID  
    ' 商品名  
    ' 単価  
    ' 数量  
    ' 金額
```

(続き)

```
' 配送済み  
If chkDeliv.Value = True Then  
    Range("I" & MaxRow).Value = "済"  
Else  
    Range("I" & MaxRow).Value = ""  
End If
```

```
' 請求済み  
If chkInv.Value = True Then  
    Range("J" & MaxRow).Value = "済"  
Else  
    Range("J" & MaxRow).Value = ""  
End If
```

```
' 入金済み  
If chkPay.Value = True Then  
    Range("K" & MaxRow).Value = "済"  
Else  
    Range("K" & MaxRow).Value = ""  
End If
```



[実習の結果]

チェックが入っている項目だけに「済」が代入された

	A	B	C	D		G	H	I	J	K
1										
2			新規追加 1							
3										
4	受注ID	日付	受注元	商品		数量	金額	配送済み	請求済み	入金済み
5	1	2018/6/1	VBA商事			2	4,000	済		
6	2	2018/7/3	VBA商事			3	9,000	済		
7	3	2018/7/4	パワポ建設			2	20,000	済		
8	4	2018/7/6	エクセル運輸			3	30,000	済		
9	5	2018/7/11	VBA商事			7	6,300	済		
10	6	2018/7/10	表計算コーポ			4	8,000			
11	7	2018/7/9	エクセル運輸			5	10,000	済	済	4
12	123	2018/10/20	会社A			3	30,000	済	済	
13										

入力フォーム

受注ID: 123

日付: 2018 / 10 / 20

受注元: 会社A

商品ID: 5

商品名: コーヒー

単価: 10000

数量: 3

金額: 30000

配送済み **2**

請求済み

入金済み

新規追加 **3**



[補足]新規追加した後、IDに戻るには？

新規追加したら、次のデータ入力のため受注IDのテキストボックスに戻る

フォーカスを移動する

The image shows two side-by-side screenshots of an input form titled '入力フォーム'. The form contains fields for '受注ID', '日付', '受注元', '商品ID', '商品名', '単価', '数量', and '金額', along with checkboxes for '配送済み', '請求済み', and '入金済み', and a '新規追加' button. In the left screenshot, the '新規追加' button is highlighted with a red box. A red arrow points from this button to the '受注ID' text box in the right screenshot, which is also highlighted with a red box. The '受注ID' field contains the value '123'.

```
Private Sub btnAdd_Click()
```

```
Dim MaxRow As Integer  
MaxRow = Cells(Rows.Count, 1).End(xlUp).Row + 1
```

```
' 受注ID  
Range("A" & MaxRow).Value = txtOdrID.Value
```

```
' 日付  
Range("B" & MaxRow).Value = txtYear.Value & "/" & _  
txtMonth.Value & "/" & _  
txtDay.Value
```

(中略)

```
' 入金済み  
If chkPay.Value = True Then  
    Range("K" & MaxRow).Value = "済"  
Else  
    Range("K" & MaxRow).Value = ""  
End If
```

```
txtOdrID.SetFocus
```

```
End Sub
```

受注IDにフォーカスを移動



よし！
これで入力の仕組みができたぞう！



(わあ、すごく早く入力できるようになった！
でも、**ここって…ちょっと不便かも…**)



ああ～ちょっと待って、わかったわかった！
どうにかするから！（汗）
…クマさん、どうしたらいいの？



うむ。ようやくフォームらしい動きを実装できたな。



っスね～。なんか、**手作業するのが面倒なところ**もあるっスね。
こことか、こことか、こことか…



…やはり、そういう意見が挙がったか。
では、その悩み、次の章で解決してやろう！

【第3章】

はじめから自動データ入力 & コンボボックス (フォームの初期化)





うん。例えば…「受注ID」の欄。
これって、わざわざ自分で入力しなくても、
はじめから連番で入力されてたらいいのに。



…というわけで、フォームの基本的な作りができた。
たしかに便利になったが、
いろいろ問題も見えてきたな？



なるほどっスね～。
でも、そんな事できるっスか？



そうだな。それに、どうせなら「日付」もだ。
はじめから日付が自動入力されていた方が
便利だろう。



うっひょ～便利！
楽しみっスねえ！



うむ、できるのだ。
こんな風に、あらかじめフォームに自動処理させる
ことを「**フォームの初期化**」という。



「初期化を制する者は、フォームを制す！」
(コホン) …ではいくぞ！



フォームの初期化処理 (Initialize)

UserForm_Initialize()について知っておこう

ボタンを押す

	B	C	D
		新規追加	
D	日付	受注元	商品ID
1	2018/6/1	VBA商事	

```
Sub ボタン1_Click()  
    UserForm1.Show  
End Sub
```

UserForm_Initialize() (初期化処理)

```
Private Sub UserForm_Initialize()  
    ' 初期化処理  
End Sub
```

あらかじめ処理したいこと

- 受注IDを自動挿入
- 日付を自動挿入
- Comboboxに要素を追加など

フォームが起動

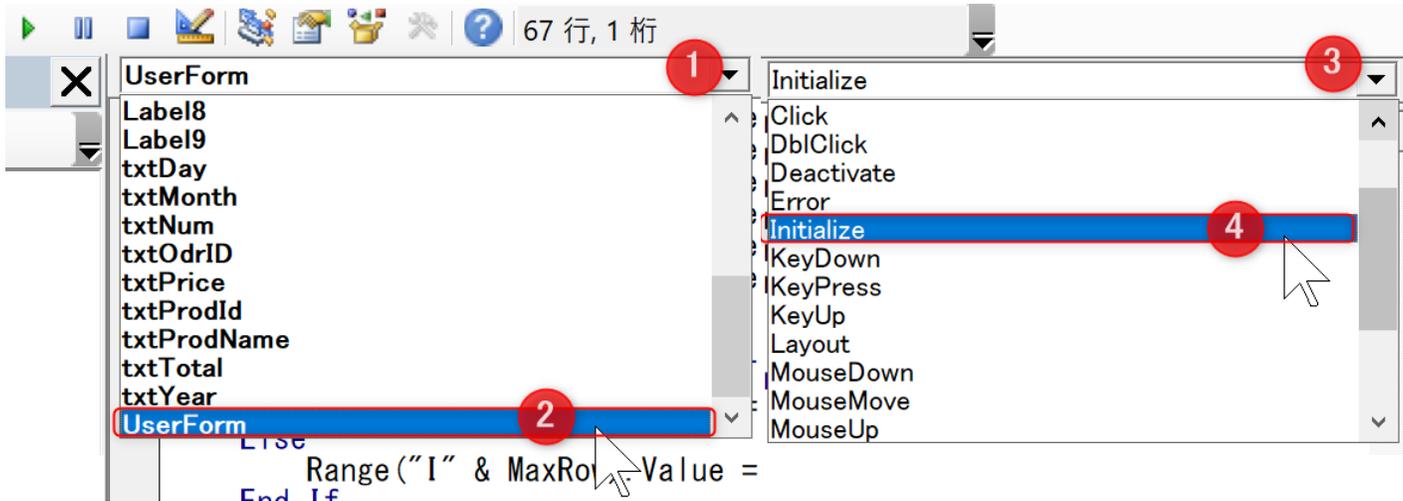
入力フォーム

受注ID	<input type="text"/>
日付	<input type="text"/> / <input type="text"/> / <input type="text"/>
受注元	<input type="text"/>
商品ID	<input type="text"/>
商品名	<input type="text"/>
単価	<input type="text"/>
数量	<input type="text"/>
金額	<input type="text"/>
<input type="checkbox"/> 配送済み	
<input type="checkbox"/> 請求済み	
<input type="checkbox"/> 入金済み	
<input type="button" value="新規追加"/>	



[実習] フォームの初期化処理を挿入

UserForm_Initialize()を挿入してみよう！



```
Private Sub UserForm_Initialize()  
End Sub
```

自動的にコードが挿入される
(※キーボードで手打ちしても可)



フォームが起動したら受注IDを自動挿入

最新データに+1した値をフォームに挿入する

	A	B	C	D	E
1					
2			新規追加		
3					
4	受注ID	日付			
5	1	2018/6/1	VI		
6	2	2018/7/3	VI		
7	3	2018/7/4	パ		
8	4	2018/7/6	工		
9	5	2018/7/			
10	6	2018/7/			
11	7	2018/7/			
12					
13					
14					
15					
16					
17					
18					

入力フォーム

受注ID

日付 / /

受注元

単価

数量

金額

配送済み

請求済み

+1

プログラムの流れ

A列の最終行にある値を取得



その値に+1



受注ID欄に挿入する



[実習] フォーム起動時、受注IDを挿入しよう

A列の最終行のデータを取得し、+1の値をフォームに挿入しよう

```
Private Sub UserForm_Initialize()
```

```
Dim MaxRow As Integer
```

変数「MaxRow」を宣言

```
MaxRow = Cells(Rows.Count, 1).End(xlUp).Row
```

A列の最終行の行番号

```
txtOdrID.Value = Range("A" & MaxRow).Value + 1
```

受注IDのTextbox

A列の最終行の値 + 1

```
End Sub
```



[実習の結果]

受注IDが自動挿入された！

	A	B	C	D	E
1					
2			新規追加		
3					
4	受注ID	日付			
5	1	2018/6/1	V		
6	2	2018/7/3	V		
7	3	2018/7/4	パ		
8	4	2018/7/6	工		
9	5	2018/7/			
10	6	2018/7/			
11	7	2018/7/9	工		
12					
13					
14					
15					
16					
17					
18					

入力フォーム

受注ID

日付 / /

受注元

単価

数量

金額

配送済み

請求済み

+1



[実習+α]エラーを未然に回避しよう

このままではエラーの原因になるので、回避する仕組みを作っておきましょう

データが1件も無い場合、エラーになる！

	A	B	C	D
1				
2				
3				
4	受注ID	日付	受注元	商品
5				
6				
7				
8				
9				

新規追加

最終行が4より大きい行の場合のみ、受注IDを挿入する

```
Private Sub UserForm_Initialize()
```

```
Dim MaxRow As Integer
```

```
MaxRow = Cells(Rows.Count, 1).End(xlUp).Row
```

```
If MaxRow > 4 Then 最終行が4より大きい場合のみ実行
```

```
[TAB]----->txtOdrID.Value = Range("A" & MaxRow).Value + 1
```

```
End If
```

```
End Sub
```

Microsoft Visual Basic

実行時エラー '13':

型が一致しません。

継続(C)

終了(E)

デバッグ(D)

ヘルプ(H)



[発展]新規追加後、受注IDが更新されない？

「新規追加」を押したあと、受注IDにフォーカスが戻るものの、数値が更新されない問題

受注IDが「8」のまま更新されない！？

The image shows two side-by-side screenshots of an input form titled '入力フォーム'. The form contains the following fields: 受注ID (8), 日付 (2018 / 7 / 10), 受注元 (VBA商事), 商品ID (3), 商品名 (紅茶葉), 単価 (2000), 数量 (3), 金額 (6000). At the bottom, there are three checkboxes: 配送済み, 請求済み, and 入金済み. A red box highlights the '新規追加' button in both screenshots. A red arrow points from the button in the left screenshot to the '受注ID' field in the right screenshot, which still contains the value '8'.

```
Private Sub btnAdd_Click()
```

```
Dim MaxRow As Integer
```

```
MaxRow = Cells(Rows.Count, 1).End(xlUp).Row + 1
```

```
' 受注ID
```

```
Range("A" & MaxRow).Value = txt0drID.Value
```

```
' 日付
```

```
Range("B" & MaxRow).Value = txtYear.Value & "/" & _
```

(中略)

```
' 入金済み
```

```
If chkPay.Value = True Then
```

```
Range("K" & MaxRow).Value = "済"
```

```
Else
```

```
Range("K" & MaxRow).Value = ""
```

```
End If
```

```
txt0drID.SetFocus
```

```
Call UserForm_Initialize
```

```
End Sub
```

「Call」で初期化処理を呼び出す



フォーム起動時、現在の日付を自動挿入

年月日の欄に、自動的に値を挿入するためには？

新規追加

受注ID	日付	
1	2018/6/1	VBA
2	2018/7/3	VBA
3	2018/7/4	パワ
4	2018/7/6	エク
5	2018/7/11	VBA
6	2018/7/10	表計
7	2018/7/9	エク

入力フォーム

受注ID

日付

受注元

商品ID

商品名

プログラムの流れ

本日の日付を取得(Date関数)

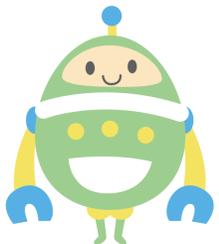


年月日のデータだけを抽出



各Textboxに挿入する

VBA関数



(引数)



戻り値





Date関数などで日付を取得、変換する

現在の日付を取得するDate関数、日付型を変換するYear,Month,Day関数の使い方

現在の日付を取得するVBA関数「Date」(引数なし)

Date関数の利用例

MsgBox Date

データから年、月、日に変換するYear,Month,Day関数

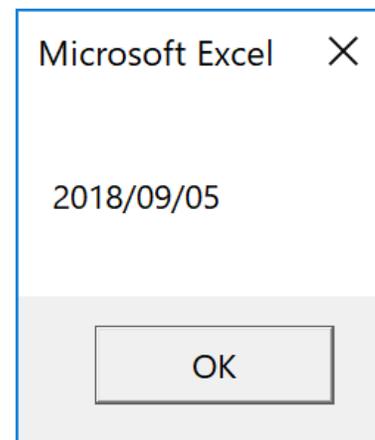
Year関数、Month関数、Day関数を利用すると

MsgBox Year(Date)

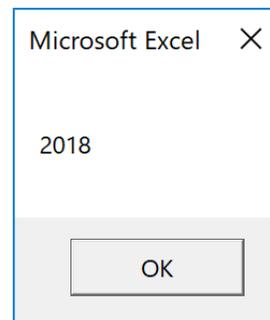
MsgBox Month(Date)

MsgBox Day(Date)

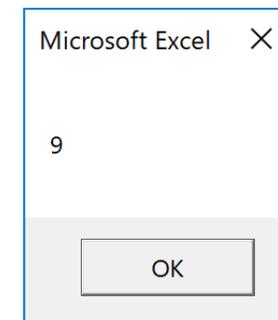
日付



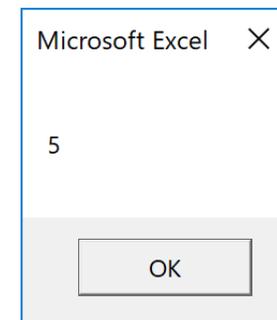
年



月



日





[実習]現在の日付を年、月、日で挿入しよう

現在の日付を取得するDate関数、日付型を変換するYear,Month,Day関数の使い方

```
Private Sub UserForm_Initialize()
```

コメント追加

```
' 最終行を取得
```

```
Dim MaxRow As Integer
```

```
MaxRow = Cells(Rows.Count, 1).End(xlUp).Row
```

```
' 受注ID
```

```
If MaxRow > 4 Then
```

```
    txtOdrID.Value = Range("A" & MaxRow).Value + 1
```

```
End If
```

```
' 年、月、日
```

```
txtYear.Value = Year(Date)
```

年のデータを挿入

```
txtMonth.Value = Month(Date)
```

月のデータを挿入

```
txtDay.Value = Day(Date)
```

日のデータを挿入

```
End Sub
```



[実習の結果]

現在の日付がそれぞれ年、月、日の欄に挿入された！

新規追加

受注ID	日付	
1	2018/6/1	VBA
2	2018/7/3	VBA
3	2018/7/4	パワ
4	2018/7/6	エク
5	2018/7/11	VBA
6	2018/7/10	表計
7	2018/7/9	エク

入力フォーム

受注ID: 8

日付: 2018 / 9 / 30

受注元: [dropdown]

商品ID: [input]

商品名: [input]



Comboboxの項目を追加するには？

.Valueプロパティで値を取得し、セル範囲に代入する

新規追加

入力フォーム

受注ID: 8

日付: 2018 / 9 / 5

受注元:

商品ID:

商品名:

単価:

受注元 dropdown menu items:
VBA商事
エクセル運輸
パワポ建設
関数カンパニー
表計算コーポ

	A	B
1	受注元リスト	
2	会社名	
3	VBA商事	
4	エクセル運輸	
5	パワポ建設	
6	関数カンパニー	
7	表計算コーポ	
8		
9		
10		
11		
12		
13		
14		
15		

受注一覧 リスト

ワークシート「リスト」の
A3:A7のセル範囲



[実習]

ワークシート「リスト」から会社名を取得して、comboBoxに追加しましょう

```
Private Sub UserForm_Initialize()
```

```
' 最終行を取得
```

```
Dim MaxRow As Integer
```

```
MaxRow = Cells(Rows.Count, 1).End(xlUp).Row
```

```
' 受注ID
```

```
If MaxRow > 4 Then
```

```
    txtOdrID.Value = Range("A" & MaxRow).Value + 1
```

```
End If
```

```
' 年、月、日
```

```
txtYear.Value = Year(Date)
```

```
txtMonth.Value = Month(Date)
```

```
txtDay.Value = Day(Date)
```

```
' comboBoxに会社名リストを追加
```

```
Dim i As Integer
```

```
For i = 3 To 7
```

i=3から始め、7まで繰り返す

```
    cmbComp.AddItem Worksheets("リスト").Range("A" & i).Value
```

```
Next i
```

ワークシート「リスト」のA列 i 行の値をcmbCompに加える

```
End Sub
```

iの値	セル
3	A3
4	A4
5	A5
6	A6
7	A7

※ワークシート「リスト」を指定するため、「Worksheets("リスト").Range…」のように指定する



[実習の結果]

フォームを起動したら、「受注元」から会社名を選べるようになった

入力フォーム

受注ID	<input type="text" value="11"/>
日付	<input type="text" value="2018"/> / <input type="text" value="9"/> / <input type="text" value="24"/>
受注元	<input type="text" value=""/> <div style="border: 1px solid red; padding: 2px;"><ul style="list-style-type: none">VBA商事エクセル運輸パワポ建設関数カンパニー表計算コーポ</div>
商品ID	
商品名	
単価	<input type="text"/>



え…？何か忘れたことあったっけ？



げげっ！！…増えた分は
ComboBoxに追加されないっス！



待ちたまえ！
これでComboBoxにアイテムを追加できた。
だが、**何か忘れていないか？**



油断するな！
今の状態なら、会社名はA3:A7までに収まっている。
だが、**もし会社名がもう1つ増えたらどうなる？**



そう。ではどう改良しなければいけないか？
…答えは、**VBAのマストテクニック、**
「最終行の取得」だ！



[発展]会社名データが増えても対応する

もし、会社名リストのデータが追加されたとしても対応できるか？

新規追加

入力フォーム

受注ID: 8

日付: 2018 / 9 / 5

受注元: [dropdown menu]

商品ID: [dropdown menu]

商品名: [dropdown menu]

単価: [input field]

Dropdown menu options: VBA商事, エクセル運輸, パワポ建設, 関数カンパニー, 表計算コーポ

	A	B
1	受注元リスト	
2	会社名	
3	VBA商事	
4	エクセル運輸	
5	パワポ建設	
6	関数カンパニー	
7	表計算コーポ	
8	ワード商店	
9		
10		

受注一覧 リスト

現在は3~7行目しか追加されない仕様

データが追加されたのに…。

(つづき)

**3～最終行までの
データをcomboBoxに追加
するように改善しよう**

最終行

	A	B
1	受注元リスト	
2	会社名	
3	VBA商事	
4	エクセル運輸	
5	パワポ建設	
6	関数カンパニー	
7	表計算コーポ	
8	ワード商店	
9		
10		

受注一覧 リスト



[実習]会社リストの最終行に対応しよう！

最終行から会社名を取得して、comboBoxに追加しましょう

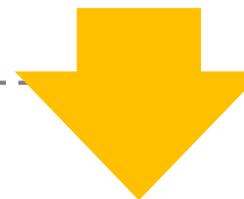
【前】

```
' comboBoxに会社名リストを追加  
Dim i As Integer  
  
For i = 3 To 7  
    cmbComp.AddItem Worksheets("リスト").Range("A" & i).Value  
Next i
```

【後】

```
' 会社名リストの最終行を取得  
Dim cMaxRow As Integer      変数「cMaxRow」を宣言  
cMaxRow = Worksheets("リスト").Cells(Rows.Count, 1).End(xlUp).Row  
  
' comboBoxに会社名リストを追加  
Dim i As Integer  
For i = 3 To cMaxRow      3～最終行(cMaxRow)まで繰り返すように変更  
    cmbComp.AddItem Worksheets("リスト").Range("A" & i).Value  
Next i
```

cMaxRowに最終行の行番号を代入

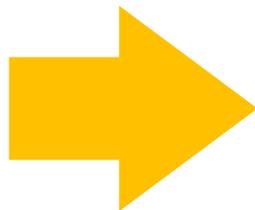




[実習の結果]

会社名が増えても、それに対応してcomboBoxのアイテムが追加されるようになった

	A
1	受注元リスト
2	会社名
3	VBA商事
4	エクセル運輸
5	パワポ建設
6	関数カンパニー
7	表計算コーポ
8	ワード商店
9	アクセス工業
10	



受注ID	<input type="text" value="11"/>
日付	<input type="text" value="2018"/> / <input type="text" value="9"/> / <input type="text" value="24"/>
受注元	<input type="text" value=""/>
商品ID	<input type="text" value=""/>
商品名	<input type="text" value=""/>
単価	<input type="text" value=""/>
数量	<input type="text" value=""/>

受注元 dropdown menu items:

- VBA商事
- エクセル運輸
- パワポ建設
- 関数カンパニー
- 表計算コーポ
- ワード商店
- アクセス工業



いや～、便利っスねえ、
コンブボックス！



(えっと、あのう…)



うひょ～、それ便利っスねえ！



やったー！次のレッスンに期待だぞう！



…それを言うなら**コン「ボ」**だろう？
うむ。ユーザーフォームの便利機能を使えば、
どんどん**入力が楽**になっていくぞ。



ん？ノアちゃん、何だって？
『**「単価」と「数量」を打ちこんだら
金額も自動計算されたらいいのに**』って？



安心したまえ。
そんなことも教えようと思っていた。
…さらに**驚くような機能**も見せてやろう！

【第4章】

フォームが自動で計算し、VLOOKUPも！？
自動計算と、ワークシート関数の活用





お、なんだか難しくなってきたな…
(ドキドキ)



マジっすか！？うほ～楽勝～！
(自分、Excelの数式とか関数も
苦手なんっすけどね…)



うほお、ありがたいっす～！



…というわけで、
今回は、フォームが自動的に計算をしたり、
関数を実行してくれるようにする。



まあ、そう身構えなくともよい。
仕組み的には、普段のExcelとほとんど同じ
だからな。



おい、心の声も聞こえてるぞ。
わかったわかった。
数式や関数が苦手なキミたちにも
分かりやすいよう、丁寧に説明していこう。



金額を自動計算させたい！

単価と数量だけを入力したら、「金額」が自動的に計算されるようにしたい

商品ID	<input type="text"/>
商品名	<input type="text" value="商品A"/>
単価	<input type="text" value="2000"/>
数量	<input type="text" value="3"/>
金額	<input type="text" value="6000"/>
	<input type="checkbox"/> 配送済み
	<input type="checkbox"/> 請求済み
	<input type="checkbox"/> 入金済み
	<input type="button" value="新規追加"/>



① 「単価」と「数量」だけを入力



② 「金額」が自動計算されるように！



Change イベント処理を利用する

Change(変更)を受けたら計算を実行するようにする方法

商品ID	<input type="text"/>
商品名	<input type="text" value="商品A"/>
単価	<input type="text" value="2000"/>
数量	<input type="text" value="3"/>
金額	<input type="text" value="6000"/>
	<input type="checkbox"/> 配送済み
	<input type="checkbox"/> 請求済み
	<input type="checkbox"/> 入金済み
	<input type="button" value="新規追加"/>

① (Change イベント)

「単価」と「数量」が**変更された**



② 計算をして、「金額」を挿入

txtPriceの値 × txtNumの値

※ユーザーが何らかの操作を実行することを「イベント」と呼ぶ。

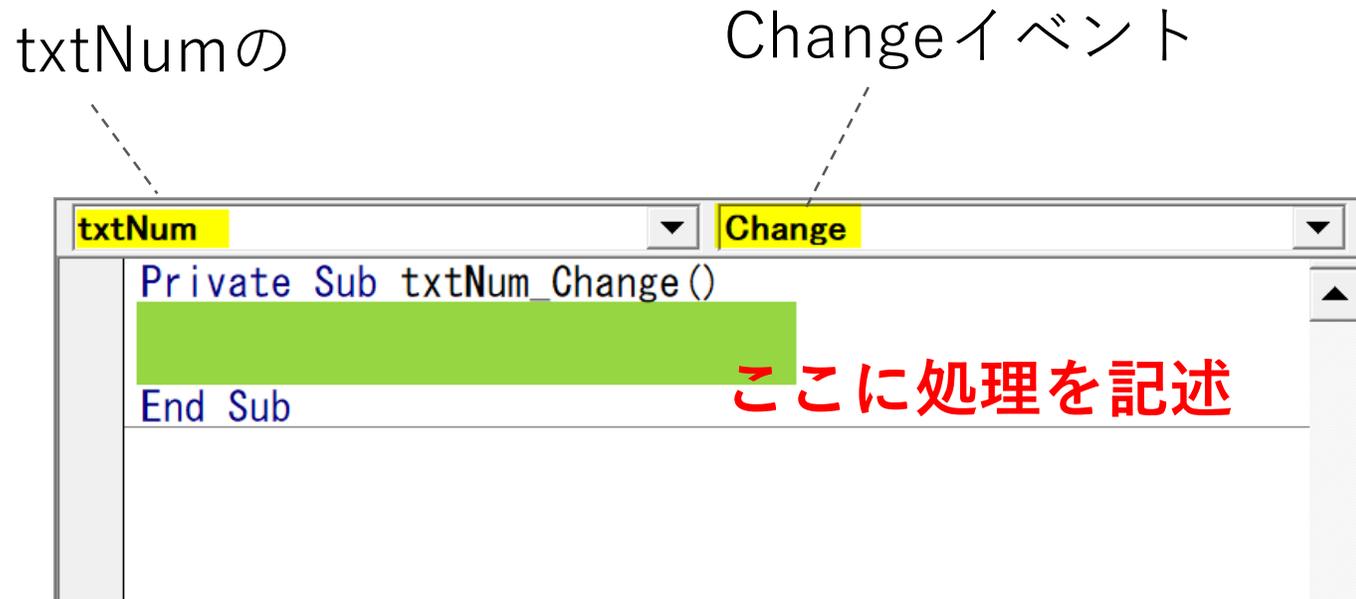
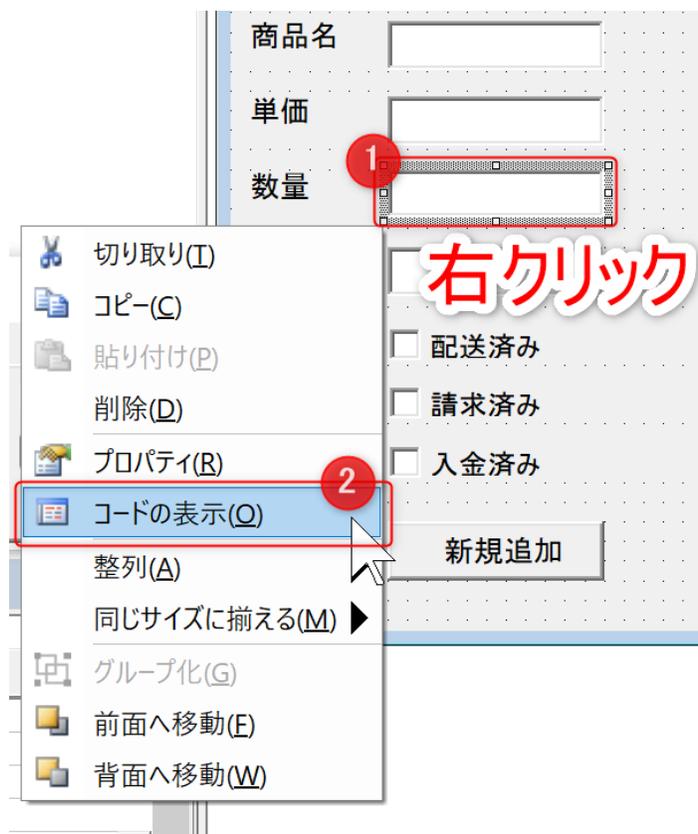


Change イベント処理を挿入するには？

txtNumが変更された（Changeイベント）ときの処理を挿入する

①右クリックして「コードの表示」

②Change イベント処理が挿入される





[実習]数量が変更されたら計算をする

txtNumのChangeイベントで、単価×数量を計算しよう

```
txtNum Change
Private Sub txtNum_Change ()
    MsgBox txtPrice.Value * txtNum.Value
End Sub
```

[単価]の値

[数量]の値

[×]乗算

算術演算子

種類	演算子
加算(足し算)	+
減算(引き算)	-
乗算(掛け算)	*
除算(割り算)	/



[実習の結果]

「数量」を入力したらメッセージボックスで計算結果が表示された！

受注ID	<input type="text" value="8"/>
日付	<input type="text" value="2018"/> / <input type="text" value="10"/> / <input type="text" value="1"/>
受注元	<input type="text" value=""/> ▼
商品ID	<input type="text"/>
商品名	<input type="text"/>
単価	<input type="text" value="1000"/>
数量	<input type="text" value="2"/>
金額	<input type="text"/>

配送済み
 請求済み
 入金済み

Microsoft Excel ×

2000



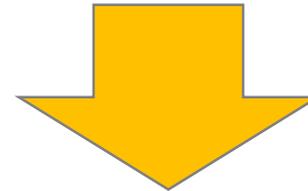
[実習②] 計算結果を[金額](txtTotal)へ代入

単価×数量を計算したら、txtTotalへ代入するように変更しよう

商品ID	<input type="text"/>
商品名	<input type="text" value="商品A"/>
単価	<input type="text" value="2000"/>
数量	<input type="text" value="3"/>
金額	<input type="text" value="6000"/>
	<input type="checkbox"/> 配送済み
	<input type="checkbox"/> 請求済み
	<input type="checkbox"/> 入金済み
	<input type="button" value="新規追加"/>

削除

```
txtNum Change  
Private Sub txtNum_Change()  
    MsgBox txtPrice.Value * txtNum.Value  
End Sub
```



記述

```
Private Sub txtNum_Change()  
    txtTotal.Value = txtPrice.Value * txtNum.Value  
End Sub
```



空白の時、エラーが起きてしまう！

単価 × 数量を計算させたい…しかし、どちらかが未入力で空白になっているとエラーに？

エラーになってしまう

受注元

商品ID

商品名

単価

数量

金額

配送済み
 請求済み
 入金済み

新規追加

Microsoft Visual Basic

実行時エラー '13':
型が一致しません。

継続(C) 終了(E) デバッグ(D) ヘルプ(H)

txtPriceの値 × txtNumの値

“(空白)

3



空白 × 3は計算できず、エラー



空白のエラーを防ぐには？

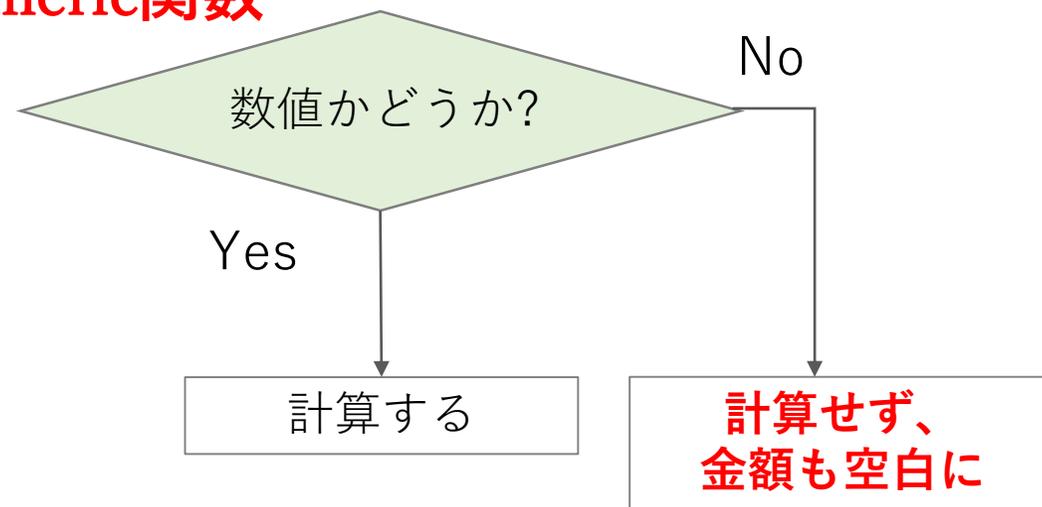
IsNumeric関数を使って、数値かどうかを確認してみよう

単価	<input type="text" value="1000"/>
数量	<input type="text" value="abc"/>
金額	<input type="text"/>

Blue arrows indicate data flow from the 'Quantity' field to the 'Amount' field.



IsNumeric関数



”(空白)や、文字列だった場合
計算をせず、金額も空白にする

VBA関数



(引数)

戻り値

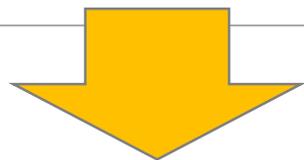




[実習③] IsNumeric関数でエラー回避

単価や数量が数値ではない（空白や文字列）場合、計算をせず空白を代入させよう

```
Private Sub txtNum_Change()  
    txtTotal.Value = txtPrice.Value * txtNum.Value  
End Sub
```



```
Private Sub txtNum_Change()  
    If IsNumeric(txtNum.Value) = True _  
        And IsNumeric(txtPrice.Value) = True Then  
        txtTotal.Value = txtPrice.Value * txtNum.Value  
    Else  
        txtTotal.Value = ""  
    End If  
End Sub
```

txtNumの値は数値である

txtPriceの値も数値であるなら

(かつ)

(そうでなければ)

金額に""(空白)を代入



他にもあるコントロールのイベント

※種類が多数あるので、代表的なものをご紹介します

種類	演算子
Activate	オブジェクトがアクティブ(有効)になった
Change	値や項目が変更された
Click	クリックされた
KeyDown	キーを押された
KeyPress	文字キーを押された
KeyUp	キーを離した
SpinDown	下向きや左向き矢印のスピンボタンを押された
SpinUp	上向きや右向き矢印のスピンボタンを押された
その他多数	



[実習④] [単価]もChangeイベントを設定

「txtPrice」にもChangeイベント処理を追加し、同様の計算を記述しよう

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み
 請求済み
 入金済み

Microsoft Excel ×
2000

txtPriceのChangeイベント処理を追加
(処理内容は、txtNumと全く同じ)

```
txtPrice 
Private Sub txtPrice_Change()
    If IsNumeric(txtNum.Value) = True _
    And IsNumeric(txtPrice.Value) = True Then
        txtTotal.Value = txtPrice.Value * txtNum.Value
    Else
        txtTotal.Value = ""
    End If
End Sub
```



[応用] 同じ処理は、関数化しておこう

入力したら、「金額」が自動的に計算されるようにしたい

```
Private Sub txtNum_Change()
```

```
If IsNumeric(txtNum.Value) = True _  
And IsNumeric(txtPrice.Value) = True Then  
    txtTotal.Value = txtPrice.Value * txtNum.Value  
Else  
    txtTotal.Value = ""  
End If
```

```
End Sub
```

```
Private Sub txtPrice_Change()
```

```
If IsNumeric(txtNum.Value) = True _  
And IsNumeric(txtPrice.Value) = True Then  
    txtTotal.Value = txtPrice.Value * txtNum.Value  
Else  
    txtTotal.Value = ""  
End If
```

```
End Sub
```

同じ処理が記述されている

【デメリット】

修正時：

2か所修正しなければならない

閲覧時：

同じ処理かどうか判断しにくい

など

つづき

```
Private Sub txtNum_Change()
```

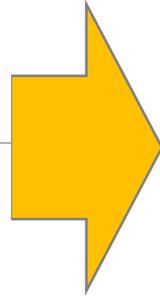
```
If IsNumeric(txtNum.Value) = True _  
And IsNumeric(txtPrice.Value) = True Then  
    txtTotal.Value = txtPrice.Value * txtNum.Value  
Else  
    txtTotal.Value = ""  
End If
```

```
End Sub
```

```
Private Sub txtPrice_Change()
```

```
If IsNumeric(txtNum.Value) = True _  
And IsNumeric(txtPrice.Value) = True Then  
    txtTotal.Value = txtPrice.Value * txtNum.Value  
Else  
    txtTotal.Value = ""  
End If
```

```
End Sub
```



```
Private Sub txtNum_Change()
```

CalcTotalを呼び出す

```
Call CalcTotal
```

```
End Sub
```

```
Private Sub txtPrice_Change()
```

CalcTotalを呼び出す

```
Call CalcTotal
```

```
End Sub
```

新しい関数に処理を記述

```
Private Sub CalcTotal() [CalcTotal]
```

```
If IsNumeric(txtNum.Value) = True _  
And IsNumeric(txtPrice.Value) = True Then  
    txtTotal.Value = txtPrice.Value * txtNum.Value  
Else  
    txtTotal.Value = ""  
End If
```

```
End Sub
```



VLOOKUP関数、導入への道！

IDを入力したら商品名や単価も自動入力されたい

商品ID	<input type="text" value="3"/>
商品名	<input type="text" value="紅茶葉"/>
単価	<input type="text" value="2000"/>
数量	<input type="text"/>

商品IDを入力



VLOOKUP関数

商品名、単価も自動入力される

【VLOOKUP関数】

検索値をもとに、表から必要なデータを参照する関数。
(VBAではなく)Excelのワークシートで利用されるのが普通。
このような関数は特に「ワークシート関数」と呼ばれる。



その前に、VLOOKUP関数を確認！

まずはExcelワークシート上でVLOOKUP関数はどう使えばいいか確認しましょう

(例)商品IDに「4」と入力された場合、商品名が挿入されるようにしたい。
VLOOKUP関数をやって利用すればいいのでしょうか？

	A	B	C	D	E	F	G		A	B	C	D	E
3													
4	受注ID	日付	受注元	商品ID	商品名	単価			受注元リスト		商品リスト		
5	1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000			2	会社名	商品ID	商品名	単価
6	2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000			3	VBA商事	0	オフィス用コーヒー	2000
7	3	2018/7/4	パワポ建設	3	コーヒー豆	10,000			4	エクセル運輸	1	オフィス用カフェラテ	3000
8	4	2018/7/6	エクセル運輸	3	コーヒー豆	10,000			5	パワポ建設	2	コーヒー豆	10000
9	5	2018/7/11	VBA商事	7	業務用ストロー	900			6	関数カンパニー	3	紅茶葉	2000
10	6	2018/7/10	表計算コーポ	3	紅茶葉	2,000			7	表計算コーポ	4	緑茶	1500
11	7	2018/7/9	エクセル運輸	3	紅茶葉	2,000			8	ワード商店	5	特製クッキー	2500
12				4					9	アクセス工業	6	紙製カップ	800
13									10		7	業務用ストロー	900
14									11		8	業務用マドラー	1300
15									12				
									13				
									14				

シート「受注一覧」

シート「リスト」

※E12のセルに入力する場合

※本来は絶対参照にするため「\$C\$3:\$E\$13」とする
(今は気にせず置いておきましょう)

=VLOOKUP(**D12** , **リスト!C3:E13** , **2** , FALSE)

検索値

検索範囲

列番号

検索の型(FALSE:完全一致)

	A	B	C	D	E	F	G		A	B	C	D	E
3					受注一覧表				1	受注元リスト		商品リスト	
4	受注ID	日付	受注元	商品ID	商品名	単価			2	会社名	商品ID	商品名	単価
5	1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000			3	VBA商事	0	オフィス用コーヒー	2000
6	2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000			4	エクセル運輸	1	オフィス用カフェラテ	3000
7	3	2018/7/4	パワポ建設	3	コーヒー豆	10,000			5	パワポ建設	2	コーヒー豆	10000
8	4	2018/7/6	エクセル運輸	3	コーヒー豆	10,000			6	関数カンパニー	3	紅茶葉	2000
9	5	2018/7/11	VBA商事	7	業務用ストロー	900			7	表計算コーポ	4	緑茶	1500
10	6	2018/7/10	表計算コーポ	3	紅茶葉	2,000			8	ワード商店	5	特製クッキー	2500
11	7	2018/7/9	エクセル運輸	3	紅茶葉	2,000			9	アクセス工業	6	紙製カップ	800
12				4					10		7	業務用ストロー	900
13									11		8	業務用マドラー	1300
14									12				
15									13				

シート「受注一覧」

シート「リスト」



VBAからVLOOKUP関数を使う場合は？

ワークシート関数を呼び出す / 範囲指定の方法にも注意！

(False:完全一致)

```
WorksheetFunction.VLookup(txtProdId.Value, Worksheets("リスト").Range("C3:E12"), 2, False)
```

ワークシート関数

検索値

検索範囲

列番号

商品ID

商品名

単価

数量

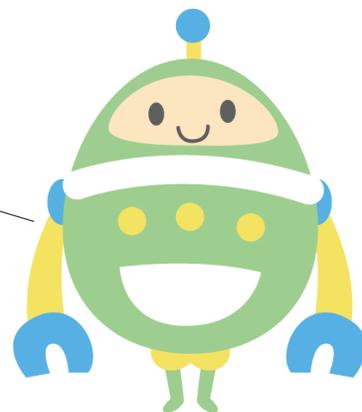


	A	B	C	D	E
1	受注元リスト		商品リスト		
2	会社名		商品ID	商品名	単価
3	VBA商事		0	オフィス用コーヒー	2000
4	エクセル運輸		1	オフィス用カフェラテ	3000
5	パワポ建設		2	コーヒー豆	10000
6	関数カンパニー		3	紅茶葉	2000
7	表計算コーポ		4	緑茶	1500
8	ワード商店		5	特製クッキー	2500
9	アクセス工業		6	紙製カップ	800
10			7	業務用ストロー	900
11			8	業務用マドラー	1300
12					
13					
14					



(つづき) このままでは**型違い**でエラーになってしまうため…

商品ID	<input type="text" value="3"/>
商品名	<input type="text" value="紅茶葉"/>
単価	<input type="text" value="2000"/>
数量	<input type="text"/>



VLOOKUP関数

商品IDのテキストボックスからは
文字列データとして参照される。
数値型データじゃないと困るよ!

CInt(引数)で**文字列を数値に変換**してあげよう

ワークシート関数

検索値

検索範囲

列番号

```
WorksheetFunction.VLookup(txtProdId.Value, Worksheets("リスト").Range("C3:E12"), 2, False)
```

CInt(txtProdId.Value)

(FALSE:完全一致)



[実習]VLOOKUP関数を利用しよう

ワークシート関数のVLOOKUP関数を利用する

①商品IDのテキストボックスで「コードの表示」



②Changeイベント処理に記述

```
txtProdId Change  
  
Private Sub txtProdId_Change()  
    txtProdName.Value = _  
        WorksheetFunction.VLookup(CInt(txtProdId.Value), Worksheets("リスト").Range("C3:E12"), 2, False)  
End Sub
```



[実習]VLOOKUP関数を利用しよう

ワークシート関数のVLOOKUP関数を利用する

①商品IDのテキストボックスで「コードの表示」



②Changeイベント処理に記述

```
txtProdId Change  
  
Private Sub txtProdId_Change()  
    商品名  
    txtProdName.Value = 改行 (False:完全一致)  
    WorksheetFunction.VLookup(検索値, Worksheets("リスト").Range("C3:E12"), 2, False)  
    検索範囲 列番号  
End Sub  
  
ワークシート関数のVLOOKUP
```



[実習の結果]

商品IDを入力すると、商品名が自動的に挿入された！

入力フォーム

受注ID	<input type="text" value="8"/>
日付	<input type="text" value="2018"/> / <input type="text" value="10"/> / <input type="text" value="3"/>
受注元	<input type="text"/>
商品ID	<input type="text" value="1"/>
商品名	<input type="text" value="オフィス用カフェラテ"/>
単価	<input type="text"/>
数量	<input type="text"/>
金額	<input type="text"/>

配送済み
 請求済み
 入金済み



VLOOKUP関数には**エラー**が付きモノ！

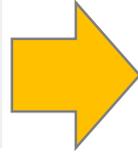
関数のエラーに対策する方法を学びましょう

文字列を入力

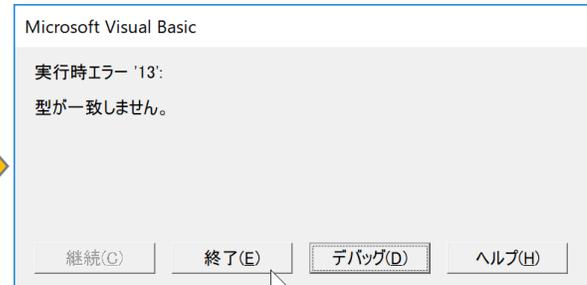
商品ID

商品名

単価



エラー

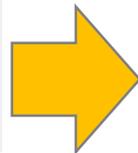


リストに無いIDを入力

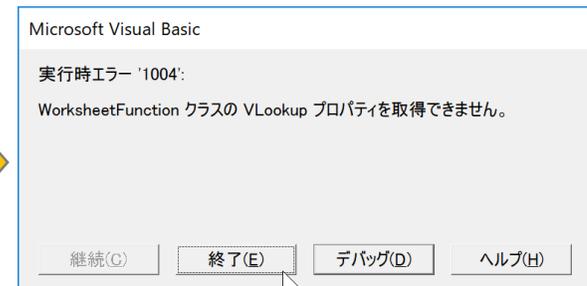
商品ID

商品名

単価



エラー



ユーザーが戸惑う



変な画面が出てきた
どうしよう…

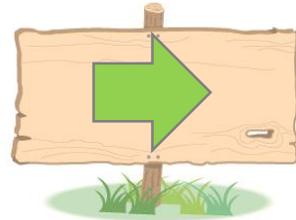


エラーに対策するためには？

On Error GoToステートメントを利用する

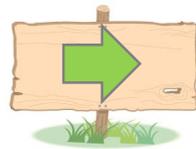
On Error GoToステートメント

この後、
エラーになったら
ラベルに行ってね



VLOOKUP関数

(おわり)



ラベル：

空白を代入

エラー
になった

On Error GoTo <ラベル名>

処理内容

Exit Sub

<ラベル名>：

処理内容

エラー
になった



[実習] エラー時は""(空白)を代入しよう

On Error GoToステートメントを利用して、VLOOKUP関数のエラーに対策しよう

```
Private Sub txtProdId_Change()
```

```
On Error GoTo vlookError
```

```
txtProdName.Value = _  
WorksheetFunction.VLookup(CInt(txtProdId.Value), Worksheets("リスト").Range("C3:E12"), 2, False)
```

```
Exit Sub
```

終了

```
vlookError:
```



ラベル「vlookError」

```
txtProdName.Value = ""
```

エラーになってしまったので商品名を""(空白)にしておく

```
End Sub
```

※ラベル名は自由

エラーになったら、ラベル「vlookError」へ行ってね



[実習の結果]

VLOOKUP関数がエラーになってもマクロが中断されず、商品名に空白が代入された！

商品ID

a

商品名

商品ID

20

商品名



商品リストがもっと増えてしまったら…？

今のままでは、商品リストの行がもっと増えた場合、VLOOKUP関数が対応できません。

もしも商品リストがC3:E12の範囲を超えて増えたら？

	A	B	C	D	E
1	受注元リスト		商品リスト		
2	会社名		商品ID	商品名	単価
3	VBA商事		0	オフィス用コーヒー	2000
4	エクセル運輸		1	オフィス用カフェラテ	3000
5	パワポ建設		2	コーヒー豆	10000
6	関数カンパニー		3	紅茶葉	2000
7	表計算コーポ		4	緑茶	1500
8	ワード商店		5	特製クッキー	2500
9	アクセス工業		6	紙製カップ	800
10			7	業務用ストロー	900
11			8	業務用マドラー	1300
12			9	カレーセット	2000
13			10	特上カレーセット	2500
14			11	究極カレーセット	3000
15			12	ラストカレーセット	4000
16					
17					

いま、**C3:E12**が検索範囲になっているから
それより下のデータは検索できないよ！



VLOOKUP関数



最終行を取得し、VLOOKUPに組み込もう

最終行までをVLOOKUPの検索範囲に指定してあげれば、データが増えてもうまくいく！

	A	B	C	D	E
1	受注元リスト		商品リスト		
2	会社名		商品ID	商品名	単価
3	VBA商事		0	オフィス用コーヒー	2000
4	エクセル運輸		1	オフィス用カフェラテ	3000
5	パワポ建設		2	コーヒー豆	10000
6	関数カンパニー		3	紅茶葉	2000
7	表計算コーポ		4	緑茶	1500
8	ワード商店		5	特製クッキー	2500
9	アクセス工業		6	紙製カップ	800
10			7	業務用ストロー	900
11			8	業務用マドラー	1300
12			9	カレーセット	2000
13			10	特上カレーセット	2500
14			11	究極カレーセット	3000
15			12	ラストカレーセット	4000
16					
17					

最終行を取得する



最終行までを
VLOOKUPの検索範囲にする

ここが最終行

これなら安心だ！





[実習]VLOOKUPの検索範囲を最終行に

変数MaxRowに最終行を取得し、VLOOKUP関数の検索範囲に利用しよう

```
Private Sub txtProdId_Change()
```

```
On Error GoTo vlookError
```

変数MaxRowを宣言し、シート「リスト」の商品リストの最終行を取得する

```
' 最終行を取得
```

```
Dim MaxRow As Integer
```

3列目(C列)

```
MaxRow = Worksheets("リスト").Cells(Rows.Count, 3).End(xlUp).Row
```

```
' VLOOKUP関数
```

```
txtProdName.Value = _
```

```
WorksheetFunction.VLookup(CInt(txtProdId.Value), Worksheets("リスト").Range("C3:E" & MaxRow), 2, False)
```

```
Exit Sub
```

```
vlookError:
```

```
txtProdName.Value = ""
```

```
End Sub
```

開始：C列の3行目

終了：E列のMaxRow行目

"E"

&

MaxRow



[実習の結果]

商品リストの行が増えても、VLOOKUPで検索できるようになった！

商品ID

商品名

商品ID	商品名	単価
0	オフィス用コーヒー	2000
1	オフィス用カフェラテ	3000
2	コーヒー豆	10000
3	紅茶葉	2000
4	緑茶	1500
5	特製クッキー	2500
6	紙製カップ	800
7	業務用ストロー	900
8	業務用マドラー	1300
9	カレーセット	2000
10	特上カレーセット	2500
11	究極カレーセット	3000
12	ラストカレーセット	4000

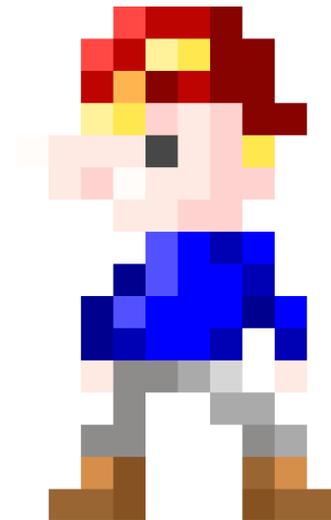


[実習] 「単価」もVLOOKUP関数で検索

どうすればいいでしょうか？挑戦してみよう！

商品ID	<input type="text" value="2"/>
商品名	<input type="text" value="コーヒー豆"/>
単価	<input type="text" value="10000"/>

商品IDを入力したら、
「単価」も自動入力される
ようにしたいっすねえ…





[解答] VLOOKUP処理とエラー処理を複製

「txtProdName」のための処理をコピーして、「txtPrice」用にアレンジしましょう

```
Private Sub txtProdId_Change()
```

```
On Error GoTo vlookError
```

```
' 最終行を取得
```

```
Dim MaxRow As Integer
```

```
MaxRow = Worksheets("リスト").Cells(Rows.Count, 3).End(xlUp).Row
```

```
' VLOOKUP関数
```

```
txtProdName.Value = _  
WorksheetFunction.VLookup(CInt(txtProdId.Value), Worksheets("リスト").Range("C3:E" & MaxRow), 2, False)
```

コピー/貼り付け

```
txtPrice.Value = _  
WorksheetFunction.VLookup(CInt(txtProdId.Value), Worksheets("リスト").Range("C3:E" & MaxRow), 3, False)
```

変更

```
Exit Sub
```

```
vlookError:
```

```
txtProdName.Value = ""
```

```
txtPrice.Value = ""
```

コピー/貼り付け

```
End Sub
```

変更

「3」列目に変更

商品リスト		
商品ID	商品名	単価
0	オフィス用コーヒー	2000
1	オフィス用カフェラテ	3000
2	コーヒー豆	10000
3	紅茶葉	2000



[実習の結果]

商品IDを入力すると、「単価」も自動入力されるようになった！

商品ID	<input type="text" value="5"/>
商品名	<input type="text" value="特製クッキー"/>
単価	<input type="text" value="2500"/>



へえ～！
普段、Excelで使ってる関数を
VBAでも利用できるなんて初めて知ったよ！



特に今回は、
VLOOKUPが便利だったっスね～！



せっかく使えるモノは、
積極的に利用した方が楽っスね！



そういうわけだ。
普段Excelで使える関数を「**ワークシート関数**」、
VBAだけで使える関数を「VBA関数」と区別する。
覚えておくといい。



うむ。
もしVLOOKUP関数が無かったら、
どうやって商品IDから商品名や単価を検索するか
自分で1からマクロを考える手間がかかるからな。

【第5章】

マウス操作でもサクサク入力できる！？
「スピンボタン」を活用しよう





確かに！これならノアちゃんの
事務的な負担も相当減るよねえ！



(えっと、あのう…)



さすがクマさん、女子にやさしいッ！



どうだ。ここまでで
フォームによる自動操作も充実してきた。
かなりデータ入力が楽になったんじゃないか？



ん？何っすかノア先輩？
…「正直、キーボード入力が苦手」！？



タイピングを練習しろよ！
…と言いたいところだが、私も鬼じゃない。
マウスだけでもデータ入力が楽々になる機能、
「スピンボタン」の利用方法を教えてやろう！



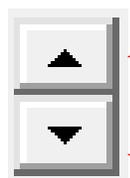
スピンドボタンについて知っておこう！

マウスでも、キーボードの上下キーでも操作できる便利機能

商品ID	<input type="text" value="2"/>	
商品名	<input type="text" value="コーヒー豆"/>	
単価	<input type="text" value="10000"/>	

クリックすることで商品IDを増減

キーボードの上下キーで商品IDを増減



SpinUpイベント

▲(上キー)ボタンが押された

SpinUpイベント処理

商品IDの数値を+1

SpinDownイベント

▼(下キー)ボタンが押された

SpinDownイベント処理

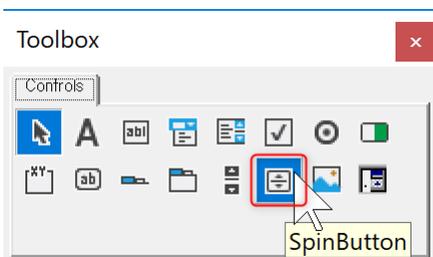
商品IDの数値を-1



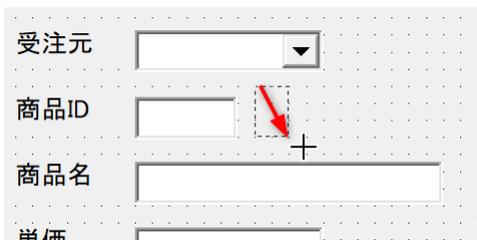
[実習] スピンボタンを作ろう

商品ID、数量の右側にスピンボタンを作成する

① SpinButtonをクリック



② ドラッグして作成



入力フォーム

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

新規追加

③ オブジェクト名を「spnID」に設定

④ オブジェクト名を「spnNum」に設定

商品ID

商品名

単価

数量

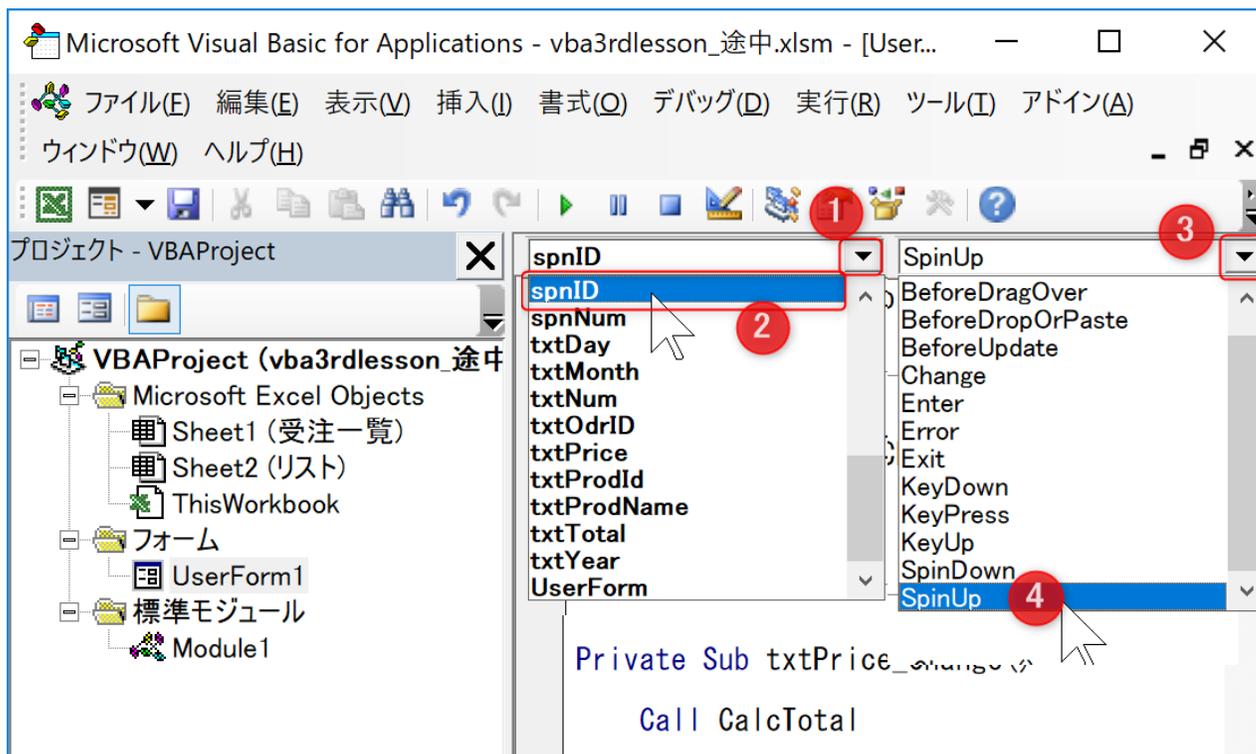
⑤ 「TabIndex」プロパティ隣のTextBoxと同じに変更しておく



[実習] スピンボタンのイベント処理を作る

まずはSpinUpイベント処理を作ろう

①オブジェクト名「spnID」、プロシージャ名「SpinUp」



②イベント処理が挿入される
(※途中で不要な処理が挿入されていたら削除)

```
Private Sub spnID_Change ()
```

削除

```
End Sub
```

```
Private Sub spnID_SpinUp ()
```

```
End Sub
```

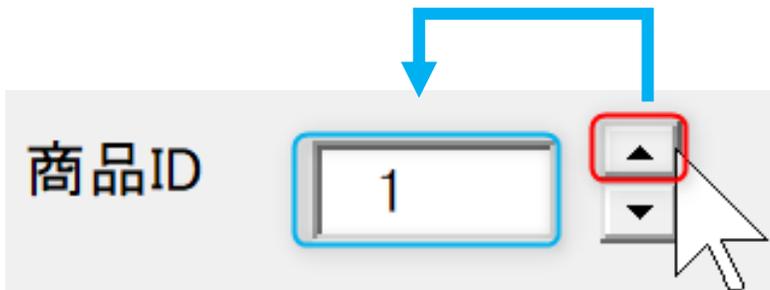
spnIDのSpinUpイベント処理



[実習]SpinUpイベントでIDを増加しよう

SpinUpイベントに対して、商品IDの数値を+1

商品IDの数値を+1



答え（ただしこのままでは問題がある）

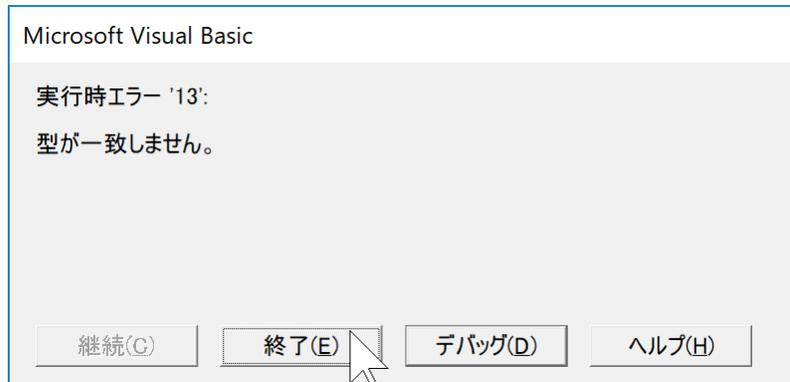
```
Private Sub spnID_SpinUp()  
    txtProdId.Value = txtProdId.Value + 1  
End Sub
```



エラーに対策するには？

商品IDのテキストボックスが""(空白)だったり、文字列が入っていた場合のエラーに対策する

"" (空白)や文字列("a"など)が入力されていた場合、エラー！



IsNumeric関数で条件判断しよう (前章で登場)

IsNumeric関数



数値データかどうか？

Yes

そのまま

No

0で書き換える



[実習] IsNumeric関数でエラー対策しよう！

商品IDに文字列や""(空白)が入っていた場合、0に書きかえる

もしIsNumeric関数の戻り値が「False」なら
(つまり文字列や""(空白)である場合)

```
Private Sub spnID_SpinUp()
```

```
'もし数値じゃなければ0を代入
```

```
If IsNumeric(txtProdId.Value) = False Then
```

```
    txtProdId.Value = 0
```

```
End If
```

```
txtProdId.Value = txtProdId.Value + 1
```

```
End Sub
```

0に書き換える



[実習の結果]

商品IDに数値でないデータが入っていた場合も **エラーにならなくなった!**

"" (空白)や文字列("a"など)が入力されていた場合

商品ID



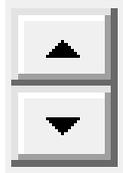
強制的に「0」が挿入され、そこに**+1**

商品ID



[課題] SpinDown イベント処理も作ろう！

SpinUp イベントだけでなく、SpinDown イベントに対して商品IDの数値を-1させよう



SpinDown イベント
▼(下キー)ボタンが押された



SpinDown イベント処理
商品IDの数値を **-1**

spnID_SpinUpのコードをコピー貼り付けで複製し、内容を書き換えればいい

```
Private Sub spnID_SpinDown()  
    'もし数値じゃなければ0を代入  
    If IsNumeric(txtProdId.Value) = False Then  
        txtProdId.Value = 0  
    End If  
  
    txtProdId.Value = txtProdId.Value - 1  
  
End Sub
```



[課題]数量のスピンドボタンも完成させよう！

数量のスピンドボタン (spnNum)にもSpinUp,SpinDownイベント処理を作成する

入力フォーム

受注ID	<input type="text" value="8"/>
日付	<input type="text" value="2018"/> / <input type="text" value="10"/> / <input type="text" value="4"/>
受注元	<input type="text" value=""/>
商品ID	<input type="text" value=""/>
商品名	<input type="text" value=""/>
単価	<input type="text" value=""/>
数量	<input type="text" value=""/>
金額	<input type="text" value=""/>

配送済み
 請求済み
 入金済み

spnNum

SpinUpイベント

SpinDownイベント



数量を+1

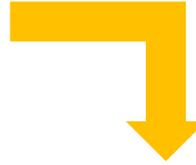
数量を-1

(つづき)

```
Private Sub spnID_SpinUp()  
    'もし数値じゃなければ0を代入  
    If IsNumeric(txtProdId.Value) = False Then  
        txtProdId.Value = 0  
    End If  
  
    txtProdId.Value = txtProdId.Value + 1  
  
End Sub
```

```
Private Sub spnID_SpinDown()  
    'もし数値じゃなければ0を代入  
    If IsNumeric(txtProdId.Value) = False Then  
        txtProdId.Value = 0  
    End If  
  
    txtProdId.Value = txtProdId.Value - 1  
  
End Sub
```

①商品IDのスピンボタン (spnID) のための SpinUp イベント処理と SpinDown イベント処理をコピーして、下の行に貼り付ける



②個数のスピンボタン (spnNum) のための内容に一部書きかえる

```
Private Sub spnNum_SpinUp()  
    'もし数値じゃなければ0を代入  
    If IsNumeric(txtNum.Value) = False Then  
        txtNum.Value = 0  
    End If  
  
    txtNum.Value = txtNum.Value + 1  
  
End Sub
```

```
Private Sub spnNum_SpinDown()  
    'もし数値じゃなければ0を代入  
    If IsNumeric(txtNum.Value) = False Then  
        txtNum.Value = 0  
    End If  
  
    txtNum.Value = txtNum.Value - 1  
  
End Sub
```



[課題の結果]

数量もスピンドタンで操作できるようになった！

単価	<input type="text" value="2000"/>	
数量	<input type="text" value="3"/>	<input type="button" value="▲"/>
		<input type="button" value="▼"/>
金額	<input type="text" value="6000"/>	



いや～便利っすね！
「すっぴんボタン」！



まあまあ、いいじゃない。
それよりホラ、ついに
データ登録フォームが完成したぞう！



(わあ、店長たち、すごい…！！)



…きたきた…
うう、どんな試練なんだろう？



…わざと言ってるんだよな？
「ス・ピ・ン ボタン」だ！



うむ。ここまでで「登録フォーム」の
機能はだいたい完成したといえる。
…だが、**この後は
諸君のスキルを試させてもらう！**



ふっ。覚悟するがいい！

【最終章】

これまでのスキルを応用して挑戦しよう！
「更新フォーム」を作ってみよう！





お？データの更新っスか。
もう登録フォームができてるから、
ちょっと改良すれば楽勝っスね～！



なんだって～！？
自分の力で考える！？



ひ～！
がんばりまーす！



さあ、ここからは応用編だ。
今度は、**データを更新するフォーム**を
作ってもらう。



…と、思うだろう？
ところが**やってみると意外と難しい**わけだ。
しかも、**今回は最初から答えを教えない。**
まず、自分で考えてみてもらおう！



うむ。
今までのように、レッスンをまねるだけでなく、
自分で考えてマクロを作れることが
本当に欲しいスキルというわけだ。



「更新フォーム」を作ってみよう

入力フォームの応用版。でも少し違う点に注意しよう

入力フォーム

入力フォーム

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み
 請求済み
 入金済み



- **入力**するための物
- ボタンを押すと新規データを追加

更新フォーム

更新フォーム

受注ID

日付 / /

受注元

商品ID

商品名

単価

数量

金額

配送済み
 請求済み
 入金済み

- **更新**するための物
- ボタンを押すと既存データを更新



「更新フォーム」の作り方の流れ

入力フォーム

入力フォーム

受注ID 8

日付 2018 / 10 / 10

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

新規追加

複製

入力フォーム(複製)

入力フォーム

受注ID 8

日付 2018 / 10 / 10

受注元

商品ID

商品名

単価

数量

金額

配送済み

請求済み

入金済み

新規追加

変更

更新フォーム

更新フォーム

受注ID 7

日付 2018 / 7 / 9

受注元 関数カンパ

商品ID 9

商品名 カレーセット

単価 2000

数量 5

金額 10000

配送済み

請求済み

入金済み

更新



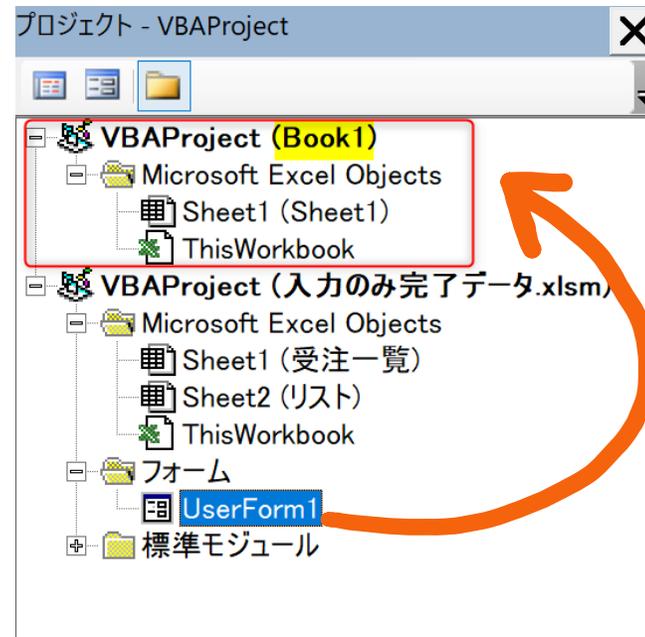
フォーム複製するには？

VBEには、1つのブックに**フォームを複製する機能が無い**。そこで…

①新規Excelブックを作成



②(VBEにて)新規ブックにUserForm1を複製

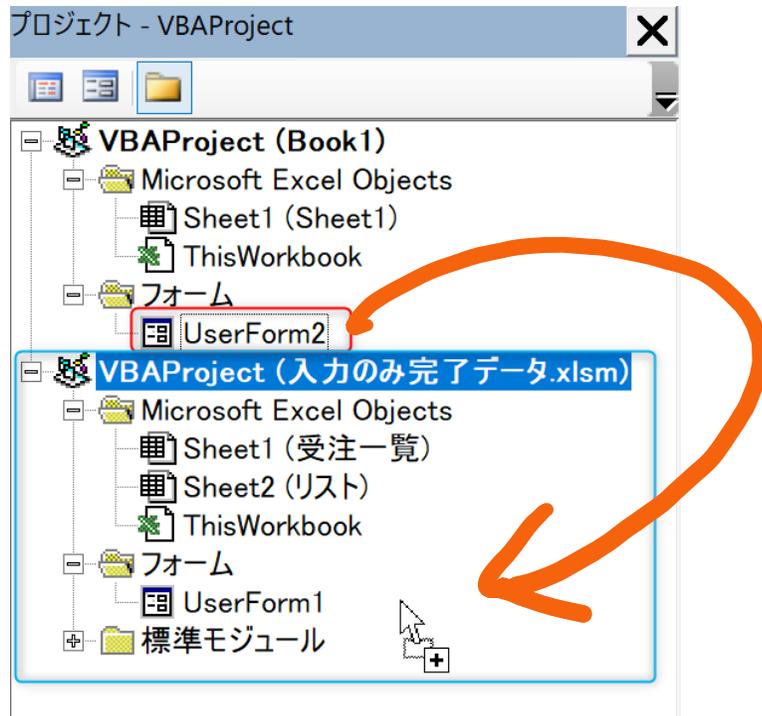


(つづき)

③ フォームのオブジェクト名を
「UserForm2」に変更

④ 「UserForm2」を
もとのプロジェクトに複製

⑤ 1つのプロジェクト内に
UserForm1と2を用意できた！



※新規ブックは保存せずに
閉じておきましょう



①プロパティを変更する

まずは見た目の動作を中心に、プロパティを変更しよう

Captionを変更

Enableプロパティを
「**False**」に変更
(操作不能にする)

更新フォーム

受注ID	<input type="text" value="8"/>	<input type="button" value="↑"/>	<input type="button" value="↓"/>		
日付	<input type="text" value="2018"/>	/	<input type="text" value="10"/>	/	<input type="text" value="10"/>
受注元	<input type="text"/>	<input type="button" value="▼"/>			
商品ID	<input type="text"/>	<input type="button" value="↑"/>	<input type="button" value="↓"/>		
商品名	<input type="text"/>				
単価	<input type="text"/>				
数量	<input type="text"/>	<input type="button" value="↑"/>	<input type="button" value="↓"/>		
金額	<input type="text"/>				
<input type="checkbox"/> 配送済み					
<input type="checkbox"/> 請求済み					
<input type="checkbox"/> 入金済み					
<input type="button" value="更新"/>					

スピンドット
「spnOdrID」を追加

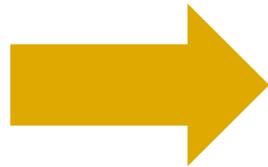
Captionを変更



②初期化(Initialize)処理を変更する

フォームのInitialize (初期化処理) を変更しよう

入力フォームの初期化



更新フォームの初期化

最終ID+1の数

6	2018/7/10	表計算コーポ	3	紅茶葉
7	2018/7/9	関数カンパニ	9	カレーセット

本日の日付

入力フォーム

受注ID: 8

日付: 2018 / 10 / 10

受注元: [dropdown]

商品ID: [dropdown]

商品名: [text]

単価: [text]

数量: [spin]

金額: [text]

配送済み
 請求済み
 入金済み

新規追加

最終データをすべて読み込む

6	2018/7/10	表計算コーポ	3	紅茶葉	2,000	4	8,000	
7	2018/7/9	関数カンパニ	9	カレーセット	2,000	5	10,000	済

更新フォーム

受注ID: 7

日付: 2018 / 7 / 9

受注元: 関数カンパニ

商品ID: 9

商品名: カレーセット

単価: 2000

数量: 5

金額: 10000

配送済み
 請求済み
 入金済み

更新



③ スピンボタンの機能を作る

スピンボタンを押すたびにIDを変え、IDに合うデータを読み込ませよう

受注ID	日付	受注元	商品ID	商品名	単価
1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000
2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000
3	2018/7/4	パワポ建設	3	コーヒー豆	10,000
4	2018/7/6	エクセル運輸	3	コーヒー豆	10,000
5	2018/7/11	VBA商事	7	業務用ストロー	900
6	2018/7/10	表計算コーポ	3	紅茶葉	2,000
7	2018/7/9	関数カンパニ	9	カレーセット	2,000

受注ID	日付	受注元	商品ID	商品名	単価
1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000
2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000
3	2018/7/4	パワポ建設	3	コーヒー豆	10,000
4	2018/7/6	エクセル運輸	3	コーヒー豆	10,000
5	2018/7/11	VBA商事	7	業務用ストロー	900
6	2018/7/10	表計算コーポ	3	紅茶葉	2,000
7	2018/7/9	関数カンパニ	9	カレーセット	2,000

受注ID「7」
のデータ

更新フォーム

受注ID: 7

日付: 2018 / 7 /

受注元: 関数カンパニ

商品ID: 9

商品名: カレーセット

単価: 2000

数量: 5

金額: 10000

配送済み
 請求済み
 入金済み

更新

7

受注ID「6」
のデータ

更新フォーム

受注ID: 6

日付: 2018 / 7 /

受注元: 表計算コーポ

商品ID: 3

商品名: 紅茶葉

単価: 2000

数量: 4

金額: 8000

配送済み
 請求済み
 入金済み

更新

6



④ スピンボタンのエラーに対策する

存在しない受注IDが入力されたとき、エラーを回避する仕組みを作ろう

受注ID	日付	受注元	商品
1	2018/6/1	VBA商事	
2	2018/7/3	VBA商事	
3	2018/7/4	パワポ建設	
4	2018/7/6	エクセル運輸	
5	2018/7/11	VBA商事	
6	2018/7/10	表計算コーポ	
7	2018/7/9	関数カンパニ	

受注一覧表

更新フォーム

受注ID

日付 / /

金額

Microsoft Excel

データが見つかりませんでした

OK



⑤ ボタンを押したらデータを更新する

「更新」ボタンを押したとき、フォームからExcelシートにデータを上書きしよう

受注ID	日付	受注元	商品ID	商品名	単価	数量
1	2018/6/1	VBA商事	1	オフィス用コーヒー	2,000	2
2	2018/7/3	VBA商事	2	オフィス用カフェラテ	3,000	3
3	2018/7/4	パワポ建設	3	コーヒー豆	10,000	2
4	2018/7/6	エクセル運輸	4	...	10,000	3
5	2018/7/11	VBA	5	...	900	7
6	2018/7/10	表計	6	...	2,000	4
7	2018/7/9	関数	7	...	2,000	5

更新フォーム

受注ID: 3

日付: 2018 / 7 / 4

受注元: VBA商事

商品ID: 3

商品名: コーヒー豆

単価: 10000

数量: 2

金額: 20000

配送済み
 請求済み
 入金済み

更新



ふう〜。
作ってみると結構、難しかったなあ〜
「データ更新フォーム」。



(店長たちが頑張ってくれた
フォーム…大事に使おう ^^)



うんうん。よお〜し！
これからもマクロVBAのスキルを
磨いて、みんなの仕事を効率的に
するぞう！



うむ。
「これまでの応用で作れる」と分かって
いても、**やってみるとなかなか難しいこ
と**が分かっただろう？

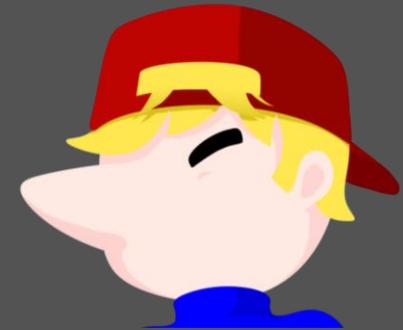


ノア先輩、仕事がサクサク進むよう
になって、喜んでるっスね〜！



**「ユーザー（使う人）」のことを考えて
マクロを作る。**
これも大事なテーマの一つだったな。

これからも、VBAの実践スキルを
いっしょに磨いていこう！





ご受講ありがとうございました！



- ✓ 生放送レッスンの配信
- ✓ 第4弾、第5弾、
VBAエキスパート試験対策など
今後の作品も予定
- ✓ よろしくお願ひします！