

エクセルの関数を使いこなすことができれば、複雑な計算についても簡単に処理することができます。ここでは、経理業務に役立つ6つの関数を取り上げて、その使い方を解説します。

経理実務に 本当に役立つ エクセルの 関数6選

BizImprove
Excel業務改善コンサルタント
小野 眸

端数を四捨五入する ROUND関数

●端数処理をした場合の計算結果
には注意が必要

数式の計算結果が小数点付きの数値になったとき、どう対処していますか。

セルの書式設定を変更して、小数点以下の桁数を0にして端数処理をしないでしょうか。

セルの書式設定で小数点以下の桁数を0にすると、たしかに四捨五入されて表示されます。しかし、これは見た目が変わるだけで、データそのものは小数点付きのままです。

たとえば、図表1の上のような

図表1

	A	B	C	D
1				
2		金額	料率	手数料
3	A支店	1,274	7%	89.18
4	B支店	1,339	6%	80.34
5	合計	2,613		169.52
6				

セルの書式設定で小数点以下の桁数を0にすると…

	A	B	C	D
1				
2		金額	料率	手数料
3	A支店	1,274	7%	89
4	B支店	1,339	6%	80
5	合計	2,613		170
6				

間違った計算結果が表示される場合がある

表をつくり、計算結果を表示してみます。

次に、D3～D5のセルについて書式設定で小数点以下の桁数を0に設定すると、上から89、80、170と表示されます（下の表）。

これでは、結果的に89 + 80 = 170という計算ミスが生じています。安易に書式設定を変更するだけでは、このようなトラブルが起こりやすくなるのです。

こうした弊害を避けるためには、きちんと関数を使って計算することが大切です。

方針としては、D3、D4の2つのセルだけを四捨五入して端数処理し、その結果をD5で合計します。

こうすれば、正しい計算結果が整数で表示されます。

その際、使うのはROUND関数です。

ROUND関数の引数は、数値と桁数です。数値には四捨五入の対象となるセル番地または数式を設定します（次ページ図表2）。桁数は四捨五入する位置を指定します。たとえば、桁数に0を指定すると、小数点第一位が四捨五入され、1を指定すると小数点第二位

が四捨五入されます（図表3）。
●**切上げ、切捨ては別の関数で設定する**

なお、端数を切上げで計算したい場合はROUNDUP関数、切捨てで計算したい場合はROUNDDOWN関数を使います。それぞれ、引数の考え方はROUND関数と同じです。

条件付きの計算ができる IF関数

エクセルでよく使う関数の一つに、IF関数があります。条件によって結果を変えられるIF関数は便利な関数ですが、思

図表2

関数の引数			
ROUND			
数値	B3*C3	=	89.18
桁数	0	=	0
		=	89
数値を指定した桁数に四捨五入した値を返します。			
数値 には四捨五入の対象となる数値を指定します。			
数式の結果 = 89			
この関数のヘルプ(H)			

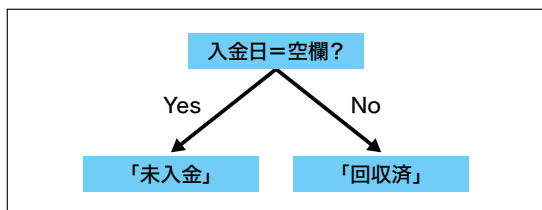
図表3

ROUND関数の桁数について	
例) 135.68	
「1」	⇒ 「135.7」
「0」	⇒ 「136」
「-1」	⇒ 「140」

図表4

	A	B	C	D
1				
2	取引先名	金額	入金日	回収
3	C商事	16,000	1月31日	回収済
4	D食品	24,000		未入金
5	E販売	18,000	1月20日	回収済

図表5



図表6

関数の引数			
IF			
論理式	C3=""	=	FALSE
値が真の場合	"未入金"	=	"未入金"
値が偽の場合	"回収済"	=	"回収済"
		=	"回収済"
論理式の結果（真または偽）に応じて、指定された値を返します			
論理式 には結果が真または偽になる値、もしくは数式を指定します			
数式の結果 = 回収済			

うように使いこなせない難しい関数とも言われています。

IF関数を使った数式をつくるには、ロジックを考える工程と引数を入力する工程があります。

エクセルの画面に向かい、ロジックを考えながら引数を入力すると難易度が上がってしまいます。そこで、初めてIF関数をつくる人にもおすすめの、ロジックを図で考える方法です。

図表4は、取引先ごとの金額と入金日が入力された表です。「回収」の列に「入金日が空欄なら未入金、それ以外は回収済」と表示する数式をつくりま

す。エクセルの画面に向かう前に、

ロジックを考えます。

まず条件は「入金日=空欄？」となります。条件がYesなら「未入金」、Noなら「回収済」を表示するロジックです（図表5）。

ロジックを考えると、セル番地を意識せずデータの項目名で考えます。

ロジックを考えたら、数式をつくる前にロジックが合っているか検証します。

ロジックが間違っていないことが確認できたら、エクセルの画面に向かってロジックの図を参考に、項目名をセル番地に置き換えて引数を設定し数式をつくりま

●**ロジックの図でIF関数を可視化する**

IF関数は作成されたものを解読するのも大変です。そこで、IF関数のロジックを図式化したものをシート内に保存しておけば、ロジックが可視化されるため、数式の修正が楽になります。

複数条件に合うデータを集計する SUMIFS関数

複数の条件に合致したデータを集計するのは、SUMIFS関数です。

図表7

	A	B	C	D	E	F	G	H
1								
2	日付	内容	部署	社員	金額		部署	交通費
3	1月21日	交通費	営業部	社員A	20,000		営業部	34,000
4	1月22日	交通費	総務部	社員B	35,000			
5	1月23日	交通費	営業部	社員C	5,000			
6	1月25日	消耗品費	経理部	社員D	20,000			
7	1月25日	消耗品費	営業部	社員E	12,000			
8	1月26日	消耗品費	総務部	社員F	4,600			
9	1月26日	交通費	営業部	社員G	9,000			

指定した範囲を無条件に集計するのがSUM関数、複数の条件を指定できるのがSUMIFS関数と考えると理解しやすいでしょう。SUMIFS関数には、1つ以上の条件を指定できます。

図表7は、経費の明細を入力した一覧表です。

ここからたとえば、「営業部」の「交通費」を合計したいとき、

図表8

関数の引数

SUMIFS

合計対象範囲 E3:E9 = {20000;35000;5000;20000;12000;4600;9000}

条件範囲1 C3:C9 = {"営業部";"総務部";"営業部";"経理部";"営業部";"総務部";"営業部"}

条件1 G3 = "営業部"

条件範囲2 B3:B9 = {"交通費";"交通費";"交通費";"消耗品費";"消耗品費";"消耗品費";"交通費"}

条件2 H2 = "交通費"

= 34000

特定の条件に一致する数値の合計を求めます

条件2: には、計算の対象となるセルを定義する条件を数値、式、または文字列で指定します

数式の結果 = 34,000

SUMIFS関数を使います。

最初の引数「合計対象範囲」は条件に合致したデータを集計するセル範囲、2番目の引数「条件範囲1」は条件1を探す対象範囲、そして3番目の引数「条件1」に条件を指定します。

指定する条件が2つ以上ある場合は、条件範囲2と条件2、条件範囲nと条件nの引数を設定します(図表8)。

●合計対象範囲…金額(E3～E9セル)

図表9

部署別内容別	内容			
部署名	交通費	消耗品費	総計	
営業部	34,000	12,000	46,000	
経理部	0	20,000	20,000	
総務部	35,000	4,600	39,600	
総計	69,000	36,600	105,600	

ル)

●条件範囲1…部署(C3～C9セル)

●条件1…G3セルの入力データ(営業部)

●条件範囲2…内容(B3～B9セル)

●条件2…H2セルの入力データ(交通費)

なお、条件1～nはセル番地だけでなく、数式、数値、文字列でも指定できます。

また、完成例の数式は相対参照で作成していますが、完成したSUMIFS関数の数式を縦や横にコピーする場合、絶対参照や複合参照などセルの参照形式を変更する必要がありますので、注意しましょう。

●SUMIFS関数の検算はピボットテーブルが便利

SUMIFS関数に限らず、数式を作成したら必ず検算を行ないましょう。

SUMIFS関数の検算にはピボットテーブルがおすすめです。なぜなら、数式は人がつくりますが、ピボットテーブルはエクセルが自動で集計してくれるからです。ちなみに、先ほどの表をピボットテーブルで部署別内容別に集計すると、図表9のような結果になります。

入力箇所を減らす
VLOOKUP関数

●別表からデータを参照するVLOOKUP関数

VLOOKUP関数は難易度の高い関数の1つですが、世の中で最も利用されている関数のベスト3に入ると言われるほど職種業種を問わず広く利用されています。

VLOOKUP関数を一言で言うとは、別の表から必要な情報を参照できる関数です。

つまり、VLOOKUP関数を使用するには、参照したいデータが入力された表が別途必要です。

図表10

	A	B	C	D	E	F	G	H	I
1									
2		売上一覧				入金一覧			
3	コード	取引先名	売上額	入金額		日付	コード	取引先名	入金額
4	H01	A商事	23,600	23,600		1月20日	H02	B食品	24,000
5	H02	B食品	24,300	24,000		1月20日	H05	E販売	15,000
6	H03	C商事	18,500	18,500		1月25日	H03	C商事	18,500
7	H04	D食品	14,700	14,700		1月31日	H01	A商事	23,600
8	H05	E販売	15,240	15,000		1月31日	H04	D食品	14,700

図表11

関数の引数

VLOOKUP

検索値 A4 = "H01"

範囲 \$G\$4:\$I\$8 = {"H02","B食品",24000;"H05","E販売"...}

列番号 3 = 3

検索方法 FALSE = FALSE

= 23600

指定された範囲の 1 列目で特定の値を検索し、指定した列と同じ行にある値を返します。テーブルは昇順で並べ替えておく必要があります。

検索値 には範囲の先頭列で検索する値を指定します。検索値には、値、セル参照、または文字列を指定します。

数式の結果 = 23,600

この関数のヘルプ(H)

OK キャンセル

図表12

	A	B	C	D	E	F	G
1							
2		売上一覧				割引率一覧	
3	取引先名	売上額	割引率	割引額		金額	割引率
4	F商事	15,000	0%	0		0	0%
5	G販売	20,000	3%	600		20,000	3%
6	H産業	68,000	4%	2720		50,000	4%
7						100,000	5%

図表13

関数の引数

VLOOKUP

検索値 B4 = 15000

範囲 \$F\$4:\$G\$7 = {0,0;20000,0.03;50000,0.04;100000,0.05}

列番号 2 = 2

検索方法 TRUE = TRUE

= 0

指定された範囲の 1 列目で特定の値を検索し、指定した列と同じ行にある値を返します。テーブルは昇順で並べ替えておく必要があります。

検索値 には範囲の先頭列で検索する値を指定します。検索値には、値、セル参照、または文字列を指定します。

別の表からデータを参照するため
に同じ項目が両方の表に必要で、
その項目を検索値として利用し、
必要なデータを参照します。

● 図表10の左側の表は、取引先ご
との売上額が入力されています。
入金額の列はVLOOKUP関数
を使って右側の入金一覧を参照し
入金額を表示しています。

● この例では、コードが検索値で
す。VLOOKUP関数の引数の
内容は次のとおりです(図表11)。
● 検索値・両方の表に存在する項
目(コード)

● 範囲・参照する範囲(入金一覧

のコード)入金額)

● 列番号・参照する範囲の列番号
を数値で指定(直前の引数で指
定した範囲の3列目)

● 検索方法・TRUE(近似値を
含めて検索)またはFALSE
(完全一致で検索)

● VLOOKUP関数を使いこな
すポイントは、次の3つです。

- ① 参照する別表の最左列は検索
値に使用する項目(例:コード)
- ② 検索値のデータを統一(例:
半角文字と全角文字では同じも
のとは認識されない)
- ③ 検索値として使用する項目は

別表内で重複不可(例:入金一
覧のコード)

● TRUE検索も使いこなせば、
VLOOKUP関数の達人

● 図表12は、売上金額に応じた割
引率を参照するVLOOKUP関
数です。参照する表は、右側の表
です。

● 検索値・両方の表に存在する項
目(売上額)

● 範囲・参照する範囲(割引率一
覧)

● 列番号・参照する範囲の列番号
を数値で指定(直前の引数で指
定した範囲の2列目)

● 検索方法・TRUE(近似値を
含めて検索)またはFALSE
(完全一致で検索)

たとえば、F商事の割引率がV
LOOKUP関数でどのように検
索されているかというと、参照す
る範囲の最左列を上から順番に検
索値5000を検索します。近似値
検索なので0を超えて20000は超
えないので割引率0%という計算
結果になります(図表13)。

● VLOOKUP関数の近似値で
の検索を使いこなすには、参照す
る表のつくり方がポイントです。

- ① △△円までが〇〇、□□円ま
でが××...という表にする
- ② 最小がないとエラーになる
(例では0円)
- ③ 検索値を昇順に並べる(例で
は金額)

複雑な条件もスッキリ指定できる
IFS関数

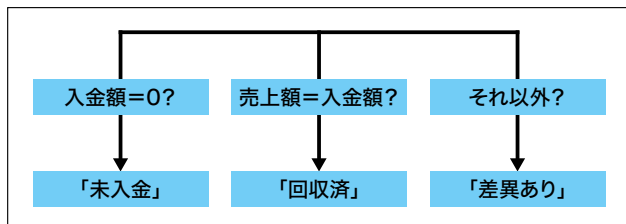
● IFS関数はIFS関数が進化し
た関数

IFS関数は複数の条件を指定し
たネスト(入れ子)にして利用す
ることが多いため、数式が複雑に
なり難易度が上がってしまいま
す。それを解消できるのがIFS

図表14

	A	B	C	D	E
1					
2					
3	コード	取引先名	売上額	入金額	状況
4	H01	A商事	23,600	0	未入金
5	H02	B食品	24,300	24,300	回収済
6	H03	C商事	18,500	15,000	差異あり
7	H04	D食品	14,700	14,700	回収済
8	H05	E販売	15,240	0	未入金

図表15



図表16

関数の引数

IFS

論理式1: D4=0 = TRUE

値が真の場合1: "未入金" = "未入金"

論理式2: C4=D4 = FALSE

値が真の場合2: "回収済" = "回収済"

論理式3: TRUE = TRUE

値が真の場合3: "未入金"

1 つ以上の条件が満たされるかどうかを確認し、最初の真条件に対応する値を返します

論理式1: は 真または偽と計算できる値または式です

図表17

	A	B	C	D	E	F
1						
2						
3	日付	コード	取引先名	売上額	入金額	売上月
4	2022/1/20	H01	A商事	23,600	23,600	1月
5	2022/1/25	H02	B食品	24,300	24,300	1月
6	2022/2/10	H03	C商事	18,500	15,000	2月
7	2022/2/20	H04	D食品	14,700	14,700	2月
8	2022/3/20	H05	E販売	15,240	0	3月
9	2022/3/21	H06	F商事	19,600	0	3月

図表18

関数の引数

TEXT

値: A4 = 44581

表示形式: "m月" = "m月"

数値に指定した書式を設定し、文字列に変換した結果を返します。

数値の結果 = 1月

この関数のヘルプ(H)

OK キャンセル

関数です。

IFS 関数は、エクセル2019から新たに導入されました。

図表14は、取引先ごとの売上額と入金額が入力された表です。

「状況」の列に「入金額が0なら未入金、売上額と入金額が同じなら回収済、それ以外を差異あり」と表示したいとき、IFS関数を使います。

IFS関数も使えますが、IFS関数のほうがわかりやすくメンテナンスしやすい数式になります。IFS関数も、IFS関数と同じようにロジックを図で考えると

つくりやすくなります(図表15)。

IFS関数の引数は論理式と値が真の場合をセットで考え、最後にそれ以外を指定します。それ以外の論理式は、「TRUE」と指定します(図表16)。

表示形式を自由に設定できるTEXT関数

入力された数値データや計算結果を任意の文字列や文章で表示したいとき、TEXT関数を使うと便利です。

TEXT関数は、数値や日付な

どの表示形式を自由に設定することができま

図表17のA～E列は、取引先ごとの売上額と入金額が入力された表です。この表を元にして、F列に1月、2月...と「売上月」を入れることを考えます(要は、

「2022/1/20」などの表示を「1月」に変換します)。

TEXT関数の最初の引数「値」には数値、数式、対象のセルを指定します。2番目の引数は、表示形式を指定します。

日付の表示形式は年をY、月をm、日をdで設定します(図表18)。

引数に設定する表示形式は、セルの書式設定のユーザー定義を参考にするとよいでしょう(数値を千円、百万円単位の表示にしたり位取りのカンマを入れたりすることがデータ上でできます)。

エクセルの関数については、苦手意識を感じている人が多いかも知れません。しかしエクセルには、今回紹介したもの以外にも有益な関数がたくさんあります。これらを存分に使いこなして、作業効率アップや業務改善に役立てていただきたいと思います。