

Chương 3: Giới Thiệu Và Sử Dụng Các Hàm

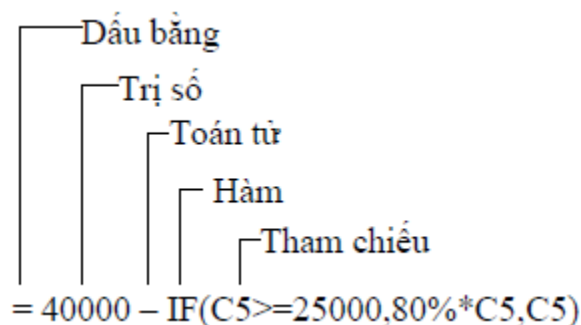
3.1 Giới thiệu công thức và hàm:

Giới thiệu công thức (Formula)

Công thức giúp bảng tính hữu ích hơn rất nhiều, nếu không có các công thức thì bảng tính cũng giống như trình soạn thảo văn bản. Chúng ta dùng công thức để tính toán từ các dữ liệu lưu trữ trên bảng tính, khi dữ liệu thay đổi các công thức này sẽ tự động cập nhật các thay đổi và tính ra kết quả mới giúp chúng ta đỡ tốn công sức tính lại nhiều lần. Vậy công thức có các thành phần gì?

Công thức trong Excel được nhận dạng là do nó bắt đầu là dấu = và sau đó là sự kết hợp của các toán tử, các trị số, các địa chỉ tham chiếu và các hàm.

Ví dụ:



Ví dụ về công thức

Các toán tử trong công thức

Toán tử	Chức năng	Ví dụ	Kết quả
+	Cộng	=3+3	3 cộng 3 là 6

-	Trừ	=45-4	45 trừ 4 còn 41
*	Nhân	=150*.05	150 nhân 0.50 thành 7.5
/	Chia	=3/3	3 chia 3 là 1
^	Lũy thừa	=2^4 =16^(1/4)	2 lũy thừa 4 thành 16 Lấy căn bậc 4 của 16 thành 2
&	Nối chuỗi	=”Lê” & “Thanh”	Nối chuỗi “Lê” và “Thanh” lại thành “Lê Thanh”
=	Bằng	=A1=B1	Ví dụ ô A1=3, ô B1=6 Kết quả: FALSE
>	Lớn hơn	=A1>B1	Ví dụ ô A1=3, ô B1=6 Kết quả: FALSE
<	Nhỏ hơn	=A1<B1	Ví dụ ô A1=3, ô B1=6 Kết quả: TRUE
>=	Lớn hơn hoặc bằng	=A1>=B1	Ví dụ ô A1=3, ô B1=6 Kết quả: FALSE
<=	Nhỏ hơn hoặc bằng	=A1<=B1	Ví dụ ô A1=3, ô B1=6 Kết quả: TRUE
<>	Khác	=A1<>B1	Ví dụ ô A1=3, ô B1=6 Kết

			quả: TRUE
,	Dấu cách các tham chiếu	=Sum(A1,B1)	Ví dụ ô A1=3, ô B1=6 Kết quả: 9
:	Tham chiếu mảng	=Sum(A1:B1)	Ví dụ ô A1=3, ô B1=6 Kết quả: 9
khoảng trắng	Trả về các ô giao giữa 2 vùng	=B1:B6 A3:D3	Trả về giá trị của ô

Thứ tự ưu tiên của các toán tử

<i>Toán tử</i>	<i>Mô tả</i>	<i>Ưu tiên</i>
: (hai chấm) (1 khoảng trắng) , (dấu phẩy)	Toán tử tham chiếu	1
–	Số âm (ví dụ –1)	2
%	Phần trăm	3
^	Lũy thừa	4
* và /	Nhân và chia	5
+ và –	Cộng và trừ	6

&	Nối chuỗi	7
= < > <= >= <>	So sánh	8

Giới thiệu hàm (Function)

Hàm trong Excel được lập trình sẵn dùng tính toán hoặc thực hiện một chức năng nào đó. Việc sử dụng thành thạo các hàm sẽ giúp chúng ta tiết kiệm được rất nhiều thời gian so với tính toán thủ công không dùng hàm. Các hàm trong Excel rất đa dạng bao trùm nhiều lĩnh vực, có những hàm không yêu cầu đối số, có những hàm yêu cầu một hoặc nhiều đối số, và các đối số có thể là bắt buộc hoặc tự chọn.

Ví dụ:

=Rand(): hàm không có đối số

=If(A1>=5,"Đạt","Rớt"): hàm 3 đối số

=PMT(10%,4,1000,,1): hàm nhiều đối số và đối số tùy chọn

Trong Excel 2010 có các hàm chính như sau:

- Hàm ngoại: Call, Registered.ID,...
- Hàm lấy dữ liệu từ SSAS: Cubeset, Cubevalue,...
- Hàm dữ liệu: Dmin, Dmax, Dcount,...
- Hàm ngày và thời gian: Time, Now, Date,....
- Hàm kỹ thuật: Dec2Bin, Dec2Hex, Dec2Oct,...
- Hàm tài chính: Npv, Pv, Fv, Rate,...

- Hàm thông tin: Cell, Thông tin, IsNa,...
- Hàm luận lý: If, And, Or,...
- Hàm tham chiếu và tìm kiếm: Choose, Vlookup, Offset,...
- Hàm toán và lượng giác: Log, Mmult, Round,...
- Hàm thống kê: Stdev, Var, CountIf,...
- Hàm văn bản: Asc, Find, Text,...

Chúng ta sẽ cùng nghiên cứu cách sử dụng từng hàm trong các nhóm hàm trên ở các phần sau.

Nhập công thức và hàm

Nhập công thức trong Excel rất đơn giản, muốn nhập công thức vào ô nào bạn chỉ việc nhập dấu = và sau đó là sự kết hợp của các toán tử, các trị số, các địa chỉ tham chiếu và các hàm. Bạn có thể nhìn vào thanh Formula để thấy được trọn công thức. Một điều hết sức lưu ý khi làm việc trên bảng tính là tránh nhập trực tiếp các con số, giá trị vào công thức mà bạn nên dùng đến tham chiếu.

Ví dụ:

	A	B	C	D	E	F	G	H	I
1	Năm	0	1	2	3	4	5		r
2	Dòng tiền	-2,000	-100	600	800	1,100	900		10%
3									
4	NPV	=NPV(I2,\$C\$2:\$G\$2)+B2							

Minh họa dùng tham chiếu trong hàm

Trong ví dụ trên, ở đối số thứ nhất của hàm NPV chúng ta không nhập trực suất chiết tính 10% vào hàm mà nên tham chiếu đến địa chỉ ô chứa nó là **I2**, vì nếu lỗi

suất có thay đổi thì ta chỉ cần nhập giá trị mới vào ô **I2** thì chúng ta sẽ thu được kết quả NPV mới ngay không cần phải chỉnh sửa lại công thức.

Giả sử các ô **C2:G2** được đặt tên là **DongTien**, và ô **I2** đặt tên là **LaiSuat** (Xem lại cách đặt tên vùng ở bài số 1) thì trong quá trình nhập công thức bạn có thể làm như sau:

B1. Tại ô **B4** nhập vào **=NPV(**

B2. Nhấn **F3**, cửa sổ **Paste Name** hiện ra

B3. Chọn **LaiSuat** và nhấn **OK**

B4. Nhập dấu **phẩy** (,) và gõ **F3**

B5. Chọn **DongTien** và nhấn **OK**


B6. Nhập dấu đóng ngoặc rồi nhập dấu **+**

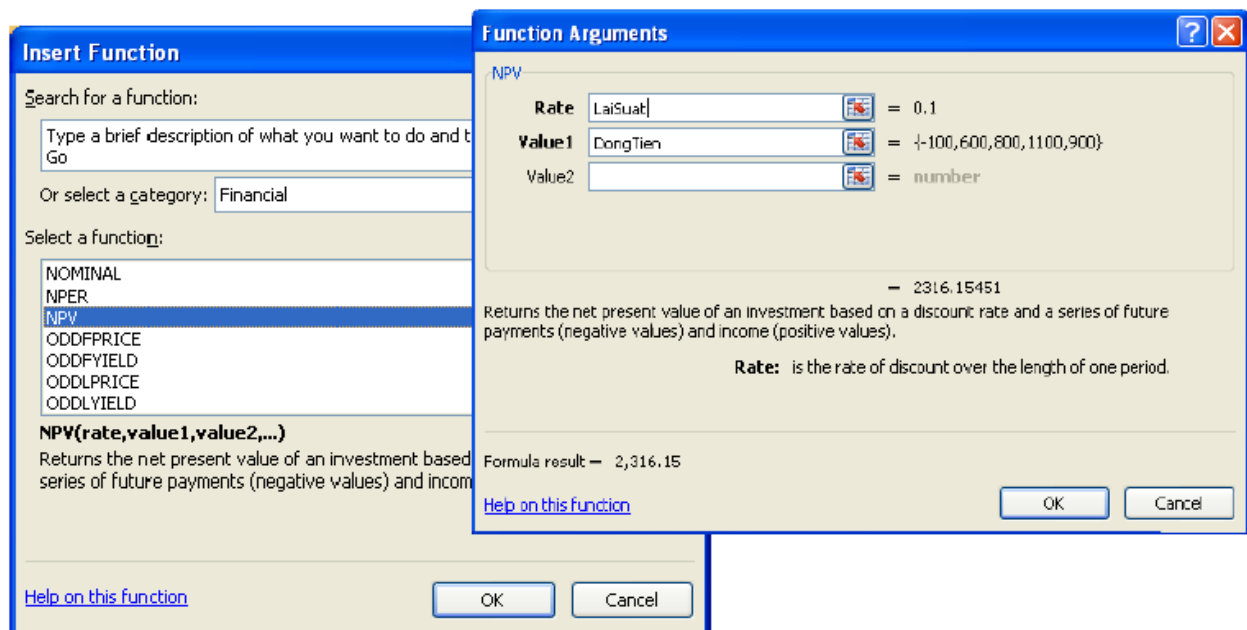
B7. Nhấp chuột vào ô **B2**

B8. Nhấn phím **Enter**

	A	B	C	D	E	F	G	H	I
1	Năm	0	1	2	3	4	5		r
2	Dòng tiền	-2,000	-100	600	800	1,100	900		10%
3									
4	NPV	=NPV(LaiSuat,							
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									

Chèn tên vùng vào công thức

Một trong những cách dễ dàng nhất để sử dụng hàm trong Excel là sử dụng thư viện hàm. Khi bạn muốn sử dụng hàm nào chỉ việc vào thanh **Ribbon** □ chọn nhóm **Formulas** □ **Function Library** -> chọn nhóm hàm -> chọn hàm cần sử dụng. Ngoài ra bạn có thể nhấn vào nút  để gọi hộp thoại **Insert Function** một cách nhanh chóng và khi cần tìm hiểu về hàm này bạn chỉ cần nhấn vào **Help on this function**.



Hộp thoại Insert Function

Tham chiếu trong công thức

Các tham chiếu sử dụng trong công thức giúp cho chúng ta khỏi tốn công sửa chữa các công thức khi các giá trị tính toán có sự thay đổi. Có 3 loại tham chiếu sau:

- Tham chiếu địa chỉ tương đối: Các dòng và cột tham chiếu sẽ thay đổi khi chúng ta sao chép hoặc di dời công thức đến vị trí khác một lượng tương ứng với số dòng và số cột mà ta di dời. Ví dụ A5:B7, C4
- Tham chiếu địa chỉ tuyệt đối: Các dòng và cột tham chiếu không thay đổi khi ta di dời hay sao chép công thức. Ví dụ \$A\$5:\$B\$7, \$C\$4
- Tham chiếu hỗn hợp: Phối hợp tham chiếu địa chỉ tương đối và tuyệt đối. Ví dụ A\$5 nghĩa là cột A tương đối và dòng 5 tuyệt đối.

Lưu ý: Dấu \$ trước thứ tự cột là cố định cột và trước thứ tự dòng là cố định dòng. Nhấn phím **F4** nhiều lần để (tuyệt đối) cố định/ bỏ cố định dòng hoặc cột.

Nhấn phím **F4** nhiều lần để (tuyệt đối) cố định/ bỏ cố định dòng hoặc cột.

Ví dụ: Tính thành tiền bằng Số lượng nhân Giá. Đổi sang giá trị Thành tiền sang VND. Tính tổng các cột Thành tiền và cột VND.

	A	B	C	D	E		
1	Người mua	Số lượng	Giá	Thành tiền	Tiền VND		
2	Bình	4	\$2.50				
3	Bình	2	\$2.50				
4	Bình	1	\$2.50				
5	Bình	3	\$2.50				
6	Đức	1	\$1.25				
7	Đức	4	\$1.25				
8	Đức	6	\$1.25				
9	Hoa	2	\$1.00				
10	Hoa	2	\$1.00				
11	Hoa	3	\$1.00				
12	Thanh	1	\$1.25				
13	Thanh	2	\$1.25				
14	Thanh	2	\$1.25				
15	Tổng						
16							
17	Tỷ giá	16,000					

	A	B	C	D	E	F	G
1	Người mua	Số lượng	Giá	Thành tiền	Tiền VND		
2	Bình	4	\$2.50	\$10.00	160,000		D2=B2*C2
3	Bình	2	\$2.50	\$5.00	80,000		D3=B3*C3
4	Bình	1	\$2.50	\$2.50	40,000		D4=B4*C4
5	Bình	3	\$2.50	\$7.50	120,000		D5=B5*C5
6	Đức	1	\$1.25	\$1.25	20,000		D6=B6*C6
7	Đức	4	\$1.25	\$5.00	80,000		D7=B7*C7
8	Đức	6	\$1.25	\$7.50	120,000		D8=B8*C8
9	Hoa	2	\$1.00	\$2.00	32,000		D9=B9*C9
10	Hoa	2	\$1.00	\$2.00	32,000		D10=B10*C10
11	Hoa	3	\$1.00	\$3.00	48,000		D11=B11*C11
12	Thanh	1	\$1.25	\$1.25	20,000		D12=B12*C12
13	Thanh	2	\$1.25	\$2.50	40,000		D13=B13*C13
14	Thanh	2	\$1.25	\$2.50	40,000		D14=B14*C14
15	Tổng			\$52.00	832,000		
16							
17	Tỷ giá	16,000		D15=SUM(D2:D14)			E2=D2*B\$17

Minh họa địa chỉ tương đối và tuyệt đối

B1. Tại ô **D2** nhập vào **=B2*C2** và **Enter**. Sau đó quét chọn cả vùng **D2:D14** và gõ **<Ctrl+D>**. Vào các ô **D3, D4... D14** ta thấy công thức các dòng tự động được thay đổi tương ứng với khoảng cách so với ô **D2**. Trường hợp này chúng ta dùng địa chỉ tương đối của **B2*C2** là vì chúng ta muốn khi sao chép công thức xuống phía dưới thì địa chỉ các ô tính toán sẽ tự động thay đổi theo.

B2. Tại ô **E2** nhập vào **=D2*B\$17** và **Enter**, sau đó chép công thức xuống các ô **E3:E14**. Chúng ta cần cố định dòng 17 trong địa chỉ tỷ giá **B17** vì ta muốn khi sao chép công thức xuống thì các công thức sao chép vẫn tham chiếu đến ô **B17** để tính toán.

B3. Tại ô **D15** nhập vào **=Sum(D2:D14)** và chép sang ô **E15**.

Lưu ý:

- Tham chiếu đến địa chỉ ở worksheet khác nhưng cùng workbook thì có dạng
- **Tên_sheet!Địa_chi_ô**. Ví dụ:

=A2*Sheet2!A2

=A2*'Thong so'!B4

Khi tên sheet có chứa khoảng trắng thì để trong cặp nháy đơn ' '

- Tham chiếu đến địa chỉ trong workbook khác thì có dạng

[Tên_Workbook]Tên_sheet!Địa_chi_ô.

Ví dụ:

=A2*[Bai2.xlsx]Sheet3!A4

=A2*'Bai tap 2.xlsx]Sheet3'!A4

Khi tên Sheet hay Workbook có chứa khoản trắng để trong cặp nháy đơn ' '

=A2*'C:\Tai lieu\[Bai tap 2.xlsx]Sheet3'!A4

Khi tham chiếu đến workbook khác mà workbook này không mở

=A2*'\DataServer\Excel\[Bai tap 2.xlsx]Sheet3'!A4

Khi tham chiếu đến tài nguyên chia sẻ trên máy chủ trong mạng

Các lỗi thông dụng (Formulas errors)

Các lỗi thông dụng

Lỗi	Giải thích
#DIV/0!	Trong công thức có chứa phép chia cho 0 (zero) hoặc chia ô rỗng
#NAME?	Do đánh sai tên hàm hay tham chiếu hoặc đánh thiếu dấu nháy
#N/A	Công thức tham chiếu đến ô mà có dùng hàm NA để kiểm tra sự tồn tại của dữ liệu hoặc hàm không có kết quả
#NULL!	Hàm sử dụng dữ liệu giao nhau của 2 vùng mà 2 vùng này không có phần chung nên phần giao rỗng
#NUM!	Vấn đề đối với giá trị, ví dụ như dùng nhầm số âm trong khi đúng phải là số dương
#REF!	Tham chiếu bị lỗi, thường là do ô tham chiếu trong hàm bị xóa
#VALUE!	Công thức tính toán có chứa kiểu dữ liệu không đúng.

3.2 Các hàm trong excel

a. Nhóm hàm về thống kê

AVEDEV (number1, number2, ...) : Tính trung bình độ lệch tuyệt đối các điểm dữ liệu theo trung bình của chúng. Thường dùng làm thước đo về sự biến đổi của tập số liệu

AVERAGE (number1, number2, ...) : Tính trung bình cộng

AVERAGEA (number1, number2, ...) : Tính trung bình cộng của các giá trị, bao gồm cả những giá trị logic

AVERAGEIF (range, criteria1) : Tính trung bình cộng của các giá trị trong một mảng theo một điều kiện

AVERAGEIFS (range, criteria1, criteria2, ...) : Tính trung bình cộng của các giá trị trong một mảng theo nhiều điều kiện

COUNT (value1, value2, ...) : Đếm số ô trong danh sách

COUNTA (value1, value2, ...) : Đếm số ô có chứa giá trị (không rỗng) trong danh sách

COUNTBLANK (range) : Đếm các ô rỗng trong một vùng

COUNTIF (range, criteria) : Đếm số ô thỏa một điều kiện cho trước bên trong một dãy

COUNTIFS (range1, criteria1, range2, criteria2, ...) : Đếm số ô thỏa nhiều điều kiện cho trước

DEVSQ (number1, number2, ...) : Tính bình phương độ lệch các điểm dữ liệu từ trung bình mẫu của chúng, rồi cộng các bình phương đó lại.

FREQUENCY (data_array, bins_array) : Tính xem có bao nhiêu giá trị thường xuyên xuất hiện bên trong một dãy giá trị, rồi trả về một mảng đứng các số. Luôn sử dụng hàm này ở dạng công thức mảng

GEOMEAN (number1, number2, ...) : Trả về trung bình nhân của một dãy các số dương. Thường dùng để tính mức tăng trưởng trung bình, trong đó lãi kép có các lãi biến đổi được cho trước...

HARMEAN (number1, number2, ...) : Trả về trung bình điều hòa (nghịch đảo của trung bình cộng) của các số

KURT (number1, number2, ...) : Tính độ nhọn của tập số liệu, biểu thị mức nhọn hay mức phẳng tương đối của một phân bố so với phân bố chuẩn

LARGE (array, k) : Trả về giá trị lớn nhất thứ k trong một tập số liệu

MAX (number1, number2, ...) : Trả về giá trị lớn nhất của một tập giá trị

MAXA (number1, number2, ...) : Trả về giá trị lớn nhất của một tập giá trị, bao gồm cả các giá trị logic và text

MEDIAN (number1, number2, ...) : Tính trung bình vị của các số.

MIN (number1, number2, ...) : Trả về giá trị nhỏ nhất của một tập giá trị

MINA (number1, number2, ...) : Trả về giá trị nhỏ nhất của một tập giá trị, bao gồm cả các giá trị logic và text

MODE (number1, number2, ...) : Trả về giá trị xuất hiện nhiều nhất trong một mảng giá trị

PERCENTILE (array, k) : Tìm phân vị thứ k của các giá trị trong một mảng dữ liệu

PERCENTRANK (array, x, significance) : Trả về thứ hạng (vị trí tương đối) của một trị trong một mảng dữ liệu, là số phần trăm của mảng dữ liệu đó

PERMUT (number, number_chosen) : Trả về hoán vị của các đối tượng.

QUARTILE (array, quart) : Tính điểm tứ phân vị của tập dữ liệu. Thường được dùng trong khảo sát dữ liệu để chia các tập hợp thành nhiều nhóm...

RANK (number, ref, order) : Tính thứ hạng của một số trong danh sách các số

SKREW (number1, number2, ...) : Trả về độ lệch của phân phối, mô tả độ không đối xứng của phân phối quanh trị trung bình của nó

SMALL (array, k) : Trả về giá trị nhỏ nhất thứ k trong một tập số

STDEV (number1, number2, ...) : Ước lượng độ lệch chuẩn trên cơ sở mẫu

STDEVA (value1, value2, ...) : Ước lượng độ lệch chuẩn trên cơ sở mẫu, bao gồm cả những giá trị logic

STDEVP (number1, number2, ...) : Tính độ lệch chuẩn theo toàn thể tập hợp

STDEVPA (value1, value2, ...) : Tính độ lệch chuẩn theo toàn thể tập hợp, kể cả chữ và các giá trị logic

VAR (number1, number2, ...) : Trả về phương sai dựa trên mẫu

VARA (value1, value2, ...) : Trả về phương sai dựa trên mẫu, bao gồm cả các trị logic và text

VARP (number1, number2, ...) : Trả về phương sai dựa trên toàn thể tập hợp

VARPA (value1, value2, ...) : Trả về phương sai dựa trên toàn thể tập hợp, bao gồm cả các trị logic và text.

TRIMMEAN (array, percent) : Tính trung bình phần trong của một tập dữ liệu, bằng cách loại tỷ lệ phần trăm của các điểm dữ liệu ở đầu và ở cuối tập dữ liệu.

b. Nhóm hàm về phân phối xác suất

BETADIST (x, alpha, beta, A, B) : Trả về giá trị của hàm tính mật độ phân phối xác suất tích lũy beta.

BETAINV (probability, alpha, beta, A, B) : Trả về nghịch đảo của hàm tính mật độ phân phối xác suất tích lũy beta.

BINOMDIST (number_s, trials, probability_s, cumulative) : Trả về xác suất của những lần thử thành công của phân phối nhị phân.

CHIDIST (x, degrees_freedom) : Trả về xác suất một phía của phân phối chi-squared.

CHIINV (probability, degrees_freedom) : Trả về nghịch đảo của xác suất một phía của phân phối chi-squared.

CHITEST (actual_range, expected_range) : Trả về giá trị của xác suất từ phân phối chi-squared và số bậc tự do tương ứng.

CONFIDENCE (alpha, standard_dev, size) : Tính khoảng tin cậy cho một kỳ vọng lý thuyết

CRITBINOM (trials, probability_s, alpha) : Trả về giá trị nhỏ nhất sao cho phân phối nhị thức tích lũy lớn hơn hay bằng giá trị tiêu chuẩn. Thường dùng để bảo đảm các ứng dụng đạt chất lượng...

EXPONDIST (x, lambda, cumulative) : Tính phân phối mũ. Thường dùng để mô phỏng thời gian giữa các biến cố...

FDIST (x, degrees_freedom1, degrees_freedom2) : Tính phân phối xác suất F.

Thường dùng để tìm xem hai tập số liệu có nhiều mức độ khác nhau hay không...

FINV (probability, degrees_freedom1, degrees_freedom2) : Tính nghịch đảo của phân phối xác suất F. Thường dùng để so sánh độ biến thiên trong hai tập số liệu

FTEST (array1, array2) : Trả về kết quả của một phép thử F. Thường dùng để xác định xem hai mẫu có các phương sai khác nhau hay không...

FISHER (x) : Trả về phép biến đổi Fisher tại x. Thường dùng để kiểm tra giả thuyết dựa trên hệ số tương quan...

FISHERINV (y) : Tính nghịch đảo phép biến đổi Fisher. Thường dùng để phân tích mối tương quan giữa các mảng số liệu...

GAMMADIST (x, alpha, beta, cumulative) : Trả về phân phối tích lũy gamma. Có thể dùng để nghiên cứu có phân bố lệch

GAMMAINV (probability, alpha, beta) : Trả về nghịch đảo của phân phối tích lũy gamma.

GAMMLN (x) : Tính logarit tự nhiên của hàm gamma

HYPGEOMDIST (number1, number2, ...) : Trả về phân phối siêu bội (xác suất của một số lần thành công nào đó...)

LOGINV (probability, mean, standard_dev) : Tính nghịch đảo của hàm phân phối tích lũy lognormal của x (LOGNORMDIST)

LOGNORMDIST (x, mean, standard_dev) : Trả về phân phối tích lũy lognormal của x, trong đó logarit tự nhiên của x thường được phân phối với các tham số mean và standard_dev.

NEGBINOMDIST (number_f, number_s, probability_s) : Trả về phân phối nhị thức âm (trả về xác suất mà sẽ có number_f lần thất bại trước khi có number_s lần thành công, khi xác suất không đổi của một lần thành công là probability_s)

NORMDIST (x, mean, standard_dev, cumulative) : Trả về phân phối chuẩn (normal distribution). Thường được sử dụng trong việc thống kê, gồm cả việc kiểm tra giả thuyết

NORMINV (probability, mean, standard_dev) : Tính nghịch đảo phân phối tích lũy chuẩn

NORMSDIST (z) : Trả về hàm phân phối tích lũy chuẩn tắc (standard normal cumulative distribution function), là phân phối có trị trung bình cộng là zero (0) và độ lệch chuẩn là 1

NORMSINV (probability) : Tính nghịch đảo của hàm phân phối tích lũy chuẩn tắc

POISSON (x, mean, cumulative) : Trả về phân phối poisson. Thường dùng để ước tính số lượng biến cố sẽ xảy ra trong một khoảng thời gian nhất định

PROB (x_range, prob_range, lower_limit, upper_limit) : Tính xác suất của các trị trong dãy nằm giữa hai giới hạn

STANDARDIZE (x, mean, standard_dev) : Trả về trị chuẩn hóa từ phân phối biểu thị bởi mean và standard_dev

TDIST (x, degrees_freedom, tails) : Trả về xác suất của phân phối Student (phân phối t), trong đó x là giá trị tính từ t và được dùng để tính xác suất.

TINV (probability, degrees_freedom) : Trả về giá trị t của phân phối Student.

TTEST (array1, array2, tails, type) : Tính xác suất kết hợp với phép thử Student.

WEIBULL (x, alpha, beta, cumulative) : Trả về phân phối Weibull. Thường sử dụng trong phân tích độ tin cậy, như tính tuổi thọ trung bình của một thiết bị.

ZTEST (array, x, sigma) : Trả về xác suất một phía của phép thử z.

c. Nhóm hàm về tương quan và hồi quy tuyến tính

CORREL (array1, array2) : Tính hệ số tương quan giữa hai mảng để xác định mối quan hệ của hai đặc tính

COVAR (array1, array2) : Tính tích số các độ lệch của mỗi cặp điểm dữ liệu, rồi tính trung bình các tích số đó

FORECAST (x, known_y's, known_x's) : Tính toán hay dự đoán một giá trị tương lai bằng cách sử dụng các giá trị hiện có, bằng phương pháp hồi quy tuyến tính

GROWTH (known_y's, known_x's, new_x's, const) : Tính toán sự tăng trưởng dự kiến theo hàm mũ, bằng cách sử dụng các dữ kiện hiện có.

INTERCEPT (known_y's, known_x's) : Tìm điểm giao nhau của một đường thẳng với trục y bằng cách sử dụng các trị x và y cho trước

LINEST (known_y's, known_x's, const, stats) : Tính thống kê cho một đường bằng cách dùng phương pháp bình phương tối thiểu (least squares) để tính đường thẳng thích hợp nhất với dữ liệu, rồi trả về mảng mô tả đường thẳng đó. Luôn dùng hàm này ở dạng công thức mảng.

LOGEST (known_y's, known_x's, const, stats) : Dùng trong phân tích hồi quy. Hàm sẽ tính đường cong hàm mũ phù hợp với dữ liệu được cung cấp, rồi trả về mảng giá trị mô tả đường cong đó. Luôn dùng hàm này ở dạng công thức mảng

PEARSON (array1, array2) : Tính hệ số tương quan momen tích pearson (r), một chỉ mục không thứ nguyên, trong khoảng từ -1 đến 1, phản ánh sự mở rộng quan hệ tuyến tính giữa hai tập số liệu

RSQ (known_y's, known_x's) : Tính bình phương hệ số tương quan momen tích Pearson (r), thông qua các điểm dữ liệu trong known_y's và known_x's

SLOPE (known_y's, known_x's) : Tính hệ số góc của đường hồi quy tuyến tính thông qua các điểm dữ liệu

STEYX (known_y's, known_x's) : Trả về sai số chuẩn của trị dự đoán y đối với mỗi trị x trong hồi quy.

TREND (known_y's, known_x's, new_x's, const) : Trả về các trị theo xu thế tuyến tính

d. Các hàm tài chính - financial functions

ACCRINT (issue, first_interest, settlement, rate, par, frequency, basis, calc_method) : Tính lãi tích lũy cho một chứng khoán trả lãi theo định kỳ

ACCRINTM (issue, settlement, rate, par, basis) : Tính lãi tích lũy đối với chứng khoán trả lãi theo kỳ hạn

AMORDEGRC (cost, date_purchased, first_period, salvage, period, rate, basis) : Tính khấu hao trong mỗi tài khóa kế toán tùy theo thời hạn sử dụng của tài sản (sử dụng trong các hệ thống kế toán theo kiểu Pháp)

AMORLINC (cost, date_purchased, first_period, salvage, period, rate, basis) : Tính khấu hao trong mỗi tài khóa kế toán (sử dụng trong các hệ thống kế toán theo kiểu Pháp)

COUPDAYBS (settlement, maturity, frequency, basis) : Tính số ngày kể từ đầu kỳ lãi tới ngày kết toán

COUPDAYS (settlement, maturity, frequency, basis) : Tính số ngày trong kỳ lãi bao gồm cả ngày kết toán

COUPDAYSCN (settlement, maturity, frequency, basis) : Tính số ngày từ ngày kết toán tới ngày tính lãi kế tiếp

COUPNCD (settlement, maturity, frequency, basis) : Trả về một con số thể hiện ngày tính lãi kế tiếp kể từ sau ngày kết toán

COUPNUM (settlement, maturity, frequency, basis) : Tính số lần lãi suất phải trả trong khoảng từ ngày kết toán đến ngày đáo hạn

COUPPCD (settlement, maturity, frequency, basis) : Trả về một con số thể hiện ngày thanh toán lãi lần trước, trước ngày kết toán

CUMIPMT (rate, nper, pv, start_period, end_period, type) : Tính lợi tức tích lũy phải trả đối với khoản vay trong khoảng thời gian giữa start_period và end_period

CUMPRINC (rate, nper, pv, start_period, end_period, type) : Trả về tiền vốn tích lũy phải trả đối với khoản vay trong khoảng thời gian giữa start_period và end_period

DB (cost, salvage, life, period, month) : Tính khấu hao cho một tài sản sử dụng phương pháp số dư giảm dần theo một mức cố định (fixed-declining balance method) trong một khoảng thời gian xác định.

DDB (cost, salvage, life, period, factor) : Tính khấu hao cho một tài sản sử dụng phương pháp số dư giảm dần kép (double-declining balance method), hay giảm dần theo một tỷ lệ nào đó, trong một khoảng thời gian xác định.

DISC (settlement, maturity, pr, redemption, basis) : Tính tỷ lệ chiết khấu của một chứng khoán

DOLLARDE (fractional_dollar, fraction) : Chuyển đổi giá dollar ở dạng phân số sang giá dollar ở dạng thập phân

DOLLARFR (decimal_dollar, fraction) : Chuyển đổi giá dollar ở dạng thập phân số sang giá dollar ở dạng phân số

DURATION (settlement, maturity, coupon, yld, frequency, basis) : Tính thời hạn hiệu lực Macauley dựa trên đồng mệnh giá \$100 (thời hạn hiệu lực là trung bình trọng giá trị hiện tại của dòng luân chuyển tiền mặt và được dùng làm thước đo về sự phản hồi làm thay đổi lợi nhuận của giá trị trái phiếu)

EFFECT (nominal_rate, npery) : Tính lãi suất thực tế hàng năm, biết trước lãi suất danh nghĩa hàng năm và tổng số kỳ thanh toán lãi kép mỗi năm

FV (rate, nper, pmt, pv, type) : Tính giá trị kỳ hạn của sự đầu tư dựa trên việc chi trả cố định theo kỳ và lãi suất cố định

FVSCCHEDULE (principal, schedule) : Tính giá trị kỳ hạn của một vốn ban đầu sau khi áp dụng một chuỗi các lãi suất kép (tính giá trị kỳ hạn cho một đầu tư có lãi suất thay đổi)

INTRATE (settlement, maturity, investment, redemption, basis) : Tính lãi suất cho một chứng khoán đầu tư toàn bộ

IPMT (rate, per, nper, pv, fv, type) : Trả về khoản thanh toán lãi cho một đầu tư dựa trên việc chi trả cố định theo kỳ và dựa trên lãi suất không đổi

IRR (values, guess) : Tính lợi suất nội hàm cho một chuỗi các lưu động tiền mặt được thể hiện bởi các trị số

ISPMT (rate, per, nper, pv) : Tính số tiền lãi đã trả tại một kỳ nào đó đối với một khoản vay có lãi suất không đổi, sau khi đã trừ số tiền gốc phải trả cho kỳ đó.

MDURATION (settlement, maturity, coupon, yld, frequency, basis) : Tính thời hạn Macauley sửa đổi cho chứng khoán dựa trên đồng mệnh giá \$100

MIRR (values, finance_rate, reinvest_rate) : Tính tỷ suất doanh lợi nội tại trong một chuỗi luân chuyển tiền mặt theo chu kỳ

NOMINAL (effect_rate, npery) : Tính lãi suất danh nghĩa hằng năm, biết trước lãi suất thực tế và các kỳ tính lãi kép mỗi năm

NPER (rate, pmt, pv, fv, type) : Tính số kỳ hạn để trả khoản vay trong đầu tư dựa trên từng chu kỳ, số tiền trả và tỷ suất lợi tức cố định

NPV (rate, value1, value2, ...) : Tính hiện giá ròng của một khoản đầu tư bằng cách sử dụng tỷ lệ chiết khấu với các chi khoản trả kỳ hạn (trị âm) và thu nhập (trị dương)

ODDFPRICE (settlement, maturity, issue, first_coupon, rate, yld, redemption, frequency, basis) : Tính giá trị trên mỗi đồng mệnh giá \$100 của chứng khoán có kỳ đầu tiên lẻ (ngắn hạn hay dài hạn)

ODDFYIELD (settlement, maturity, issue, first_coupon, rate, pr, redemption, frequency, basis) : Trả về lợi nhuận của một chứng khoán có kỳ tính lãi đầu tiên là lẻ (ngắn hạn hay dài hạn)

ODDLPRICE (settlement, maturity, last_interest, rate, yld, redemption, frequency, basis) : Tính giá trị trên mỗi đồng mệnh giá \$100 của chứng khoán có kỳ tính lãi phiếu cuối cùng là lẻ (ngắn hạn hay dài hạn)

ODDLYIELD (settlement, maturity, last_interest, rate, pr, redemption, frequency, basis) : Tính lợi nhuận của chứng khoán có kỳ cuối cùng là lẻ (ngắn hạn hay dài hạn)

PMT (rate, nper, pv, fv, type) : Tính tiền phải trả đối với khoản vay có lãi suất không đổi và chi trả đều đặn

PPMT (rate, per, nper, pv, fv, type) : Tính khoản vốn thanh toán trong một kỳ hạn đã cho đối với một khoản đầu tư, trong đó việc chi trả được thực hiện đều đặn theo định kỳ với một lãi suất không đổi

PRICE (settlement, maturity, rate, yld, redemption, frequency, basis) : Tính giá trị chứng khoán trên đồng mệnh giá \$100, thanh toán lợi tức theo chu kỳ

PRICEDISC (settlement, maturity, discount, redemption, basis) : Tính giá trị trên đồng mệnh giá \$100 của một chứng khoán đã chiết khấu

PRICEMAT (settlement, maturity, issue, rate, yld, basis) : Tính giá trị trên đồng mệnh giá \$100 của một chứng khoán phải thanh toán lãi vào ngày đáo hạn

PV (rate, nper, pmt, fv, type) : Tính giá trị hiện tại của một khoản đầu tư

RATE (nper, pmt, pv, fv, type, guess) : Tính lãi suất mỗi kỳ trong một niên kim

REVEICED (settlement, maturity, investment, discount, basis) : Tính số tiền nhận được vào kỳ hạn thanh toán cho một chứng khoán đầu tư toàn bộ

SLN (cost, salvage, life) : Tính chi phí khấu hao (theo phương pháp đường thẳng) của một tài sản trong một kỳ

SYD (cost, salvage, life, per) : Tính khấu hao theo giá trị còn lại của tài sản trong định kỳ xác định

TBILLEQ (settlement, maturity, discount) : Tính lợi nhuận tương ứng với trái phiếu cho trái phiếu kho bạc

TBILLPRICE (settlement, maturity, discount) : Tính giá trị đồng mệnh giá \$100 cho trái phiếu kho bạc

TBILLYIELD (settlement, maturity, pr) : Tính lợi nhuận cho trái phiếu kho bạc

VDB (cost, salvage, life, start_period, end_period, factor, no_switch) : Tính khấu hao tài sản sử dụng trong nhiều kỳ

XIRR (values, dates, guess) : Tính lợi suất nội hàm cho một loạt lưu động tiền mặt không định kỳ

XNPV (rate, values, dates) : Tính tỷ giá ròng cho một dãy lưu động tiền mặt không định kỳ

YIELD (settlement, maturity, rate, pr, redemption, frequency, basis) : Tính lợi nhuận đối với chứng khoán trả lãi theo định kỳ

YIELDDISC (settlement, maturity, pr, redemption, basis) : Tính lợi nhuận hằng năm cho chứng khoán đã chiết khấu

YIELDMAT (settlement, maturity, issue, rate, pr, basis) : Tính lợi nhuận hằng năm của chứng khoán trả lãi vào ngày đáo hạn

e. Danh mục các Hàm Quản lý Cơ sở dữ liệu và Danh sách

DAVERAGE (database, field, criteria) : Tính trung bình các giá trị trong một cột của danh sách hay các trị của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DCOUNT (database, field, criteria) : Đếm các ô chứa số liệu trong một cột của danh sách hay các trị của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DCOUNTA (database, field, criteria) : Đếm các ô "không rỗng" trong một cột của danh sách hay các trị của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DGET (database, field, criteria) : Trích một giá trị từ một cột của một danh sách hay cơ sở dữ liệu, khớp với điều kiện được chỉ định.

DMAX (database, field, criteria) : Trả về trị lớn nhất trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DMIN (database, field, criteria) : Trả về trị nhỏ nhất trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DPRODUCT (database, field, criteria) : Nhân các giá trị trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DSTDEV (database, field, criteria) : Ước lượng độ lệch chuẩn của một tập hợp theo mẫu, bằng cách sử dụng các số liệu trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DSTDEVP (database, field, criteria) : Tính độ lệch chuẩn của một tập hợp theo toàn thể các tập hợp, bằng cách sử dụng các số liệu trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DSUM (database, field, criteria) : Cộng các số trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DVAR (database, field, criteria) : Ước lượng sự biến thiên của một tập hợp dựa trên một mẫu, bằng cách sử dụng các số liệu trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

DVARP (database, field, criteria) : Tính toán sự biến thiên của một tập hợp dựa trên toàn thể tập hợp, bằng cách sử dụng các số liệu trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

GETPIVOTDATA (data_field, pivot_table, field1, item1, field2, item2,...) : Trả về dữ liệu được lưu giữ trong báo cáo PivotTable. Có thể dùng GETPIVOTDATA để lấy dữ liệu tổng kết từ một báo cáo PivotTable, với điều kiện là phải thấy được dữ liệu tổng kết từ trong báo cáo đó.

Nói chung về Các Hàm Quản lý Cơ sở dữ liệu và Danh sách

Microsoft Excel cung cấp nhiều hàm dùng cho việc phân tích dữ liệu trong danh sách hay cơ sở dữ liệu. Những hàm này bắt đầu bằng chữ D.

Mỗi hàm đều sử dụng 3 đối số là database, field và criteria; những đối số này là các tham chiếu đến các mảng trong bảng tính.

* database : Là một dãy các ô tạo nên danh sách hay cơ sở dữ liệu. Một cơ sở dữ liệu là một danh sách dữ liệu gồm những mẫu tin, và có cột dữ liệu là các trường (field). Dòng đầu tiên trong danh sách luôn chứa tên trường.

* field : Cho biết cột nào được sử dụng trong hàm. field có thể được cho ở dạng text với tên cột được để trong cặp dấu ngoặc kép (như "Age", "Yield"...) hay là số đại diện cho vị trí của cột (1, 2, ...)

* criteria : Là một dãy các ô chứa điều kiện. Có thể dùng bất cứ dãy nào cho phần điều kiện này, miễn là dãy đó có ít nhất một tên cột và một ô bên dưới tên cột để làm điều kiện cho hàm (xem thêm bài Một số ví dụ về cách dùng Criteria để nhập điều kiện ở sau đây).

HÀM NGÀY THÁNG VÀ THỜI GIAN

Một số lưu ý khi sử dụng ngày tháng và thời gian trong Excel:

Excel hỗ trợ tính toán ngày tháng cho Windows và Macintosh. Windows dùng hệ ngày bắt đầu từ 1900. Macintosh dùng hệ ngày bắt đầu từ 1904. Tài liệu này được diễn giải theo hệ ngày 1900 dành cho Windows.

Hệ thống ngày giờ Excel phụ thuộc vào thiết lập trong Regional Options của Control Panel. Mặc định là hệ thống của Mỹ "Tháng/Ngày/Năm" (M/d/yyyy). Bạn có thể sửa lại thành hệ thống ngày của VN "Ngày/Tháng/Năm" (dd/MM/yyyy).

Khi bạn nhập một giá trị ngày tháng không hợp lệ nó sẽ trở thành một chuỗi văn bản. Công thức tham chiếu tới giá trị đó sẽ trả về lỗi.

=NOW() Cho hiện ngày giờ của hệ thống

=TODAY() Cho ngày của hệ thống

=DAY(D) Cho giá trị ngày của D (Trả về thứ tự của ngày trong tháng từ một giá trị kiểu ngày tháng)

=MONTH(D) Cho giá trị tháng của D

=YEAR(D) Cho giá trị năm của D

=DAYS360(BTNT1, BTNT2) Tính số ngày giữa 2 mốc ngày tháng dựa trên cơ sở một năm có 360 ngày.

=EDATE Trả về mốc thời gian xảy ra trước hoặc sau mốc chỉ định

=EOMONTH Trả về ngày cuối cùng của tháng xảy ra trước hoặc sau mốc chỉ định

Hàm HOUR()

Cho biết số chỉ giờ trong một giá trị thời gian

Cú pháp: = HOUR(serial_number)

serial_number: Biểu thức thời gian hoặc là một con số chỉ giá trị thời gian

Ví dụ: HOUR(0.5) = 12 (giờ)

Hàm MINUTE()

Cho biết số chỉ phút trong một giá trị thời gian

Cú pháp: = MINUTE(serial_number)

serial_number: Biểu thức thời gian hoặc là một con số chỉ giá trị thời gian

Ví dụ: Bây giờ là 10:20 PM, MINUTE(NOW()) = 20 (phút)

=MONTH Trả về số tháng của một giá trị kiểu ngày tháng.

=NETWORKDAYS Trả về số ngày làm việc trong mốc thời gian đưa ra sau khi trừ đi ngày nghỉ và ngày lễ.

=NOW Trả về ngày giờ hiện tại trong hệ thống của bạn.

Hàm SECOND()

Cho biết số chỉ giây trong một giá trị thời gian

Cú pháp: = SECOND(serial_number)

serial_number: Biểu thức thời gian hoặc là một con số chỉ giá trị thời gian

Ví dụ: SECOND("2:45:30 PM") = 30 (giây)

Hàm TIME()

Trả về một giá trị thời gian nào đó

Cú pháp: = TIME(hour, minute, second)

hour: Số chỉ giờ, là một con số từ 0 đến 23. Nếu lớn hơn 23, Excel sẽ tự trừ đi một bội số của 24.

minute: Số chỉ phút, là một con số từ 0 đến 59. Nếu lớn hơn 59, Excel sẽ tính lại và tăng số giờ lên tương ứng.

second: Số chỉ giây, là một con số từ 0 đến 59. Nếu lớn hơn 59, Excel sẽ tính lại và tăng số phút, số giờ lên tương ứng.

Ví dụ:

TIME(14, 45, 30) = 2:45:30 PM

TIME(14, 65, 30) = 3:05:30 PM

TIME(25, 85, 75) = 2:26:15 AM

* Cũng như DATE(), hàm TIME() rất hữu dụng khi hour, minute, second là những công thức mà không phải là một con số, nó sẽ giúp chúng ta tính toán chính xác hơn

Hàm TIMEVALUE()

Chuyển đổi một chuỗi văn bản có dạng thời gian thành một giá trị thời gian để có thể tính toán được

Cú pháp: = TIMEVALUE(time_text)

time_text: Chuỗi văn bản cần chuyển đổi

Ví dụ:

TIMEVALUE("26:15") = 0.09375 (= 2:15:00 AM)

=WEEKDAY Trả về số thứ tự của ngày trong tuần từ giá trị ngày tháng.

=WEEKNUM Trả về số thứ tự của tuần trong năm từ giá trị ngày tháng.

=WORKDAY Trả về ngày làm việc xảy ra trước hoặc sau mốc thời gian đưa ra.

Trong công việc hằng ngày, chắc hẳn chúng ta hay nghĩ đến chuyện việc làm này của mình mất hết mấy phần trăm của một năm, ví dụ, một ngày ngủ hết 6 tiếng, là 1/4 ngày, vậy một năm chúng ta ngủ hết 25% (hìc) thời gian...

Hoặc một nhân viên của công ty xin nghỉ việc vào tháng 5, lương tính theo năm, vậy công ty phải trả cho người đó bao nhiêu phần trăm lương khi cho nghỉ việc?

Excel có một hàm để tính tỷ lệ của một khoảng thời gian trong một năm, và cho phép tính theo nhiều kiểu (năm 365 ngày, hay năm 360 ngày, tính theo kiểu Mỹ hay theo kiểu châu Âu...):

Hàm YEARFRAC()

(Dịch từ chữ Year: năm, và Frac = Fraction: tỷ lệ)

Cú pháp: = YEARFRAC(start_date, end_date [, basis])

start_date, end_date: Ngày tháng đại diện cho ngày bắt đầu và ngày kết thúc của khoảng thời gian cần tính toán. Nên nhập bằng hàm DATE(), hoặc dùng một kết quả trả về của một công thức khác, vì có thể sẽ xảy ra lỗi nếu bạn nhập trực tiếp ngày tháng dưới dạng text.

basis: Một con số, quy định kiểu tính:

* 0 : (hoặc không nhập) Tính toán theo kiểu Bắc Mỹ, một năm có 360 ngày chia cho 12 tháng, một tháng có 30 ngày.

* 1 : Tính toán theo số ngày thực tế của năm và số ngày thực tế của từng tháng

* 2 : Tính toán theo một năm có 360 ngày, nhưng số ngày là số ngày thực tế của từng tháng

* 3 : Tính toán theo một năm có 365 ngày, và số ngày là số ngày thực tế của từng tháng

* 4 : Tính toán theo kiểu Châu Âu, mỗi tháng có 30 ngày (nếu start_date hoặc end_date mà rơi vào ngày 31 của một tháng thì chúng sẽ được đổi thành ngày 30 của tháng đó)

Ví dụ: Tính tỷ lệ giữa ngày 15/3/2010 và ngày 30/7/2010 so với 1 năm:

`YEARFRAC("15/3/2010", "30/7/2010") = 37%`

Tính số ngày chênh lệch theo kiểu một năm có 360 ngày

Hiện nay, vẫn còn một số hệ thống kế toán dùng kiểu tính thời gian là một tháng coi như có 30 ngày và một năm coi như có 360 ngày!

Gặp trường hợp này, việc tính toán thời gian sẽ không đơn giản, vì thực tế thì số ngày trong mỗi tháng đâu có giống nhau.

Có lẽ vì nghĩ đến chuyện đó, nên Excel có một hàm dành riêng cho các hệ thống kế toán dựa trên cơ sở một năm có 360 ngày, đó là hàm DAYS360.

Hàm DAYS360()

Cú pháp: `= DAYS360(start_date, end_date [, method])`

start_date, end_date: Ngày tháng đại diện cho ngày bắt đầu và ngày kết thúc của khoảng thời gian cần tính toán. Nên nhập bằng hàm DATE(), hoặc dùng một kết quả trả về của một công thức khác, vì có thể sẽ xảy ra lỗi nếu bạn nhập trực tiếp ngày tháng dưới dạng text.

method: Một giá trị logic (TRUE, FALSE) để chỉ cách tính toán, theo kiểu châu Âu hay theo kiểu Mỹ.

* FALSE: (hoặc không nhập) Tính toán theo kiểu Mỹ: Nếu start_date là ngày 31 của tháng, thì nó được đổi thành ngày 30 của tháng đó. Nếu end_date là ngày 31 của tháng và start_date nhỏ hơn 30, thì end_date được đổi thành ngày 1 của tháng kế tiếp.

* TRUE: Tính toán theo kiểu châu Âu: Hễ start_date hoặc end_date mà rơi vào ngày 31 của một tháng thì chúng sẽ được đổi thành ngày 30 của tháng đó.

Ví dụ: So sánh số ngày chênh lệch giữa 01/01/2008 và 31/5/2008 theo kiểu một năm có 360 ngày và theo kiểu thường (dùng hàm DATEDIF)

`DAYS360("01/01/2008", "31/5/2008") = 150`

`DAYS360("01/01/2008", "31/5/2008", TRUE) = 149`

`DATEDIF("01/01/2008", "31/5/2008", "d") = 151`

Tính số ngày làm việc giữa hai khoảng thời gian

Bình thường, nếu lấy ngày tháng trừ ngày tháng, kết quả sẽ bao gồm luôn những ngày lễ, ngày nghỉ, v.v... Còn nếu tính số ngày làm việc trong một khoảng thời gian, thì phải trừ bớt đi những ngày không làm việc.

Trong Excel có một hàm chuyên để tính toán những ngày làm việc giữa hai khoảng thời gian mà không bao gồm các ngày thứ Bảy, Chủ Nhật và những ngày nghỉ khác được chỉ định: Hàm NETWORKDAYS (đúng nguyên nghĩa của nó: net workdays).

Dĩ nhiên hàm này chỉ thích hợp với những cơ quan làm việc 5 ngày một tuần, chứ như chúng ta, làm tuốt, có khi là 365 ngày một năm (hic) thì hàm này vô tác dụng!

Hàm NETWORKDAYS()

Cú pháp: = NETWORKDAYS(start_date, end_date [, holidays])

start_date, end_date: Ngày tháng đại diện cho ngày bắt đầu và ngày kết thúc công việc. Nên nhập bằng hàm DATE(), hoặc dùng một kết quả trả về của một công thức khác, vì có thể sẽ xảy ra lỗi nếu bạn nhập trực tiếp ngày tháng dưới dạng text.

holidays: Danh sách những ngày nghỉ ngoài những ngày thứ Bảy và Chủ Nhật.

Danh sách này có thể là một vùng đã được đặt tên. Nếu nhập trực tiếp thì phải bỏ trong cặp dấu móc {}.

Ví dụ: Công thức tính số ngày làm việc giữa ngày 1/12/2010 và ngày 10/1/2011, trong đó có nghỉ ngày Noel (25/12) và ngày Tết Tây (1/1):

```
= NETWORKDAYS("01/12/2010", "10/01/2011", {"12/25/2007", "1/1/2011"})
```

Hàm DATEDIF()

Có lẽ cách dễ nhất khi muốn tính toán ngày tháng năm là dùng hàm DATEDIF().

Cú pháp: = DATEDIF(start_day, end_day, unit)

start_day: Ngày đầu

end_day: Ngày cuối (phải lớn hơn ngày đầu)

unit: Chọn loại kết quả trả về (khi dùng trong hàm phải gõ trong dấu ngoặc kép)

y : số năm chênh lệch giữa ngày đầu và ngày cuối

m : số tháng chênh lệch giữa ngày đầu và ngày cuối

d : số ngày chênh lệch giữa ngày đầu và ngày cuối

md : số ngày chênh lệch giữa ngày đầu và tháng ngày cuối, mà không phụ thuộc vào số năm và số tháng

ym : số tháng chênh lệch giữa ngày đầu và ngày cuối, mà không phụ thuộc vào số năm và số ngày

yd : số ngày chênh lệch giữa ngày đầu và ngày cuối, mà không phụ thuộc vào số năm

Ví dụ:

DATEDIF("01/01/2000", "31/12/2100", "y") = 100 (năm)

DATEDIF("01/01/2000", "31/12/2100", "m") = 1211 (tháng)

DATEDIF("01/01/2000", "31/12/2100", "d") = 36889 (ngày)

DATEDIF("01/01/2000", "31/12/2100", "md") = 30 (= ngày 31 - ngày 1)

DATEDIF("01/01/2000", "31/12/2100", "ym") = 11 (= tháng 12 - tháng 1)

DATEDIF("01/01/2000", "31/12/2100", "yd") = 365 (= ngày 31/12 - ngày 1/1)

*Tính tuổi (2):

Ở bài trước, tôi đã đưa ra một cái công thức để tính tuổi dài thòng như vậy:

= YEAR(NOW() - YEAR(Birthdate) - (DATE(YEAR(NOW()),
MONTH(Birthdate), DAY(Birthdate)) > TODAY()))

Đó là khi chưa biết đến hàm DATEDIF().

Bây giờ, với DATEDIF(), công thức trên chỉ ngắn như vậy thôi, mà ra kết quả vẫn chính xác:

= DATEDIF(Birthdate, TODAY(), "y")

Ví dụ, hôm nay là ngày 09/01/2007:

* Với ngày sinh là 05/01/1969 (đã tổ chức sinh nhật rồi), $\text{DATEDIF}("05/01/1969", \text{TODAY}(), "y") = 39$

* Nhưng với ngày sinh là 11/1/1969 (chưa tổ chức sinh nhật), $\text{DATEDIF}("11/01/1969", \text{TODAY}(), "y") = 38$

*Tìm một giờ, phút, giây nào đó tính từ lúc này

Như tôi đã nói ở trên, hàm $\text{TIME}()$ sẽ tự động điều chỉnh kết quả của một giá trị thời gian khi những thông số trong hàm không hợp lý (giờ > 24, phút và giây > 60). Và do đó, khi cần tính toán hoặc tìm một giá trị thời gian nào đó kể từ lúc này (hoặc bất kỳ lúc nào), người ta thường sử dụng hàm $\text{TIME}()$.

Ví dụ, công thức sau đây sẽ cho ra kết quả là thời gian vào 12 tiếng nữa kể từ lúc này:

$= \text{TIME}(\text{HOUR}(\text{NOW}()) + 12, \text{MINUTE}(\text{NOW}()), \text{SECOND}(\text{NOW}()))$

Không giống như hàm $\text{DATE}()$, bạn không thể đơn giản cộng thêm giờ, phút, hay giây trong hàm $\text{TIME}()$. Ví dụ công thức sau đây chỉ làm mỗi chuyện là tăng thêm 1 ngày vào ngày tháng năm và thời gian hiện tại:

$= \text{NOW}() + 1$

Nếu bạn muốn cộng thêm giờ, phút, hay giây vào một giá trị thời gian, bạn phải tính thời gian cộng thêm đó theo một tỷ lệ của một ngày. Ví dụ, bởi vì một ngày thì có 24 giờ, nên một giờ được tính như là $1/24$. Cũng vậy, bởi vì một giờ thì có 60 phút, nên một phút sẽ được tính như là $1/24/60$ (của một ngày). Và cuối cùng, bởi vì có 60 giây trong một phút, nên 1 giây trong một ngày sẽ được tính bằng $1/24/60/60$.

*Tính tổng thời gian

Khi tính tổng thời gian, bạn nên phân biệt hai trường hợp sau đây:

* Cộng thêm giờ, phút, giây: Ví dụ, bây giờ là 8 giờ, cộng thêm 2 tiếng nữa, là 10 giờ... Hoặc bây giờ là 23 giờ, cộng thêm 3 tiếng nữa là 2 giờ sáng (chớ không phải 26 giờ)... Nếu cộng kiểu này thì bạn cứ cộng bình thường, dùng hàm TIME() và nếu cần thì theo bảng hướng dẫn ở trên.

* Cộng tổng thời gian làm việc: Mỗi ngày tôi làm việc 18 tiếng, vậy hai ngày tôi làm mấy tiếng? là 36 tiếng. Nhưng nếu bạn dùng format bình thường dạng thời gian thì Excel nó sẽ tự quy ra $(36-24) = 12:00$... Để được kết quả là 36:00, bạn phải định dạng thời gian cho ô theo kiểu:

[h]:mm:ss (giờ nằm trong một cặp dấu ngoặc vuông)

Lấy lại định dạng ngày tháng

Đôi khi, bạn nhận được một cái database mà không hiểu vì một lý do gì đó, cell chứa ngày tháng năm nó như sau: 20070823,

nghĩa là nó được định dạng theo kiểu YYYYMMDD.

Trường hợp này, không thể dùng Format Cell của Excel để định dạng lại, mà bạn phải dùng đến hàm DATE(year, month, day)

và các hàm xử lý text, ví dụ hàm LEFT(), MID() và RIGHT() để lấy các trị số ngày, tháng, năm cho hàm DATE():

Tôi giả sử con số 20070823 này đang nằm ở cell A1, thì công thức LEFT(A1, 4) sẽ cho ta trị số của năm, MID(A1, 3, 2) sẽ cho trị số của tháng

và RIGHT(A1, 2) sẽ cho trị số của ngày.

Giả sử trong Control Panel của bạn định dạng ngày tháng năm theo kiểu dd/mm/yyyy, và cell nhận kết quả của bạn đã được định dạng theo kiểu ngày tháng năm,

bạn dùng hàm DATE(year, month, day) với 3 tham số là 3 công thức vừa làm ở trên, ta sẽ có:

=DATE(LEFT(A1, 4), MID(A1, 5, 2), RIGHT(A1, 2))→ 23/8/2007

HÀM TÌM KIẾM VÀ THAM CHIẾU

Bao gồm các hàm tìm kiếm và tham chiếu rất hữu ích khi bạn làm việc với CSDL lớn trong EXCEL như kế toán, tính lương, thuế...

=ADDRESS(row_num, column_num [, abs_num] [, a1] [, sheet_text]) Dùng để tạo địa chỉ dạng chuỗi ký tự, theo chỉ số dòng và chỉ số cột được cung cấp.

row_num: Số thứ tự dòng của địa chỉ

column_num: Số thứ tự cột của địa chỉ

abs_num: Kiểu địa chỉ (tuyệt đối hay tương đối) - mặc định là tuyệt đối

a1: Dạng địa chỉ (=TRUE: dạng A1; =FALSE: dạng R1C1) - mặc định là TRUE

sheet_text: Tên của sheet chứa địa chỉ - mặc định là không có

=AREAS(reference) Trả về số vùng tham chiếu trong một tham chiếu. Mỗi vùng tham chiếu có thể là một ô rời rạc hoặc là một dãy ô liên tục trong bảng tính.

reference: Là một hoặc nhiều vùng tham chiếu mà bạn cần đếm. Nếu muốn đưa nhiều vùng rời rạc nhau vào công thức thì bạn phân cách chúng bằng dấu phẩy, nếu không ngăn bằng dấu phẩy, Excel sẽ không đếm (xem ví dụ 3 dưới đây)

=CHOOSE Trả về giá trị trong mảng giá trị tại vị trí được chỉ định.

=COLUMN(reference) Trả về số thứ tự cột của ô đầu tiên ở góc trên bên trái của vùng tham chiếu.

reference: Là ô hoặc một vùng nhiều ô. Nếu reference bỏ trống thì COLUMN() trả về số thứ tự cột của ô chứa công thức.

=COLUMNS(array) Trả về số cột của vùng tham chiếu.

array: Là ô hoặc một vùng nhiều ô, mảng tham chiếu

=HYPERLINK(link_location, friendly_name) Tạo một siêu liên kết, một liên kết để mở một tài liệu...

link_location: Đường dẫn đến tài liệu cần mở, nhập ở dạng chuỗi ký tự.

Link_location có thể chỉ đến một ô đặc biệt, tên của một dãy ô trong một trang bảng tính hoặc một bảng tính, hoặc chỉ đến một đánh dấu (bookmark) trong Microsoft Excel. Link_location cũng có thể chỉ đến một tập tin lưu trên ổ cứng, hoặc một đường dẫn truy xuất nội bộ trên một máy chủ hoặc một đường dẫn tài nguyên URL trên mạng intranet, internet.

- Link_location có thể là một chuỗi ký tự đặt trong dấu nháy kép, hoặc một ô nào đó chứa đường dẫn dưới dạng chuỗi ký tự.

- Nếu link_location liên kết đến một tài nguyên không tồn tại, sẽ xuất hiện lỗi khi bạn kích vào ô chứa hàm HYPERLINK() này.

friendly_name: Là nội dung hiển thị trong ô chứa hàm HYPERLINK(), có thể là một số, hoặc chuỗi ký tự. Nội dung này sẽ hiển thị bằng màu xanh và có gạch chân, nếu không có friendly_name thì chính link_location sẽ hiển thị.

- Friendly_name có thể là một giá trị, một chuỗi ký tự, một tên mảng, hoặc một ô liên kết đến một giá trị hoặc một chuỗi văn bản.

- Nếu firendly_name liên kết đến một giá trị bị lỗi, thì chính tên cái lỗi đó sẽ được hiển thị để thay thế cho nội dung bạn cần.

Ví dụ:

=HYPERLINK("http://example.microsoft.com/report/budget report.xls", "Click for report")

sẽ hiển thị: Click for report, và khi bạn click vào ô này, Excel sẽ mở bảng tính budget report.xls của trang web http://example.microsoft.com/report

=HYPERLINK("F:\GPE\book2.xls",F10)

sẽ hiển thị nội dung của ô F10 và khi bạn click vào ô này, Excel sẽ mở bảng tính book2.xls ở thư mục GPE của ổ F

=HYPERLINK("F:\GPE\book2.xls",F10)

sẽ hiển thị nội dung của ô F10 và khi bạn click vào ô này, Excel sẽ mở bảng tính book2.xls ở thư mục GPE của ổ F

=MATCH Trả về vị trí của một giá trị trong bảng dữ liệu

=INDEX Trả về một giá trị trong bảng dữ liệu tương ứng với chỉ mục của nó.

Tìm kiếm trong nhiều cột

Đôi khi, các dữ liệu dùng để tìm kiếm không nằm ở một cột mà trong nhiều cột.

Ví dụ, bạn có một danh sách đã được tách sẵn họ và tên riêng ra hai cột.

Người ta yêu cầu dựa vào họ và tên để xác định chức vụ.

Chúng ta có thể dùng một cột phụ để nối họ và tên lại rồi tìm theo cột phụ... Nhưng có lẽ không cần, vì tôi sẽ dùng hàm INDEX() và MATCH() với công thức mảng.

Hàm MATCH() có một cú pháp ít người biết đến, đó là:

=MATCH(value1 & value2, array1 & array2, match_type)

value1 & value2 là các dữ liệu để tìm (ví dụ họ và tên)

array1 & array2 là các cột (hoặc hàng) chứa các dữ liệu dùng để tìm kiếm đó

Và đây là cú pháp dùng INDEX() ghép với MATCH() trong công thức mảng:

{=INDEX(reference, MATCH(value1 & value2, array1 & array2, match_type))}

=INDIRECT(ref_text [, a1])

ref_text là tham chiếu tới một ô (có thể là dạng A1 hoặc dạng R1C1), là tên định nghĩa của một tham chiếu, hoặc là một tham chiếu dạng chuỗi.

-Nếu ref_text không hợp lệ, INDIRECT() sẽ báo lỗi #REF!

-Nếu ref_text chứa tham chiếu đến một bảng tính khác thì bảng tính này phải đang mở, nếu không, INDIRECT() cũng báo lỗi #REF!

a1 là giá trị logic xác định dạng tham chiếu bên trong ref_text.

a1 = TRUE (hoặc là 1, hoặc không nhập) là kiểu tham chiếu A1

a1 = FALSE (hoặc là 2) là kiểu tham chiếu R1C1

Trả về giá trị của một tham chiếu từ chuỗi ký tự. Tham chiếu được trả về ngay tức thời để hiển thị nội dung của chúng. Cũng có thể dùng hàm INDIRECT khi muốn thay đổi tham chiếu tới một ô bên trong một công thức mà không cần thay đổi công thức đó.

=LOOKUP Dò tìm một giá trị

Hàm LOOKUP

Dùng để dò tìm một giá trị từ một dòng hoặc một cột trong một dãy ô hoặc một mảng giá trị.

Hàm LOOKUP() có hai dạng: Vec-tơ (vector form) và Mảng (array form)

* Dạng Vec-tơ: LOOKUP() tìm kiếm trên một dòng hoặc một cột, nếu tìm thấy sẽ trả về giá trị của ô cùng vị trí trên dòng (hoặc cột) được chỉ định.

* Dạng Mảng: LOOKUP() tìm kiếm trên dòng (hoặc cột) đầu tiên của một mảng giá trị, nếu tìm thấy sẽ trả về giá trị của ô cùng vị trí trên dòng (hoặc cột) cuối cùng trong mảng đó.

Vector form (dạng vec-tơ)

Cú pháp: LOOKUP(lookup_value, lookup_vector, result_vector)

lookup_value: Là giá trị LOOKUP() sẽ tìm kiếm trong lookup_vector. Nó có thể là một số, một ký tự, một giá trị logic, một tên đã được định nghĩa của một vùng ô hoặc một tham chiếu đến một giá trị.

lookup_vector: Là một vùng mà chỉ gồm một dòng (hoặc một cột) có chứa lookup_value. Những giá trị chứa trong vùng này có thể là một số, một ký tự hoặc một giá trị logic.

- lookup_vector phải được sắp xếp theo thứ tự tăng dần, nếu không, LOOKUP() có thể cho kết quả không chính xác.

- Nếu không tìm thấy lookup_value trong lookup_vector thì LOOKUP() sẽ lấy giá trị lớn nhất mà nhỏ hơn hoặc bằng lookup_value.

- Nếu lookup_value nhỏ hơn giá trị nhỏ nhất trong lookup_vector thì LOOKUP() sẽ báo lỗi #NA!

result_vector: Là một vùng mà chỉ gồm một dòng (hoặc một cột) chứa giá trị trả về. Kích thước của result_vector bắt buộc phải bằng kích thước của lookup_vector.

^^ Array form (dạng mảng)

Cú pháp: LOOKUP(lookup_value, array)

lookup_value: Là giá trị LOOKUP() sẽ tìm kiếm trong array. Nó có thể là một số, một ký tự, một giá trị logic, một tên đã được định nghĩa của một vùng ô hoặc một tham chiếu đến một giá trị.

- Nếu không tìm thấy lookup_value trong array thì LOOKUP() sẽ lấy giá trị lớn nhất mà nhỏ hơn hoặc bằng lookup_value.

- Nếu lookup_value nhỏ hơn giá trị nhỏ nhất trong cột hoặc hàng đầu tiên trong array thì LOOKUP() sẽ báo lỗi #NA!

array: Là một vùng chứa lookup_value, có thể là số, ký tự, hoặc giá trị logic.

Dạng mảng của LOOKUP() gần tương đương như hàm VLOOKUP() hoặc HLOOKUP().

Khác biệt ở chỗ VLOOKUP() và HLOOKUP() tìm kiếm trên cột (hoặc dòng) đầu tiên, còn LOOKUP() tìm kiếm trên cột hoặc trên dòng tùy thuộc vào dạng mảng được khai báo:

- Nếu array là mảng có số cột nhiều hơn số dòng thì LOOKUP() sẽ tìm trên dòng đầu tiên.

- Nếu array là mảng có số dòng nhiều hơn số cột thì LOOKUP() sẽ tìm trên cột đầu tiên.

- Trường hợp array là mảng có số dòng bằng số cột thì LOOKUP() sẽ tìm trên cột đầu tiên.
- VLOOKUP() và HLOOKUP() lấy kết quả trên cột (hoặc) dòng được chỉ định, còn LOOKUP() luôn luôn lấy kết quả trên dòng (hoặc cột) cuối cùng.
- Các giá trị trên dòng (hoặc cột) đầu tiên của array phải được sắp xếp theo thứ tự tăng dần, nếu không, LOOKUP() có thể cho kết quả không chính xác.

Hàm GETPIVOTDATA()

Trả về dữ liệu được lưu giữ trong báo cáo PivotTable. Có thể dùng GETPIVOTDATA() để lấy dữ liệu tổng kết từ một báo cáo PivotTable, với điều kiện là phải thấy được dữ liệu tổng kết từ trong báo cáo đó.

Để nhanh chóng nhập công thức GETPIVOTDATA(), bạn có thể gõ dấu = vào ô muốn nhận dữ liệu tổng kết, rồi nhấp vào ô chứa số liệu tổng kết của báo cáo PivotTable.

Cú pháp: = GETPIVOTDATA(data_field, pivot_table, field1, item1, field2, item2,...)

data_field : Tên trường chứa dữ liệu tổng kết của báo cáo PivotTable. Tên này phải được đặt trong một cặp dấu nháy kép.

pivot_data : Tham chiếu đến một ô, dãy ô, hoặc tên dãy ô bên trong một báo cáo PivotTable. Thông tin này được dùng để xác định báo cáo PivotTable nào chứa dữ liệu trả về.

field1, item1, field2, item2,... : Có thể có từ 1 đến 126 (với Excel 2003 trở về trước thì con số này chỉ là 14) cặp tên field và item mô tả dữ liệu muốn trả về. Những cặp này có thể thuộc bất cứ loại nào. Nếu tên field và item không là ngày tháng

hoặc số liệu, cần phải đặt chúng trong cặp dấu nháy kép. Đối với loại báo cáo OLAP PivotTable, items có thể bao gồm đầy đủ đường dẫn lần tên của item. Một cặp field và item của OLAP PivotTable có thể như sau:

```
"[Product]", "[Product].[All Products].[Foods].[Baked Goods]"
```

Lưu ý:

- * Các field hay item tính toán và các phép tính tự tạo có thể được đặt trong phép tính GETPIVOTDATA.
- * Nếu pivot_table là một dãy có chứa nhiều hơn 1 báo cáo PivotTable, dữ liệu trả về sẽ là từ báo cáo được tạo ra sau cùng.
- * Nếu đối số field và các đối số của item mô tả chỉ mỗi một ô, giá trị của ô đó sẽ được trả về mà không cần biết giá trị đó là chuỗi, là số, là lỗi, hay là một thứ gì đó...
- * Nếu item chứa ngày tháng, giá trị phải được biểu diễn dưới dạng một chuỗi số hoặc được thiết lập bằng cách dùng hàm DATE() để giá trị đó sẽ không biến đổi khi bảng tính được mở ở một máy khác, có hệ thống định dạng ngày tháng khác với nơi tạo ra nó. Ví dụ, một item tham chiếu tới ngày 5 tháng 3 năm 1999 có thể được nhập là 36224 hay DATE(1999, 3, 5). Thời gian có thể được nhập như một giá trị thập phân hoặc bằng cách dùng hàm TIME().
- * Nếu pivot_table không phải là một dãy có chứa báo cáo PivotTable, GETPIVOTDATA() sẽ trả về lỗi #REF!
- * Nếu các đối số miêu tả một field không thể thấy được, hoặc nếu chúng gồm một trường không hiển thị, GETPIVOTDATA() cũng sẽ trả về lỗi #REF!

Hàm DGET()

Trích một giá trị từ một cột của một danh sách hay cơ sở dữ liệu, khớp với điều kiện được chỉ định.

Cú pháp: = DGET(database, field, criteria)

=OFFSET(reference, rows, cols [, height] [, width]) Trả về một vùng tham chiếu từ một vùng xuất phát.

Đây là một trong những hàm rất hay của Excel, và được ứng dụng rất nhiều.

Nó dùng để tham chiếu đến một vùng nào đó, bắt đầu từ một ô, hoặc một dãy ô, với một khoảng cách được chỉ định.

reference: Là vùng mà bạn muốn làm điểm xuất phát để tham chiếu đến vùng khác. Reference phải chỉ đến một ô hoặc một dãy ô liên tục, nếu không, hàm sẽ báo lỗi #VALUE!

rows: Số dòng dời lên (hoặc xuống) tính từ reference, nhập số dương nếu muốn dời xuống, hoặc số âm nếu muốn dời lên.

cols: Số cột dời sang phải trái (hoặc phải) tính từ reference, nhập số dương nếu muốn dời sang phải, hoặc số âm nếu muốn dời sang trái.

height: Là số dòng (độ cao) của vùng tham chiếu cần trả về.

width: Là số cột (độ rộng) của vùng tham chiếu cần trả về.

Ghi chú:

* Nếu số dòng (rows) hoặc cột (cols) vượt ra ngoài phạm vi bảng tính, hàm sẽ báo lỗi #REF!

* Độ cao (height) và độ rộng (width) nếu không nhập, thì xem như nó bằng với độ cao và độ rộng của vùng xuất phát (reference)

=ROW(reference) Trả về số thứ tự dòng của ô đầu tiên trong dãy ô. Trả về số thứ tự dòng của ô đầu tiên ở góc trên bên trái của vùng tham chiếu.

reference: Là ô hoặc một vùng nhiều ô. Nếu reference bỏ trống thì ROW() trả về số thứ tự cột của ô chứa công thức.

=ROWS(array) Trả về số dòng của vùng tham chiếu.

array: Là ô hoặc một vùng nhiều ô, mảng tham chiếu

=TRANSPOSE(array) Chuyển một vùng dữ liệu ngang thành dọc và ngược lại.

Hàm TRANSPOSE() luôn luôn được nhập ở dạng công thức mảng (nhấn Ctrl-Shift-Enter sau khi nhập)

array: Là mảng dữ liệu cần hoán chuyển

Nếu số cột trong vùng nhập công thức nhiều hơn số hàng của array, hoặc số hàng trong vùng nhập công thức nhiều hơn số cột của array, hàm sẽ báo lỗi #NA tại những ô bị dư ra.

=VLOOKUP(giá trị tìm, vùng cần tìm, cột cần lấy, cách tìm) Dò tìm một giá trị trên cột đầu tiên và trả về ...

Hàm tìm kiếm và tham chiếu theo cột.

Vùng cần tìm: thường để ở chế độ giá trị tuyệt đối: \$

Cột cần lấy: ở vùng cần tìm.

Cách tìm: Tìm theo 2 giá trị:

0: Cột bên trái của vùng cần tìm không sắp xếp (ngầm định)

1: Sắp xếp tăng dần.

=HLOOKUP(giá trị tìm, vùng cần tìm, hàng cần lấy, cách tìm) Dò tìm một giá trị trên hàng đầu tiên và trả về ... - giống hàm VLOOKUP

f. HÀM TOÁN HỌC VÀ LƯỢNG GIÁC

Bao gồm các hàm về toán học và lượng giác giúp bạn có thể giải một bài toán đại số, giải tích, hoặc lượng giác từ tiểu học đến đại học...

Lưu ý đến quy cách hiển thị số của VN và của US. Để luôn nhập đúng một giá trị kiểu số bạn hãy sử dụng bàn phím số.

=ABS Tính trị tuyệt đối của một số

=ACOS Tính nghịch đảo cosin

=ACOSH Tính nghịch đảo cosin hyperbol

=ASIN Tính nghịch đảo sin

=ASINH Tính nghịch đảo sin hyperbol

=ATAN Tính nghịch đảo tang

=ATAN2 Tính nghịch đảo tang với tọa độ

=ATANH Tính nghịch đảo tang hyperbol

=CEILING Là tròn đến bội số gần nhất

=COMBIN Tính tổ hợp từ số phần tử chọn

=COS Tính cosin của một góc

=COSH Tính cosin hyperbol

=DEGREES Đổi radians sang độ

=EVEN Làm tròn một số đến số nguyên chẵn gần nhất.

- =EXP Tính lũy thừa cơ số e
- =FACT Tính giai thừa của một số
- =FACTDOUBLE Tính lũy thừa cấp 2
- =FLOOR Làm tròn xuống đến bội số gần nhất do bạn chỉ.
- =GCD Tìm ước số chung lớn nhất
- =INT(X) Làm tròn xuống số nguyên gần nhất (Hàm lấy giá trị là phần nguyên-
Hàm cho giá trị là phần nguyên của X)
- =LCM Tìm bội số chung nhỏ nhất
- =LN Tính logarit cơ số tự nhiên của một số
- =LOG Tính logarit
- =LOG10 Tính logarit cơ số 10
- =MDETERM Tính định thức của ma trận
- =MINVERSE Tìm ma trận nghịch đảo
- =MMULT Tính tích 2 ma trận
- =MOD(a,b) Hàm cho giá trị là phần dư của phép chia a:b
- =MROUND Làm tròn một số đến bội số của số khác.
- =MULTINOMIAL Tỷ lệ giai thừa tổng với tích các giai thừa của các số.
- =ODD Làm tròn đến một số nguyên lẻ gần nhất.
- =PI Trả về giá trị pi
- =POWER Tính lũy thừa của một số

=PRODUCT Tính tích các số

=QUOTIENT Lấy phần nguyên của phép chia

=RADIANS Đổi độ sang radians.

=RAND Trả về một số ngẫu nhiên trong khoảng 0 và 1

=RANDBETWEEN Trả về một số ngẫu nhiên trong khoảng do bạn chỉ định

Hàm ROMAN()

Dùng để chuyển đổi một số dạng Ả-rập sang dạng số La-mã

Cú pháp: = ROMAN(number, form)

number: Số cần chuyển đổi

form: dạng chuyển đổi

0 (hoặc TRUE, hoặc không nhập): Dạng cổ điển

1 cho đến 3: Dạng cổ điển nhưng được rút gọn, số càng lớn rút gọn càng nhiều (xem thêm ở ví dụ)

4 (hoặc FALSE): Dạng hiện đại

Chú ý:

* number phải là số dương, nếu $number < 0$ hàm sẽ báo lỗi #VALUE!

* Nếu number là số thập phân, ROMAN() chỉ chuyển đổi phần nguyên của nó

* Hàm ROMAN() chỉ xử lý được tới số lớn nhất là 3999, nếu $number > 3999$ hàm sẽ báo lỗi #VALUE!

* Sau khi đã chuyển đổi, kết quả sẽ là một dữ liệu dạng text, và không thể tính toán với nó được nữa

Ví dụ:

ROMAN(499, 0) = CDXCIX = ROMAN(499) = ROMAN(499, TRUE)

ROMAN(499, 1) = LDVLIIV

ROMAN(499, 2) = XDIX

ROMAN(499, 3) = VDIV

ROMAN(499, 4) = ID = ROMAN(499, FALSE)

ROMAN(2008) = MMVIII

=ROUND(X,n) Hàm làm tròn n số của X

Nếu n dương (n>0) sẽ làm tròn số bên phải kể từ vị trí dấu chấm thập phân.

Nếu n âm (n<0) sẽ làm tròn số bên trái kể từ vị trí dấu chấm thập phân.

=ROUNDDOWN Làm tròn một số hướng xuống zero

=ROUNDUP Làm tròn một số hướng ra xa zero.

=SIN Tính sin của một góc

=SINH Tính sin hyperbol của một số

=SUM Tính tổng của các số

=DSUM (vùng dữ liệu, cột giá trị, vùng tiêu chuẩn) Hàm tính tổng trong cột giá trị thoả mãn điều kiện của vùng tiêu chuẩn trong vùng dữ liệu.

VD: Tính tổng tiền lương những người 26 tuổi.

Hàm SUMIF()

Tính tổng các ô trong một vùng thoả một điều kiện cho trước.

Cú pháp: = SUMIF(range, criteria, sum_range)

Range : Dãy các ô để tính tổng, có thể là ô chứa số, tên, mảng, hay tham chiếu đến các ô chứa số. Ô rỗng và ô chứa giá trị text sẽ được bỏ qua.

Criteria : Điều kiện để tính tổng. Có thể ở dạng số, biểu thức, hoặc text. Ví dụ, criteria có thể là 32, "32", "> 32", hoặc "apple", v.v...

Sum_range : Là vùng thực sự để tính tổng. Nếu bỏ qua, Excel sẽ coi như sum_range = range.

Lưu ý:

* Sum_range không nhất thiết phải cùng kích thước với range. Vùng thực sự để tính tổng được xác định bằng ô đầu tiên phía trên bên trái của sum_range, và bao gồm thêm những ô tương ứng với kích thước của range. Ví dụ:

- Nếu Range là A1:A5, Sum_range là B1:B5, thì vùng thực sự để tính tổng là B1:B5

- Nếu Range là A1:A5, Sum_range là B1:B3, thì vùng thực sự để tính tổng là B1:B5

- Nếu Range là A1:B4, Sum_range là C1:D4, thì vùng thực sự để tính tổng là C1:D4

- Nếu Range là A1:B4, Sum_range là C1:D2, thì vùng thực sự để tính tổng là C1:D4

* Có thể dùng các ký tự đại diện trong điều kiện: dấu ? đại diện cho một ký tự, dấu

* đại diện cho nhiều ký tự (nếu như điều kiện là tìm những dấu ? hoặc *, thì gõ thêm dấu ~ ở trước dấu ? hay *).

* Khi điều kiện để tính tổng là những ký tự, SUMIF() không phân biệt chữ thường hay chữ hoa.

Hàm SUMIFS()

Tính tổng các ô trong một vùng thỏa nhiều điều kiện cho trước.

Cú pháp: = SUMIFS(sum_range, criteria_range1, criteria1, criteria_range2, criteria2, ...)

Sum_range : Dãy các ô để tính tổng, có thể là ô chứa số, tên, mảng, hay tham chiếu đến các ô chứa số. Ô rỗng và ô chứa giá trị text sẽ được bỏ qua.

Criteria_range1, criteria_range2... : Có thể có từ 1 đến 127 vùng dùng để liên kết với các điều kiện cho vùng.

Criteria1, criteria2... : Có thể có từ 1 đến 127 điều kiện để tính tổng. Chúng có thể ở dạng số, biểu thức, hoặc text. Ví dụ, criteria có thể là 32, "32", "> 32", hoặc "apple", v.v...

Lưu ý:

* Mỗi ô trong sum_range chỉ được tính tổng nếu tất cả các điều kiện tương ứng với ô đó đều đúng. Nếu thỏa các điều kiện, nó sẽ bằng 1, còn không, thì nó bằng 0.

* Không giống như những đối số range và criteria của hàm SUMIF, trong hàm SUMIFS, mỗi vùng criteria_range phải có cùng kích thước và hình dạng giống như sum_range.

* Có thể dùng các ký tự đại diện trong các điều kiện: dấu ? đại diện cho một ký tự, dấu * đại diện cho nhiều ký tự (nếu như điều kiện là tìm những dấu ? hoặc *, thì gõ thêm dấu ~ ở trước dấu ? hay *)

* Khi điều kiện để đếm là những ký tự, SUMIFS() không phân biệt chữ thường hay chữ hoa.

Hàm SUMSQ()

Dùng để tính tổng các bình phương của các số

Cú pháp: = SUMSQ(number1, number2, ...)

number1, number2, ... : Có thể dùng đến 255 tham số (với Excel 2003 trở về trước, con số này chỉ là 30)

Các tham số (number) có thể là một số, là một mảng, một tên, hay là một tham chiếu đến một ô chứa số, v.v...

Ví dụ: SUMSQ(3, 4) = (3²) + (4²) = 9 + 16 = 25

Ba hàm sau đây có cách dùng và cú pháp tương tự nhau:

Hàm SUMX2MY2(), Hàm SUMXPY2() và Hàm SUMXMY2()

Để dễ nhớ tên của ba hàm này, bạn đọc chúng từ trái sang phải với các quy ước sau:

SUM = Tổng, M (Minus) = Trừ (hiệu số), P (Plus) = Cộng (tổng số), 2 = Bình phương, X và Y là hai mảng gì đó, có chứa nhiều phần tử x và y

Vậy, định nghĩa và cách tính toán của 3 hàm này là:

= SUMX2MY2: Tổng của hiệu hai bình phương của các phần tử tương ứng trong 2 mảng dữ liệu

= SUMX2PY2: Tổng của tổng hai bình phương của các phần tử tương ứng trong 2 mảng dữ liệu

= SUMXMY2: Tổng của bình phương của hiệu các phần tử tương ứng trong 2 mảng dữ liệu

Cú pháp:

= SUMX2MY2(array_x, array_y)

= SUMX2PY2(array_x, array_y)

= SUMXMY2(array_x, array_y)

array_x và array_y là các dãy ô hoặc giá trị kiểu mảng

Lưu ý:

* array_x và array_y bắt buộc phải có cùng kích thước, nếu không, hàm sẽ báo lỗi #NA!

* Nếu trong array_x hoặc array_y có những giá trị kiểu text, kiểu logic hoặc rỗng, thì sẽ được bỏ qua (không tính), tuy nhiên các giá trị = 0 vẫn được tính.

=TAN Tính tang của một góc

=TANH Tính tang hyperbol của một số

=TRUNC Cắt bớt phần thập phân của số

Hàm SUMPRODUCT() Tính tổng các tích các phần tử tương ứng trong các mảng giá trị

Sum = Tổng-Product = Tích

SUMPRODUCT = Tổng của tích (các mảng dữ liệu)

Cú pháp: = SUMPRODUCT(array1, array2, ...)

array1, array2, ... : Có thể dùng từ 2 tới 255 mảng (với Excel 2003 trở về trước thì con số này chỉ là 30) và các mảng này phải cùng kích thước với nhau

Lưu ý:

* Nếu các mảng không cùng kích thước, SUMPRODUCT sẽ báo lỗi #VALUE!

* Bất kỳ một phần tử nào trong mảng không phải là dữ liệu kiểu số, sẽ được SUMPRODUCT coi như bằng 0 (zero)

Hàm SUBTOTAL là một hàm rất linh hoạt nhưng cũng là một trong các hàm hơi khó sử dụng của Excel. Điều khó hiểu thứ nhất chính là cái tên của nó, vì nó thực sự làm được nhiều thứ hơn ý nghĩa của tên hàm. Đối số thứ nhất của hàm bắt buộc bạn phải nhớ con số đại diện cho phép tính cần thực hiện trên tập số liệu (trong Excel 2010 có tính năng AutoComplete giúp chúng ta khỏi nhớ các con số này). Hàm SUBTOTAL được Microsoft nâng cấp kể từ phiên bản Excel 2003 với sự gia tăng các tùy chọn cho đối số thứ nhất của hàm, tuy nhiên điều này dẫn đến sự không tương thích với các phiên bản cũ nếu chúng ta sử dụng các tính năng mới bổ sung này.

Đối số đầu tiên của của hàm SUBTOTAL xác định hàm thực sự nào sẽ được sử dụng khi tính toán (xem trong danh sách bên dưới). Ví dụ nếu đối số là 1 thì hàm SUBTOTAL hoạt động giống nhưng hàm AVERAGE, nếu đối số thứ nhất là 9 thì hàm hàm SUBTOTAL hoạt động giống nhưng hàm SUM.

SUBTOTAL là hàm tính toán cho một nhóm con trong một danh sách hoặc bảng dữ liệu tùy theo phép tính mà bạn chọn lựa trong đối số thứ nhất.

Cú pháp: = SUBTOTAL(function_num, ref1, ref2,...)

Function_num: Các con số từ 1 đến 11 (hay 101 đến 111) qui định hàm nào sẽ được dùng để tính toán trong SUBTOTAL

Ref1, ref2: Các vùng địa chỉ tham chiếu mà bạn muốn thực hiện phép tính trên đó.

Trong Excel 2010, bạn có thể dùng đến 254 ref (với Excel 2003 trở về trước thì con số này chỉ là 29)

Ghi chú:

* Nếu có hàm SUBTOTAL khác đặt lồng trong các đối số ref1, ref2,... thì các hàm lồng này sẽ bị bỏ qua không được tính, nhằm tránh trường hợp tính toán 2 lần.

* Đối số function_num nếu từ 1 đến 11 thì hàm SUBTOTAL tính toán bao gồm cả các giá trị ẩn trong tập số liệu (hàng ẩn). Đối số function_num nếu từ 101 đến 111 thì hàm SUBTOTAL chỉ tính toán cho các giá trị không ẩn trong tập số liệu (bỏ qua các giá trị ẩn).

* Hàm SUBTOTAL sẽ bỏ qua không tính toán tất cả các hàng bị ẩn bởi lệnh Filter (Auto Filter) mà không phụ thuộc vào đối số function_num được dùng (1 giống 101...).

* Hàm SUBTOTAL được thiết kế để tính toán cho các cột số liệu theo chiều dọc, nó không được thiết kế để tính theo chiều ngang.

* Hàm này chỉ tính toán cho dữ liệu 2-D, do vậy nếu dữ liệu tham chiếu dạng 3-D (Ví dụ về tham chiếu 3-D: =SUM(Sheet2:Sheet13!B5) thì hàm SUBTOTAL báo lỗi #VALUE!

Hàm SQRT()

Dùng để tính căn bậc hai của một số

Cú pháp: = SQRT(number)

number: Số thực, dương (nếu number < 0 hàm sẽ báo lỗi #NUM!)

Ví dụ: Giả sử ở ô A2, có con số -16

SQRT(16) = 4

SQRT(A2) = #NUM!

SQRT(ABS(A2)) = 4

Hàm SQRTPI()

Dùng để tính căn bậc hai của một số nhân với Pi (= 3.14159265358979)

Cú pháp: = SQRTPI(number)

number: Số thực, dương nhân với Pi (nếu number < 0 hàm sẽ báo lỗi #NUM!)

Ví dụ: Giả sử ở ô A2, có con số -16

SQRTPI(1) = 1.772454 (căn bậc hai của Pi)

SQRTPI(2) = 2.506628 (căn bậc hai của 2*Pi)

Hàm SIGN()

Trả về dấu của số: 1 nếu là số dương, 0 (zero) nếu là số 0 và -1 nếu là số âm.

Cú pháp: = SIGN(number)

Ví dụ:

SIGN(10) = 1

SIGN(4-4) = 0

SIGN(-0.057) = -1

Hàm SERIESSUM()

Dùng để tính tổng lũy thừa của một chuỗi số, theo công thức sau đây:

$$\text{series}(x, n, m, a) = a_1 * x^n + a_2 * x^{(n+m)} + a_3 * x^{(n+2m)} + \dots + a_i * x^{(n+(i-1)m)}$$

Cú pháp: =SERIESSUM(x, n, m, coefficients)

x : giá trị nhập vào cho chuỗi lũy thừa

n : lũy thừa khởi tạo để tăng tới x

m : bước tăng cho mỗi phần tử trong chuỗi

coefficients : tập hợp hệ số sẽ được nhân với mỗi lũy thừa của x

Các thông số này phải là các dữ liệu kiểu số, nếu không, hàm sẽ báo lỗi #VALUE!

Ví dụ:

$$\text{SERIESSUM}(5, 0, 2, \{1, 2, 3, 4\}) = 64,426$$

Diễn giải chi tiết: (x = 5, n = 0, m = 2, coefficients = 1, 2, 3, 4)

$$=1*5^0 + 2*5^{(0+2)} + 3*5^{(0+2*2)} + 4*5^{(0+3*2)} = 64426$$

HÀM THỐNG KÊ

Bao gồm các hàm số giúp bạn giải quyết các bài toán thống kê từ đơn giản đến phức tạp.

=AVEDEV Tính bình quân độ phân cực

=AVERAGE(vùng) Tính trung bình cộng các số.

=AVERAGEA Tính trung bình cộng các giá trị

=DAVERAGE (vùng dữ liệu, cột cần tính, vùng tiêu chuẩn) Hàm tính trung bình các giá trị trong cột thoả mãn điều kiện của vùng tiêu chuẩn trong vùng dữ liệu.

Hàm AVERAGEIF()

Trả về trung bình cộng (số học) của tất cả các ô được chọn thỏa mãn một điều kiện cho trước.

Cú pháp: = AVERAGEIF(range, criteria, average_range)

range : Là một hoặc nhiều ô cần tính trung bình, có thể bao gồm các con số, các tên vùng, các mảng hoặc các tham chiếu đến các giá trị...

criteria : Là điều kiện dưới dạng một số, một biểu thức, địa chỉ ô hoặc chuỗi, để qui định việc tính trung bình cho những ô nào...

average_range : Là tập hợp các ô thật sự được tính trung bình. Nếu bỏ trống thì Excel dùng range để tính.

Lưu ý:

- * Các ô trong range nếu có chứa những giá trị luận lý (TRUE hoặc FALSE) thì sẽ được bỏ qua.
- * Những ô rỗng trong average_range cũng sẽ được bỏ qua.
- * Nếu range rỗng hoặc có chứa dữ liệu text, AVERAGEIF sẽ báo lỗi #DIV/0!
- * Nếu có một ô nào trong criteria rỗng, AVERAGEIF sẽ xem như nó bằng 0.
- * Nếu không có ô nào trong range thỏa mãn điều kiện của criteria, AVERAGEIF sẽ báo lỗi #DIV/0!
- * Bạn có thể các ký tự đại diện như ?, * trong criteria (dấu ? thay cho một ký tự nào đó, và dấu * thay cho một chuỗi nào đó). Khi điều kiện trong criteria là chính các dấu ? hoặc *, thì bạn gõ thêm dấu ~ trước nó.

* `average_range` không nhất thiết phải có cùng kích thước với `range`, mà các ô thực sự được tính trung bình sẽ dùng ô trên cùng bên trái của `average_range` làm ô bắt đầu, và bao gồm thêm những ô tương ứng với kích thước của `range`.

Hàm AVERAGEIFS()

Trả về trung bình cộng (số học) của tất cả các ô được chọn thỏa mãn nhiều điều kiện cho trước.

Cú pháp: = AVERAGEIFS(`average_range`, `criteria_range1`, `criteria1`, `criteria_range2`, `criteria2`, ...)

`average_range` : Vùng cần tính trung bình, có thể bao gồm các con số, các tên vùng, các mảng hoặc các tham chiếu đến các giá trị...

`criteria_range1`, `criteria_range2`, ... : Vùng chứa những điều kiện để tính trung bình. Có thể khai báo từ 1 đến 127 vùng.

`criteria1`, `criteria2`, ... : Là các điều kiện để tính trung bình. Có thể khai báo từ 1 đến 127 điều kiện, dưới dạng số, biểu thức, tham chiếu hoặc chuỗi...

Lưu ý:

* Nếu `average_range` rỗng hoặc có chứa dữ liệu text, AVERAGEIFS sẽ báo lỗi #DIV/0!

* Nếu có một ô nào trong những vùng `criteria_range` rỗng, AVERAGEIFS sẽ xem như nó bằng 0.

* Những giá trị logic: TRUE sẽ được xem là 1, và FALSE sẽ được xem là 0.

* Mỗi ô trong `average_range` chỉ được tính trung bình nếu thỏa tất cả điều kiện quy định cho ô đó

* Không giống như AVERAGEIF(), mỗi vùng criteria_range phải có cùng kích thước với average_range

* Nếu có một ô nào trong average_range không thể chuyển đổi sang dạng số, hoặc nếu không có ô nào thỏa tất cả các điều kiện, AVERAGEIFS sẽ báo lỗi #DIV/0!

* Có thể các ký tự đại diện như ?, * cho các điều kiện (dấu ? thay cho một ký tự nào đó, và dấu * thay cho một chuỗi nào đó). Khi điều kiện trong criteria là chính các dấu ? hoặc *, thì bạn gõ thêm dấu ~ trước nó.

Hàm COUNT() Đếm ô dữ liệu chứa số (đếm số ô không trống) - đếm số ô có trong vùng. Nhưng chỉ đếm những ô có kiểu dữ liệu là kiểu số.

Cú pháp=COUNT(vùng)

=COUNTA Đếm số ô chứa dữ liệu

=COUNTIF(Vùng cần đếm, ô điều kiện) Hàm đếm có điều kiện - đếm số ô có trong vùng với điều kiện đã chỉ ra ở ô điều kiện.

Hàm COUNTIF đếm ô có kiểu dữ liệu là kiểu số và cả kiểu ký tự.

=DCOUNT(vùng dữ liệu, cột cần đếm, vùng tiêu chuẩn) Đếm số ô chứa lượng giá trị số trong cột thỏa mãn đk trong vùng tiêu chuẩn tìm kiếm trong vùng dữ liệu.

VD: Đếm xem có bao nhiêu người có mức lương là 500000.

=MAX(vùng) Tìm số lớn nhất trong vùng

=MAXA Tìm giá trị lớn nhất

Hàm DMAX()

Cú pháp: = DMAX(database, field, criteria)=DMAX (vùng dữ liệu, cột giá trị, vùng tiêu chuẩn)

Hàm tính giá trị cao nhất trong cột giá trị thoả mãn điều kiện của vùng tiêu chuẩn trong vùng dữ liệu.

VD: Tìm xem những người 26 tuổi ai cao lương nhất.

=MAX(vùng) Tìm số nhỏ nhất trong vùng

=MINA Tìm giá trị nhỏ nhất

Hàm DMIN()

Cú pháp: =DMIN(database, field, criteria)=DMIN (vùng dữ liệu, cột giá trị, vùng tiêu chuẩn)

Hàm tính giá trị nhỏ nhất trong cột giá trị thoả mãn điều kiện của vùng tiêu chuẩn trong vùng dữ liệu.

VD: Tìm xem những người 26 tuổi ai thấp lương nhất.

=RANK(ô cần xếp thứ, vùng cần so sánh) Tìm vị thứ của một số trong dãy số.

Vùng cần so sánh: Thường để ở chế độ giá trị tuyệt đối: \$

Hàm DVARP()

Cú pháp: =DVARP(database, [field,] criteria)

Tính toán sự biến thiên của một tập hợp dựa trên toàn thể tập hợp, bằng cách sử dụng các số liệu trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

Hàm DVAR()

Cú pháp: =DVAR(database, [field,] criteria)

Ước lượng sự biến thiên của một tập hợp dựa trên một mẫu, bằng cách sử dụng các số liệu trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

Hàm DSUM()

Cú pháp: = DSUM(database, field, criteria)

Cộng các số trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

Hàm DSTDEVP()

Cú pháp: = DSTDEVP(database, field, criteria)

Tính độ lệch chuẩn của một tập hợp theo toàn thể các tập hợp, bằng cách sử dụng các số liệu trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

Hàm DSTDEV()

Cú pháp: = DSTDEV(database, field, criteria)

Ước lượng độ lệch chuẩn của một tập hợp theo mẫu, bằng cách sử dụng các số liệu trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

Hàm DPRODUCT()

Cú pháp: = DPRODUCT(database, field, criteria)

Nhân các giá trị trong một cột của một danh sách hay của một cơ sở dữ liệu, theo một điều kiện được chỉ định.

Đếm số ô rỗng trong bảng tính

Dùng công thức mảng: $\{=SUM(IF(ISBLANK(range), 1, 0))\}$ với range là vùng dữ liệu cần kiểm tra.

Đếm số ô chứa những giá trị không phải là kiểu số

Dùng công thức mảng: $\{=SUM(IF(ISNUMBER(range), 0, 1))\}$ với range là vùng dữ liệu cần kiểm tra.

Đếm số ô bị lỗi

Dùng công thức mảng: $\{=SUM(IF(ISERROR(range), 1, 0))\}$ với range là vùng dữ liệu cần kiểm tra.

g. HÀM XỬ LÝ VĂN BẢN VÀ DỮ LIỆU

Bao gồm các hàm xử lý chuỗi văn bản như trích lọc, tìm kiếm, thay thế, chuyển đổi chuỗi văn bản trong Excel.

Hàm ASC()

Dùng để đổi các ký tự double-byte sang các ký tự single-byte cho những ngôn ngữ sử dụng bộ ký tự double-byte.

Cú pháp: = ASC(text)

text : Là chữ hoặc tham chiếu đến một ô có chứa chữ. Nếu text không chứa bất kỳ mẫu tự nào thuộc loại double-byte, thì text sẽ không được chuyển đổi.

Ví dụ: = ASC("Excel") = Excel

=CHAR Chuyển số thành ký tự

=CLEAN Xóa ký tự không phù hợp

=CODE Trả về mã số của ký tự đầu tiên

Hàm CONCATENATE

Công dụng: Dùng nối nhiều chuỗi lại với nhau

Công thức: =CONCATENATE(text1,text2,...)

Ví dụ: =CONCATENATE("Giải pháp", "Excel", " - ", "Công cụ tuyệt vời của bạn")

→ Giải pháp Excel - Công cụ tuyệt vời của bạn

=DOLLAR Chuyển định dạng số thành tiền tệ

Hàm EXACT

Công dụng: Dùng để so sánh hai chuỗi với nhau.

Công thức: =EXACT(text1,text2)

Hàm EXACT phân biệt chữ thường và chữ hoa.

Nếu 2 chuỗi text1, text2 giống nhau hoàn toàn, hàm sẽ trả về TRUE; nếu không, sẽ trả về trị FALSE

Ví dụ:

=EXACT("Giải pháp", "Giải pháp") → TRUE

=EXACT("Giải pháp", "Giải Pháp") → FALSE

=FIXED Chuyển một số sang định dạng văn bản

=LEFT(X,n) Hàm LEFT lấy n ký tự từ bên trái sang của văn bản X

=LEN Tính độ dài một chuỗi

=LOWER Chuyển thành chữ thường.

=PROPER Chuyển ký tự đầu mỗi từ thành chữ hoa

Hàm MID Dùng để trích xuất một chuỗi con (substring) từ một chuỗi

Công thức: =MID(text, start_num, num_chars)=MID(X, m, n)

Hàm MID lấy n ký tự trong chuỗi X bắt đầu từ vị trí m.

text: chuỗi văn bản cần trích xuất

start_num: vị trí bắt đầu trích ra chuỗi con, tính từ bên trái sang

num_chars: số ký tự của chuỗi con cần trích ra

- num_chars phải là số nguyên dương

- start_num phải là số nguyên dương

- Nếu start_num lớn hơn độ dài của chuỗi thì kết quả trả về sẽ là chuỗi rỗng

Ví dụ: =MID("Karen Elizabeth Hammond", 7, 9) → Elizabeth

Hàm REPLACE

Công dụng: Dùng để thay thế một phần của chuỗi bằng một chuỗi khác, dựa vào số ký tự được chỉ định

Công thức: =REPLACE(old_text, start_num, num_chars, new_text)

old_text: chuỗi văn bản cần được xử lý

start_num: vị trí bắt đầu tìm cái sẽ thay thế, tính từ bên trái sang

num_chars: số ký tự của chuỗi cần được thay thế

new_text: chuỗi văn bản sẽ thay thế cho số ký tự đã chọn bởi start_num và num_chars

Cái khó của hàm này là xác định được bởi start_num và num_chars. Làm sao biết được bắt đầu từ đâu và thay thế bao nhiêu chữ? Tôi gợi ý nhé:

- Bạn dùng hàm FIND() hoặc SEARCH() để xác định vị trí bắt đầu (start_num)

- Dùng hàm LEN() để xác định số ký tự của chuỗi sẽ được thay thế (num_chars)

Ví dụ: để thay số 2007 bằng 2008 trong câu Expense Budget for 2007

Dùng công thức như sau:

=REPLACE(A1, FIND("2007", A1), LEN("2007"), "2008") → Expense Budget for 2008

Với A1 = Expense Budget for 2007

Hàm RIGHT Dùng để trích xuất phần bên phải của một chuỗi một hoặc nhiều ký tự tùy theo sự chỉ định của bạn

Công thức: =RIGHT(text [,num_chars])=RIGHT(X,n)

Hàm RIGHT lấy n ký tự từ bên phải sang của văn bản X

text: chuỗi văn bản cần trích xuất ký tự

num_chars: số ký tự cần trích ra phía bên phải của chuỗi text, mặc định là 1

- num_chars phải là số nguyên dương

- Nếu num_chars lớn hơn độ dài của chuỗi thì kết quả trả về sẽ là toàn bộ chuỗi text

Ví dụ: =RIGHT("Karen Elizabeth Hammond", 7) → Hammond

=REPT Lặp lại một chuỗi

=SUBSTITUTE Thay thế một chuỗi xác định

Hàm FIND và Hàm SEARCH

Công dụng: Dùng để tìm vị trí bắt đầu của một chuỗi con (substring) trong một chuỗi

Công thức:

=FIND(find_text, within_text [, start_num])

=SEARCH(find_text, within_text [, start_num])

find_text: chuỗi văn bản cần tìm (chuỗi con)

within_text: chuỗi văn bản chứa chuỗi cần tìm (chuỗi mẹ)

start_num: vị trí bắt đầu tìm trong chuỗi within_text (mặc định là 1)

Một số lưu ý:

- Kết quả của hai hàm này là một con số, chỉ vị trí bắt đầu (tính từ start_num) của find_text trong within_text
- Dùng SEARCH() khi muốn tìm một chuỗi bất kỳ. Ví dụ: SEARCH("e", "Expenses") sẽ cho kết quả là 1.
- Dùng FIND() khi muốn tìm chính xác một chuỗi có phân biệt chữ hoa, chữ thường. Ví dụ: FIND("e", "Expenses") sẽ cho kết quả là 4.
- Nếu không tìm thấy find_text, hàm sẽ báo lỗi #VALUE
- Có thể dùng những ký tự đại diện như *, ? trong find_text của hàm SEARCH()
- Với hàm SEARCH(), nếu muốn tìm chính ký tự * hoặc ? thì gõ dấu ~ trước ký tự đó (~* hoặc là ~?)

=TEXT Chuyển một số sang text.

Hàm T

Công dụng: Trả về một chuỗi nếu trị tham chiếu là chuỗi, ngược lại, sẽ trả về chuỗi rỗng - Kiểm tra dữ liệu kiểu text

Công thức: =T(value)

Hàm này ít khi được dùng.

Hàm SUBSTITUTE

Công dụng: Dùng để thay thế một chuỗi này bằng một chuỗi khác. Hàm này cũng tương tự hàm REPLACE(), nhưng dễ sử dụng hơn.

Công thức: =SUBSTITUTE(text, old_text, new_text [,instance_num])

text: chuỗi văn bản gốc, cần được xử lý

old_text: chuỗi văn bản cần được thay thế

new_text: chuỗi văn bản sẽ thay thế vào

instance_num: số lần thay thế old_text bằng new_text, nếu bỏ qua thì tất cả old_text tìm được sẽ được thay thế bằng new_text

Ví dụ: để thay số 2007 bằng 2008 trong câu Expense Budget for 2007

Dùng công thức như sau:

=SUBSTITUTE("Expense Budget for 2007", "2007", "2008") → Expense Budget for 2008

HÀM THAY THẾ CHUỖI NÀY BẰNG CHUỖI KHÁC (Substituting One Substring for Another)

Có nhiều chương trình có khả năng tìm kiếm một số đoạn văn và thay thế nó bằng đoạn văn khác.

Excel cũng có khả năng làm chuyện đó bằng cách dùng hàm. Đó là hàm REPLACE và hàm SUBSTITUTE.

=TRIM Xóa những ký tự trắng bên trong chuỗi.

=UPPER Chuyển ký tự thường thành hoa.

Hàm VALUE

Công dụng: Dùng để đổi một chuỗi đại diện cho một số thành kiểu số

Công thức: =VALUE(text)

text phải là định dạng số, ngày tháng hoặc bất kỳ một thời gian nào miễn là được Excel công nhận.

Nếu 2 chuỗi text1, text2 giống nhau hoàn toàn, hàm sẽ trả về TRUE; nếu không, sẽ trả về trị FALSE

Ví dụ:

Để trích ra số 6500 trong SQA6500, bạn có thể dùng hàm RIGHT()

=RIGHT("SQA6500", 4) → 6500

Tuy nhiên kết quả do hàm RIGHT() có được sẽ ở dạng text, bạn không thể nhân chia cộng trừ gì với cái "6500" này được.

Để có thể tính toán với "6500", bạn phải đổi nó sang dạng số:

=VALUE(RIGHT("SQA6500",4)) → 6500

Cũng là 6500, nhưng bây giờ bạn có thể cộng trừ nhân chia với nó.

*Trích xuất họ và tên (ví dụ dùng để trích cho tên tiếng Anh, bỏ qua tên đệm)

Cách làm là dùng hàm FIND() để tìm những khoảng trắng phân cách giữa họ và tên, sau đó dùng hàm LEFT() để tách phần tên, và hàm RIGHT() để tách phần họ.

Để lấy phần tên (First Name), chúng ta dùng công thức sau (giả sử họ tên nằm ở cell A2):

```
=LEFT(A2, FIND(" ", A2) - 1)
```

Nghĩa là dùng hàm FIND() để tìm vị trí của ký tự trắng đầu tiên kể từ bên trái, ví dụ nó là vị trí thứ 5,

khi đó hàm LEFT() sẽ xác định được cái tên này gồm có 4 chữ (= 5-1).

Để lấy phần họ (Last Name), chúng ta dùng công thức:

```
=RIGHT(A2, LEN(A2) - FIND(" ", A2))
```

Bạn tự dịch câu này nhé!

*Trích xuất họ, tên đệm và tên (ví dụ với tên tiếng Anh, phần tên đệm được viết tắt)

Cách làm giống như bài Trích xuất họ và tên ở trên, tuy nhiên có khác một chút, để trích thêm phần tên đệm.

Giả sử Họ và Tên (full name) nằm ở cell A2, và đang có giá trị là Karen E. Hammond

Đầu tiên, như bài trên, dùng công thức sau để tách phần Tên (first name):

```
=LEFT(A2, FIND(" ", A2) - 1) → Karen
```

Công thức FIND(" ", A2) sẽ cho kết quả là 6, là vị trí của khoảng trắng đầu tiên (sau chữ Karen).

Để tìm vị trí của khoảng trắng thứ hai, thì bạn phải gán vị trí bắt đầu tìm (start_num) là 7, hoặc là bằng kết quả của FIND(" ", A2) cộng thêm 1:

```
=FIND(" ", A2, FIND(" ", A2) + 1)
```

Rồi dùng kết quả của công thức này làm tham số cho hàm RIGHT() để trích ra phần Họ (last name):

=RIGHT(A2, LEN(A2) - FIND(" ", A2, FIND(" ", A2) + 1)) → Hammond

Để trích phần tên đệm, dùng hàm FIND() để tìm vị trí của dấu chấm (.) rồi đưa vào trong công thức của hàm MID() để tìm ký tự đứng trước dấu chấm:

=MID(A2, FIND(".", A2) - 1, 1) → E

HÀM LOGIC

Hàm AND

Có lẽ khỏi giải thích nhỉ. AND có nghĩa là VÀ. Vậy thôi. Dùng hàm này khi muốn nói đến cái này và cái này và cái này.....

Cú pháp: AND(logical1 [, logical2] [, logical3]...)

logical: Những biểu thức sẽ được xét xem đúng (TRUE) hay sai (FALSE)

Nếu tất cả các biểu thức đều đúng, hàm AND() sẽ trả về giá trị TRUE, và chỉ cần 1 trong các biểu thức sai, hàm AND() sẽ trả về giá trị FALSE.

Bạn có thể dùng hàm AND() bất cứ chỗ nào bạn muốn, nhưng thường thì hàm AND() hay được dùng chung với hàm IF().

Ví dụ:

=IF(AND(B2 > 0, C2 > 0), "1000", "No bonus")

Nếu giá trị ở B2 và ở C2 lớn hơn 0, thì (thưởng) 1.000, còn nếu chỉ cần một trong 2 ô B2 hoặc C2 nhỏ hơn 0, thì không thưởng chi cả.

=FALSE(): Có thể nhập trực tiếp FALSE vào trong công thức, Excel sẽ hiểu đó là một biểu thức có giá trị FALSE mà không cần dùng đến cú pháp của hàm này

Hàm IF

=IF (logical_test, value_if_true, value_if_false) : Dùng để kiểm tra điều kiện theo giá trị và công thức

Cú pháp: IF (điều kiện, giá trị 1, giá trị 2) Nếu điều kiện đúng thì hàm trả về giá trị 1, ngược lại hàm nhận giá trị 2

Cái lập luận: "Nếu tôi đúng thì làm cho tôi cái này, nếu tôi sai thì làm cho tôi cái kia".. Có lẽ trong chúng ta ai cũng hiểu.

Một tình huống đơn giản nhất

Cú pháp: IF(logical_test, value_is_true)

logical_test: Một biểu thức sẽ được xét xem đúng (TRUE) hay sai (FALSE)

value_is_true: giá trị trả về khi biểu thức logical_test được kiểm tra là đúng (TRUE)

Ví dụ:

=IF(A1 >= 1000, "It's big!")

Nghĩa là, nếu giá trị ở A1 lớn hơn hoặc bằng 1000, thì kết quả nhận được sẽ là "It's big!", còn không, nếu A1 nhỏ hơn 1000, kết quả sẽ là FALSE.

Một ví dụ khác, giả sử bạn có một bảng đánh giá mức độ bán ra, mua vào của một danh mục hàng hóa dài, và bạn muốn theo dõi những mặt hàng có doanh số bán ra không đạt yêu cầu để điều chỉnh chiến lược kinh doanh của mình, bằng cách gán những dấu "<" bên cạnh nó, hễ phần trăm doanh số càng thấp thì những dấu hiệu "<" càng nhiều...

Bạn có thể dùng hàm IF(), theo mẫu:

=IF(cell<0, flag)

Với cell là giá trị doanh số mà bạn muốn theo dõi, và flag là dấu hiệu để mô tả, ví dụ, cell chứa giá trị doanh số là B2:

=IF(B2<0, "<<<<<")

Để những dấu "<" tỷ lệ thuận với mức sụt giảm doanh số bán hàng, bạn có thể dùng hàm REPT(), với công thức:

REPT("<" , B2 * -100)

Ở đây, phải nhân giá trị của B2 với -100, bởi vì chúng ta chỉ xét những trường hợp B2<0

Và công thức hoàn chỉnh để thể hiện mức độ sụt giảm doanh số của từng mặt hàng sẽ là:

=IF(B2<0, REPT("<" , B2 * -100))

Những hàm IF lồng nhau

Trong cuộc sống đời thường, có mấy ai dễ dàng chấp nhận chuyện "một cái nếu", phải không các bạn.

Chúng ta thường sẽ dùng kiểu, nếu... rồi nhưng mà nếu... nhiều khi kéo dài đến vô tận!

Trong Excel cũng vậy. Giả sử, chúng ta xếp loại học tập, nếu điểm trung bình (ĐTB) lớn hơn 9 thì giỏi, vậy ĐTB nhỏ hơn 9 thì dở? Chưa, ĐTB nhỏ hơn 9 nhưng lớn hơn 7 thì khá cái đã, rồi ĐTB nhỏ hơn 7 nhưng chưa bị điểm 5 thì trung bình, chỉ khi nào ĐTB nhỏ hơn 4 thì mới gọi là dở (cái này tôi chỉ ví dụ thôi, các bạn đừng sử dụng để xếp loại nhé).

Khi đó, chúng ta sẽ dùng những hàm IF() lồng nhau, IF() này nằm trong IF() kia. Sau này chúng ta sẽ học cách ghép thêm nhiều điều kiện khác vào nữa.

Ví dụ, tôi lấy lại ví dụ đã nói ở bài trước:

```
=IF(A1 >= 1000, "Big!", "Not big")
```

Bi giờ thêm chút, A1 lớn hơn 1000 là "big" rồi, nhưng chẳng lẽ 10000 thì cũng chỉ là "big" ? Có lẽ nên tặng thêm một danh hiệu cao hơn:

```
=IF(A1 >= 1000, IF(A1 >= 10000, "Really big!!", Big!), "Not big")
```

Hoặc là, đồng ý rằng <1000 là "Not big", nhưng nó khác "Small" chứ (tui không lớn, chưa chắc tui nhỏ), vậy chúng ta thêm một định nghĩa "Small" thử xem:

```
=IF(A1 >= 1000, "Big!", IF(A1 < 100, "Small", "Not big"))
```

Bạn để ý nhé, ở đây tôi đặt cái IF "con" không giống như ở trên, sao cũng, miễn là đừng sai cú pháp của IF().

Và nếu thích, bạn có thể ghép tất cả lại:

```
=IF(A1 >= 1000, IF(A1 >= 10000, "Really big!!", Big!), IF(A1 < 100, "Small", "Not big"))
```

Chỉ cần một lưu ý, là những dấu đóng mở ngoặc đơn. Nếu bạn đóng và mở không đúng lúc hoặc không đủ, Excel sẽ không hiểu, hoặc là cho các bạn kết quả sai đấy.

Hàm IFERROR

Trong quá trình thao tác với bảng tính, không ít lần chúng ta gặp lỗi, và cũng khó mà tránh được lỗi. Ví dụ, một công thức đơn giản thôi =A/B có thể gây lỗi #DIV/0! nếu như B bằng 0, hoặc gây lỗi #NAME? nếu A hoặc B không tồn tại, gây lỗi #REF! nếu có ô nào đó liên kết với A hoặc B bị xóa đi...

Tuy nhiên, đôi lúc chúng ta lại cần phải lợi dụng chính những cái lỗi này, ví dụ sẽ đặt ra một tình huống: nếu có lỗi thì làm gì đó... Gọi nôm na là BÃY LỖI.

Có lẽ vì vậy mà hàm này có hai chữ đầu là IF; IFERROR = nếu xảy ra lỗi (thì)...

MS Excel 2003 trở về trước có hàm ISERROR(value), với value là một biểu thức.

Nếu biểu thức này gặp lỗi, ISERROR() sẽ trả về giá trị TRUE, còn nếu biểu thức không có lỗi, ISERROR() trả về giá trị FALSE.

Và chúng ta thường dùng ISERROR() kèm với IF:

=IF(ISERROR(expression), ErrorResult, expression)

Nếu như biểu thức (expression) có lỗi, công thức trên sẽ lấy giá trị ErrorResult (một ô rỗng, hoặc một thông báo lỗi, v.v.), ngược lại, sẽ lấy chính giá trị biểu thức đó.

Ví dụ: =IF(ISERROR(A/B), "", A/B)

Cái bất tiện khi phải dùng vừa IF() vừa ISERROR() là chúng ta phải nhập cái biểu thức hai lần: một lần trong hàm ISERROR() và một lần ở tham số value_is_False của IF()

Có thể cái bất tiện vừa nói trên không đáng kể, tuy nhiên cách sử dụng này làm cho công thức của chúng ta trở nên khó dùng hơn,

bởi vì nếu thay cái biểu thức(expression), thì chúng ta phải thay đổi nguyên cả công thức.

Excel 2010 dường như hiểu được sự bất tiện đó, nên đã gộp hai hàm IF() và ISERROR lại thành một, đó là IFERROR()

Cú pháp: IFERROR(value, value_if_error)

_____ value: Biểu thức có thể sẽ gây ra lỗi

_____ value_if_error: kết quả trả về nếu value gây ra lỗi

Nếu biểu thức value không gây lỗi, IFERROR() sẽ lấy biểu thức đó, còn nếu nó có lỗi thì lấy cái biểu thức value_if_error.

Ví dụ, công thức =IF(ISERROR(A/B), "", A/B) nếu dùng IFERROR() thì sẽ là =IFERROR(A/B, "")

Bạn thấy đấy, IFERROR() ngắn gọn và dễ hiểu hơn nhiều.

=NOT (logical) : Đảo ngược giá trị của các đối số

Hàm OR

OR có nghĩa là HOẶC. Dùng hàm này khi muốn nói đến cái này hoặc cái này hay cái kia... cái nào cũng được, miễn là phải có ít nhất 1 cái!

Cú pháp: OR(logical1 [, logical2] [, logical3]...)

logical: Những biểu thức sẽ được xét xem đúng (TRUE) hay sai (FALSE)

Nếu tất cả các biểu thức đều sai, hàm OR() sẽ trả về giá trị FALSE, và chỉ cần 1 trong các biểu thức đúng, hàm OR() sẽ trả về giá trị TRUE.

Giống như hàm AND(), bạn có thể dùng hàm OR() bất cứ chỗ nào bạn muốn, nhưng thường thì hàm OR() hay được dùng chung với hàm IF().

Ví dụ:

=IF(OR(B2 > 0, C2 > 0), "1000", "No bonus")

Nếu giá trị ở B2 hoặc ở C2 lớn hơn 0 (tức là chỉ cần 1 trong 2 ô lớn hơn 0), thì (thường) 1.000, còn nếu cả 2 ô B2 hoặc C2 đều nhỏ hơn 0, thì không thưởng chi cả.

=TRUE(): Có thể nhập trực tiếp TRUE vào trong công thức, Excel sẽ hiểu đó là một biểu thức có giá trị TRUE mà không cần dùng đến cú pháp của hàm này

Bỏ qua những ô bị lỗi khi chạy công thức

Ví dụ: Cột Gross Margin (cột D) của bảng tính dưới đây có chứa một số ô gặp lỗi chia cho 0 (#DIV/0!), do bên cột C có những ô trống.

Để tính trung bình cộng của cột D, kể những ô có lỗi #DIV/0!, phải dùng công thức mảng như sau:

```
{=AVERAGE(IF(ISERROR(D3:D12), "", D3:D12))}
```

(nghĩa là nếu gặp những ô có lỗi thì coi như nó bằng rỗng)

Xác định tên của cột (Determining the Column Letter)

Trong Excel có hàm COLUMN(), cho ra kết quả là số của cột (ví dụ, gõ hàm này trong cột B thì kết quả sẽ là 2).

Nhưng đôi khi bạn muốn kết quả là tên của cột chứ không muốn đó là con số (B chứ không phải là 2), thì làm sao?

Đây là một vấn đề đòi hỏi sự khôn khéo một chút, vì tên cột trong bảng tính chạy từ A đến Z, từ AA đến AZ... và cho tới tận cùng là XFD (!)

Có một hàm giúp chúng ta tìm địa chỉ tuyệt đối của một cell, đó là hàm CELL("address"), ví dụ \$A\$2, hoặc \$B\$10...

Hàm CELL(info_type [,reference])

Với info_type là một tham số đã được định nghĩa (sẽ nói kỹ hơn trong những bài sau)

Và reference là cell mà bạn chỉ định, nếu bỏ trống thì Excel sẽ lấy ngay cái cell có chứa công thức CELL().

Trong bài này, để tìm địa chỉ tuyệt đối của một cell, chúng ta sẽ dùng công thức CELL() với info_type là "address"

Tinh ý một chút, ta thấy tên của cột chính là những chữ cái nằm giữa hai dấu dollar (\$) trong cái địa chỉ tuyệt đối này.

Bắt đầu làm nhé: dùng hàm MID() trích ra chữ cái từ vị trí thứ 2 trong địa chỉ tuyệt đối của cell:

```
=MID(CELL("Address"), 2, num_chars)
```

Cái khó là cái num_chars này đây, vì tên cột thì có thể là 1, 2, hoặc 3 ký tự (ví dụ: A, AA hoặc AAA). Vận dụng hàm FIND thôi:

```
FIND("$", CELL("address",A2), 3) - 2
```

Giải thích chút nhé: Dùng hàm FIND(), tìm vị trí của dấu \$ trong cái địa chỉ tuyệt đối của cell, và bắt đầu tìm từ vị trí thứ 3 trong cái địa chỉ này.

Tại sao phải trừ đi 2? Công thức trên sẽ chỉ ra vị trí (là một con số) của dấu \$ thứ hai trong địa chỉ tuyệt đối của cell, tức là cái dấu \$ phía sau tên cột,

phải trừ đi 2 tức là trừ bớt đi 2 cái \$, lúc này kết quả sẽ chính là số ký tự của tên cột (1 chữ, 2 chữ hoặc 3 chữ)

Bây giờ, công thức hoàn chỉnh sẽ như sau:

```
=MID(CELL("Address"), 2, FIND("$", CELL("address"), 3) - 2)
```

Công thức này áp dụng cho chính cell chứa công thức.

Nếu muốn tìm tên cột tại một cell nào đó, bạn chỉ việc thêm địa chỉ (hoặc một cái gì đó tham chiếu đến địa chỉ này) của cell muốn tìm vào phía sau cái "address"

Ví dụ, muốn tìm tên của cell AGH68, bạn gõ:

```
=MID(CELL("Address", AGH68), 2, FIND("$", CELL("address", AGH68), 3) - 2)  
→ AGH
```

Lập mã số tự động

Có nhiều danh mục khách hàng hoặc danh mục hàng hóa được lập mã số bằng cách dùng vài ký tự đầu của tên khách hàng (hoặc tên hàng hóa) kết hợp với một con số.

Cách đặt mã số tự động như vậy rất dễ làm trong Excel, bằng cách sử dụng những hàm xử lý văn bản và chuỗi.

Giả sử danh mục của chúng ta nằm ở cột A và bắt đầu tại cell A2.

Trước tiên, chúng ta hãy tách 3 ký tự đầu tiên của danh mục và định dạng cho nó thành những chữ in hoa, bằng công thức:

```
UPPER(LEFT(A2, 3))
```

Tiếp theo, tận dụng chính số của những hàng trong bảng tính để lập mã số tự động:
ROW(A2),

và định dạng sao cho những con số này luôn có 4 chữ số, bằng công thức:

```
TEXT(ROW(A2), "0000")
```

Và đây là công thức hoàn chỉnh:

```
=UPPER(LEFT(A2, 3)) & TEXT(ROW(A2), "0000")
```

Hàm BETADIST()

Trả về giá trị của hàm tính mật độ phân phối xác suất tích lũy beta.

Thông thường hàm này được dùng để nghiên cứu sự biến thiên về phần trăm các mẫu, ví dụ như khoảng thời gian mà người ta dùng để xem TV trong một ngày chẳng hạn.

Cú pháp: = BETADIST(x, alpha, beta, A, B)

x : Giá trị giữa A và B, dùng để tính mật độ hàm.

alpha & beta : Tham số của phân phối.

A : Cận dưới của khoảng x, mặc định là 0.

B : Cận trên của khoảng x, mặc định là 1.

Lưu ý:

* Nếu có bất kỳ đối số nào không phải là số, BETADIST() trả về giá trị lỗi #VALUE!

* Nếu $\alpha \leq 0$ hay $\beta \leq 0$, BETADIST() trả về giá trị lỗi #NUM!

* Nếu $x < A$, $x > B$ hay $A = B$, BETADIST() trả về giá trị lỗi #NUM!

* Nếu bỏ qua A và B, nghĩa là mặc định $A = 0$ và $B = 1$, BETADIST() sẽ sử dụng phân phối tích lũy beta chuẩn hóa.

Hàm BETAINV()

Trả về nghịch đảo của hàm tính mật độ phân phối xác suất tích lũy beta.

Nghĩa là nếu $\text{xác suất} = \text{BETADIST}(x, \dots)$ thì $x = \text{BETAINV}(\text{xác suất}, \dots)$

Thường dùng trong việc lên kế hoạch dự án, để mô phỏng số lần mở rộng xác suất, biết trước thời gian bổ sung kỳ vọng và độ biến đổi.

Cú pháp: = BETAINV(probability, alpha, beta, A, B)

Probability : Xác suất của biến cố x trong phân phối xác suất tích lũy beta.

alpha & beta : Tham số của phân phối.

A : Cận dưới của khoảng x, mặc định là 0.

B : Cận trên của khoảng x, mặc định là 1.

Lưu ý:

- * Nếu có bất kỳ đối số nào không phải là số, BETAINV() trả về giá trị lỗi #VALUE!
- * Nếu $\alpha \leq 0$ hay $\beta \leq 0$, BETAINV() trả về giá trị lỗi #NUM!
- * Nếu $\text{probability} \leq 0$ hay $\text{probability} > 1$, BETAINV() trả về giá trị lỗi #NUM!
- * Nếu bỏ qua A và B, nghĩa là mặc định $A = 0$ và $B = 1$, BETAINV() sẽ sử dụng phân phối tích lũy beta chuẩn hóa.
- * BETAINV() sử dụng phương pháp lặp khi tính mật độ phân phối. Với probability cho trước, BETAINV() lặp cho tới khi kết quả chính xác trong khoảng ± 0.0000003 . Nếu BETAINV() không hội tụ sau 100 lần lặp, nó sẽ trả về giá trị lỗi #NA!

Ví dụ:

BETAINV(0.6854706, 8, 10, 1, 3) = 2

Hàm BINOMDIST()

Trả về xác suất của những lần thử thành công của phân phối nhị phân.

BINOMDIST() thường được dùng trong các bài toán có số lượng cố định các phép thử, khi kết quả của các phép thử chỉ là thành công hay thất bại, khi các phép thử là độc lập, và khi xác xuất thành công là không đổi qua các cuộc thử nghiệm.

Ví dụ, có thể dùng BINOMDIST() để tính xác suất khoảng hai phần ba đứa trẻ được sinh ra là bé trai.

Cú pháp: = BINOMDIST(number_s, trials, probability_s, cumulative)

Number_s : Số lần thử thành công trong các phép thử.

Trials : Số lần thử.

Probability_s : Xác suất thành công của mỗi phép thử.

Cumulative : Một giá trị logic để xác định hàm tính xác suất.

= 1 (TRUE) : BINOMDIST() trả về hàm tính xác suất tích lũy, là xác suất có số lần thành công number_s lớn nhất.

= 0 (FALSE) : BINOMDIST() trả về hàm tính xác suất điểm (hay là hàm khối lượng xác suất), là xác suất mà số lần thành công là number_s.

Lưu ý:

* Nếu number_s và trials là số thập phân, chúng sẽ được cắt bỏ phần lẻ để trở thành số nguyên.

* Nếu number_s, trials hay probability_s không phải là số, BINOMDIST() trả về giá trị lỗi #VALUE!

* Nếu number_s < 0 hay number_s > trials, BINOMDIST() trả về giá trị lỗi #NUM!

* Nếu $\text{probability}_s < 0$ hay $\text{probability}_s > 1$, BINOMDIST() trả về giá trị lỗi #NUM!

Ví dụ:

$\text{BINOMDIST}(6, 10, 0.5, 0) = 0.2050781$

$\text{BINOMDIST}(6, 10, 0.5, 1) = 0.828125$

Hàm CHIDIST()

Trả về xác suất một phía của phân phối chi-squared.

Phân phối chi-squared kết hợp với phép thử chi-squared dùng để so sánh các giá trị quan sát với các giá trị kỳ vọng.

Ví dụ, một thí nghiệm về di truyền có thể giả thiết rằng thế hệ kế tiếp của các cây trồng sẽ thừa hưởng một tập hợp các màu sắc nào đó; bằng cách so sánh các giá trị quan sát được với các giá trị kỳ vọng, có thể thấy được giả thiết ban đầu là đúng hay sai.

Cú pháp: = CHIDIST(x, degrees_freedom)

x : Giá trị dùng để tính phân phối.

degrees_freedom : Số bậc tự do.

Lưu ý:

* Nếu các đối số không phải là số, CHIDIST() trả về giá trị lỗi #VALUE!

* Nếu $x < 0$, CHIDIST() trả về giá trị lỗi #NUM!

* Nếu degrees_freedom không phải là số nguyên, phần thập phân của nó sẽ bị cắt bỏ để trở thành số nguyên.

* Nếu $\text{degrees_freedom} < 1$ hay $\text{degrees_freedom} > 10^{10}$, CHIDIST() trả về giá trị lỗi #NUM!

* CHIDIST() được tính toán theo công thức: $\text{CHIDIST} = P(X > x)$, với X là biến ngẫu nhiên chi-squared.

Ví dụ:

$\text{CHIDIST}(18.307, 10) = 0.050001$

Hàm NORMINV()

Trả về nghịch đảo của phân phối tích lũy chuẩn.

Cú pháp: = NORMINV(probability, mean, standard_dev)

probability : Xác suất ứng với phân phối chuẩn

mean : Giá trị trung bình cộng của phân phối

standard_dev : Độ lệch chuẩn của phân phối

Lưu ý:

* Nếu có bất kỳ đối số nào không phải là số, NORMINV() sẽ báo lỗi #VALUE!

* Nếu probability nhỏ hơn 0 hoặc lớn hơn 1, NORMINV() sẽ báo lỗi #NUM!

* Nếu standard_dev nhỏ hơn hoặc bằng 0, NORMDINV() sẽ báo lỗi #NUM!

* Nếu mean = 0 và standard_dev = 1, NORMINV() sẽ dùng phân bố chuẩn.

* NORMINV() sử dụng phương pháp lặp đi lặp lại để tính hàm. Nếu NORMINV() không hội tụ sau 100 lần lặp, hàm sẽ báo lỗi #NA!