

Funzioni @ dalla A alla Z

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

A

@@(posizione)

Consente di ottenere indirettamente il contenuto della cella specificata in *posizione*.

@360GIORNI(data_inizio;data_fine)

Calcola il numero di giorni che intercorrono tra due date, sulla base di un anno di 360 giorni, secondo gli standard del mercato mobiliare statunitense.

@ACOS(x)

Calcola l'arcocoseno (inverso del coseno) utilizzando il coseno x di un angolo.

@ACOSH(x)

Calcola l'arcocoseno (inverso del coseno) iperbolico utilizzando il coseno iperbolico x di un angolo.

@ACOT(x)

Calcola l'arcocotangente (inverso della cotangente) utilizzando la cotangente x di un angolo.

@ACOTH(x)

Calcola l'arcocotangente (inverso della cotangente) iperbolico utilizzando la cotangente iperbolica x di un angolo.

@ACSC(x)

Calcola l'arcocosecante (inverso della cosecante) utilizzando la cosecante x di un angolo.

@ACSCH(x)

Calcola l'arcocosecante (inverso della cosecante) iperbolico utilizzando la cosecante iperbolica x di un angolo.

@ADESSO

Calcola il valore data (parte intera) e il valore orario (parte decimale) corrispondente alla data e all'ora correnti dell'orologio del computer.

@ALTROMESE(data_inizio;mesi;[giorno_del_mese];[base])

Calcola il valore data della data che cade un dato numero di *mesi* prima o dopo la *data_inizio*.

@AMMORT(stima;zona)

Calcola il tasso di rendimento interno (profitto) per una serie di movimenti di cassa generati da un investimento.

@AMMORTMODIF(zona;tasso_finanziario;tasso_reinvestimento;[tipo])

Calcola il tasso di rendimento interno modificato (profitto) per una serie di movimenti di cassa generati da un investimento.

@ANNO(valore_data;tipo)

Estrae l'anno, ossia un numero intero compreso tra 1 (1 gennaio 1900) e 2958465 (31 dicembre 9999) da *valore_data*.

@AQC(costo;realizzo;vita)

Calcola la quota di deprezzamento di un bene secondo il metodo di ammortamento a quote costanti per un solo periodo. Il bene ha un *costo* iniziale, una *vita* utile prevista e un valore finale di *realizzo*.

@AQP(costo;realizzo;vita;periodo)

Calcola la quota detraibile di un bene, utilizzando il metodo di ammortamento a quote fisse decrescenti.

@ARROTDIFETTO(x;n;[direzione])

arrotonda per difetto il valore *x* al più vicino multiplo della potenza di 10 specificata da *n*.

@ARROTECCESSO(x;n;[direzione])

Arrotonda per eccesso il valore *x* al più vicino multiplo della potenza di 10 specificata da *n*.

@ARROT(x;n)

Arrotonda il valore *x* al più vicino multiplo della potenza di 10 specificata da *n*.

@ARROTM(x;multiplo;[direzione])

Arrotonda il valore *x* al più vicino *multiplo*.

@ASEC(x)

Calcola l'arcosecante (inverso della secante) utilizzando la secante *x* di un angolo.

@ASECH(x)

Calcola l'arcosecante (inverso della secante) iperbolica utilizzando la secante iperbolica *x* di un angolo.

@ASEN(x)

Calcola l'arcoseno (inverso del seno) utilizzando il seno *x* di un angolo.

@ASENH(x)

Calcola l'arcoseno (inverso del seno) iperbolico utilizzando il seno iperbolico *x* di un angolo.

@ASIMMETRIA(zona;[tipo])

Calcola l'asimmetria dei valori di una *zona*.

@ASS(x)

Calcola il valore assoluto (positivo) di *x*.

@ATAN2(x;y)

Calcola l'arcotangente utilizzando la tangente *y/x* di un angolo. Il risultato di @ATAN2 è un angolo espresso in radianti, compreso tra $-\pi$ e π , a seconda del segno di *x* e di *y*.

@ATAN(x)

Calcola l'arcotangente (inverso della tangente) utilizzando la tangente *x* di un angolo. Il risultato di @ATAN è un angolo espresso in radianti, compreso tra $-\pi/2$ e $\pi/2$.

@ATANH(x)

Calcola l'arcotangente (inverso della tangente) iperbolica utilizzando la tangente iperbolica *x*.

B

@BESSELI(x;n)

Calcola la funzione di Bessel modificata $I_n(x)$.

@BESSELJ(x;n)

Calcola la funzione di Bessel $J_n(x)$.

@BESSELK(x;n)

Calcola la funzione di Bessel $K_n(x)$.

@BESSELY(x;n)

Calcola la funzione di Bessel $Y_n(x)$, nota anche come funzione di Neumann.

@BETA(z;w)

Calcola la funzione beta.

@BETAI(a;b;x)

Calcola la funzione beta incompleta.

@BINOMIALE(*prove;successi;probabilità;[tipo]*)

Calcola la funzione generatrice della probabilità binomiale oppure la distribuzione binomiale cumulativa.

C

@CAPATTUALE(*valore_futuro;interesse;durata;[frequenza]*)

Genera il valore attuale di una somma complessiva dariceversi in un dato numero di periodi futuri (*durata*) e attualizzata a un dato tasso di *interesse*.

@CAPFUT(*capitale;interesse;durata;[frequenza]*)

Genera il valore futuro di una somma complessiva investita a un certo tasso di interesse per un certo numero di periodi.

@CAPITRATA(*capitale;interesse;durata;data_inizio;[data_fine];[tipo];[valore_finale]*)

Calcola la porzione costituita dal capitale della rata periodica di un prestito (*capitale*) dato il tasso di *interesse* per un certo numero di periodi di pagamento (*durata*).

@CAR(*x*)

Genera il carattere del set di caratteri multilingue Lotus, LMBCS (Lotus Multibyte Character Set) corrispondente al numero *x*.

@CAS

riporta un valore casuale compreso tra 0 e 1 con fino a 15 cifre significative. Ogni volta che 1-2-3 esegue un ricalcolo, la funzione @CAS produce un nuovo valore casuale.

@CELLA(*attributo;zona*)

Fornisce informazioni sulla prima cella di una *zona*.

@CHIARIF(*stringa*)

Elimina dalla *stringa* i caratteri che non vengono stampati.

@CODICE(*stringa*)

Fornisce il codice del set di caratteri multilingue Lotus, LMBCS (Lotus Multibyte Character Set), che corrisponde al primo carattere di *stringa*.

@COLLDATI(*nome_applicazione;nome_file;nome_elemento;[formato];[righe_max];[colonne_max];[fogli_max]*)

Crea un collegamento OLE con dei dati.

@COLONNE(*zona*)

Conta le colonne comprese in *zona*.

@COMBIN(*n;r*)

Calcola il coefficiente binomiale per *n* e *r*. Il coefficiente binomiale è il numero di modi in cui *r* può essere selezionato da *n*, a prescindere dall'ordine.

@CONTAPURODB(*zona_immissione;campo;[condizioni]*)

Conta le celle che contengono valori in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@CONTAPURO(*lista*)

Conta le celle di una *lista* di zone, escludendo quelle che contengono etichette.

@CONTDB(*zona_immissione;campo;[condizioni]*)

Conta le celle non vuote di un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@CONVERTDATA(*data;tipo_origine;tipo_destinazione*)

Converte una data del calendario Hijri (arabo), Farsi (iraniano) o ebraico nella data corrispondente del calendario gregoriano o viceversa. Solo per le versioni bidirezionali di 1-2-3.

@CONVERTIREF(*riferimento*)

Converte le lettere delle colonne o dei fogli da A a IV in numeri da 1 a 256 e i numeri da 1 a 256 nelle lettere corrispondenti.

@COORD(*foglio;colonna;riga;assoluto*)

Crea un riferimento di cella usando i valori *foglio*, *colonna* e *riga*.

@CORREL(*zona1;zona2*)

Calcola il coefficiente di correlazione tra i valori contenuti nella *zona1* e nella *zona2*.

@CORRISP(*contenuto_cell;zona;[tipo]*)

Fornisce la posizione relativa della cella di *zona* il cui contenuto corrisponde a *contenuto_cell*.

@COS(*x*)

Calcola il coseno dell'angolo *x*. L'angolo deve essere espresso in radianti.

@COSH(x)

Calcola il coseno iperbolico dell'angolo x . Il risultato della funzione @COSH è un valore maggiore o uguale a 1. L'angolo deve essere espresso in radianti.

@COT(x)

Calcola la cotangente dell'angolo x . L'angolo deve essere espresso in radianti.

@COTH(x)

Calcola la cotangente iperbolica dell'angolo x . L'angolo deve essere espresso in radianti.

@COUNT(lista)

Conta le celle non vuote in una *lista* di zone.

@COV(zona1;zona2;[tipo])

Calcola la covarianza della popolazione o di un campione di valori contenuti in *zona1* e *zona2*.

@CRITBINOMIALE(prove;probabilità;alfa)

Genera il numero intero più grande per cui la distribuzione binomiale cumulativa è minore o uguale ad *alfa*.

@CSC(x)

Calcola la cosecante dell'angolo x . L'angolo deve essere espresso in radianti.

@CSCH(x)

Calcola la cosecante iperbolica dell'angolo x . L'angolo deve essere espresso in radianti.

@CTERM(interesse;valore_futuro;valore_attuale)

Calcola il numero di periodi di capitalizzazione necessari affinché un investimento (*valore_attuale*) raggiunga un *valore_futuro* a un dato tasso di *interesse* fisso.

@CURSORE(attributo)

Fornisce informazioni sulla cella attiva.

@CURTOSI(zona;[tipo])

Calcola la curtosi dei valori compresi nella *zona*.

D

@D360(data_inizio;data_fine)

Calcola il numero di giorni che intercorrono tra due date, sulla base di un anno di 360 giorni.

@DAGG(nome)

Controlla il *nome* per verificare se si tratta di una applicazione aggiuntiva attualmente in memoria e genera il valore 1 (vero) oppure 0 (falso).

@DATA(anno;mese;giorno)

Calcola il valore data per l'*anno*, il *mese* e il *giorno* specificati.

@DATAT(valore_data)

Converte *valore_data* nel giorno della settimana secondo il calendario thailandese.

@DCOMPRESO(valore;limite1;limite2;[estremi])

Verifica se *valore* è compreso tra *limite1* e *limite2*. @DCOMPRESO genera 1 (vero) se *valore* cade tra *limite1* e *limite2* oppure 0 (falso) in caso contrario.

@DDB(costo;realizzo;vita;periodo)

Calcola la quota detraibile di un bene, secondo il metodo di ammortamento a doppie quote proporzionali ai valori residui.

@DECILE(x;zona)

Calcola il decile specificato.

@DECIMALE(esadecimale)

Converte un valore esadecimale nel suo equivalente decimale preceduto dal segno.

@DERR(x)

Verifica se x ha valore ERR e produce 1 (vero) o 0 (falso).

@DESTRAB(stringa;n)

Riporta gli ultimi n byte di *stringa*.

@DESTRA(stringa;n)

Riporta gli ultimi n caratteri di *stringa*.

@DESTRAT(stringa;n)

Fornisce gli ultimi n caratteri thailandesi logici di *stringa*. Questa funzione si applica alle stringhe di caratteri

tailandesi.

@DEVQ(lista)

Calcola la somma delle deviazioni quadratiche dei valori compresi in una *lista*, ricavandola dalla media dei valori stessi.

@DFCAGG(nome)

Controlla il *nome* per verificare se si tratta di una funzione LotusScript globale aggiuntiva definita e genera il valore 1 (vero) oppure 0 (falso).

@DFILE(nome_file;[tipo])

Verifica se *nome_file* è un file in memoria o sul disco e produce 1 (vero) o 0 (falso).

@DIFDATA(data_inizio;data_fine;formato)

Calcola il numero di anni, mesi o giorni che intercorrono tra *data_inizio* e *data_fine*.

@DISPARI(x)

Arrotonda il valore *x* al successivo numero dispari intero.

@DISTRIBCHI2(x;gradi_libertà;[tipo])

Calcola la distribuzione chi-quadrato.

@DISTRIBF(x;gradi_libertà1;gradi_libertà2;[tipo])

Calcola la distribuzione *F*.

@DISTRIBT(x;gradi_libertà;[tipo];[code])

Calcola la distribuzione *t* di Student.

@DMACRO(nome)

controlla il *nome* per verificare se si tratta di una subroutine LotusScript globale aggiuntiva definita e produce 1 (vero) oppure 0 (falso).

@DND(x)

Verifica se *x* ha il valore ND e produce 1 (vero) oppure 0 (falso).

@DNUMERO(x)

Verifica se *x* è un valore, ND, ERR o una cella vuota e produce 1 (vero) oppure 0 (falso).

@DSTRINGA(x)

Verifica se *x* è un testo o una cella contenente un'etichetta o una formula che produce un'etichetta e produce 1 (vero) oppure 0 (falso).

@DURATA(liquidazione;scadenza;cedola;rendimento;[frequenza];[base])

Calcola la durata annua di un titolo con pagamento periodico degli interessi.

@DURATAM(liquidazione;scadenza;cedola;rendimento;[frequenza];[aase])

Calcola la durata annua modificata di un titolo con pagamento periodico degli interessi.

@DVUOTA(posizione)

verifica se in *posizione* si trova una cella vuota e produce 1 (vero) oppure 0 (falso).

@DZONA(zona)

Verifica se *zona* è un nome di zona definito o un indirizzo di zona valido e produce 1 (vero) oppure 0 (falso).

E

@ERR

Genera il valore ERR.

@ESA(x)

Converte un numero decimale nell'equivalente esadecimale.

@ESATTO(stringa1;stringa2)

Genera 1 (vero) se *stringa1* e *stringa2* presentano un'equivalenza totale; se invece le due serie non sono identiche genera 0 (falso).

@ESP2(x)

Calcola il valore della costante *e* (circa 2,718282) elevato alla potenza $(-x^2)$.

@ESP(x)

Calcola il valore della costante *e* (approssimativamente 2,718282) elevata alla potenza *x*.

F

@FALSO

Genera il valore logico 0 (falso).

@FATTORIALE(*n*)

Calcola il fattoriale di *n*.

@FATTORIALELOGN(*n*)

Calcola il logaritmo naturale del fattoriale di *n*.

@FERRC(*x*)

Calcola la funzione d'errore complementare integrata tra *x* e ∞ (infinito).

@FERRD(*x*)

Calcola la derivata della funzione d'errore.

@FERR(*limite_inferiore*;[*limite_superiore*])

Calcola la funzione di errore integrata tra il *limite_inferiore* e il *limite_superiore*.

@FISSALUNGH(*stringa*;lunghezza;[*allineamento*])

Produce un'etichetta della *lunghezza* in caratteri indicata.

@FOGLI(*zona*)

Conta il numero di fogli in *zona*.

@FULLP(*stringa*)

Converte i caratteri a un solo byte (ASCII) di *stringa* nei corrispondenti caratteri giapponesi a due byte.

G

@GAMMA(*x*)

Calcola la funzione gamma.

@GAMMAI(*a*;x;[*complement*])

Calcola la funzione gamma incompleta.

@GAMMALOGN(*x*)

Calcola il logaritmo naturale della funzione gamma.

@GIORNI(*data_inizio*,*data_fine*,[*base*])

Calcola il numero di giorni che intercorrono tra due date, utilizzando la *base* specificata per il conteggio dei giorni.

@GIORNILAVNETTI(*data_inizio*,*data_fine*;[*festività*];[*fine_settimana*])

Calcola il numero di giorni che intercorrono tra *data_inizio* e *data_fine*, escludendo i fine settimana e le festività.

@GIORNO(*valore_data*)

Estrae il giorno del mese, da 1 a 31, da *valore_data*.

@GLAVOR(*data_inizio*;giorni;[*festività*];[*fine_settimana*])

Calcola il valore data corrispondente alla data che cade il numero di *giorni* lavorativi specificato prima o dopo *data_inizio*.

@GRADIRAD(*gradi*)

Converte i *gradi* in radianti.

@GRANDE(*zona*;n)

Individua l'*ennesimo* valore più alto della *zona*.

@GSETT(*data*)

Estrae il giorno della settimana dalla *data*, visualizzandolo sotto forma di numero intero compreso tra 0 (lunedì) e 6 (domenica).

H

@HALFP(*stringa*)

Converte i caratteri giapponesi a due byte di *stringa* nei corrispondenti caratteri (ASCII) a un solo byte.

I

@IND(*zona*;colonna;*riga*;[*foglio*])

Riporta il contenuto della cella situata nel punto di intersezione della *colonna*, *riga* e *foglio* di una zona.

@INDEX(*zona*;intest_colonna;intest_riga;[intest_foglio])

Riporta il contenuto di una cella situata all'intersezione identificata da *intest_colonna*, *intest_riga* e *intest_foglio*.

@INFODATA(*data*;attributo)

Fornisce informazioni sulla *data*.

@INFO(*attributo*)

Riporta informazioni riguardanti la sessione di 1-2-3 in corso.

@INFOSCENARIO(*opzione;nome;[autore]*)

Fornisce informazioni su un gruppo di versioni.

@INFOVERSIONE(*opzione;zona_versione;nome[;autore]*)

Fornisce informazioni su una versione.

@INT(*x*)

Riporta la parte intera di *x*.

@INTRATA(*capitale;interesse;durata;data_inizio;[data_fine];[tipo];[valore_futuro]*)

Calcola la porzione costituita dagli interessi cumulativi della rata periodica da pagare per un prestito, per un certo numero di periodi di pagamento (*durata*).

@IPAGAM(*capitale;interesse;durata;rata*)

Calcola la porzione costituita dagli interessi di una rata costante.

@IPS(*capitale;interesse;durata;rata*)

Calcola la porzione costituita dagli interessi di una rata, per cui la parte di capitale rimane invariata in ogni periodo.

J

K

L

@LIVELLO(*elemento;zona;[ordine]*)

Calcola la grandezza di un valore di una *zona* in relazione agli altri valori contenuti nella stessa *zona*.

@LIVELLOPERC(*x;zona;[cifre_decimali]*)

Individua il percentile di *x* tra i valori compresi in una *zona*.

@LOG(*x*)

Calcola il logaritmo decimale (in base 10) di *x*.

@LOGN(*x*)

Calcola il logaritmo naturale (in base *e*) di *x*.

@LUNGH(*stringa*)

Conta i byte di *stringa*.

@LUNGH(*stringa*)

Conta i caratteri in *stringa*.

@LUNGHT(*stringa*)

Conta un certo numero di caratteri thailandesi logici di *stringa*. Questa funzione si applica alle stringhe di caratteri thailandesi.

M

@MAIUSC(*stringa*)

Converte in maiuscole tutte le lettere di *stringa*.

@MATURATO2(*liquidazione;scadenza;cedola;[pari];[frequenza];[emissione];[first];[tipo]*)

Calcola gli interessi maturati sui titoli con pagamento periodico degli interessi, in base alle convenzioni giapponesi.

@MATURATO(*liquidazione;emissione;primo_interesse;cedola;[pari];[frequenza];[base]*)

Calcola gli interessi maturati sui titoli che prevedono pagamenti periodici degli interessi.

@MAXDB(*zona_immissione;campo;[condizioni]*)

Individua il valore più alto in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@MAX(*lista*)

Determina il valore più alto nella *lista*.

@MAXPURO(*lista*)

Determina il valore più alto nella *lista*, ignorando le celle contenenti etichette.

@MEDIAARMON(*lista*)

Calcola la media armonica dei valori compresi in una *lista*.

@MEDIADB(zona_immissione;campo:[condizioni])

Calcola la media dei valori contenuti in un *campo* di una tabella di database che soddisfano le *condizioni* specificate.

@MEDIADEV(lista)

Calcola la media delle deviazioni assolute dei valori compresi in una *lista*.

@MEDIAES(lista)

Calcola l'errore standard della media campionaria per i valori compresi nella *lista*.

@MEDIAGEOM(lista)

Calcola la media geometrica dei valori compresi in una *lista*.

@MEDIA(lista)

Calcola la media aritmetica di una *lista* di valori.

@MEDIANA(lista)

Calcola il valore mediano di una *lista*.

@MEDIANAPURA(lista)

Calcola il valore mediano di una *lista*, ignorando le celle vuote, le etichette e le formule che producono etichette.

@MEDIAPOND(zona_dati;fattori_peso:[tipo])

Calcola la media ponderata dei valori contenuti in *zona_dati*.

@MEDIAPURA(lista)

Calcola la media aritmetica di una *lista* di valori, ignorando tutte le celle contenenti etichette.

@MESE(valore_data)

Estrae il mese, ossia un valore compreso tra 1 e 12, da *valore_data*.

@MEZZOB(stringa;numero_inizio;n)

Copia *n* byte di *stringa*, a partire dal byte situato nella posizione specificata da *numero_inizio*.

@MEZZO(stringa;numero_inizio;n)

Copia *n* caratteri di *stringa* a partire dal carattere situato nella posizione specificata da *numero_inizio*.

@MEZZOT(stringa;numero_inizio;n)

Copia *n* caratteri tailandesi logici di *stringa*, a partire dalla posizione specificata di *numero_inizio*. Questa funzione si applica alle stringhe di caratteri thailandesi.

@MINDB(zona_immissione;campo:[condizioni])

Individua il valore più basso in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@MIN(lista)

Determina il valore più basso in una *lista*.

@MINPURO(lista)

Determina il valore più basso in una *lista*, ignorando tutte le celle contenenti etichette.

@MINUSC(stringa)

Converte in minuscole tutte le lettere di *stringa*.

@MINUTI(valore_orario)

Estrae i minuti, ossia un valore compreso tra 0 e 59, da *valore_orario*.

@MOD(x;y)

Calcola il resto (modulo) di x/y . Il segno del risultato è uguale a quello di *x*.

@MODULO(x;y)

Calcola il resto (modulo) di x/y . Il segno del risultato è uguale a quello di *y*.

N

@N(zona)

Riporta il contenuto della prima cella di *zona* sotto forma di valore. Se la cella contiene un'etichetta, @N produce il valore 0.

@ND

Produce il valore ND (non disponibile).

@NOMEPRO(stringa)

Converte in maiuscola la prima lettera di ogni parola della *stringa*, trasformando le restanti in minuscole.

@NOMEZONA(cella)

Riporta il nome della zona in cui è collocata la *cella*.

@NORMALE(*x*;[*media*];[*deviazione_standard*];[*tipo*])

Calcola la funzione di distribuzione normale per *x*.

@NUMEROA(*stringa_numerica*)

Converte una *stringa-numerica* in caratteri thailandesi in una stringa in caratteri numerici arabi.

@NUMEROT(*caratteri_numerici*)

Converte i *caratteri_numerici* espressi in numeri arabi in una stringa di caratteri numerici thailandesi.

@NUMPERIODI(*versamenti*;[*interesse*];[*valore_futuro*];[*tipo*];[*valore_attuale*])

Calcola il tempo necessario per effettuare una serie di *versamenti* di uguale importo del *valore_attuale* specificato fino ad accumulare un capitale di *valore_futuro*, dato un tasso d'*interesse* periodico.

O

@OGGI

Calcola il valore data che corrisponde alla data corrente del computer.

@ORA(*valore_orario*)

Estrae l'ora, ossia un valore compreso tra 0 (mezzanotte) e 23 (23:00 o 11:00 PM), da un *valore_orario*.

@ORARIO(*ora*;[*minuti*];[*secondi*])

Calcola il valore orario per l'*ora*, i *minuti* e i *secondi* specificati.

@ORDDATA(*stringa*)

Calcola il valore data della data specificata in *stringa*.

@ORDORA(*stringa*)

Calcola il valore orario dell'ora specificata nella *stringa*.

P

@PAGAM2(*capitale*;[*interesse*];[*durata*])

Calcola le rate di rimborso di un prestito (*capitale*) a un dato tasso di *interesse* per un numero specificato di periodi di pagamento (*durata*), presupponendo una rendita anticipata.

@PAGAMC(*capitale*;[*interesse*];[*durata*])

Calcola le rate per il rimborso di un prestito (*capitale*) a un dato tasso d'*interesse* per un numero specificato di periodi di pagamento (*durata*), secondo le convenzioni canadesi dei vincoli ipotecari.

@PAGAM(*capitale*;[*interesse*];[*durata*])

Calcola le rate per il rimborso di un prestito (*capitale*) a un dato tasso d'*interesse* per un numero specificato di periodi di pagamento (*durata*), con il sistema della rendita immediata.

@PAGAMENTO(*capitale*;[*interesse*];[*durata*];[*tipo*];[*valore_futuro*])

Calcola le rate per il rimborso di un prestito (*capitale*) a un dato tasso d'*interesse* per un numero specificato di periodi di pagamento (*durata*).

@PARI(*x*)

Arrotonda il valore *x* al successivo numero pari intero.

@PERCENTILE(*x*;[*zona*])

Calcola l'*x*-esimo percentile campionario tra i valori di una *zona*.

@PERMUT(*n*;[*r*])

Calcola il numero di sequenze ordinate (permutazioni) di *r* oggetti selezionabili dall'insieme totale di *n* oggetti.

@PICCOLO(*zona*;[*n*])

Individua l'*ennesimo* valore più basso nella *zona*.

@PI

Produce il valore π (calcolato a 3,14159265358979).

@POISSON(*x*;[*media_arrivi*];[*cumulativa*])

Calcola la distribuzione di Poisson.

@PREVISIONE(*x*;[*zona_y*];[*zona_x*])

Genera il valore previsto di *x* sulla base di una tendenza lineare tra i valori contenuti nella *zona_y* e nella *zona_x*.

@PREZZO2(*liquidazione*;[*scadenza*];[*cedola*];[*rendimento*];[*rimborso*];[*base*])

Calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a ¥100, in base alle convenzioni giapponesi.

@PREZZO(*liquidazione*;[*scadenza*];[*cedola*];[*rendimento*];[*rimborso*];[*frequenza*];[*base*])

Calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a \$100.

@PROD(*lista*)

Moltiplica i valori compresi in una *lista*.

Q

@QUARTILE(*x;zona*)

Genera un dato quartile.

@QUOZIENTE(*x;y*)

Calcola il risultato di x/y , fornendone però solo l'intero.

R

@RADGRADI(*radianti*)

Converte i *radianti* in gradi.

@RADQ(*x*)

Calcola la radice quadrata positiva di *x*.

@RADQPI(*x*)

Calcola la radice quadrata di $x^*\pi$.

@REGRESSIONE(*zona_x;zona_y;attributo;[calcolo]*)

Esegue una regressione lineare multipla, generando la statistica specificata.

@RENDIMENTO2(*liquidazione;scadenza;cedola;prezzo;[rimborso];[base]*)

Calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi, secondo le convenzioni giapponesi.

@RENDIMENTO(*liquidazione;scadenza;cedola;prezzo;[rimborso];[frequenza];[base]*)

Calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi.

@RICERB(*stringa_ricerca;stringa;numero_inizio*)

Calcola la posizione (in byte) della *stringa* in cui 1-2-3 trova la prima occorrenza di *stringa_ricerca*, incominciando dalla posizione indicata da *numero_inizio*.

@RICER(*stringa_ricerca;stringa;numero_inizio*)

Calcola la posizione all'interno della *stringa* in cui 1-2-3 trova la prima occorrenza di *stringa_ricerca*, iniziando dalla posizione indicata da *numero_inizio*.

@RICERT(*stringa_ricerca;stringa;colonna_inizio*)

Calcola la posizione dei caratteri thailandesi logici in *stringa* in cui 1-2-3 trova la prima occorrenza della *stringa_ricerca*, a partire dal carattere thailandese che si trova nella *colonna_inizio*. Questa funzione si applica alle stringhe di caratteri thailandesi.

@RIGHE(*zona*)

Conta il numero di righe in *zona*.

@RIPETI(*stringa;n*)

Duplica la *stringa* per il numero di volte specificato da *n*.

S

@S(*zona*)

Riporta il contenuto della prima cella di una *zona* sotto forma di etichetta.

@SCANSMAX(*lista_zone*)

Genera un riferimento assoluto alla cella che contiene il valore più alto di una serie di zone.

@SCANSMIN(*lista_zone*)

Genera un riferimento assoluto alla cella contenente il valore più piccolo di una serie di zone.

@SCANSOR(*x;zona;scarto_riga*)

Riporta il contenuto della cella di una tabella di consultazione orizzontale, che si trova all'intersezione della riga specificata e della colonna specificata da *x*.

@SCANSVER(*x;zona;scarto_colonna*)

Determina il contenuto della cella situata nella colonna specificata di una tabella di consultazione verticale.

@SCARTOC(*lista*)

Calcola la deviazione standard campionaria di una *lista* di valori.

@SCARTOCPURO(*lista*)

Calcola la deviazione standard campionaria dei valori di una *lista*, ignorando le celle contenenti etichette.

@SCARTODBC(*zona_immissione;campo;[condizioni]*)

Calcola la deviazione standard di un campione di valori contenuti in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@SCARTODB(*zona_immissione;campo;[condizioni]*)

Calcola la deviazione standard di una popolazione di valori contenuti in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@SCARTO(*lista*)

Calcola la deviazione standard di una popolazione di valori situati in *lista*.

@SCARTOPURO(*lista*)

Calcola la deviazione standard della popolazione di valori compresi in *lista*, ignorando le celle contenenti etichette.

@SCELTA(*x;lista*)

Genera l'*n*-esimo elemento di *lista*, dove *n* è il valore di *x*.

@SE(*condizione;x;y*)

Valuta la *condizione* e riporta *x* se la *condizione* è vera oppure *y* se la *condizione* è falsa.

@SEC(*x*)

Calcola la secante dell'angolo *x*.

@SECH(*x*)

Calcola la secante iperbolica dell'angolo *x*.

@SECONDI(*valore_orario*)

Estrae i secondi, ossia un numero intero compreso fra 0 e 59, da *valore_orario*.

@SEGNO(*x*)

Produce 1 se *x* è un valore positivo, 0 se *x* è 0, e -1 se *x* è un valore negativo.

@SELDB(*zona_immissione;campo;[condizioni]*)

Ricava un valore o un'etichetta da un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@SEN(*x*)

Calcola il seno dell'angolo *x*.

@SENH(*x*)

Calcola il seno iperbolico dell'angolo *x*.

@SINISTRAB(*stringa;n*)

Fornisce i primi *n* byte di *stringa*.

@SINISTRA(*stringa;n*)

Fornisce i primi *n* caratteri di *stringa*.

@SINISTRAT(*stringa;n*)

Fornisce i primi *n* caratteri thailandesi logici di *stringa*. Questa funzione si applica alle stringhe di caratteri thailandesi.

@SOMMADB(*zona_immissione;campo;[condizioni]*)

Calcola la somma dei valori situati in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@SOMMA(*lista*)

Somma i valori di *lista*.

@SOMMANEGATIVI(*lista*)

Esegue la somma dei soli valori negativi contenuti in *lista*.

@SOMMAN(*scarto;n;lista*)

Somma ogni *ennesimo* valore di *lista*, partendo da *scarto*.

@SOMMAPOSITIVI(*lista*)

Esegue la somma dei soli valori positivi contenuti in *lista*.

@SOMMAPROD(*lista*)

Moltiplica i valori situati nelle celle corrispondenti di una *lista* di zone, sommandone i prodotti.

@SOMMAQUADRATI(*lista*)

Calcola la somma dei quadrati dei valori compresi nella *lista*.

@SOMMASERIE(*x;n;m;coefficienti*)

Calcola la somma di una serie di potenze.

@SOMMAXMY2(zona1;zona2)

Sottrae i valori contenuti nella *zona2* dalle celle corrispondenti della *zona1*, eleva le differenze al quadrato per poi sommare i risultati.

@SOSTITB(stringa_originale;numero_inizio;n;stringa_sostit)

Sostituisce *n* byte di *stringa_originale* con *stringa_sostit*, a partire da *numero_inizio*.

@SOSTIT(stringa_originale;numero_inizio;n;stringa_sostit)

Sostituisce *n* caratteri di *stringa_originale* con *stringa_sostit*, a partire da *numero_inizio*.

@SOSTITI(stringa_originale;numero_inizio;n;stringa_sostit)

Sostituisce *n* caratteri logici thailandesi di *stringa_originale* con *stringa_sostit*, a partire da *numero_inizio*. Questa funzione si applica alle stringhe di caratteri thailandesi.

@STRINGADATA(data)

Converte un valore *data* nella data corrispondente sotto forma di etichetta con il formato Data Internazionale standard.

@STRINGADATAT(valore_data)

Converte *valore_data* nella corrispondente stringa che esprime la data nel formato thailandese abbreviato.

@STRINGADATATL(valore_data)

Converte il *valore_data* nella corrispondente stringa di data nel formato thailandese esteso.

@STRINGA(x;n)

Converte il valore *x* in un'etichetta, applicandole il formato specificato da *n*.

@STRINGANUM(numero;tipo)

Converte *numero* nel corrispondente testo in caratteri giapponesi, utilizzando il formato specificato in *tipo*.

@STRINGANUMT(numero)

Converte *numero* nell'equivalente stringa numerica in caratteri thailandesi.

@SYD(costo;realizzo;vita;periodo)

Calcola la quota detraibile per il deprezzamento di un bene, per un *periodo* specificato.

T

@TAN(x)

Calcola la tangente dell'angolo *x*.

@TANH(x)

Calcola la tangente iperbolica dell'angolo *x*.

@TASSO(valore_futuro;valore_attuale;durata)

Calcola il tasso d'interesse periodico necessario affinché un investimento raggiunga un *valore_futuro* nel corso dei periodi di capitalizzazione specificati in *durata*.

@TASSOINT(durata;rata;valore_attuale;[tipo];[valore_futuro];[stima])

Calcola il tasso di interesse periodico necessario affinché un investimento (*valore_attuale*) raggiunga il *valore_futuro* nel corso del periodo di capitalizzazione indicato da *durata*.

@TERM2(versamenti;interesse;valore_futuro)

Calcola il tempo necessario per accumulare un capitale di *valore_futuro*, effettuando *versamenti* periodici di uguale importo, dato un tasso d'*interesse* periodico, secondo la convenzione della rendita anticipata.

@TERM(versamenti;interesse;valore_futuro)

Calcola il tempo necessario per accumulare un capitale di *valore_futuro*, effettuando *versamenti* periodici di uguale importo, dato un tasso d'*interesse* periodico, con il sistema della rendita immediata.

@TESTCHI2(zona1;[zona2])

Esegue un test chi-quadrato di indipendenza sui dati della *zona1* o un test chi-quadrato di bontà di adattamento sui dati contenuti in *zona1* e *zona2*.

@TESTF(zona1;zona2)

Esegue un test *f*, riportando la probabilità ad essa associata.

@TESTT(zona1;zona2;[tipo];[code])

Esegue un test *t* di Student sui dati contenuti in *zona1* e *zona2*, e riporta la probabilità associata.

@TESTZ(zona1;media1;deviazione_standard1;[code];[zona2];[media2];[deviazione_standard2])

Esegue un test *z* su una o due popolazioni, riportando la probabilità associata.

@TOTGENER(*lista*)

Somma tutte le celle comprese in una *lista* contenenti la funzione @TOTPARZ.

@TOTPARZ(*lista*)

Somma i valori di una *lista*. Questa funzione Può servire per indicare quali celle dovrà sommare la funzione @TOTGENER.

@TRONCADEC(*x*;[*n*])

Tronca il valore *x* alla cifra decimale specificata da *n*.

@TRONCA(*stringa*)

Elimina eventuali spazi iniziali, finali e consecutivi dalla *stringa*.

U

@ULTIMOSCENARIO(*nome_file*)

Riporta il nome del gruppo di versioni di una cartella di lavoro visualizzato per ultimo durante la sessione di 1-2-3 in corso.

V

@VALATT2(*versamenti*; *interesse*; *durata*)

Calcola il valore attuale di un investimento, sulla base di una serie di *versamenti* di uguale importo, attualizzati a un tasso d'*interesse* periodico per la *durata* specificata, usando la convenzione della rendita anticipata.

@VALATT(*versamenti*; *interesse*; *durata*)

Calcola il valore attuale di un investimento, sulla base di una serie di *versamenti* di uguale importo, attualizzati a un tasso d'*interesse* periodico per la *durata* specificata, con il sistema della rendita immediata.

@VALATTUALE(*versamenti*; *interesse*; *durata*; [*tipo*]; [*valore_futuro*])

Determina il valore attuale di un investimento dal *valore_futuro* specificato, per una rendita immediata o anticipata.

@VALFUT2(*versamenti*; *interessi*; *durata*)

Calcola il valore futuro di un investimento sulla base di una serie di *versamenti* di uguale importo, di un tasso di *interesse* periodico e del numero di periodi specificati da *durata*, presupponendo una rendita anticipata.

@VALFUT(*versamenti*; *interesse*; *durata*)

Calcola il valore futuro di un investimento sulla base di una serie di *versamenti* di uguale importo, di un tasso d'*interesse* periodico e del numero di periodi specificati da *durata*, secondo il sistema della rendita immediata.

@VALFUTURO(*versamenti*; *interesse*; *durata*; [*tipo*]; [*valore_attuale*])

Calcola il valore futuro di un investimento del *valore_attuale*, con il sistema della rendita immediata o anticipata.

@VALORE(*stringa*)

Converte un numero immesso come *stringa* nel valore corrispondente.

@VAN(*interesse*; *zona*; [*tipo*])

Calcola il valore attuale netto di una serie di movimenti di cassa futuri (*zona*), scontati a un tasso d'*interesse* periodico costante.

@VARC(*lista*)

Calcola la varianza campionaria di una *lista* di valori.

@VARCPURA(*lista*)

Calcola la varianza campionaria di una *lista* di valori, ignorando le celle contenenti etichette.

@VARDBC(*zona_immissione*; *campo*; [*condizioni*])

Calcola la varianza di un campione di valori contenuti in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@VARDB(*zona_immissione*; *campo*; [*condizioni*])

Calcola la varianza di una popolazione di valori contenuti in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

@VAR(*lista*)

Calcola la varianza della popolazione di una *lista* di valori.

@VARPURA(*lista*)

Calcola la varianza della popolazione di una *lista* di valori, ignorando le celle contenenti etichette.

@VDB(*costo*; *realizzo*; *vita*; *periodo_iniz*; *periodo_fin*; [*tasso_ammortamento*]; [*commutatore*])

Calcola la detrazione per l'ammortamento di un bene per il periodo di tempo specificato utilizzando il metodo a

quote proporzionali con tasso variabile.

@VERO

Genera il valore logico 1 (vero).

@VERSIONEATTIVA(zona)

Riporta il nome della versione in uso nella zona.

@VERSIONEDATI(opzione;cella;zona_versione;nome;[autore])

Riporta il contenuto di una cella di una versione.

W

X

@XAMMORT(stima;flusso_di_cassa;date)

Calcola il tasso di rendimento interno per una serie di movimenti di cassa in entrata e in uscita.

@XVAN(tasso;flussi_di_cassa;date)

Calcola il valore attuale netto di una serie di flussi di cassa in uscita e in entrata.

Y

Z

@ASS

@ASS(x) calcola il valore assoluto (positivo) di x.

Argomenti

x può assumere qualsiasi valore.

NOTA

Usare -@ASS quando si vuole che il risultato della funzione @ sia negativo.

Esempi

@ASS(A5) = 25, se la cella A5 contiene il valore 25, -25 o una formula che assume come valore 25 o -25.

-@ASS(A5) = -25, se la cella A5 contiene il valore 25, -25 o una formula che assume come valore 25 o -25.

@ASS(INIZIO-FINE) = 5,6, quando INIZIO e FINE contengono una qualsiasi combinazione di numeri negativi o positivi che differiscono di 5,6 o -5,6.

@MATURATO

@MATURATO(*liquidazione*; *emissione*; *primo_interesse*; *cedola*; [*pari*]; [*frequenza*]; [*base*]) calcola gli interessi maturati sui titoli che prevedono pagamenti periodici degli interessi. @MATURATO supporta cedole a breve, a medio e a lungo periodo.

Argomenti

liquidazione è il giorno dei compensi di un titolo e consiste in un valore data. Se *liquidazione* è inferiore a *emissione*, la funzione @MATURATO assume il valore ERR.

emissione è la data di emissione o di inizio del titolo, e consiste in un valore data.

primo_interesse è la prima scadenza di pagamento degli interessi del titolo e consiste in un valore data. Se *primo_interesse* è inferiore o uguale a *emissione*, la funzione @MATURATO assume il valore ERR.

cedola è il tasso nominale annuale di un titolo e consiste in un valore positivo o uguale a zero.

pari è un argomento facoltativo che specifica il valore di parità o nominale di un titolo, cioè il capitale che deve essere pagato alla data di scadenza. *pari* è un valore positivo. Se non viene specificato l'argomento *pari*, 1-2-3 usa il valore 100.

frequenza è un argomento facoltativo che indica il numero di pagamenti delle cedole previsti in un anno, il cui valore dev'essere ricavato dalla tabella sottostante.

<i>frequenza</i>	Frequenza dei pagamenti delle cedole
1	Annuale
2	Semestrale (impostazione predefinita in caso di omissione)
4	Trimestrale
12	Mensile

base è un argomento facoltativo che indica il tipo di base per il conteggio dei giorni da usare, il cui valore dev'essere ricavato dalla tabella sottostante.

<i>base</i>	Base per il conteggio dei giorni
0	30/360 (impostazione predefinita in caso di omissione)
1	Effettivo/effettivo
2	Effettivo/360
3	Effettivo/365
4	Europeo 30/360

Non è consentito usare un argomento facoltativo senza specificare quelli che lo precedono.

Esempi

Un'obbligazione ha come data di liquidazione il 1° luglio 1993, come data di emissione il 1° dicembre 1992 e come data del primo interesse il 1 giugno 1993. Il tasso nominale d'interesse semestrale è pari al 5,50%. L'obbligazione ha un valore nominale di L. 100.000 e una base per il conteggio dei giorni di 30/360.

Per calcolare gli interessi maturati sull'obbligazione, si dovrebbe usare la seguente funzione:

@MATURATO(@DATA(93;7;1);@DATA(92;12;1);@DATA(93;6;1);0,055;100000;2;0) = 458,33

Funzioni @ affini

@MATURATO2 calcola gli interessi maturati sui titoli con pagamento periodico degli interessi, in base alle convenzioni giapponesi.

@PREZZO calcola il prezzo di un'obbligazione con valore nominale pari a \$100. @RENDIMENTO calcola il rendimento delle obbligazioni con pagamento periodico degli interessi. @DURATA calcola la durata annua e

@DURATAM calcola la durata annua modificata di un titolo con pagamento periodico degli interessi.

@ACOS

@ACOS(x) calcola l'arcocoseno (inverso del coseno) utilizzando il coseno x di un angolo. Il risultato di **@ACOS** è un angolo, espresso in radiani, da 0 a π che rappresenta un angolo compreso fra 0 e 180 gradi.

Argomenti

x è il coseno di un angolo e può essere un valore compreso tra -1 e 1.

Esempi

Il coseno dell'angolo x è 0,5. Per calcolare la misura dell'angolo x , usare la funzione **@ACOS(0,5)**, il cui risultato è 1,0472 radianti. Per convertire questo valore in gradi, usare la funzione **@RADGRADI(1,0472)**; il risultato è 60 gradi.

Funzioni @ affini

@COS calcola il coseno di un angolo. @ACOSH calcola l'arcocoseno iperbolico di un valore.

@ACOSH

@ACOSH(x) calcola l'arcocoseno (inverso del coseno) iperbolico utilizzando il coseno iperbolico x di un angolo.

Argomenti

x è il coseno iperbolico di un angolo e consiste in un valore maggiore o uguale a 1.

Esempi

@ACOSH(2) = 1,316958

Funzioni @ affini

@ACOS calcola l'arcocoseno di un valore. @COSH calcola il coseno iperbolico di un angolo.

@ACOT

@ACOT(x) calcola l'arcocotangente (inverso della cotangente) utilizzando la cotangente x di un angolo. Il risultato della funzione **@ACOT** è un angolo espresso in radiani, da 0 a π , che rappresenta un angolo compreso tra 0 e 180 gradi.

Argomenti

x è la cotangente di un angolo e può essere un qualsiasi valore.

Esempi

La cotangente dell'angolo x è 1,732051. Per calcolare la misura dell'angolo x , usare la funzione **@ACOT(1,732051)**, il cui risultato è 0,523599 radianti. Per convertire questo valore in gradi, usare la funzione **@RADGRADI(0,523599)**; il risultato è 30 gradi.

Funzioni @ affini

@COT calcola la cotangente di un angolo. @ATAN calcola l'arcotangente di un valore.

@ACOTH

@ACOTH(x) calcola l'arcocotangente (inverso della cotangente) iperbolico utilizzando la cotangente iperbolica x di un angolo.

Argomenti

x è la cotangente iperbolica di un angolo e può essere un valore minore di -1 o maggiore di 1.

Esempi

@ACOTH(2) = 0,549306

Funzioni @ affini

@ACOT calcola l'arcocotangente di un valore. @COTH calcola la cotangente iperbolica di un angolo.

@ACSC

@ACSC(x) calcola l'arcocosecante (inverso della cosecante) utilizzando la cosecante x di un angolo. Il risultato della funzione **@ACSC** è un angolo espresso in radianti, da $-\pi/2$ a $\pi/2$, che rappresenta un angolo compreso tra -90 e 90 gradi.

Argomenti

x è la cosecante di un angolo e può essere un valore maggiore o uguale a 1 oppure minore o uguale a -1 .

Esempi

La cosecante dell'angolo x è $1,743447$. Per calcolare la misura dell'angolo x , usare la funzione **@ACSC(1,743447)**, il cui risultato è $0,610865$ radianti. Per convertire questo valore in gradi, usare la funzione **@RADGRADI(0,610865)**; il risultato è 35 gradi.

Funzioni @ affini

@ACSCH calcola l'arcocosecante iperbolico di un valore. **@CSC** calcola la cosecante di un angolo.

@ACSCH

@ACSCH(x) calcola l'arcocosecante (inverso della cosecante) iperbolico utilizzando la cosecante iperbolica x di un angolo.

Argomenti

x è la cosecante iperbolica di un angolo e consiste in un valore diverso da 0.

Esempi

@ACSCH(1,54) = 0,61068

Funzioni @ affini

@ACSC calcola l'arcocosecante di un valore. @CSCH calcola la cosecante iperbolica di un angolo.

@ASEC

$@ASEC(x)$ calcola l'arcosecante (inverso della secante) utilizzando la secante x di un angolo. Il risultato della funzione $@ASEC$ è un angolo espresso in radiani, da 0 a π , che rappresenta un angolo compreso tra 0 e 180 gradi.

Argomenti

x è la secante di un angolo e consiste in un valore minore o uguale a -1 oppure maggiore o uguale a 1.

Esempi

In un triangolo rettangolo la secante dell'angolo x è 2. Per calcolare la misura dell'angolo x , usare la funzione $@ASEC(2)$, il cui risultato è 1,047198 radianti. Per convertire questo valore in gradi, usare la funzione $@RADGRADI(1,047198)$; il risultato è 60 gradi.

Funzioni @ affini

@SEC calcola la secante di un angolo. @ASECH calcola l'arcosecante iperbolica di un valore.

@ASECH

@ASECH(x) calcola l'arcosecante (inverso della secante) iperbolica utilizzando la secante iperbolica x di un angolo.

Argomenti

x è la secante iperbolica di un angolo e consiste in un valore maggiore di 0 e minore o uguale a 1.

Esempi

@ASECH(0,5) = 1,316958

Funzioni @ affini

@ASEC calcola l'arcosecante di un angolo. @SECH calcola la secante iperbolica di un angolo.

@ASEN

@ASEN(x) calcola l'arcoseno (inverso del seno) utilizzando il seno x di un angolo. Il risultato di **@ASEN** è un angolo espresso in radianti, compreso tra $-1/2$ e $1/2$, ossia un angolo compreso tra -90 e 90 gradi.

Argomenti

x è il seno di un angolo, e può essere un valore oppure l'indirizzo o il nome di una cella contenente un valore compreso tra -1 e 1 .

Esempi

Il seno dell'angolo x è $0,66$. Per calcolare la misura dell'angolo x , usare la funzione **@ASEN**($0,66$), il cui risultato è $0,72082$ radianti. Per convertire questo valore in gradi, usare la funzione **@RADGRADI**($0,72082$); il risultato è $41,3$ gradi.

Funzioni @ affini

@SEN calcola il seno di un angolo. @ASENH calcola l'arcoseno iperbolico di un valore.

@ASENH

@ASENH(x) calcola l'arcoseno (inverso del seno) iperbolico utilizzando il seno iperbolico x di un angolo.

Argomenti

x è il seno iperbolico di un angolo e può assumere qualsiasi valore.

Esempi

@ASENH(2) = 1,443635

Funzioni @ affini

@SENH calcola il seno iperbolico di un angolo. @ASEN calcola l'arcoseno di un valore.

@@

@(posizione) consente di ottenere indirettamente il contenuto della cella specificata in *posizione*.

Argomenti

posizione specifica l'indirizzo o il nome di una cella contenente l'indirizzo o il nome di una cella, oppure una formula che genera l'indirizzo o il nome di una cella.

posizione agisce da puntatore a un'altra cella, il cui contenuto viene visualizzato nella cella che contiene @@. Se *posizione* non è un indirizzo di cella o un nome di zona valido, oppure se è una zona composta da più celle, @@ assume il valore ERR.

NOTE

@ è particolarmente utile nella creazione di formule condizionali, in quanto il suo riferimento indiretto è in grado di modificarne automaticamente il valore. Ad esempio, se la cella A10 contiene la formula

`@SE(D2="S";"D8";@SE(D2="N";"D9";@ERR))`

nella cella A10, la formula @@(A10) nella cella E2 riporta in E2 il contenuto della cella D8, della cella D9 oppure ERR, a seconda che D2 contenga S, N o qualcosa di diverso.

Quando *posizione* fa riferimento a una cella contenente una formula, occorre premere F9 (CALCOLA) per aggiornare la formula @@ dopo l'esecuzione del ricalcolo automatico. Se non si preme F9 (CALCOLA), la formula @@ assume il valore 0.

Esempi

@

Funzioni @ affini

@SCANSOR e @SCANSVER riportano il contenuto della cella specificata di una tabella di consultazione orizzontale o verticale. @SCELTA assume un valore o un'etichetta da una lista e @IND fornisce il contenuto della cella posizionata nel punto di intersezione della colonna, riga e foglio specificati.

@ATAN

@ATAN(x) calcola l'arcotangente (inverso della tangente) utilizzando la tangente x di un angolo. Il risultato di @ATAN è un angolo espresso in radiani, compreso tra $-\pi/2$ e $\pi/2$, ossia un angolo compreso tra -90 e 90 gradi.

Argomenti

x è la tangente di un angolo, e può essere un valore.

Esempi

La tangente dell'angolo x è $2/1$ o 2 . Per calcolare la misura dell'angolo x , usare la funzione @ATAN(2), il cui risultato è $1,10715$ radianti. Per convertire questo valore in gradi, usare la funzione @RADGRADI(1,10715); il risultato è $63,4$ gradi.

Funzioni @ affini

@ATANH calcola l'arcotangente iperbolico di un valore.

@ATAN2 calcola la misura di un angolo la cui tangente è y/x . @TAN calcola la tangente di un angolo.

@ATAN2

@ATAN2(x;y) calcola l'arcotangente utilizzando la tangente y/x di un angolo. Il risultato di @ATAN2 è un angolo espresso in radianti, compreso tra $-\pi$ e π , ossia un angolo compreso tra -180 e 180 gradi, a seconda del segno di x e y (v. lista seguente).

Argomenti

x e y possono essere valori. Se y è pari a 0, @ATAN2 dà 0; se sia x che y sono pari a 0, @ATAN2 assume il valore ERR.

NOTE

La tabella seguente elenca le gamme di valori per @ATAN2.

- Se sia x che y sono positivi, il risultato è compreso tra 0 e $\pi/2$ (quadrante I).
- Se x è negativo e y è positivo, il risultato è compreso fra $\pi/2$ e π (quadrante II).
- Se sia x che y sono negativi, il risultato è compreso tra $-\pi$ e $-\pi/2$ (quadrante III).
- Se x è positivo e y è negativo, il risultato è compreso tra $-\pi/2$ e 0 (quadrante IV).

Quando x e y sono entrambi positivi (quadrante I) e quando x è positivo e y è negativo (quadrante IV), i risultati sono uguali a quelli generati dalla funzione @ATAN.

Esempi

In un triangolo rettangolo i due cateti misurano 1 e 2. Per calcolare la misura dell'angolo acuto più grande, usare la funzione @ATAN2(1;2), il cui risultato è 1,10715 radianti. Per convertire questo valore in gradi, usare la funzione @RADGRADI(1,10715); il risultato è 63,4.

Funzioni @ affini

@ATAN calcola l'arcotangente utilizzando la tangente x di un angolo. @TAN calcola la tangente di un angolo.

@ATANH

@ATANH(x) calcola l'arcotangente (inverso della tangente) iperbolica utilizzando la tangente iperbolica x.

Argomenti

x è la tangente iperbolica di un angolo e consiste in un valore compreso tra -1 e 1.

Esempi

@ATANH(0,544736) = 0,610865

Funzioni @ affini

@ATAN calcola l'arcotangente di un valore. @ATAN2 calcola la misura di un angolo la cui tangente è y/x. @TANH calcola la tangente iperbolica di un angolo.

@MEDIADEV

@MEDIADEV(*lista*) calcola la media delle deviazioni assolute dei valori compresi nella *lista*.

Argomenti

lista può contenere i seguenti elementi in qualunque combinazione: numeri, formule numeriche e indirizzi o nomi di zone contenenti numeri o formule numeriche. Tali elementi devono essere divisi tra loro dagli appositi separatori di argomenti.

Vedere anche Argomenti delle funzioni @ statistiche.

Esempi

@MEDIADEV

Funzioni @ affini

@SCARTO e @SCARTOPURO calcolano la deviazione standard dei valori compresi in una lista. @DEVQ calcola la somma delle deviazioni quadratiche dei valori compresi in una lista.

@MEDIA, @MEDIAPURA

@MEDIA(*lista*) calcola la media aritmetica di una *lista* di valori.

@MEDIAPURA(*lista*) calcola la media aritmetica di una *lista* di valori, ignorando tutte le celle contenenti etichette.

Argomenti

lista può contenere i seguenti elementi in qualunque combinazione: numeri, formule numeriche e indirizzi o nomi di zone contenenti numeri o formule numeriche. Gli elementi della *lista* devono essere divisi tra loro dagli appositi separatori di argomenti.

Vedere anche Argomenti delle funzioni @ statistiche.

Esempi

@MEDIA(A2..A5) = 252,75, dove A2..A5 contiene i valori 160, 227, 397 e 227.

@MEDIA(A1..A5) = 202,20, dove A1..A5 contiene i valori 160, 227, 397 e 227 e l'etichetta Gennaio (@MEDIA considera l'etichetta come valore zero, includendola nel calcolo).

@MEDIAPURA(A1..A5) = 252,75. La funzione @MEDIAPURA ignora l'etichetta Gennaio.

Funzioni @ affini

@MEDIADB calcola la media dei valori contenuti nel campo di una tabella di database che soddisfano le condizioni specificate. @MEDIAGEOM calcola la media geometrica, mentre @MEDIARARMON calcola la media armonica dei valori compresi in una lista. @MEDIANA calcola il valore mediano di una lista di valori.

@BESSELI, @BESSELJ, @BESSELK, @BESSELY

@BESSELI($x;n$) calcola la funzione di Bessel modificata di prima specie e di ordine n $I_n(x)$.

@BESSELJ($x;n$) calcola la funzione di Bessel di prima specie e di ordine n $J_n(x)$.

@BESSELK($x;n$) calcola la funzione di Bessel modificata di seconda specie e di ordine n $K_n(x)$.

@BESSELY($x;n$) calcola la funzione di Bessel di seconda specie e di ordine n $Y_n(x)$, nota anche come funzione di Neumann.

Argomenti

x è il valore a cui deve essere calcolata la funzione; può essere un valore qualsiasi.

n è l'ordine della funzione; può essere un qualsiasi numero intero positivo. Nel caso specifico di @BESSELI e @BESSELJ, n può anche assumere il valore zero.

NOTE

@BESSELI, @BESSELJ, @BESSELK e @BESSELY calcolano le rispettive funzioni con un'approssimazione di $\pm 5 \cdot 10^{-8}$.

Le funzioni di Bessel vengono spesso usate per la risoluzione dei problemi di simmetria cilindrica relativi al movimento ondulatorio, alla fluidodinamica, all'elasticità e alla diffusione.

Esempi

@BESSELI(2;2) = 0,688948

@BESSELJ(1;0) = 0,765198

@BESSELK(3;0) = 0,03474

@BESSELY(1;1) = -0,781213

@BETA

@BETA(z;w) calcola la funzione beta.

Argomenti

z e w possono assumere qualsiasi valore reale diverso da zero e dai numeri interi negativi.

NOTA

Il risultato della funzione @BETA è calcolato con un'approssimazione fino ad almeno la sesta cifra significativa.

Esempi

@BETA(0,5;0,5) = 3,141593

Funzioni @ affini

@BETA calcola la funzione beta incompleta. @GAMMA calcola la funzione gamma.

@BETAI

@BETAI($a;b;x$) calcola la funzione beta incompleta.

Argomenti

a e b possono assumere qualsiasi valore.

x deve essere un valore compreso tra 0 e 1.

NOTA

Il risultato della funzione @BETAI è calcolato con un'approssimazione fino ad almeno la sesta cifra significativa.

Esempi

@BETAI(5;0,5;0,668271) = 0,050012

Funzioni @ affini

@BETA calcola la funzione beta.

@BINOMIALE

@BINOMIALE(*prove*; *successi*; *probabilità*; [*tipo*]) calcola la funzione generatrice della probabilità binomiale oppure la distribuzione binomiale cumulativa.

Argomenti

prove è il numero di tentativi indipendenti e può essere un qualsiasi numero intero.

successi è il numero di *prove* riuscite e può essere un qualsiasi numero intero minore o uguale a *prove*, oppure 0.

Se *prove* e *successi* non sono numeri interi, 1-2-3 li tronca per renderli tali.

probabilità è la probabilità di successo per ogni tentativo e può essere un qualsiasi valore compreso tra 0 e 1.

tipo è un argomento facoltativo che indica se 1-2-3 deve calcolare la funzione generatrice della probabilità oppure la distribuzione binomiale cumulativa.

tipo	1-2-3 calcola
0	la probabilità che si verifichi esattamente un certo numero di <i>successi</i> sul numero di tentativi (impostazione predefinita in caso di omissione dell'argomento)
1	la probabilità che si verifichi un certo numero massimo di <i>successi</i> sul numero di tentativi
2	la probabilità che si verifichi un certo numero minimo di <i>successi</i> sul numero di tentativi

NOTA

@BINOMIALE calcola la distribuzione binomiale cumulativa con un'approssimazione di $\pm 3 \cdot 10^{-7}$.

Esempi

Sono stati selezionati a caso dieci consumatori di cola per partecipare a un test anonimo di degustazione. Ogni soggetto riceve un bicchiere di cola A e uno di cola B. I bicchieri sono identici, ad eccezione di un codice scritto sul fondo che identifica il tipo di cola. Partendo dal presupposto che i consumatori di cola non hanno la tendenza a preferire una marca rispetto a un'altra, la probabilità che un soggetto scelga il prodotto A è del 50%.

Per calcolare la probabilità che sette partecipanti su dieci preferiscano la cola A, occorre usare la seguente funzione:

@BINOMIALE(10;7;0,5) = 0,117188 pari a 11,72%

Per calcolare la probabilità che almeno sette partecipanti su dieci preferiscano la cola A, occorre usare la seguente funzione:

@BINOMIALE(10;7;0,5;2) = 0,171875 pari a 17,19%

Funzioni @ affini

@CRITBINOMIALE calcola il valore massimo per cui la distribuzione binomiale cumulativa è minore o uguale a una determinata condizione. @COMBIN calcola il numero di combinazioni di un determinato numero di valori.

@PERMUT calcola il numero di permutazioni di una lista di valori.

@CELLA, @CURSORE

@CELLA(*attributo;zona*) fornisce informazioni sulla prima cella della *zona*.

@CURSORE(*attributo*) fornisce informazioni sulla cella attiva.

Argomenti

zona è l'indirizzo o il nome di una zona.

attributo può essere uno degli elementi riportati qui di seguito, racchiuso tra virgolette (" "), oppure l'indirizzo o il nome di una cella contenente uno di tali elementi.

attributo	Risultato
allineamento_orizz	Numero corrispondente al tipo di allineamento orizzontale applicato ai dati della cella: 0 per l'allineamento predefinito (etichette allineate a sinistra, valori a destra) 1 per l'allineamento a Sinistra 2 per la centratura (Centro) 3 per l'allineamento a Destra 4 per la spaziatura uniforme (Spazia uniformemente)
altezza	Altezza di riga, espressa in punti
bordo_destro	Numero intero, compreso tra 0 e 15, che indica il tipo di linea del bordo destro della cella
bordoinf	Numero intero compreso tra 0 (bordo assente) e 8, che indica il tipo di linea del bordo inferiore della cella
bordo_sinistro	Numero intero, compreso tra 0 (bordo assente) e 8, che indica il tipo di linea del bordo sinistro della cella
carattere_tipografico	Carattere tipografico applicato ai dati della cella
col	Lettera della colonna espressa da un valore compreso tra 1 e 256 (1 per la colonna A, 2 per la B e così via)
colore	1 se la cella è impostata per visualizzare i numeri negativi in un certo colore 0 se la cella non è impostata per rappresentare i numeri negativi a colori
colore_bordo_destro	Numero intero, compreso tra 0 e 239, che indica il colore del bordo destro della cella
colore_bordo_inferiore	Numero intero compreso tra 0 e 239, che indica il colore del bordo inferiore della cella
colore_bordo_sinistro	Numero intero, compreso tra 0 e 239, che indica il colore del bordo sinistro della cella
colore_motivo	Numero intero, compreso tra 0 e 239, che indica il colore del motivo selezionato dalla tavolozza dei colori
colore_sfondo	Colore dello sfondo, rappresentato da un numero intero compreso tra 0 e

	239, corrispondente a quello usato nella tavolozza dei colori
contenuto	Contenuto della cella
coord	Indirizzo assoluto della cella per esteso (lettera del foglio, lettera della colonna e numero della riga)
corpo_font	Corpo, espresso in punti, applicato ai dati della cella
corsivo	1 se alla cella è stato applicato il formato corsivo 0 se alla cella non è stato applicato il formato corsivo
data_file	Valore corrispondente alla data e all'ora in cui è stata salvata per l'ultima volta la cartella di lavoro che contiene la cella, composto sia da un <u>valore_data</u> (parte intera del numero) che da un <u>valore_orario</u> (parte decimale).
foglio	Lettera del foglio, espressa con un numero da 1 a 256 (1 per il foglio A, 2 per il foglio B e così via)
formato	Formato della cella: da V0 a V15 per il formato Valuta, con 0 -15 cifre decimali da F0 a F15 per il formato Fisso, con 0 - 15 cifre decimali G per il formato Generico, per un'etichetta o per una cella vuota da P0 a P15 per il formato Percentuale, con 0 - 15 cifre decimali da S0 a S15 per il formato Scientifico, con 0 - 15 cifre decimali da .0 a .15 per il formato Punto (.) con 0 - 15 cifre decimali + per il formato +/- D1 per il formato Data 31-Dic-92 D2 per il formato Data 31-Dic D3 per il formato Data Dic-92 D4 per il formato Data 12/31/92 D5 per il formato Data 12/31 D6 per il formato Ora 11:59:59 AM D7 per il formato Ora 11:59 AM D8 per il formato Ora 23:59:59 D9 per il formato Ora 59:59 T per Testo N per Nascondi E per Etichetta A per Automatico -- per il formato Colore per i numeri negativi () per Parentesi
grassetto	1 se alla cella è stato applicato il formato grassetto 0 se alla cella non è stato applicato il formato grassetto
indirizzo	Indirizzo assoluto in forma abbreviata (solo la lettera della colonna e il numero della riga)

larghezza	Larghezza della colonna
motivo	Numero intero, compreso tra 0 e 63, che indica il motivo
nomearch	Nome della cartella di lavoro contenente la cella, compreso il percorso
nomefoglio	Nome del foglio oppure, se il foglio non è stato denominato, lettera del foglio
orientamento	Numero intero, compreso tra 0 e 4, corrispondente all'orientamento dei testi ruotati
parentesi	1 se alla cella è stato assegnato il formato Parentesi 0 se la cella non è stato assegnato il formato Parentesi
prefisso	' se la cella contiene un'etichetta allineata a sinistra " se la cella contiene un'etichetta allineata a destra ^ se la cella contiene un'etichetta centrata \ se la cella contiene un'etichetta ripetuta se la cella contiene un'etichetta non stampabile Nessun prefisso di etichetta se la cella è vuota o contiene un valore
protezione	1 se la cella è protetta 0 se la cella non è protetta
riga	Numero di riga, da 1 a 8192
rotazione	Angolo di rotazione del testo ruotato, espresso in gradi, da 0 a 90
sucolonne	1 se i dati contenuti nella cella sono allineati in colonne 0 se i dati della cella non sono allineati in colonne
tipo	Tipo di dati contenuti nella cella: s se la cella è vuota v se la cella contiene un valore numerico, una <u>valore numerico</u> o una <u>formula di testo</u> e se la cella contiene un'etichetta
tipodati	Tipo di dati contenuti nella cella: s se la cella è vuota v se la cella contiene un numero o una formula numerica e se la cella contiene un'etichetta o una formula di testo r se la cella contiene il valore <u>ERRn</u> se la cella contiene il valore <u>ND</u>
tipoformula	Tipo della formula contenuta nella cella: s se la cella è vuota v se la cella contiene un numero e se la cella contiene un'etichetta fv e la cella contiene una formula numerica

	fe se la cella contiene una formula di testo
	fr se la cella contiene a formula che produce ERR
	fn se la cella contiene a formula che produce ND
colore_testo	Colore dei dati della cella, rappresentato dal numero intero compreso tra 0 e 239 usato per individuare il colore nella tavolozza dei colori
bordo_sup	Numero intero compreso tra 0 (nessun bordo) e 8, che specifica il tipo di linea applicato al bordo superiore della cella
colore_bordo_sup	Numero intero compreso tra 0 e 239, che specifica il colore della linea usata per il bordo superiore della cella
sottolineatura	Tipo di sottolineatura, espresso come numero intero compreso tra 0 (nessuna sottolineatura) e 3
allineamento_vert	Allineamento verticale dei dati della cella: 0 se l'allineamento è rispetto al bordo inferiore 1 se l'allineamento è rispetto al centro 2 se l'allineamento è rispetto al bordo superiore
a_capo	1 se i dati vanno a capo entro la cella 0 se i dati non vanno a capo entro la cella

NOTE

Per eseguire un ricalcolo prima di usare @CELLA o @CURSORE ed essere quindi certi che i risultati siano corretti, premere F9 (CALCOLA).

@CELLA e @CURSORE possono essere utili nelle macro e in combinazione con @SE. Usare la funzione @CELLA per controllare l'input durante una macro ed evitare certi tipi di immissione, oppure per indirizzare l'esecuzione delle macro mediante sottoprogrammi basati sulle immissioni dell'utente. Inoltre @CELLA consente a un'applicazione automatizzata di cambiare gli attributi di cella sulla base delle immissioni dell'utente.

Usare la funzione @CURSORE per individuare la posizione del cursor oppure per calcolare una formula basata sul contenuto della cella attiva. L'elaborazione può essere poi indirizzata a seconda del contenuto o del tipo di cella.

Esempi

Nell'esempio sottostante, la funzione @CELLA viene usata insieme a @SE e @ERR per generare un errore (ERR) se l'utente non digita un valore nella cella denominata IMP e per fornire il contenuto di IMP (un valore) se l'utente immette un valore.

```
@SE(@CELLA("tipo";IMP)="v";IMP;@ERR)
```

Nell'esempio sottostante, la funzione @CURSORE viene usata in una macro intesa a verificare l'eventuale esistenza di una cella vuota in una lista di elementi. In questo caso specifico, se 1-2-3 incontra una cella vuota, emette un segnale acustico e salta a un sottoprogramma.

```
{SE @CURSORE("tipo")="s"}{BIP}{SALTO Istruzione2}
```

Funzioni @ affini

@INFO fornisce informazioni sulla sessione in corso di 1-2-3.

@CAR

@CAR(x) genera il carattere del set di caratteri multilingue Lotus, LMBCS (Lotus Multibyte Character Set) corrispondente al numero x.

Argomenti

x può essere un numero intero oppure l'indirizzo o il nome di una cella contenente un numero intero. I valori che non corrispondono ai codici dei caratteri LMBCS generano il valore ERR. Se x non è un numero intero, @CAR elimina le cifre decimali per renderlo tale.

NOTA

Se il monitor del computer non è in grado di visualizzare il carattere corrispondente a x, 1-2-3 visualizza un carattere il più possibile simile a quello indicato. Se nessuno dei caratteri assomiglia al carattere desiderato, verrà visualizzato un rettangolo pieno a indicare un carattere non visualizzabile. In tal caso conviene accertarsi che la stampante sia in grado di stampare i caratteri immessi.

Esempi

@CAR(156) = £ (simbolo della sterlina britannica).

@CAR(D9) = A, se la cella D9 contiene il valore 65.

Funzioni @ affini

@CODICE fornisce il codice LMBCS corrispondente a un carattere.

@DISTRIBCHI2

@DISTRIBCHI2(x;gradi_libertà;[tipo]) calcola la distribuzione chi-quadrato.

Argomenti

x è il valore a cui deve essere calcolata la distribuzione chi-quadrato. Il valore di x dipende dal *tipo* immesso.

<i>tipo</i>	x
0	valore critico o valore limite della variabile casuale della
1	probabilità (livello di significatività), il cui valore è compreso tra 0

gradi_libertà è il numero di gradi di libertà per il campione e consiste in un numero intero positivo. Se *gradi_libertà* non è un numero intero, 1-2-3 lo tronca per renderlo tale.

tipo è un argomento facoltativo che specifica come 1-2-3 deve calcolare @DISTRIBCHI2.

<i>tipo</i>	1-2-3 calcola
0	il livello di significatività corrispondente a x (impostazione)
1	il valore critico corrispondente al livello di significatività x

NOTE

@DISTRIBCHI2 calcola la distribuzione cumulativa chi-quadrato con un'approssimazione del $\pm 3 \cdot 10^{-7}$. Se @DISTRIBCHI2 non riesce a eseguire il calcolo con un'approssimazione dello 0,0000001 dopo 100 iterazioni, produce il risultato ERR.

La distribuzione chi-quadrato è una distribuzione continua a un solo parametro, derivata da un caso particolare della distribuzione gamma.

@DISTRIBCHI2 può servire a verificare la validità di un'ipotesi operando un confronto tra valori osservati e valori previsti.

Esempi

@DISTRIBCHI2(12,592;6) = 0,05

@DISTRIBCHI2(0,05;6;1) = 12,59159

Funzioni @ affini

@TESTCHI2 calcola la probabilità associata a un test chi-quadrato. @DISTRIBF calcola la distribuzione F.

@DISTRIBT calcola la distribuzione t di Student.

@SCELTA

@SCELTA(x;lista) assume il valore o l'etichetta corrispondente a *x* nella *lista*.

Argomenti

x è un valore e rappresenta lo scarto corrispondente alla posizione della voce situata in *lista*.

lista può essere un gruppo di valori e di etichette oppure gli indirizzi o i nomi delle celle contenenti valori ed etichette, divisi tra loro dagli appositi separatori di argomenti. 1-2-3 numera tutte le immissioni nella *lista* e poi sceglie l'immissione che corrisponde al valore di *x*.

Esempi

Un foglio di lavoro contiene una lista di etichette situata nella zona A1..A4, nonché i rispettivi scarti 0, 1, 2 e 3, posti nella zona B1..B4. **@SCELTA(B3;A1;A2;A3;A4)** genera l'etichetta in A3 che corrisponde alla voce avente uno scarto pari a 2 (2 è il valore della cella B3) in *lista*.

Funzioni @ affini

@SCANSOR e **@SCANSVER** identificano le immissioni situate in tabelle di consultazione orizzontali e verticali.

@IND fornisce il contenuto di una cella posizionata nel punto di intersezione della colonna, riga e foglio di lavoro.

CORRISP riporta la posizione relativa di una cella con il contenuto specificato. **INDX** riporta il contenuto di una cella

specificata dai titoli di colonna, riga e foglio di lavoro. **@SCANSMAX** produce un riferimento assoluto alla cella che contiene il valore più alto di una serie di zone. **@SCANSMIN** genera un riferimento assoluto alla cella che contiene il valore inferiore di una serie di zone.

@CHIARIF

@CHIARIF(*stringa*) elimina dalla *stringa* i caratteri che non vengono stampati.

Argomenti

stringa può essere un testo racchiuso tra virgolette (" "), l'indirizzo o il nome di una cella contenente un'etichetta oppure una formula o una funzione @ che produce una stringa di testo.

Esempi

Si supponga di aver importato dei dati da un programma di elaborazione testi in 1-2-3. La cella A45 contiene la seguente etichetta:

®Secondo, dobbiamo agire subito.¬

@CHIARIF(A45) = Secondo, dobbiamo agire subito.

Funzioni @ affini

@CAR genera il carattere del set di caratteri multilingue Lotus LMBCS corrispondente a un certo codice. @TRONCA elimina gli spazi iniziali, finali e consecutivi in una stringa di testo.

@CODICE

@CODICE(*stringa*) fornisce il codice del set di caratteri multilingue Lotus (LMBCS (Lotus Multibyte Character Set)) che corrisponde al primo carattere di *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), l'indirizzo o il nome di una cella contenente un'etichetta, oppure una formula o una funzione @ che genera una stringa di testo.

Esempi

@CODICE("A") = 65.

@CODICE(C5) = 77, se la cella C5 contiene l'etichetta Monsignor Bellini (77 è il codice che corrisponde a M nel set di caratteri LMBCS).

Funzioni @ affini

@CAR fornisce il carattere LMBCS che corrisponde a un certo codice.

@COLONNE

@COLONNE(*zona*) conta le colonne comprese in *zona*.

Argomenti

zona può essere l'indirizzo o il nome di una zona.

NOTA

@COLONNE può essere usata anche assieme a {PER} in una macro che svolge la stessa operazione su una serie di colonne per determinare quando dovrà terminare.

Esempi

@COLONNE(D9..J25) = 7, poiché *zona* contiene le colonne da D a J (sette colonne)

@COLONNE(PUNTEGGIO) = 2, se PUNTEGGIO è il nome della zona B3..C45

Funzioni @ affini

@CONVERTIRIF converte le lettere di colonna e di foglio da A a IV nei numeri compresi tra 1 e 256. @RIGHE conta il numero di righe, mentre @FOGLI conta il numero di fogli contenuti in una zona.

@COMBIN

@COMBIN($n;r$) calcola il coefficiente binomiale per n e r . Il coefficiente binomiale è il numero di modi in cui r può essere preso da n , a prescindere dall'ordine.

Argomenti

n è il numero di valori e può essere un qualsiasi numero intero positivo oppure zero.

r è il numero di valori di cui deve essere composta ogni combinazione e può essere un qualsiasi numero intero positivo oppure zero, purché minore o uguale a n .

Se n e r non sono numeri interi, 1-2-3 li tronca per renderli tali.

NOTA

@COMBIN calcola il coefficiente binomiale con un'approssimazione di $\pm 3 \cdot 10^{-7}$.

Esempi

Un'urna contiene cinque palline di colore diverso. Estraendo tre palline a caso, si può calcolare il numero di possibili combinazioni di colori con la seguente funzione:

@COMBIN(5;3) = 10

Funzioni @ affini

@BINOMIALE calcola la funzione generatrice di probabilità binomiale oppure la distribuzione binomiale cumulativa.

@CRITBINOMIALE calcola il valore minimo per cui la distribuzione binomiale cumulativa è minore o uguale a una determinata condizione. @PERMUT calcola il numero di permutazioni per una lista di valori.

@COORD

@COORD(*foglio;colonna;riga;assoluto*) crea un riferimento di cella usando i valori *foglio*, *colonna* e *riga*.

Argomenti

foglio e *colonna* possono essere numeri interi da 1 a 256, e corrispondono alle lettere del foglio e delle colonne (1 per il foglio o la colonna A, 2 per il foglio o la colonna B e così via).

riga può essere un qualsiasi numero intero da 1 a 8192, e corrisponde al numero della riga.

assoluto può essere un numero intero da 1 a 8.

Se *foglio*, *colonna*, *riga* e *assoluto* non sono numeri interi, 1-2-3 li tronca per renderli tali.

NOTE

@COORD crea un riferimento di cella relativo, assoluto o misto, in funzione del valore di *assoluto*. La tabella seguente riporta i valori di *assoluto* e il loro effetto sull'indirizzo di cella A1 nel foglio A.

<i>assoluto</i>	Valore di @COORD(1;1;1; <i>assoluto</i>)
1	\$A:\$A\$1
2	\$A:A\$1
3	\$A:\$A1
4	\$A:A1
5	A:\$A\$1
6	A:A\$1
7	A:\$A1
8	A:A1

La funzione @COORD può risultare utile in combinazione con @IND, @SCANSVER o @SCANSOR per la creazione di indirizzi di cella ricavati da tabelle di valori della cartella di lavoro attiva. @COORD insieme a @@ produce il valore situato all'indirizzo di cella creato da @COORD.

Esempi

@COORD(3;7;25;8) genera l'indirizzo di cella relativo C:G25.

@(@COORD(C1;D1;E4;8)) assume il valore della cella A:A4 (C1 contiene 1, D1 contiene 1 e E4 contiene 4).

Funzioni @ affini

@CONVERTIREF converte le lettere delle colonne o dei fogli di 1-2-3 da A a IV in numeri da 1 a 256.

@CORREL

@CORREL(*zona1*; *zona2*) calcola il coefficiente di correlazione tra i valori contenuti nella *zona1* e nella *zona2*.

Argomenti

zona1 e *zona2* possono essere nomi o indirizzi di zone. *zona1* e *zona2* devono avere grandezza uguale, altrimenti la funzione @CORREL assume il valore ERR.

1-2-3 correla le celle delle due zone in base all'ordine esistente all'interno delle zone stesse. Le zone sono ordinate dall'alto verso il basso, da sinistra a destra e dal primo foglio all'ultimo.

NOTA

Sia la correlazione che la covarianza misurano la relazione tra due serie di dati. Le statistiche basate sulla correlazione, a differenza di quelle basate sulla covarianza, sono tuttavia indipendenti dall'unità di misura.

Esempi

@CORREL

Funzioni @ affini

@COV calcola la covarianza dei valori contenuti in due zone.

@COS

@COS(x) calcola il coseno dell'angolo x . Il coseno è dato dal rapporto tra il cateto adiacente a un angolo acuto di un triangolo rettangolo e l'ipotenusa. Il risultato di **@COS** è un valore compreso fra -1 e 1.

Argomenti

x è un angolo espresso in radianti e può assumere valori da -2^63 to 2^63 .

Esempi

@COS(@GRADIRAD(30)) = 0,866, che è il coseno di un angolo di 30 gradi.

Funzioni @ affini

@ACOS calcola l'arcocoseno di un valore per trovare la misura di un angolo.

@COSH calcola il coseno iperbolico di un valore.

@COSH

@COSH(x) calcola il coseno iperbolico dell'angolo x . Il risultato della funzione @COSH è un valore maggiore o uguale a 1.

Argomenti

x può essere un qualsiasi valore compreso all'incirca tra -709,7827 e 709,7827.

Esempi

@COSH(@GRADIRAD(30)) = 1,140238, che è il coseno iperbolico di un angolo di 30 gradi.

Funzioni @ affini

@ACOS calcola l'arcocoseno (inverso del coseno) di un valore. @COS calcola il coseno di un angolo.

@COT

@COT(x) calcola la cotangente dell'angolo x . La cotangente è data dal rapporto tra il lato adiacente un angolo acuto di un triangolo rettangolo e il lato opposto.

Argomenti

x è un angolo espresso in radianti e può essere un qualsiasi valore compreso tra -2^63 e 2^63 .

Esempi

@COT(@GRADIRAD(30)) = 1,73205, che è la cotangente di un angolo di 30 gradi.

Funzioni @ affini

@ACOT calcola l'arcocotangente di un valore. **@COTH** calcola la cotangente iperbolica di un angolo. **@TAN** calcola la tangente di un angolo.

@COTH

@COTH(x) calcola la cotangente iperbolica dell'angolo x.

Argomenti

x può essere un qualsiasi valore compreso all'incirca tra -11355,1371 e 11355,1371, eccetto lo zero.

Esempi

@COTH(@GRADIRAD(30)) = 2,081283, che è la cotangente iperbolica di un angolo di 30 gradi.

Funzioni @ affini

@ACOTH calcola l'arcocotangente iperbolico di un valore. @COT calcola la cotangente di un angolo. @TANH calcola la tangente iperbolica di un angolo.

@CONT, @CONTAPURO

@CONT(*lista*) conta le celle non vuote in una *lista* di zone.

@CONTAPURO(*lista*) conta le celle di una *lista* di zone, escludendo quelle che contengono etichette.

Argomenti

lista può essere una qualsiasi combinazione di indirizzi o nomi di zone, divisi fra loro dagli appositi separatori di argomenti.

Vedere anche Argomenti delle funzioni @ statistiche.

NOTE

@CONT conta le celle comprese nella *lista* che contengono una qualsiasi immissione, comprese etichette, prefissi di etichetta o i valori ERR e ND.

@CONT e @CONTAPURO sono utili per interrompere (o deviare) una macro che esegue una funzione su una serie di zone quando il cursore raggiunge una zona priva di immissioni.

Esempi

@CONT(A2..A3;A5) = 1, se la zona A2..A3 è vuota e sia che A5 sia vuota o meno, perché A5 è l'indirizzo di una cella.

{SE @CONTAPURO(SETTEMBRE)=0}{SALTO TOT} passa il controllo a una macro denominata TOT se la zona SETTEMBRE è vuota o contiene un'etichetta, un prefisso d'etichetta o una formula di testo.

Funzioni @ affini

@CONTDB e @CONTAPURODB contano le celle non vuote del campo di una tabella di database che soddisfano le condizioni specificate.

@COV

@COV(zona1;zona2;[tipo]) calcola la covarianza della popolazione o di un campione di valori contenuti in *zona1* e *zona2*.

Argomenti

zona1 e *zona2* possono essere nomi o indirizzi di zone. Le due zone devono avere uguale grandezza, altrimenti la funzione @COV assume il valore ERR.

1-2-3 correla le celle delle due zone in base all'ordine esistente all'interno delle zone stesse. Le zone sono ordinate dall'alto verso il basso, da sinistra a destra e dal primo foglio all'ultimo.

tipo è un argomento facoltativo che specifica se deve essere calcolata la covarianza della popolazione o di un campione.

<i>tipo</i>	1-2-3 calcola
0	la covarianza della popolazione (impostazione predefinita in caso di omissione dell'argomento)
1	la covarianza campionaria

NOTE

La covarianza è la media dei prodotti delle deviazioni dei valori corrispondenti nelle liste indicate.

Sia la correlazione che la covarianza misurano la relazione tra due serie di dati. Le statistiche basate sulla correlazione, a differenza di quelle basate sulla covarianza, sono tuttavia indipendenti dall'unità di misura.

Esempi

@COV

Funzioni @ affini

@VAR e @VARPURA calcolano la varianza della popolazione. @VARC e @VARCPURA calcolano la varianza campionaria dei valori compresi in una lista. @CORREL calcola il coefficiente di correlazione tra valori corrispondenti in due zone.

@CSC

@CSC(x) calcola la cosecante dell'angolo x . La cosecante è il reciproco del seno. Il risultato di **@CSC** è un valore maggiore o uguale a 1, oppure minore o uguale a -1.

Argomenti

x è un angolo espresso in radianti e può essere un valore qualsiasi compreso tra $-2^{\wedge}63$ e $2^{\wedge}63$.

Esempi

@CSC(@GRADIRAD(30)) = 2, che è la cosecante di un angolo di 30 gradi.

Funzioni @ affini

@ACSC calcola l'arcocosecante di un valore. **@CSCH** calcola la cosecante iperbolica di un angolo. **@SEN** calcola il seno di un angolo.

@CSCH

@CSCH(x) calcola la cosecante iperbolica dell'angolo x . La cosecante iperbolica è il reciproco del seno iperbolico.

Argomenti

x può essere un qualunque valore compreso all'incirca fra $-709,7827$ e $709,7827$.

Esempi

@CSCH(@GRADIRAD(30)) = 1,825306, che è la cosecante iperbolica di un angolo di 30 gradi.

Funzioni @ affini

@ACSCH calcola l'arcocosecante iperbolico di un valore. @CSC calcola la cosecante di un angolo. @SENH calcola il seno iperbolico di un angolo.

@CTERM

@CTERM(*interesse;valore_futuro;valore_attuale*) calcola quanto tempo sarà necessario affinché un investimento (*valore_attuale*) raggiunga un *valore_futuro* a un dato tasso di *interesse* fisso per ciascun periodo di capitalizzazione.

Argomenti

interesse può essere un qualsiasi valore maggiore di -1, ad eccezione di 0.

valore_futuro e *valore_attuale* possono essere valori qualsiasi, oppure gli indirizzi o i nomi di celle contenenti valori. Gli argomenti *valore_futuro* e *valore_attuale* devono avere lo stesso segno.

Esempi

Nel seguente esempio, si supponga di aver depositato 14.000.000 di lire su un conto che rende un tasso d'interesse annuo del 10% (0,10), capitalizzato mensilmente, e di voler calcolare quanti anni ci vorranno per raddoppiare l'investimento.

@CTERM(0,10/12;28000000;14000000)/12 = 6,960313

In altre parole, ci vorranno circa sette anni per raddoppiare l'investimento originale di L.14.000.000.

Dal momento che **@CTERM** fornisce un risultato espresso in anni di capitalizzazione, potrebbe essere necessario specificare il numero di periodi per i quali viene capitalizzato il tasso d'*interesse*, per ridurre questo valore alla stessa unità di tempo utilizzata dalla funzione **@CTERM**. In questo esempio, il tasso d'interesse annuo del 10% capitalizzato mensilmente viene immesso sotto forma di 0,10/12 (*interesse* diviso per il numero di periodi di capitalizzazione).

Funzioni @ affini

@TERM calcola il numero di periodi di capitalizzazione necessari affinché un investimento costituito da versamenti periodici di uguale importo raggiunga un valore specificato.

Esempio: @@

Nella seguente tabella di provvigioni sul venduto, la cella A10 contiene la formula

`@SE(C3="N";"C7";@SE(C3="B";"C8";@ERR))`

che riporta uno dei due indirizzi di cella (C7 o C8), a seconda del codice prodotto (N o B) immesso nella cella C3. La formula `@@(A10)` situata nella cella C4 immette il contenuto della cella il cui indirizzo viene generato dalla formula nella cella A10.

Se nella cella C3 viene immesso un dato diverso dai due codici, le funzioni `@SE` e `@@` generano entrambe il valore `ERR`.

A	-----	A	-----	B	-----	C	-----	
1		TABELLA DI PROVVISIIONI SUL VENDUTO						
2								
3		Codice prodotto:		N				
4		Tasso di provvigione:		5%				
5								
6		Prodotto		Codice		Tasso		
7		Nasi		N		5%		
8		Baffi finti		B		3%		
9								
10		C7						

Esempio: @MEDIADDEV

La tabella sottostante elenca una serie di vendite immobiliari effettuate nel mese di aprile. La zona denominata LISTA_ETÀ (C2..C8) contiene l'età, espressa in anni, di vari immobili. Si vuole calcolare la deviazione media delle età degli immobili compresi nella lista.

@MEDIADDEV(LISTA_ETÀ) = 15,18

A	A	B	C	D
1	INDIRIZZO	LOTTO	ETÀ	COSTO
2	P.za San Marco	0,25	48	L. 290.000.000
3	Via dei Marinai, 40	0,40	22	L. 105.000.000
4	Vicolo Stretto	0,50	21	L. 135.000.000
5	Via dei Giardini, 27	1,00	70	L. 128.000.000
6	Via Università, 468	0,50	52	L. 174.000.000
7	Via Accademia, 9	0,25	42	L. 195.000.000
8	Lungomare Est, 80	0,25	23	L. 118.000.000

Esempio: @CORREL

Si vuole stabilire se esiste una relazione tra l'altezza e il peso di dieci soggetti selezionati a caso, i cui dati sono contenuti in un foglio di lavoro.

@CORREL(A2..A11;B2..B11) = 0,384947

A ----- A ----- B -----

	ALTEZZA (cm)	PESO (kg)
1		
2	190,50	72,73
3	187,96	86,36
4	175,26	68,18
5	175,26	76,37
6	180,34	77,27
7	180,34	72,73
8	187,96	75,00
9	172,72	68,18
10	177,80	70,46
11	179,07	86,36

Esempio: @COV

Si vuole determinare se esiste una relazione tra l'altezza e il peso di dieci soggetti scelti a caso, i cui dati sono stati inseriti in un foglio di lavoro.

@COV(A2..A11;B2..B11) = 13,8872

A ----- A ----- B --

1	ALTEZZA (cm)	PESO (kg)
2	190,50	72,73
3	187,96	86,36
4	175,26	68,18
5	175,26	76,37
6	180,34	77,27
7	180,34	72,73
8	187,96	75,00
9	172,72	68,18
10	177,80	70,46
11	179,07	86,36

@MATURATO2

@MATURATO(*liquidazione*; *scadenza*; *cedola*; [*pari*]; [*frequenza*]; [*emissione*]; [*primo_interesse*]; [*tipo*]) calcola gli interessi maturati sui titoli con pagamento periodico degli interessi, in base alle convenzioni giapponesi.

Argomenti

liquidazione è il giorno dei compensi di un titolo e consiste in un valore data.

scadenza è il giorno in cui viene rimborsato un titolo e deve essere un valore data. Se *scadenza* è inferiore o uguale a *liquidazione*, la funzione @MATURATO2 genera ERR.

cedola è il tasso annuo nominale di un titolo e consiste in un valore positivo o uguale a zero, espresso da un numero decimale.

pari è un argomento facoltativo che specifica il valore di parità o nominale di un titolo, cioè il capitale che deve essere pagato alla data di scadenza. *pari* è un valore positivo. Se non viene specificato l'argomento *pari*, 1-2-3 usa il valore 100.

frequenza è un argomento facoltativo che indica il numero di pagamenti delle cedole previsti in un anno. *frequenza* può assumere i valori riportati nella tabella qui sotto:

<u>frequenza</u>	<u>Pagamento delle cedole</u>
1	Annuale
2	Semiannuale (impostazione predefinita)

emissione è un argomento facoltativo che rappresenta la data di emissione o di inizio del titolo e consiste in un valore data. Se *emissione* è maggiore di *liquidazione*, @MATURATO2 genera ERR.

primo_interesse è un argomento facoltativo che rappresenta la prima scadenza degli interessi del titolo e consiste in un valore data. Se *primo_interesse* è inferiore o uguale a *emissione* o maggiore di *scadenza*, la funzione @MATURATO2 genera ERR.

tipo è un argomento facoltativo che specifica il tipo di titolo. Scegliere il valore di *tipo* dalla seguente tabella:

<u>tipo</u>	<u>Descrizione</u>
0	titolo di stato giapponese; impostazione predefinita in caso di omissione dell'argomento
1	titolo pubblico o industriale

Non è consentito usare un argomento facoltativo senza specificare quelli che lo precedono.

Esempi

Un titolo di stato giapponese viene acquistato per ¥100.000 il 14 Giugno 1996. La data di emissione del titolo è il 1° Maggio 1990 e la data di scadenza il 20 Giugno 2000. Il tasso di interesse è del 9,00% e gli interessi vengono pagati due volte all'anno. Il primo interesse viene pagato il 21 Dicembre 1990.

Gli interessi maturati sono:

@MATURATO2(@DATA(96;6;14);@DATA(2000;6;20);0,09;100000;2;@DATA(90;5;1);@DATA(90;12;21)) = ¥4.339

Funzioni @ affini

@MATURATO calcola gli interessi maturati sui titoli che prevedono pagamenti periodici degli interessi. @PREZZO2 calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a ¥100, in base alle convenzioni giapponesi. @RENDIMENTO2 calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi, secondo le convenzioni giapponesi.

@PREZZO calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a \$100.

@RENDIMENTO1 calcola il rendimento di titoli con pagamento periodico degli interessi. @DURATA calcola la durata annua e @DURATAM calcola la durata annua modificata di un titolo con pagamento periodico degli interessi.

@COLLDATI

@COLLDATI(*nome_applicazione*; *nome_file*; *nome_elemento*; [*formato*]; [*righe_max*]; [*colonne_max*]; [*fogli_max*]) crea un collegamento OLE con dei dati.

Per modificare il collegamento, cambiare gli argomenti di @COLLDATI.

Argomenti

nome_applicazione è una stringa di testo che specifica il nome di un'applicazione Windows che supporta il collegamento server OLE. Questo argomento non viene usato da 1-2-3 97, ma serve per mantenere la compatibilità con le precedenti versioni di 1-2-3.

nome_file è una stringa di testo che specifica il nome del file dell'applicazione con cui si intende stabilire il collegamento.

nome_elemento è una stringa di testo che specifica il nome dell'elemento dell'applicazione server a cui si intende collegarsi. Tale elemento del file dell'applicazione server è quello da cui verranno trasferiti i dati attraverso il collegamento.

formato è una stringa di testo che specifica uno dei formati degli Appunti e può essere Testo, WK1 o WK3.

In caso di omissione, 1-2-3 usa il formato Testo degli Appunti.

righe_max, *colonne_max* e *fogli_max* specificano il numero massimo di righe, colonne e fogli della zona di destinazione. Se si omettono questi argomenti, 1-2-3 usa tutte le righe, colonne e fogli necessari.

Note

Quando si crea un collegamento con Modifica - Incolla speciale e si sceglie il tipo "Testo", WK1 o WK3, la funzione @COLLDATI compare nella prima cella della zona di destinazione del collegamento.

I collegamenti creati vengono denominati automaticamente "CollDati 1," "CollDati 2" e così via.

Per ulteriori informazioni, vedere [Creazione di un collegamento OLE in 1-2-3](#).

Esempi

La seguente formula della funzione @COLLDATI crea un collegamento OLE con il file di Word Pro MUTUO.LWP.
@COLLDATI("WordPro";"C:\LOTUS\WORK\WORDPRO\MUTUO.LWP";"!Collega_segnaibro1")

@CONVERTDATA

@CONVERTDATA(*data;tipo_immissione;tipo_risultato*) converte una data del calendario Hijri (arabo), Farsi (iraniano) o ebraico nella data corrispondente del calendario gregoriano o vice versa.

Note

@CONVERTDATA è una funzione specifica delle lingue bidirezionali e funziona solo su computer che supportano le versioni bidirezionali di 1-2-3 (ad esempio, quella ebraica o Farsi).

@DECILE

@DECILE(x;zona) calcola l'x-esimo decile.

Argomenti

x è un numero intero compreso tra 0 e 10, estremi compresi.

x	Risultato	Uguale a...
0	Minimo	@PERCENTILE(0,0;zona) o @MIN(zona)
1	Primo decile	@PERCENTILE(0,1;zona)
2	Secondo decile	@PERCENTILE(0,2;zona)
3	Terzo decile	@PERCENTILE(0,3;zona)
4	Quarto decile	@PERCENTILE(0,4;zona)
5	Mediano	@PERCENTILE(0,5;zona) o @MEDIAN(zona)
6	Sesto decile	@PERCENTILE(0,6;zona)
7	Settimo decile	@PERCENTILE(0,7;zona)
8	Ottavo decile	@PERCENTILE(0,8;zona)
9	Nono decile	@PERCENTILE(0,9;zona)
10	Massimo	@PERCENTILE(1,0;zona) o @MAX(zona)

zona è l'indirizzo o il nome di una zona contenente valori.

Note

@DECILE considera le celle vuote e le celle con etichette uguali a 0.

Esempi

La zona denominata DATI contiene i seguenti valori:

34
12
60
128
67
350
206

@DECILE(0;DATI) = 12,0

@DECILE(1;DATI) = 25,2

@DECILE(2;DATI) = 39,2

Funzioni @ affini

@PERCENTILE calcola l'x-esimo percentile campionario dei valori di una zona. @QUARTILE calcola un dato quartile.

@MIN riporta il valore più basso di una serie. @MAX riporta il valore più alto di una serie. @MEDIANA calcola il valore mediano di una serie.

@NUMEROA

@NUMEROA(*stringa_numerica*) converte *stringa_numerica* dai caratteri numerici thailandesi a una stringa in numeri arabi.

Argomenti

stringa_numerica è una stringa di caratteri numerici thailandesi.

Note

Usare **@NUMEROA** in combinazione con **@VALORE** per convertire i dati immessi sotto forma di stringhe numeriche thailandesi in valori numerici utilizzabili nei calcoli.

Esempi

@EDIGIT("๑๖ สิ. ๓๔") = 16 สิ. 34

Funzioni @ affini

@NUMEROT converte i *caratteri_numerici* espressi in numeri arabi in una stringa di caratteri numerici thailandesi.

@VALORE converte un numero immesso come stringa di testo nel valore corrispondente.

@RICERB

@RICERB(*stringa_ricerca;stringa;numero_inizio*) calcola la posizione (in byte) della *stringa* in cui 1-2-3 trova la prima occorrenza di *stringa_ricerca*, incominciando dalla posizione indicata da *numero_inizio*.

Argomenti

stringa_ricerca e *stringa* possono essere stringhe di testo racchiuse fra virgolette (" "), una formula che genera una stringa di testo oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta..

numero_inizio è uno scarto espresso in byte.

Note

Se 1-2-3 non trova *stringa_ricerca* in *stringa*, **@RICERB** genera il valore ERR. **@RICERB** genera ERR anche quando *numero_inizio* è maggiore del numero di byte di *stringa*, oppure se *numero_inizio* è negativo.

@RICERB fa distinzione tra maiuscole e minuscole e tra lettere accentate e non; ad esempio, **@RICERB** non troverà la *stringa_ricerca* "e" nella *stringa* "FIRENZE".

@RICERB è utile anche in combinazione con **@MEZZOB** o **@SOSTITB** per individuare ed estrarre o sostituire una stringa di testo.

Esempi

@RICERB("p";"Fatture in pagamento";0) = 12 perché la *stringa_ricerca* p si trova nella dodicesima posizione della *stringa* Fatture in pagamento.

Funzioni @ affini

@RICER calcola la posizione in caratteri di una stringa di ricerca in una stringa.

@PREVISIONE

@PREVISIONE(x;zona_y;zona_x) dà il valore previsto di *x* sulla base di una tendenza lineare tra i valori contenuti nella *zona_y* e nella *zona_x*.

Argomenti

x può essere un valore qualunque e rappresenta il valore da prevedere.

zona_y è un indirizzo di zona o il nome di una zona e rappresenta la serie di variabili dipendenti. Se *zona_y* contiene dati non numerici, **@PREVISIONE** produce il valore **ERR**.

zona_x è un indirizzo di zona o il nome di una zona e rappresenta la serie di variabili indipendenti. Se *zona_x* contiene dati non numerici, la funzione **@PREVISIONE** produce il valore **ERR**.

zona_y e *zona_x* devono avere le stesse dimensioni, ovvero contenere lo stesso numero di celle. Le celle delle due zone vengono accoppiate in ordine di successione. Le zone sono ordinate dall'alto verso il basso (verticali) e da sinistra a destra (orizzontali) e, infine, per fogli. Se la *zona_y* e la *zona_x* non sono delle stesse dimensioni, la funzione **@PREVISIONE** produce **ERR**.

Note

Si noti che questi calcoli vengono eseguiti anche dalla funzione **@REGRESSIONE**.

Esempi

Dati i seguenti valori:

ZONAY	ZONAX
250	3
545	5
550	5
450	6
605	6
615	7

@PREVISIONE(-1;ZONA_Y;ZONA_X) = -26,786

@PREVISIONE(0;ZONA_Y;ZONA_X) = 56,786

Funzioni @ affini

@REGRESSIONE esegue una regressione lineare multipla e genera la statistica specificata.

@FULLP

@FULLP(*etichetta*) converte i caratteri a un solo byte (ASCII) di *etichetta* nei corrispondenti caratteri giapponesi a due byte.

Argomenti

etichetta è una stringa di testo contenente i caratteri ASCII da convertire. Se *etichetta* non è una stringa, la funzione @FULLP produce ERR.

Note

@FULLP converte i caratteri ASCII di *etichetta* nei corrispondenti caratteri a due byte. I caratteri del DBCS o Double-byte character set (set di caratteri a due byte) non vengono invece convertiti e la funzione @FULLP li rigenera così come sono. @FULLP non converte i caratteri Katakana del set di caratteri a un solo byte nei caratteri Katakana a due byte.

Funzioni @ affini

@HALFP converte i caratteri giapponesi a due byte in caratteri a un solo byte.

@VALFUT2

@VALFUT2(*versamenti;interessi;durata*) calcola il valore futuro di un investimento sulla base di una serie di *versamenti* di uguale importo, di un tasso di *interesse* periodico e del numero di periodi specificati da *durata*, presupponendo una rendita anticipata.

Argomenti

versamenti rappresenta la cifra corrisposta in ogni periodo e può essere un valore qualunque.

durata è il numero di periodi di versamento e può essere un qualunque numero intero. I numeri non interi vengono troncati.

interesse è il tasso di interesse periodico e può essere un valore decimale o percentuale maggiore di -1.

Note

Il periodo usato per calcolare l'*interesse* deve essere uguale a quello utilizzato per *durata*; ad esempio, se si intende calcolare una rata mensile, occorre immettere l'interesse e la durata in incrementi mensili. Solitamente ciò significa che il tasso di interesse deve essere diviso per 12 e che il numero di anni di *durata* deve essere moltiplicato per 12.

Esempi

Si supponga di voler depositare ¥500,000 ogni anno per i prossimi vent'anni in un conto pensione. Questo conto rende un interesse composto del 7,5%, calcolato annualmente e corrisposto l'ultimo giorno dell'anno. A questo punto si vuol calcolare il valore che il conto assumerà tra vent'anni. Il versamento annuale viene effettuato il primo giorno dell'anno.

@VALFUT2(500000;0,075;20) = ¥23,276,266, che è il valore del conto al termine dei venti anni.

Si può ottenere lo stesso risultato con **@VALFUTURO**(500000;0.075;20;1).

Funzioni @ affini

@VALFUT calcola il valore futuro di un investimento sulla base di una serie di versamenti di uguale importo, di un tasso d'interesse periodico e del numero di periodi specificati dalla durata, presupponendo una rendita immediata.

@VALFUTURO calcola il valore futuro di un investimento di un valore attuale specificato, per una rendita immediata o anticipata.

@VALATT e **@VALATTUALE** calcolano il valore attuale di un investimento. **@VAN** calcola il valore attuale netto di un investimento, attualizzando il valore futuro a quello presente.

@CAPFUT

@CAPFUT(*capitale;interesse;durata;[frequenza]*) genera il valore futuro di una somma complessiva investita a un certo tasso di interesse per un certo numero di periodi.

Argomenti

capitale rappresenta la somma investita inizialmente e può essere un valore qualsiasi.

interesse rappresenta il tasso di interesse periodico e può essere un numero decimale o una percentuale.

durata rappresenta il numero di periodi previsti per l'investimento e può essere un valore qualunque.

frequenza è un argomento facoltativo e rappresenta la frequenza di capitalizzazione per periodo. I numeri non interi vengono troncati. L'impostazione predefinita è 1.

Non è consentito l'uso di un argomento opzionale in assenza di quelli che lo precedono.

Note

Se *durata* è 1 e *frequenza* è 1, **@CAPFUT** calcola interessi semplici.

Esempio

Si supponga che al decimo compleanno del proprio figliolo si depositino \$100 in un conto risparmio che frutta interessi del 10%. Con la funzione **@CAPFUT**, si può calcolare la cifra che conterrà il conto quando il ragazzo festeggerà il suo 18° compleanno:

@CAPFUT(100;10%;8) = \$214.36

Funzioni @ affini

@CAPFUT è adatta solo per investimenti realizzati in un'unica soluzione. **@VALFUT** e **@VALFUTURO** calcolano il valore futuro di una rendita, ovvero di un solo o più versamenti. **@CAPATTUALE** genera il valore attuale di una somma complessiva dariceversi in un dato numero di periodi futuri e attualizzata a un dato tasso di interesse.

@HALFP

@HALFP(*etichetta*) converte i caratteri giapponesi a due byte nei corrispondenti caratteri (ASCII) a un solo byte.

Argomenti

etichetta è una stringa di testo contenente i caratteri giapponesi a due byte da convertire. Se *etichetta* non è una stringa di testo, @HALFP produce ERR.

Note

@HALFP converte le lettere, i numeri e i simboli (come + e -) del set di caratteri giapponesi a due byte (DBCS) nei corrispondenti caratteri ASCII. I caratteri ASCII e i caratteri che non hanno corrispondenti valori ASCII non vengono convertiti e la funzione @HALFP li lascia così come sono. @HALFP non converte il set di caratteri giapponesi a due byte Katakana nel set Katakana a un solo byte.

Funzioni @ affini

@FULLP converte caratteri a un solo byte in caratteri giapponesi a due byte.

@DCOMPRESO

@DCOMPRESO(*valore*; *limite1*; *limite2*; [*estremi*]) verifica se *valore* è compreso tra *limite1* e *limite2*. @DCOMPRESO genera 1 (vero) se *valore* cade tra *limite1* e *limite2* oppure 0 (falso) in caso contrario.

Argomenti

valore, *limite1* e *limite2* possono essere valori numeri o stringhe di testo.

estremi è un argomento facoltativo e può assumere i seguenti valori:

<u><i>estremi</i></u>	<u>Descrizione</u>	<u>Algoritmo</u>
0	Compresi <i>limite1</i> e <i>limite2</i> (impostazione predefinita)	@MIN(<i>limite1</i> , <i>limite2</i>) <= <i>valore</i> <= @MAX(<i>limite1</i> , <i>limite2</i>)
1	Compreso il valore inferiore tra <i>limite1</i> e <i>limite2</i> ed escluso il maggiore	@MIN(<i>limite1</i> , <i>limite2</i>) <= <i>valore</i> < @MAX(<i>limite1</i> , <i>limite2</i>)
2	Escluso il valore inferiore tra <i>limite1</i> e <i>limite2</i> e compreso il maggiore	@MIN(<i>limite1</i> , <i>limite2</i>) < <i>valore</i> <= @MAX(<i>limite1</i> , <i>limite2</i>)
3	Esclusi entrambi gli estremi, <i>limite1</i> e <i>limite2</i>	@MIN(<i>limite1</i> , <i>limite2</i>) < <i>valore</i> < @MAX(<i>limite1</i> , <i>limite2</i>)

Note

Se il tipo di dati rappresentato da *limite1* è diverso da quello rappresentato da *limite2*, @DCOMPRESO produce ERR.

Se il tipo di dati rappresentato da *limite1* o *limite2* è diverso da quello di *valore*, @DCOMPRESO genera 0.

Non è ammesso l'uso di un argomento facoltativo in assenza di quelli che lo precedono.

Esempi

@DCOMPRESO(123;100;200) = 1

@DCOMPRESO("ABC";"AAA";"BBB") = 1

@DCOMPRESO(100;100;200;0) = 1

@SINISTRAB

@SINISTRAB(*stringa*; *n*) fornisce i primi *n* byte di *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

n può essere un numero intero positivo o 0. Se *n* è 0, il risultato è una stringa vuota. Se *n* è maggiore della lunghezza in byte di *stringa*, la funzione @SINISTRAB riporta l'intera *stringa*.

Note

@SINISTRAB conta come caratteri anche i segni di interpunzione e gli spazi.

@SINISTRAB viene usata per i set di caratteri a due byte (DBCS), come il set giapponese. Ad esempio, se si vuole mettere un'etichetta in un campo di un database esterno attraverso 1-2-3 e il campo ha una lunghezza limitata in byte, è possibile estrarre parti dell'etichetta per farla rientrare entro tale limite.

Se l'ultimo byte dell'etichetta generata costituisce la prima metà di un carattere a due byte, @SINISTRAB sostituisce il carattere incompleto con uno spazio.

Se la *stringa* è formata solo da caratteri a un byte, la funzione @SINISTRAB genera lo stesso risultato della funzione @SINISTRA.

Esempi

@SINISTRAB("Un solo byte";8) = " "

Funzioni @ affini

@SINISTRA fornisce i primi *n* caratteri di *stringa*. @MEZZOB ricava un certo numero di byte da una *stringa*.

DESTRAB fornisce gli ultimi *n* byte di *stringa*.

@LENGTHB

@LENGTHB(*stringa*) conta i byte di *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

Note

@LENGTHB conta come caratteri anche i segni di interpunzione e gli spazi.

@LENGTHB viene usata per i set di caratteri a due byte (DBCS), come il set giapponese. Ad esempio, se si vuole mettere un'etichetta in un campo di un database esterno attraverso 1-2-3 e il campo ha una lunghezza limitata in byte, è possibile controllare se il testo supera il limite.

Esempi

@LENGTHB(A5&G12) = numero totale di byte nelle celle A5 e G12.

Funzioni @ affini

@LUNGH conta i caratteri in *stringa*.

@MEZZOB

@MEZZOB(*stringa*; *numero_inizio*; *n*) copia *n* byte di *stringa*, a partire dal byte situato nella posizione specificata da *numero_inizio*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

numero_inizio è uno scarto in byte. Se *numero_inizio* è maggiore della lunghezza di *stringa*, la funzione @MEZZOB produce una stringa vuota.

n può essere qualsiasi numero intero positivo o uno 0. Se *n* è uguale a 0, @MEZZOB genera una stringa vuota. Se *n* è maggiore della lunghezza di *stringa*, 1-2-3 riporta tutti i byte a partire da *numero_inizio* sino alla fine di *stringa*.

Note

@MEZZOB viene usata per i set di caratteri a due byte (DBCS), come il set giapponese.

@MEZZOB conta come caratteri anche i segni di interpunzione e gli spazi.

Se non si conosce la lunghezza di *stringa*, si consiglia di usare un numero alto per *n*; 1-2-3 ignora gli spazi in eccesso, riportando l'intera *stringa* a partire da *numero_inizio*.

Se il primo byte dell'etichetta generata costituisce l'ultima metà di un carattere a due byte o se l'ultimo byte dell'etichetta generata costituisce la prima metà di un carattere a due byte, la funzione @MEZZOB sostituisce il carattere incompleto con uno spazio.

Esempi

@MEZZOB("Testo di un solo byte";7;4) = "i un"

Funzioni @ affini

@MEZZO copia *n* caratteri di una stringa, a partire dal carattere situato nella posizione specificata da *numero_inizio*.

@SINISTRAB fornisce i primi *n* byte di *stringa* @DESTRAB fornisce gli ultimi *n* byte di *stringa*.

@SOMMAN

@SOMMAN(*scarto*; *n*; *lista*) somma ogni *n*-esimo valore di *lista*, partendo da *scarto*.

@SOMMAN ignora i valori ERR e ND nelle celle/valori saltati.

Argomenti

lista può comprendere una combinazione qualsiasi dei seguenti elementi: numeri, formule numeriche e indirizzi o nomi di zona contenenti numeri o formule. Ricordarsi di separare tra loro i vari elementi di *lista* con gli appositi separatori di argomenti.

scarto può essere un numero intero qualunque.

n può essere un numero intero qualunque.

Note

@SOMMAN produce la somma di (*scarto*), (*scarto* + *n*), (*scarto* + *2n*), (*scarto* + *3n*), ... entro la *lista*.

Il risultato della funzione @SOMMAN(0;1;*lista*) è uguale a quello di @SOMMA(*lista*).

Esempi

@SOMMAN(1;3;B5..B15) produce la somma dei valori contenuti in B6, B9, B12 e B15.

Funzioni @ affini

@SOMMA somma i valori di *lista*. @SOMMADB calcola la somma dei valori che soddisfano le *condizioni* specificate.

@TOTPARZ somma i valori di una lista e indica alla funzione @TOTGENER quali valori deve sommare.

@STRINGANUM

@STRINGANUM(*numero*;*tipo*) converte *numero* nel corrispondente testo in caratteri giapponesi, utilizzando il formato specificato in *tipo*.

Argomenti

numero può essere un qualsiasi numero arrotondato compreso tra 0 e 9.999.999.999.999.999. 1-2-3 arrotonda *numero* al numero intero più vicino. Se il numero specificato non rientra nei limiti prestabiliti, la funzione @STRINGANUM produce ERR.

tipo specifica il formato e può assumere i seguenti valori:

<i>tipo</i>	Risultato
1	Notazione numerica Kanji regolare
2	Notazione numerica Kanji formale
3	Conversione semplice, cifra per cifra, in numeri Kanji

Esempi

@NUMBERSTRING(1234567890,1) = 十二億三千四百五十六万七千八百九十

@NUMBERSTRING(1234567890,2) = 拾貳億參千四百五拾六万七千八百九拾

@NUMBERSTRING(1234567890,3) = 一 二 三 四 五 六 七 八 九 〇

@PAGAM2

@PAGAM2(*capitale;interesse;durata*) calcola le rate di rimborso di un prestito (*capitale*) a un dato tasso di *interesse* per un numero specificato di periodi di pagamento (*durata*), presupponendo una rendita anticipata.

Argomenti

capitale rappresenta il valore del prestito e può assumere un valore qualsiasi.

interesse rappresenta il tasso di interesse sul prestito e può essere un numero decimale o un valore percentuale maggiore di -1.

durata rappresenta il numero di periodi di pagamento e deve essere un numero intero. I numeri non interi vengono troncati.

Note

Il periodo usato per calcolare l'*interesse* deve essere uguale a quello usato per *durata*. Ad esempio, supponendo di calcolare una rata mensile, si dovranno immettere gli interessi e la durata in incrementi mensili. Solitamente ciò significa che occorre dividere il tasso d'interesse per 12 e moltiplicare per 12 il numero di anni della *durata*.

A differenza di **@PAGAM**, **@PAGAM2** per il calcolo usa la fine del periodo, anziché l'inizio.

Esempi

Si vuole considerare la possibilità di accendere un prestito di ¥8,000 per 3 anni a un tasso di interesse annuo del 10,5%, capitalizzato mensilmente. Le rate vanno versate l'ultimo giorno di ogni mese. La seguente funzione calcola l'ammontare della rata mensile:

@PAGAM2(8000;0.105/12;36) = ¥257.76

@PAGAMENTO(8000;0.105/12;36;1) produce lo stesso risultato.

Funzioni @ affini

@PAGAM e **@PAGAMENTO** calcolano le rate di rimborso di un prestito (*capitale*) a un dato tasso di interesse per un numero specificato di periodi di pagamento, presupponendo una rendita immediata. **@IPAGAM** calcola la porzione costituita dagli interessi di una rata costante.

@INTRATA calcola la porzione costituita dagli interessi cumulativi della rata periodica da pagare per un prestito.

@CAPITRATA calcola la porzione costituita dal capitale della rata periodica da pagare per un prestito. **@TERM** calcola il numero di versamenti da effettuare per un investimento. **@TERM2** calcola il tempo necessario per accumulare un capitale di *valore_futuro*, effettuando *versamenti* periodici di uguale importo, dato un tasso d'*interesse* periodico, secondo la convenzione della rendita anticipata.

@IPAGAM

@IPAGAM(*capitale;interesse;durata;rata*) calcola la porzione costituita dagli interessi di una rata costante.

Argomenti

capitale rappresenta il valore del prestito e può assumere qualunque valore.

interesse rappresenta il tasso di interesse del prestito e può essere un numero decimale o un valore percentuale maggiore di -1.

durata rappresenta il numero di periodi di pagamento e deve essere un numero intero. I numeri non interi vengono troncati.

rata rappresenta la rata per cui si desidera calcolare gli interessi e può essere un qualunque numero positivo intero minore o uguale a *durata*. I numeri non interi vengono troncati.

Note

Il periodo usato per calcolare l'*interesse* deve essere uguale a quello utilizzato per *durata*; ad esempio, se si intende calcolare una rata mensile, occorre immettere l'interesse e la durata in incrementi mensili. Solitamente ciò significa che il tasso di interesse deve essere diviso per 12 e che il numero di anni di *durata* deve essere moltiplicato per 12.

@INTRATA(*capitale;interesse;durata;rata;rata;0*) produce lo stesso risultato della funzione

@IPAGAM(*capitale;interesse;durata;rata*).

Esempi

Si è acceso un prestito di \$8,000 per tre anni al tasso di interesse annuo del 10,5%, capitalizzato mensilmente. Le rate mensili sono di \$260,02. Per calcolare la porzione costituita dagli interessi della prima rata, usare questa funzione:

@IPAGAM(8000;0.105/12;36;1) = \$70

Funzioni @ affini

@PAGAM e @PAGAMENTO calcolano le rate per il rimborso di un prestito (*capitale*) a un dato tasso d'*interesse* per un numero specificato di periodi di pagamento, secondo il metodo della rendita immediata. @PAGAM2 calcola le rate di rimborso di un prestito a un dato tasso d'interesse per un numero specificato di periodi di pagamento, con il metodo della rendita anticipata.

@IPS calcola la porzione costituita dagli interessi di una rata per il rimborso di un prestito laddove il pagamento di capitale rimane uguale per ogni pagamento.

@INTRATA calcola la porzione costituita dagli interessi cumulativi della rata periodica da pagare per un prestito, dato il tasso di interesse per un certo numero di periodi di pagamento.

@PREZZO2

@PREZZO2(*liquidazione*; *scadenza*; *cedola*; *rendimento*; [*rimborso*]; [*base*]) calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a ¥100, in base alle convenzioni giapponesi.

Argomenti

liquidazione è il giorno in cui avviene il pagamento dei compensi e consiste in un valore data.

scadenza è il giorno in cui viene rimborsato un titolo e deve essere un valore data. Se *scadenza* è inferiore o uguale a *liquidazione*, la funzione @PREZZO2 genera ERR.

cedola è il tasso annuo nominale di un titolo e consiste in un valore positivo o uguale a zero.

rendimento è il rendimento annuale, espresso con un valore decimale positivo.

rimborso è un argomento facoltativo che specifica il valore di rimborso di un titolo con valore nominale pari a ¥100. Questo argomento deve essere costituito da un valore positivo o da zero. In caso di omissione, 1-2-3 usa un rimborso pari a 100.

base è un argomento facoltativo che indica il tipo di base per il conteggio dei giorni da usare:

base	Base per il conteggio dei giorni
0	30/360
1	effettivo/effettivo
2	effettivo/360
3	effettivo/365
4	europeo 30/360

Se si specifica la *base*, 1-2-3 richiama la funzione @PREZZO per calcolare il valore.

Esempi

Un'obbligazione ha come data di liquidazione il 18 di Luglio 1996 e come data di scadenza il 20 Luglio 2000. Il tasso nominale d'interesse semestrale è pari al 5,50% e il rendimento annuale è del 5,61%. La seguente formula calcola il prezzo dell'obbligazione:

@PREZZO2(@DATA(96;7;18),@DATA(100;6;20);0,055;0,0561;100) = ¥99,65

Funzioni @ affini

@PREZZO calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a \$100.

@MATURATO calcola gli interessi maturati sui titoli che prevedono pagamenti periodici degli interessi.

@RENDIMENTO calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi. @DURATA calcola la durata annua e @DURATAM calcola la durata annua modificata di titoli con pagamento periodico degli interessi.

@MATURATO2 calcola gli interessi maturati sui titoli con pagamento periodico degli interessi, in base alle convenzioni giapponesi. @RENDIMENTO2 calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi, secondo le convenzioni giapponesi.

@PROD

@PROD(*lista*) moltiplica i valori compresi in una *lista*.

Argomenti

La *lista* può contenere i seguenti elementi in qualunque combinazione: numeri, formule numeriche e indirizzi o nomi di zone contenenti numeri o formule numeriche. Tali elementi devono essere divisi tra loro dagli appositi separatori di argomenti.

Esempi

@PROD(2; 4; 6; 8) = 384

Funzioni @ affini

@FATTORIALE calcola il fattoriale di un valore. @SOMMA somma i valori di una lista. @SOMMAPROD calcola la somma dei prodotti di valori corrispondenti situati in zone diverse.

@MEDIANAPURA

@MEDIANAPURA(*lista*) calcola il valore mediano di una *lista*, ignorando le celle vuote, le etichette e le formule che producono etichette.

Argomenti

lista può contenere una qualsiasi combinazione dei seguenti elementi: numeri, formule numeriche e indirizzi o nomi di zona contenenti numeri o formule numeriche. Gli elementi della *lista* vanno divisi tra loro dagli appositi separatori di argomenti.

Note

Il valore mediano è il valore di mezzo di *lista*, ovvero il numero di cui esistono altrettanti numeri maggiori quanti inferiori. Se la *lista* contiene un numero dispari di valori, **@MEDIANAPURA** ordina i valori e produce il valore mediano. Se invece il numero di valori è pari, la funzione ordina i valori produce la media aritmetica dei due valori mediani.

Se la zona è vuota o contiene solo etichette o formule che producono etichette, **@MEDIANAPURA** genera **ERR**.

Esempi

@MEDIANAPURA(A2..A6) = 12, dove A2..A6 contiene i valori 5, 12, 65, 82 e 9.

@MEDIANAPURA(A1..A6) = 12, dove A1..A6 contiene i valori 5, 12, 65, 82 e 9 e l'etichetta "Gennaio". (**@MEDIANAPURA** ignora l'etichetta.) **@MEDIANA** calcolerebbe l'etichetta pari a 0 e produrrebbe quindi il valore 10,5.

Funzioni @ affini

@MEDIANA produce il valore mediano di una lista.

@VALATT2

@VALATT2(*versamenti;interesse;durata*) calcola il valore attuale di un investimento, sulla base di una serie di *versamenti* di uguale importo, attualizzati a un tasso d'*interesse* periodico per la *durata* specificata, usando la convenzione della rendita anticipata.

Argomenti

versamenti rappresenta l'ammontare dei versamenti e può assumere qualunque valore.

interesse rappresenta il tasso d'interesse di un investimento e può essere un numero decimale o un valore percentuale maggiore di -1.

durata rappresenta il numero di periodi di versamento e può essere un qualunque numero intero positivo. I numeri non interi vengono troncati.

Note

Il periodo usato per calcolare l'*interesse* deve essere uguale a quello utilizzato per *durata*; per esempio, se si intende calcolare un versamento mensile, occorre immettere l'interesse e la durata in incrementi mensili. Solitamente, ciò significa che il tasso di interesse deve essere diviso per 12 e che il numero di anni di *durata* deve essere moltiplicato per 12.

A differenza di @VALATT, @VALATT2 per i calcoli usa la fine della durata, anziché l'inizio.

Usare @VALATT2 per valutare un investimento o per confrontarlo con altri investimenti. @VALATT2 è utile anche per confrontare tra loro diversi tipi di investimenti -- ad esempio per valutare se, giunti allo scadere di una polizza pensionistica, sia meglio versare l'intero capitale maturato oppure optare per una serie di versamenti rateali. Per creare una tabella di ammortamento, usare @VALATT2 con @PAGAM2.

@VALATT2 è complementare a @PAGAM in quanto @VALATT2 informa sull'importo massimo che si può prendere a prestito, dato un tetto massimo per la rata mensile. Viceversa, @PAGAM informa sull'entità delle rate mensili, dato il prestito che si desidera accendere.

Esempi

Si supponga di aver vinto un premio di ¥1.000.000 che può essere riscosso in 20 rate di ¥50.000 con scadenza a fine anno oppure in un'unica soluzione di ¥400,000 anziché con una rendita di ¥1.000.000. Si vuole scoprire quale possibilità sia più conveniente dato l'attuale corso dello yen.

Accettando la rateizzazione annuale di ¥50.000, si prevede di investire questa somma a un interesse del 8%, capitalizzato annualmente.

@VALATT2(50000;0,08;20) produce il valore ¥530,180; ciò significa che l'importo di ¥1,000,000 riscosso in 20 anni ha un valore attuale di ¥530,180, al valore attuale dello yen.

Si può ottenere la stessa risposta anche con la funzione @VALATTUALE(50000;0,08;20;1)

Funzioni @ affini

@VALATT calcola il valore attuale di un investimento, sulla base di una serie di *versamenti* di uguale importo, attualizzati a un tasso d'*interesse* periodico per la *durata* specificata, secondo la convenzione della rendita immediata. @VALATTUALE determina il valore attuale di un investimento dal *valore_futuro* specificato, per una rendita immediata o anticipata.

@VALFUT e @VALFUTURO calcolano il valore futuro di un investimento sulla base di una serie di versamenti di uguale importo. @VALFUT2 calcola il valore futuro di un investimento sulla base di una serie di *versamenti* di uguale importo, di un tasso di *interesse* periodico e del numero di periodi specificati da *durata*, presupponendo una rendita anticipata. @VAN calcola il valore attuale netto di un investimento, attualizzando il valore futuro a quello presente.

@PAGAM e @PAGAMENTO calcolano le rate di rimborso di un prestito (*capitale*) a un dato tasso di interesse per un numero specificato di periodi di pagamento. @PAGAM2 calcola le rate di rimborso di un prestito (*capitale*) a un dato tasso di *interesse* per un numero specificato di periodi di pagamento (*durata*), presupponendo una rendita anticipata.

@CAPATTUALE

@CAPATTUALE(*valore_futuro;interesse;durata;[frequenza]*) genera il valore attuale di una somma complessiva dariceversi in un dato numero di periodi futuri e attualizzata a un dato tasso di interesse.

Argomenti

valore_futuro rappresenta la somma da riceversi in futuro e può assumere qualsiasi valore.

interesse rappresenta il tasso di attualizzazione periodico e può essere un numero decimale o un valore percentuale.

durata rappresenta il numero di periodi futuri durante i quali si riceverà il capitale e può assumere qualunque valore.

frequenza è un argomento facoltativo e rappresenta la frequenza di capitalizzazione per periodo di pagamento. I numeri non interi vengono troncati. L'impostazione predefinita di frequenza è 1.

Note

Se *durata* è 1 e *frequenza* è 1, **@CAPATTUALE** calcola interessi semplici.

Esempi

Si supponga che sia necessario pagare una retta scolastica di \$10,000 per i prossimi dieci anni e di beneficiare di un interesse del 10% sul proprio capitale. La funzione **@CAPATTUALE** può servire a calcolare la somma da investire oggi:

@CAPATTUALE(10000,10%,10) = \$3.855,43

Funzioni @ affini

@VALATT e **@VALATTUALE** calcola il valore attuale di una rendita costituita da uno o più versamenti. **@CAPFUT** genera il valore futuro di una somma complessiva investita a un certo tasso di interesse per un certo numero di periodi.

@QUARTILE

@QUARTILE(x;zona) genera un dato quartile.

Argomenti

x è un numero intero compreso tra 0 e 4, estremi compresi. Usare uno dei valori compresi nella seguente tabella:

x	Risultato	Uguale a...
0	Minimo	@PERCENTILE(0,0;zona) o @MIN(zona)
1	Primo quartile	@PERCENTILE(0,25;zona)
2	Mediano	@PERCENTILE(0,50;zona) o @MEDIAN(zona)
3	Terzo quartile	@PERCENTILE(0,75;zona)
4	Massimo	@PERCENTILE(1,00;zona) o @MAX(zona)

zona è l'indirizzo o il nome di una zona contenente valori.

Note

@QUARTILE considera le celle vuote e le celle con etichette uguali a 0.

Esempi

La zona denominata DATI contiene i seguenti valori:

34
12
60
128
67
350
206

@QUARTILE(0;DATI) = 12

@QUARTILE(1;DATI) = 47

@QUARTILE(2;DATI) = 67

@QUARTILE(3;DATI) = 167

@QUARTILE(4;DATI) = 350

Funzioni @ affini

@PERCENTILE calcola l'x-esimo percentile campionario dei valori di una zona. @DECILE calcola l'x-esimo decile.

@SOSTITB

@SOSTITB(*stringa_originale*;numero_inizio;n;*stringa_sostit*) sostituisce *n* byte di *stringa_originale* con *stringa_sostit*, a partire da *numero_inizio*.

Argomenti

stringa_originale e *stringa_sostit* possono essere stringhe di testo racchiuse fra virgolette (" "), formule che generano stringhe di testo, oppure gli indirizzi o i nomi di celle contenenti etichette o formule che generano etichette.

numero_inizio è lo scarto di dati in *stringa_originale*; può essere qualsiasi valore positivo o 0. Se *numero_inizio* è maggiore della lunghezza in byte di *stringa_originale*, @SOSTITB aggiunge *stringa_sostit* in fondo a *stringa_originale*.

n può essere qualsiasi numero intero positivo o 0. Se *n* è 0, @SOSTITB aggiunge *stringa_sostit* da *numero_inizio* senza cancellare i caratteri di *stringa_originale*.

Note

Per la posizione e la lunghezza della stringa di testo, @SOSTITB si regola sui byte.

@SOSTITB considera i segni di interpunzione e gli spazi come caratteri. Se si usa @SOSTITB per aggiungere o inserire una stringa di testo, occorre ricordarsi di lasciare gli spazi necessari.

Se i dati sostituiti di *stringa_originale* iniziano o terminano al centro di un carattere a due byte, @SOSTITB sostituisce ogni carattere a due byte incompleto della nuova stringa di testo con uno spazio.

Usare @RICERB con @SOSTITB per cercare e sostituire un'etichetta o per calcolare un *numero_inizio* ignoto.

@SOSTITB è utile per sostituire una serie di caratteri con un'altra serie, ad esempio, per cambiare il prefisso in un database di numeri telefonici.

Esempi

@SOSTITB(CELLA;@RICERB("-";CELLA;0),1,"") copia l'etichetta di CELLA (4-24) sotto forma di 4/24.

Funzioni @ affini

@SOSTIT sostituisce *n* caratteri di *stringa_originale* con *stringa_sostit*, a partire da *numero_inizio*.

@DESTRAB

@DESTRAB(*stringa*; *n*) riporta gli ultimi *n* byte di *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

n può essere un numero intero positivo o 0. Se *n* è pari a 0, il risultato sarà una stringa vuota. Se *n* è maggiore della lunghezza di *stringa*, @DESTRAB riporta l'intera *stringa*.

Note

La funzione @DESTRAB viene usata per i set di caratteri a due byte (DBCS), come il set giapponese.

@DESTRAB conta come caratteri anche i segni di interpunzione e gli spazi.

Se il primo byte della nuova etichetta costituisce l'ultima metà di un carattere a due byte, @DESTRAB sostituisce il carattere incompleto con uno spazio.

Si consiglia di usare @DESTRAB con @RICERB qualora non si conosca il valore esatto di *n* o quando *n* è variabile.

Esempi

@DESTRAB("Un solo byte",4) = "byte"

Funzioni @ affini

@DESTRA riporta gli ultimi *n* caratteri di *stringa*. @SINISTRAB fornisce i primi *n* byte di *stringa*. @MEZZOB produce un certo numero di byte dall'interno di *stringa*.

@IPS

@IPS(*capitale;interesse;durata;rata*) calcola la porzione costituita dagli interessi di una rata, per cui la parte di capitale rimane invariata in ogni periodo.

Argomenti

capitale rappresenta il valore del prestito e può assumere qualsiasi valore.

interesse rappresenta il tasso di interessi sul prestito e può essere un numero decimale o un valore percentuale maggiore di -1.

durata rappresenta il numero di rate e può essere un qualunque numero intero. I numeri non interi vengono troncati.

rata rappresenta lo specifico versamento per cui si vuole calcolare la porzione costituita dagli interessi. *rata* può essere un qualunque numero intero. I numeri non interi vengono troncati. Il conteggio delle rate parte da 0 (zero).

Note

Il periodo usato per calcolare l'*interesse* deve essere uguale a quello utilizzato per *durata*; ad esempio, se si intende calcolare una rata mensile, occorre immettere l'interesse e la durata in incrementi mensili. Solitamente ciò significa che il tasso di interesse deve essere diviso per 12 e che il numero di anni di *durata* deve essere moltiplicato per 12.

Esempi

È stato acceso un prestito di \$8000 per 3 anni al tasso d'interesse del 10,5%, capitalizzato mensilmente. La seguente funzione calcola la porzione costituita dagli interessi della prima rata:

$$\text{@IPS}(8000;0.105/12;36;0) = \$70$$

Questa funzione invece calcola l'ammontare del primo versamento o rata:

$$\text{@IPS}(8000;0.105/12;36;0) + 8000/36 = \$292,22$$

Funzioni @ affini

@IPAGAM calcola la porzione costituita dagli interessi di una rata costante, con versamento sempre nello stesso periodo e per cui la parte costituita dal capitale aumenta al diminuire degli interessi.

@STRINGADATAT

@STRINGADATAT(*valore_data*) converte *valore_data* nella corrispondente stringa che esprime la data nel formato thailandese abbreviato.

Argomenti

valore_data è un valore data.

Esempi

`@TDATESTRING(@DATE(91,8,16)) = 16 สค. 34`

Funzioni @ affini

@STRINGADATATL converte il *valore_data* nella corrispondente stringa di data nel formato thailandese esteso.

@NUMEROT

@NUMEROT(*caratteri_numerici*) converte i *caratteri_numerici* espressi in numeri arabi in una stringa di caratteri numerici thailandesi.

Argomenti

caratteri_numerici è una stringa di *numerali arabi*.

Esempi

`@TDIGIT("16 สค. 34") = ๑๖ สค. ๓๔`

Funzioni @ affini

@NUMEROA converte *caratteri_numerici* espressi in caratteri numerici thailandesi in una stringa in numeri arabi.

@DATAT

@DATAT(*valore_data*) converte *valore_data* nel giorno della settimana secondo il calendario thailandese.

Argomenti

valore_data è un valore data.

Esempi

@TDOW(@DATE(91,8,16)) = ศุกร์

@TERM2

@TERM2(*versamenti;interesse;valore_futuro*) calcola il tempo necessario per accumulare un capitale di *valore_futuro*, effettuando *versamenti* periodici di uguale importo, dato un tasso d'*interesse* periodico, secondo la convenzione della rendita anticipata.

Argomenti

versamenti rappresenta il valore degli investimenti di uguale importo; può assumere qualunque valore ad eccezione di 0.

interesse rappresenta il tasso d'interesse periodico e può essere un numero decimale o un valore percentuale maggiore di -1.

valore_futuro rappresenta l'importo che si vuole raggiungere e può assumere qualsiasi valore.

Note

Per calcolare il tempo necessario per rimborsare un prestito, usare un argomento *valore_futuro* negativo con la funzione **@TERM2**. Ad esempio, per sapere quanto tempo si impiegherà per restituire un prestito di \$10,000 a un tasso d'interesse annuo del 10%, versando \$1,174 all'anno, si usi la seguente formula:.

@ASS(@TERM2(1174.6;0.1;-10000)) calcola che per restituire il prestito occorrono 16 anni.

A differenza di **@TERM**, **@TERM2** per il calcolo usa la fine della durata, anziché l'inizio.

Esempi

Si supponga di voler depositare l'importo di ¥200,000 alla fine di ogni anno su un conto di risparmio che rende un tasso d'interesse di 7,5% capitalizzato annualmente e di voler stabilire quanto tempo si impiegherà per raggiungere l'importo di ¥10,000,000:

@TERM2(200000;0.075;10000000) = 20,76 anni

Si può ottenere lo stesso risultato con **@NUMPERIODI(200000;0,075;10000000;1)**

Funzioni @ affini

@TERM calcola il tempo necessario per accumulare un capitale di *valore_futuro*, effettuando *versamenti* periodici di uguale importo, dato un tasso d'*interesse* periodico, presupponendo una rendita immediata. **@NUMPERIODI** calcola il tempo necessario per accumulare un *valore_futuro* dato un tasso di interesse periodico e versamenti di uguale importo, con il sistema della rendita immediata o anticipata.

@CTERM calcola il numero di periodi di capitalizzazione necessari affinché un singolo investimento possa raggiungere un *valore_futuro*.

@RICERT

@RICERT(*stringa_ricerca*; *stringa*; *colonna_inizio*) calcola la posizione dei caratteri thailandesi logici in *stringa*, in cui 1-2-3 trova la prima occorrenza della *stringa_ricerca*, a partire dal carattere thailandese che si trova nella *colonna_inizio*. Questa funzione si applica ai caratteri thailandesi, che possono essere costituiti da 1 a 3 byte.

Argomenti

stringa_ricerca e *stringa* possono essere stringhe di testo racchiuse fra virgolette (" "), una formula che genera una stringa di testo oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta. *colonna_inizio* è uno scarto.

Note

La funzione @RICERT può essere usata in combinazione con @MEZZOT o @SOSTITT per individuare la posizione ed estrarre o sostituire una stringa di testo.

Esempi

@TFIND("ไทย", "บริษัท ไทยซอฟต์แวร์ จำกัด", 0) = 5

Funzioni @ affini

@RICER svolge la stessa funzione di @RICERT ma si applica a stringhe di testo non thailandese.

@STRINGADATATL

@STRINGADATATL(*valore_data*) converte il *valore_data* nella corrispondente stringa di data nel formato thailandese esteso.

Argomenti

valore_data è un valore data.

Esempi

@TLDATESTRING(@DATE(91,8,16)) = 16 สิงหาคม 2534

Funzioni @ affini

@STRINGADATAT converte *valore_data* in una stringa con il formato data thailandese abbreviato.

@SINISTRAT

@SINISTRAT(*stringa*; *n*) fornisce i primi *n* caratteri thailandesi logici di *stringa*. Questa funzione si applica ai caratteri thailandesi logici, che possono essere forniti da 1 a 3 byte.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

n può essere un numero intero positivo o 0. Se *n* è 0, il risultato è una stringa vuota. Se *n* è maggiore della lunghezza della *stringa*, @SINISTRAT riporta l'intera *stringa*.

Esempi

@TLEFT("บริษัท วิทยุสื่อสาร จำกัด",4) = บริษัท

Funzioni @ affini

@SINISTRA svolge la stessa funzione di @SINISTRAT ma si applica a stringhe di testo non thailandese. @MEZZOT copia *n* caratteri thailandesi di *stringa*. @DESTRA fornisce gli ultimi *n* caratteri thailandesi di *stringa*.

@LUNGHT

@LUNGHT(*stringa*) conta un certo numero di caratteri thailandesi logici di *stringa*. Questa funzione si applica ai caratteri thailandesi logici, che possono essere composti da 1 a 3 byte.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

Esempi

@TLENGTH("บริษัท วิทยุขอฟท์ จำกัด") = 17

Funzioni @ affini

@LUNGH svolge la stessa funzione di @LUNGHT, ma viene applicata a stringhe di testo non thailandese.

@MEZZOT

@MEZZOT(*stringa*; *numero_inizio*; *n*) copia *n* caratteri thailandesi logici di *stringa*, a partire dalla posizione specificata di *numero_inizio*. Questa funzione si applica ai caratteri thailandesi logici, che possono essere composti da 1 a 3 byte.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

numero_inizio è uno scarto. Se *numero_inizio* supera la lunghezza di *stringa*, la funzione @MEZZOT produce una stringa vuota.

n è un numero intero positivo o 0. Se *n* è 0, la funzione @MEZZOT produce una stringa vuota. Se *n* supera la lunghezza di *stringa*, @MEZZOT riporta tutti i caratteri da *numero_inizio* fino alla fine di *stringa*.

Esempi

@TMID("บริษัท วิทยุคอมพ์ จำกัด",5,7) = วิทยุคอมพ์

Funzioni @ affini

@MID esegue la stessa funzione di @MEZZOT ma si applica a stringhe di testo non thailandese. @DESTRAT fornisce gli ultimi *n* caratteri thailandesi di *stringa*. @SINISTRAT fornisce i primi *n* caratteri thailandesi di *stringa*.

@STRINGANUMT

@STRINGANUMT(*numero*) converte *numero* nell'equivalente stringa numerica in caratteri thailandesi.

Argomenti

numero deve essere un numero intero o un numero a virgola mobile.

Esempi

@TNUMBERSTRING(1000000) = หนึ่งล้านบาทถ้วน

Funzioni @ affini

@STRINGANUM converte *numero* nell'equivalente numero in caratteri, memorizzato sotto forma di etichetta, in un formato non thailandese.

@SOSTITT

@SOSTITT(*stringa_originale*; *numero_inizio*; *n*; *stringa_sostit*) sostituisce *n* caratteri logici thailandesi di *stringa_originale* con *stringa_sostit*, a partire da *numero_inizio*. Questa funzione si applica ai caratteri thailandesi logici, che possono essere composti da 1 a 3 byte.

Argomenti

stringa_originale e *stringa_sostit* possono essere stringhe di testo racchiuse fra virgolette (" "), formule che generano stringhe di testo, oppure gli indirizzi o i nomi di celle contenenti etichette o formule che generano etichette.

numero_inizio è uno scarto di colonna in *stringa_originale*; può essere qualsiasi valore positivo o 0. Se *numero_inizio* è maggiore della lunghezza di *stringa_originale*, @SOSTITT aggiunge *stringa_sostit* in fondo a *stringa_originale*.

n può essere qualsiasi numero intero positivo o 0. Se *n* è uguale a 0, @SOSTITT inserisce *stringa_sostit* da *numero_inizio* senza cancellare alcuna colonna da *stringa_originale*.

Esempi

@TREPLACE("บริษัท จำกัด",5,0,"ไทยซอฟท์") = บริษัท ไทยซอฟท์ จำกัด

Funzioni @ affini

@SOSTIT esegue la stessa funzione di @SOSTITT, ma si applica a stringhe di testo non thailandese.

@DESTRAT

@DESTRAT(*stringa*; *n*) fornisce gli ultimi *n* caratteri thailandesi logici di *stringa*. Questa funzione si applica ai caratteri thailandesi logici, che possono essere forniti da 1 a 3 byte.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

può essere un numero intero positivo o 0. Se *n* è 0, il risultato è una stringa vuota. Se *n* è maggiore della lunghezza della *stringa*, @DESTRAT riporta l'intera *stringa*.

Esempi

@TRIGHT("บริษัท ไทยซอฟต์แวร์ จำกัด",4) = จำกัด

Funzioni @ affini

@DESTRA svolge la stessa funzione di @DESTRAT, ma si applica a stringhe di testo non thailandese. @MEZZOT genera un certo numero di caratteri thailandesi da *stringa*. @SINISTRAT fornisce i primi *n* caratteri thailandesi logici di *stringa*.

@XAMMORT

@XAMMORT(*stima;flusso_di_cassa;date*) calcola il tasso di rendimento interno per una serie di movimenti di cassa in entrata e in uscita.

Argomenti

stima può essere un valore decimale o percentuale che rappresenta il tasso di rendimento interno ipotizzato. Nella maggior parte dei casi, *stima* dovrebbe essere un valore percentuale compreso tra 0 (0%) e 1 (100%). 10% è spesso una stima affidabile. In presenza di movimenti di cassa molto consistenti, occorre rendere il più accurato possibile l'argomento *stima*.

flussi_di_cassa può essere l'indirizzo o il nome di una zona contenente i movimenti di cassa. 1-2-3 considera i numeri negativi come uscite e i numeri positivi come entrate di cassa. Solitamente, il primo movimento di cassa di una zona è un numero negativo (uscita) che rappresenta l'investimento iniziale. 1-2-3 assegna il valore 0 a tutte le celle vuote e a tutte le etichette situate nella zona, includendole nei calcoli.

date può essere l'indirizzo o il nome di una zona contenente le date in cui avvengono i corrispondenti flussi di cassa. Ogni *date* può essere un qualunque valore data e deve coincidere con i tempi del corrispondente flusso nella zona dei *flussi_di_cassa*. Le *date* devono essere in ordine crescente.

Le zone dei *flussi_di_cassa* e delle *date* devono avere le stesse dimensioni, ovvero contenere lo stesso numero di celle. Le celle delle due zone vengono infatti accoppiate in base all'ordine di successione. Le zone sono ordinate dall'alto verso il basso (colonne), poi da sinistra a destra (righe) e infine foglio per foglio. Se le zone dei *flussi_di_cassa* e delle *date* non hanno le stesse dimensioni, la funzione @XAMMORT produce ERR.

Note

I flussi di cassa possono essere effettuati a intervalli irregolari. Ogni flusso di cassa della zona dei *flussi_di_cassa* viene accoppiato con una data della zona delle *date*. Il primo flusso di cassa e la prima data si riferiscono all'inizio del programma di investimento e il primo flusso di cassa non viene attualizzato. I flussi di cassa successivi si basano sul tasso di attualizzazione annuale e sul momento in cui si verifica il flusso, indicato dalla data corrispondente.

@AMMORT attualizza il primo flusso di cassa, mentre @XAMMORT non lo fa e usa la prima data come inizio del programma. Per riprodurre la convenzione di attualizzazione utilizzata da @AMMORT, usare 0 come primo flusso di cassa nella funzione @XAMMORT.

La funzione @XAMMORT è adatta a quelle situazioni in cui i flussi di cassa si verificano a intervalli irregolari, mentre @AMMORT è utile quando gli intervalli dei flussi di cassa sono uguali tra loro.

Esempi

Il 13 Settembre 1996 si investono \$50.000 e successivamente si ricevono i seguenti versamenti: \$1.000 il 31 Gennaio 1997 e \$53.000 il 14 Giugno 1997. Tali dati sono contenuti nelle seguenti zone:

STIMA	FLUSSI DI CASSA	DATE
0,10	-50000	13/09/96
	1000	31/01/97
	53000	14/06/97

@XAMMORT(STIMA;FLUSSI DI CASSA;DATE) = 10,90%

Funzioni @ affini

@XVAN calcola il valore attuale netto di una serie di flussi di cassa in uscita e in entrata. @AMMORT calcola il tasso di rendimento interno (profitto) per una serie di movimenti di cassa generati da un investimento. @AMMORTMODIF calcola il tasso di rendimento interno modificato.

@XVAN

@XVAN(*tasso;flussi_di_cassa;date*) calcola il valore attuale netto di una serie di flussi di cassa in uscita e in entrata.

Argomenti

tasso può assumere un valore maggiore di -1 e rappresenta il tasso di attualizzazione.

flussi_di_cassa è l'indirizzo o il nome di una zona contenente i movimenti di cassa. 1-2-3 considera i numeri negativi come uscite e i numeri positivi come entrate di cassa. Solitamente, il primo movimento di cassa di una zona è un numero negativo (uscita) che rappresenta l'investimento iniziale. 1-2-3 assegna il valore 0 a tutte le celle vuote e a tutte le etichette situate nella zona, includendole nei calcoli.

date può essere l'indirizzo o il nome di una zona contenente le date in cui avvengono i corrispondenti flussi di cassa. Ogni data può essere un qualunque valore data e deve coincidere con i tempi del corrispondente flusso nella zona dei *flussi_di_cassa*. Le date devono essere in ordine crescente.

Le zone dei *flussi_di_cassa* e delle *date* devono avere le stesse dimensioni, ovvero contenere lo stesso numero di celle. Le celle delle due zone vengono infatti accoppiate in base all'ordine di successione. Le zone sono ordinate dall'alto verso il basso (colonne), poi da sinistra a destra (righe) e infine foglio per foglio. Se le zone dei *flussi_di_cassa* e delle *date* non hanno le stesse dimensioni, la funzione @XVAN produce ERR.

Note

I flussi di cassa possono essere effettuati a intervalli irregolari. Ogni flusso di cassa della zona dei *flussi_di_cassa* viene accoppiato con una data della zona delle *date*. Il primo flusso di cassa e la prima data si riferiscono all'inizio del programma di investimento e il primo flusso di cassa non viene attualizzato. I flussi di cassa successivi si basano sul tasso di attualizzazione annuale e sul momento in cui si verifica il flusso, indicato dalla data corrispondente.

Esempi

In questo esempio, la funzione @XVAN viene usata per attualizzare in dollari correnti (1 Gennaio 1996) una serie di ripartizioni investite a un tasso annuo del 11,5%. L'esempio usa i seguenti dati: la lista dei *flussi_di_cassa* è contenuta in una zona denominata RIPARTIZIONI e l'elenco delle *date*, corrispondenti ai tempi dei flussi di cassa, si trova nella zona DATA:

RIPARTIZIONI	DATE
\$0,00	01/01/96
\$250.000,00	01/04/96
\$250.000,00	01/05/96
\$300.000,00	01/06/96
\$500.000,00	01/07/96
\$600.000,00	01/08/96
\$900.000,00	01/09/96
\$300.000,00	01/10/96
\$250.000,00	01/11/96
\$750.000,00	01/01/97

@XVAN(0.115;RIPARTIZIONI;DATE) = \$3.821.809,20

Funzioni @ affini

@NPV calcola il valore attuale netto di un investimento, attualizzando il valore futuro a quello presente. @XAMMORT calcola il tasso di rendimento interno per una serie di movimenti di cassa in entrata e in uscita.

@RENDIMENTO2

@RENDIMENTO2(*liquidazione*; *scadenza*; *cedola*; *prezzo*; [*rimborso*]; [*base*]) calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi, secondo le convenzioni giapponesi.

Argomenti

liquidazione è il giorno dei compensi di un titolo e consiste in un valore data

scadenza è il giorno in cui viene rimborsato un titolo; deve essere un valore data. Se *scadenza* è inferiore o uguale a *liquidazione*, la funzione @RENDIMENTO2 genera ERR.

cedola è il tasso nominale annuale di un titolo e consiste in un valore positivo o uguale a zero.

prezzo è il prezzo del titolo in yen per ¥100 di valore nominale e può assumere qualsiasi valore positivo.

rimborso è un argomento facoltativo che specifica il valore di rimborso di un titolo con valore nominale pari a ¥100. Questo argomento deve essere costituito da un valore positivo o zero. In caso di omissione, @RENDIMENTO2 usa 100.

base è un argomento facoltativo che specifica il tipo di base per il conteggio dei giorni da usare:

<u>base</u>	<u>Base per il conteggio dei giorni</u>
0	30/360
1	effettivo/effettivo
2	effettivo/360
3	effettivo/365
4	europeo 30/360

Se si specifica la *base*, 1-2-3 richiama la funzione @RENDIMENTO per calcolare il valore.

Esempi

Un'obbligazione ha come data di liquidazione il 1° luglio 1993 e come data di scadenza il 1° dicembre 1998. Il tasso nominale d'interesse semestrale è pari al 5,50%. Il costo è di ¥99,5 e il valore di rimborso di ¥100. Si vuole stabilire il rendimento dell'obbligazione:

@RENDIMENTO2(@DATA(93;7;1);@DATA(98;12;1);0,055;99,5;100) = 0,056072

Funzioni @ affini

@RENDIMENTO calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi.

@MATURATO2 calcola gli interessi maturati sui titoli con pagamento periodico degli interessi, in base alle convenzioni giapponesi. @PREZZO2 calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a ¥100, in base alle convenzioni giapponesi.

@MATURATO calcola gli interessi maturati sui titoli che prevedono pagamenti periodici degli interessi. @PREZZO calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a \$100. @DURATA calcola la durata annua di titoli con pagamento periodico degli interessi. @DURATAM calcola la durata annua modificata di titoli con pagamento periodico degli interessi.

@TESTCHI2

@TESTCHI2(*zona1*;*[zona2]*) esegue un test chi quadrato di indipendenza sui dati della *zona1* o un test chi quadrato di bontà di adattamento sui dati contenuti in *zona1* e *zona2*.

Argomenti

zona1 e *zona2* devono avere le stesse dimensioni. Se hanno dimensioni diverse o se una delle due zone contiene etichette, formule di testo o celle vuote, **@TESTCHI2** produce ERR.

Nota

@TESTCHI2 calcola la probabilità associata a un test chi quadrato con un'approssimazione di $\pm 3 \cdot 10^{-7}$.

Esempi

@TESTCHI2: test per l'indipendenza.

@TESTCHI2: test per la bontà di adattamento.

Funzioni @ affini

@DISTRIBCHI2 calcola la distribuzione chi quadrato. **@TESTF** esegue un test *F*, **@TTEST** esegue un test *t* di Student, mentre **@TESTZ** esegue un test *z*.

@CRITBINOMIALE

@CRITBINOMIALE(*prove;probabilità;alfa*) riporta il numero intero più grande per cui la distribuzione binomiale cumulativa è minore o uguale ad *alfa*.

Argomenti

prove rappresenta il numero di prove di Bernoulli; può consistere in un qualsiasi numero intero positivo o 0.

probabilità rappresenta la probabilità di successo di una sola prova di Bernoulli; deve essere un valore compreso tra 0 e 1.

alfa rappresenta il numero designato di successi; deve essere un valore compreso tra 0 e 1.

Nota

@CRITBINOMIALE calcola la distribuzione binomiale cumulativa con un'approssimazione di $\pm 3 \cdot 10^{-7}$.

Esempi

Si supponga di dirigere un piccolo stabilimento per la produzione di filtri dell'olio. I filtri vengono lavorati in lotti di 100; la probabilità che un filtro sia privo di difetti è pari al 85%. Si vuole avere la sicurezza del 99% che almeno un certo numero di filtri sia esente da difetti:

@CRITBINOMIALE(100;0,85;0,01) = 76 filtri esenti da difetti

Funzioni @ affini

@BINOMIALE calcola la funzione generatrice della probabilità binomiale oppure la distribuzione binomiale cumulativa. **@COMBIN** calcola il coefficiente binomiale. **@PERMUT** calcola il numero di permutazioni per una lista di valori.

@MEDIADB

`@MEDIADB(zona_immissione;campo;condizioni)` calcola la media dei valori contenuti in un *campo* di una tabella di database che soddisfano le *condizioni* specificate.

Argomenti

Consultare [Argomenti delle funzioni @ di database](#).

Esempi

`@MEDIADB`

Funzioni @ affini

`@MEDIA` e `@MEDIAPURA` calcolano la media aritmetica dei valori compresi in una lista.

@CONTDB, @CONTAPURODB

@CONTDB(*zona_immissione;campo;condizioni*) conta le celle non vuote di un *campo* di una tabella di database che soddisfano le *condizioni* specificate.

@CONTAPURODB(*zona_immissione;campo;[condizioni]*) conta le celle contenenti valori di un *campo* di una tabella di database che soddisfano le *condizioni* specificate.

Argomenti

Consultare [Argomenti delle funzioni @ di database](#).

Esempi

[@CONTDB](#)

Funzioni @ affini

[@CONTA](#) e [@CONTAPURO](#) contano le celle di una lista di zone.

@SELDB

@SELDB(*zona_immissione;campo;condizioni*) ricava un valore o un'etichetta da un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

Argomenti

Consultare [Argomenti delle funzioni @ di database](#).

Nota

Se le *condizioni* specificate sono soddisfatte da più di un'immissione, @SELDB produce ERR.

@SELDB può essere utile quando si tratta di estrarre un valore da un solo record che soddisfa condizioni specifiche -- per esempio, la matricola di un dipendente. Con la funzione @SELDB si possono ricavare automaticamente questo tipo di informazioni per usarle in una macro come argomenti di una funzione @ o come variabili di una formula.

Esempi

@SELDB

Funzioni @ affini

@SCANSOR e @SCANSVER riportano il contenuto della cella specificata di una tabella di consultazione orizzontale o verticale. @SCELTA individua un'immissione in una lista. @IND riporta il contenuto di una cella di una tabella situata nel punto di intersezione del foglio di lavoro, della riga e della colonna specificati. @INDX riporta il contenuto di una cella situata in una tabella nel punto di intersezione dei titoli di colonna, riga e foglio di lavoro specificati. @@ riporta indirettamente il contenuto della cella specificata.

@MAXDB

`@MAXDB(zona_immissione;campo;condizioni)` individua il valore più alto in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

Argomenti

Consultare [Argomenti delle funzioni @ di database](#).

Nota

La funzione `@MAXDB` può essere usata per individuare la data o l'orario più recente in una lista di date e orari.

Esempi

[@MAXDB](#)

Funzioni @ affini

[@MAX](#) e [@MAXPURO](#) individuano il valore più alto in una lista.

@MINDB

@MINDB(*zona_immissione;campo;condizioni*) individua il valore più basso in un *campo* di una tabella di database che soddisfa le *condizioni* specificate.

Argomenti

Consultare [Argomenti delle funzioni @ di database](#).

Nota

La funzione **@MINDB** può servire a individuare la data o l'ora più remota in una lista di date o orari.

Esempi

[@MINDB](#)

Funzioni @ affini

[@MIN](#) e [@MINPURO](#) individuano il valore più basso in una lista.

@SCARTODB, @SCARTODBC

@SCARTODB(*zona_immissione;campo;condizioni*) calcola la deviazione standard di una popolazione di valori contenuti in un *campo* di una tabella di database che soddisfano le *condizioni* specificate.

@SCARTODBC(*zona_immissione;campo;condizioni*) calcola la deviazione standard di un campione di valori contenuti in un *campo* di una tabella di database che soddisfano le *condizioni* specificate.

Argomenti

Consultare Argomenti delle funzioni @ di database.

Note

@SCARTODB usa il metodo n (o della popolazione) per calcolare la deviazione standard di una popolazione di dati. Il metodo n parte dal presupposto che i valori selezionati costituiscano l'intera popolazione di dati; se invece i valori rappresentano solo un campione della popolazione, la deviazione standard risulta falsata dagli errori di campionamento.

La deviazione standard è uguale alla radice quadrata della varianza di tutti i singoli valori rispetto alla media aritmetica dei valori stessi.

Esempi

@SCARTODB e @SCARTODBC

Funzioni @ affini

@SCARTO e @SCARTOPURO calcolano la deviazione standard dell'intera popolazione di valori contenuti in una *zona*. @SCARTOC e @SCARTOCPURO calcolano la deviazione standard di valori campione. @VARDB calcola la varianza di una popolazione di valori che soddisfano le condizioni specificate.

@SOMMADB

@SOMMADB(*zona_immissione;campo;condizioni*) calcola la somma dei valori situati in un *campo* di una tabella di database che soddisfano le *condizioni* specificate.

Argomenti

Consultare [Argomenti delle funzioni @ di database](#).

Esempi

@SOMMADB

Funzioni @ affini

@SOMMA somma i valori di una lista. @SOMMAPOSITIVI esegue la somma dei soli valori positivi contenuti in una lista. @SOMMANEGATIVI esegue la somma dei soli valori negativi contenuti in una lista.

@VARDB, @VARDBC

@VARDB(*zona_immissione;campo;condizioni*) calcola la varianza di una popolazione di valori contenuti in un *campo* di una tabella di database che soddisfano le *condizioni* specificate.

@VARDBC(*zona_immissione;campo;condizioni*) calcola la varianza di un campione di valori contenuti in un *campo* di una tabella di database che soddisfano le *condizioni* specificate.

Argomenti

Consultare [Argomenti delle funzioni @ di database](#).

Note

La varianza misura il grado di scostamento dei singoli valori dalla media di tutti i valori della lista. Quanto minore è la varianza, tanto più piccolo sarà lo scostamento dei singoli valori dalla media aritmetica, e tanto più attendibile la media stessa. Una varianza pari a 0 indica che tutti i valori della lista sono uguali. Il calcolo della varianza è utile in molti test statistici di analisi della varianza.

@VARDB usa il metodo n (o della popolazione) per calcolare la varianza. Il metodo n parte dal presupposto che i valori selezionati costituiscano l'intera popolazione dei dati. Se i valori costituiscono soltanto un campione della popolazione di dati, la varianza risulta falsata dagli errori di campionamento.

La varianza è il quadrato della deviazione standard.

Esempi

[@VARDB](#) e [@VARDBC](#)

Funzioni @ affini

[@VAR](#) e [@VARPURA](#) calcolano la varianza di una popolazione di valori compresi in una lista.

[@SCARTODB](#) calcola la deviazione standard di una popolazione di valori che soddisfano le condizioni specificate.

@TESTF

@TESTF(*zona1*; *zona2*) esegue un test *f*, riportando la probabilità associata.

Argomenti

zona1 e *zona2* sono zone contenenti i dati da esaminare; non occorre che abbiano uguali dimensioni.

Note

@TESTF esegue il calcolo della probabilità associata a un test *F* con un'approssimazione di $\pm 3 \cdot 10^{-7}$.

La funzione @TESTF può servire a determinare se due campioni presentano la stessa varianza.

Esempi

@TESTF

Funzioni @ affini

@DISTRIBF calcola la distribuzione *F*. @TESTCHI2 esegue un test chi quadrato, @TESTT esegue un test *t* di Student, mentre @TESTZ esegue un test *z*.

@TOTGENER

@TOTGENER(*lista*) somma tutte le celle comprese in una *lista* contenenti la funzione @TOTPARZ.

Argomenti

lista può essere una qualsiasi combinazione di zone, divise tra loro dagli appositi separatori di argomenti.

Vedere anche Argomenti delle funzioni @ statistiche.

Esempi

@TOTGENER

Funzioni @ affini

@SOMMA somma i valori compresi in una lista. @SOMMADB somma i valori situati in una tabella di database che soddisfano le condizioni specificate. @SOMMAPOSITIVI esegue la somma dei soli valori positivi contenuti in una lista. @SOMMANEGATIVI esegue la somma dei soli valori negativi contenuti in una lista.

@DFILE

@DFILE(*nome_file*;*tipo*) va a cercare *nome_file* nella memoria e sul disco: se lo trova, riporta 1 (vero); in caso contrario, riporta 0 (falso).

Argomenti

nome_file è il nome completo, comprensivo di estensione, del file da cercare; va immesso come qualunque altra stringa di testo. A meno che 1-2-3 non debba andarlo a cercare solo nella directory attiva, occorre specificare anche il percorso del file.

tipo specifica se *nome_file* deve essere cercato nella memoria o sul disco. Se *tipo* è 0, 1-2-3 esegue la ricerca nella memoria; se è 1, sul disco. In caso di omissione dell'argomento, 1-2-3 usa il valore 0.

Esempi

@DFILE("C:\123W\BUDGET\COSTI_93.123";1) = 1, se il file COSTI_93.123 si trova in C:\123W\BUDGET.

@CORRISP

@CORRISP(*contenuto_cella*; *zona*; [*tipo*]) segnala, sotto forma di scarto, la posizione, all'interno della *zona*, della cella il cui contenuto corrisponde a *contenuto_cella*.

Argomenti

contenuto_cella può essere un valore o una stringa di testo, nel qual caso può contenere anche caratteri jolly.

zona è il nome o l'indirizzo di una zona.

tipo è un argomento facoltativo che specifica come 1-2-3 deve eseguire il confronto tra *contenuto_cella* e il contenuto delle celle comprese nella *zona*.

tip o	@CORRISP riporta la posizione relativa
0	della prima cella il cui contenuto corrisponde a <i>contenuto_cella</i>
1	della cella contenente il valore più alto tra quelli minori o uguali a <i>contenuto_cella</i> ; disporre i dati della <i>zona</i> in ordine crescente (impostazione standard nel caso di omissione dell'argomento)
2	della cella contenente il valore più basso tra quelli maggiori o uguali a <i>contenuto_cella</i> ; disporre i dati della <i>zona</i> in ordine decrescente

Note

1-2-3 esegue la ricerca nella *zona*, procedendo dall'alto verso il basso e da sinistra verso destra. Se si specifica una zona composta da più fogli di lavoro, 1-2-3 va a cercare nel primo foglio della zona, prosegue poi con il secondo e così via fino a incontrare una cella dal contenuto corrispondente oppure la fine della zona.

Se 1-2-3 non riesce a trovare una cella con il *contenuto_cella*, genera il valore ERR.

Se *tipo* è 1 e la prima cella della *zona* contiene un valore maggiore di *contenuto_cella*, @CORRISP produce ERR.

Se *tipo* è 2 e la prima cella della *zona* contiene un valore minore di *contenuto_cella*, @CORRISP produce ERR.

Esempi

@CORRISP

Funzioni @ affini

@SCANSOR e @SCANSVER cercano il contenuto delle celle nelle tabelle di consultazione orizzontali e verticali. @IND cerca il contenuto di una cella quando sono specificati gli scarti di riga e di colonna. @SCELTA sostituisce una tabella di consultazione che occupa una sola riga.

@SCANSMAX produce un riferimento assoluto alla cella che contiene il valore più alto di una serie di zone. @SCANSMIN genera un riferimento assoluto alla cella che contiene il valore inferiore di una serie di zone

@NORMALE

@NORMALE(x;[media];[deviazione_standard];[tipo]) calcola la funzione di distribuzione normale per x.

Argomenti

x è il valore limite della distribuzione normale cumulativa; può assumere qualsiasi valore. Se x è negativo, 1-2-3 lo converte nel valore assoluto (positivo) corrispondente.

media è un argomento facoltativo che specifica la media della distribuzione; deve essere un numero positivo oppure zero. In caso di omissione dell'argomento, 1-2-3 usa una *media* uguale a zero.

deviazione_standard è un argomento facoltativo che specifica la deviazione standard della distribuzione. La *deviazione_standard* deve essere un valore positivo oppure zero. In caso di omissione dell'argomento, 1-2-3 usa una *deviazione_standard* uguale a 1.

tipo è un argomento facoltativo che specifica la funzione che si desidera calcolare con @NORMALE.

<i>tipo</i>	@NORMALE calcola
0	la funzione di distribuzione cumulativa; impostazione predefinita in caso di omissione dell'argomento
1	l'inverso della distribuzione cumulativa
2	la funzione di densità di probabilità

Non è consentito l'uso di un argomento facoltativo senza specificare quelli che lo precedono.

Nota

@NORMALE calcola la funzione di distribuzione cumulativa con un'approssimazione di $\pm 7,5 \cdot 10^{-8}$ e l'inverso della distribuzione cumulativa con un'approssimazione di $\pm 4,5 \cdot 10^{-4}$.

Esempi

@NORMALE(1,96) = 0,9750

@NORMALE(0,975;0;1;1) = 1,96

@NORMALE(1,96;0;1;2) = 0,058441

Funzioni @ affini

@DISTRIBCHI2 calcola la distribuzione chi quadrato. @DISTRIBF calcola la distribuzione F.

@POISSON calcola la distribuzione di Poisson. @DISTRIBT calcola la distribuzione t di Student.

@NOMEZONA

@NOMEZONA(*cella*) riporta il nome della zona in cui è collocata la *cella*.

Argomenti

cella è un indirizzo di cella.

Note

Se si specifica una cella appartenente a diverse zone denominate in sovrapposizione, 1-2-3 riporta il nome della prima zona incontrata.

Se la *cella* non appartiene a una zona denominata, @NOMEZONA produce ERR.

La funzione @NOMEZONA può essere usata solo per le cartelle di lavoro in memoria.

Esempi

@NOMEZONA(A:A2) riporta VENDITE se la cella A:A2 appartiene alla zona omonima.

@ULTIMOSCENARIO

@ULTIMOSCENARIO(*[nome_file]*) riporta il nome del gruppo di versioni di una cartella di lavoro visualizzato per ultimo durante la sessione di 1-2-3 in corso.

Argomenti

nome_file è il nome, comprensivo di estensione, del file della cartella di lavoro da esaminare, immesso sotto forma di testo. A meno che 1-2-3 non debba andare a cercare il file della cartella di lavoro solo nella directory attiva, occorre specificare anche il percorso.

Note

Se durante la sessione di 1-2-3 in corso non è stato visualizzato alcun gruppo di versioni in *nome_file*, @ULTIMOSCENARIO produce ERR.

Esempi

@ULTIMOSCENARIO("d:\123v5w\fogli\luglio.wk4") riporta il nome dell'ultimo gruppo di versioni visualizzato nel file della cartella di lavoro LUGLIO.WK4 residente nella directory D:\123V5W\FOGLI.

@MEDIAES

@MEDIAES(*lista*) calcola l'errore standard della media campionaria per i valori compresi nella *lista*.

Argomenti

lista può contenere i seguenti elementi in qualunque combinazione: numeri, formule numeriche e indirizzi o nomi di zone contenenti numeri o formule numeriche. Tali elementi devono essere divisi tra loro dagli appositi separatori di argomenti.

Esempi

@MEDIAES(2;6;8;5;3;9;1;2) = 1,052209

Funzioni @ affini

@MEDIAGEOM calcola la media geometrica dei valori compresi in una lista. @MEDIAARMON calcola la media armonica dei valori compresi in una lista. @SCARTO e @SCARTOPURO calcolano la deviazione standard dei valori compresi in una lista.

@INFOSCENARIO, @INFOVERSIONE

@INFOSCENARIO(*opzione;nome;[autore]*) fornisce informazioni su un gruppo di versioni.

@INFOVERSIONE(*opzione;zona_versione;nome;[autore]*) fornisce informazioni su una versione.

Argomenti

opzione è una stringa di testo che specifica le informazioni richieste a 1-2-3.

<i>opzione</i>	1-2-3 indica
autore	il nome della persona che ha creato la versione o il gruppo di versioni
modifiche_di	il nome dell'ultima persona che ha modificato la versione o il gruppo di versioni
data_creazione	la data e l'ora in cui è stata creata la versione o il gruppo di versioni (sotto forma di valore <u>data</u> e <u>orario</u>)
data_modifica	la data e l'ora in cui ha avuto luogo l'ultima modifica della versione o del gruppo di versioni (sotto forma di valore <u>data</u> e <u>orario</u>)
commento	il commento immesso per la versione o il gruppo di versioni; se supera i 512 caratteri (byte), 1-2-3 lo tronca
nascosto	0 (falso) se la versione o il gruppo di versioni non sono nascosti; 1 (vero) se sono nascosti
protetto	0 (falso) se la versione o il gruppo di versioni non sono protetti; 1 (vero) se sono protetti

nome è una stringa che specifica il nome della versione o del gruppo di versioni desiderato. Se esistono più versioni o gruppi di versioni che recano lo stesso *nome*, 1-2-3 usa l'ultimo creato.

autore è una stringa che specifica il nome dell'utente che ha creato la versione o il gruppo di versioni. 1-2-3 ricorre all'argomento *autore* per determinare la versione o il gruppo di versioni da usare.

zona_versione è il nome della zona che contiene la versione; deve riferirsi a una zona denominata.

Esempi

@INFOSCENARIO("commento";"Ottimale";"Marco_Dilani") riporta il commento immesso per l'ultimo gruppo di versioni (denominato Ottimale) creato da Marco Dilani

@INFOSCENARIO("autore";"Vendite") riporta il nome dell'utente che ha creato l'ultimo gruppo di versioni denominato Vendite

@INFOVERSIONE("data_creazione";ZONAVENDITE;"Ottimale";"Marco_Dilani") riporta la data e l'ora in cui Marco Dilani ha creato l'ultima versione di Ottimale per ZONAVENDITE

@INFOVERSIONE("data_modifica";ZONAVENDITE;"Accessori") riporta la data e l'ora in cui è stata effettuata l'ultima modifica della versione Accessori di ZONAVENDITE

@TESTT

@TESTT(*zona1*; *zona2*; [*tipo*]; [*code*]) esegue un test *t* di Student sui dati contenuti in *zona1* e *zona2*, e riporta la probabilità associata.

Argomenti

zona1 e *zona2* sono zone contenenti valori.

tipo è un argomento facoltativo che specifica il tipo di test *t* che si desidera svolgere.

tipo	1-2-3 esegue
0	un test <i>t</i> per i campioni estratti da popolazioni aventi varianza uguale (omoscedastici); non occorre che <i>zona1</i> e <i>zona</i> contengano lo stesso numero di celle; impostazione predefinita in caso di omissione dell'argomento
1	un test <i>t</i> per i campioni estratti da popolazioni aventi varianze diverse (eteroscedastici); non occorre che <i>zona1</i> e <i>zona</i> contengano lo stesso numero di celle
2	un test <i>t</i> accoppiato; <i>zona1</i> e <i>zona2</i> devono contenere lo stesso numero di celle

code è un argomento facoltativo che specifica la direzione del test *t*.

code	1-2-3 esegue
1	un test <i>t</i> a una coda
2	un test <i>t</i> a due code; impostazione predefinita in caso di omissione dell'argomento

Non è consentito l'uso di un argomento facoltativo senza specificare quelli che lo precedono.

Nota

@TESTT calcola la probabilità associata a un test *t* con un'approssimazione di $\pm 3 \cdot 10^{-7}$.

Esempi

@TESTT

Funzioni @ affini

@DISTRIBT calcola la distribuzione *t* di Student. @DISTRIBCHI2 esegue un test chi quadrato, @TESTF esegue un test *F*, mentre @TESTZ esegue un test *z*.

@VERSIONEATTIVA

@VERSIONEATTIVA(*zona*) riporta il nome della versione in uso nella *zona*.

Argomenti

zona è il nome o l'indirizzo della zona per cui si vuole trovare il nome della versione.

Nota

Se non è attiva alcuna versione, @VERSIONEATTIVA genera ERR.

Esempi

@VERSIONEATTIVA(UTILI) riporta il nome della versione in uso nella zona UTILI.

@DATIVERSIONE

@DATIVERSIONE(*opzione*; *cella*; *zona_versione*; *nome*; [*autore*]) riporta il contenuto di una cella di una versione.

Argomenti

opzione è una stringa di testo che specifica i criteri seguiti da 1-2-3 a seconda del contenuto di *cella*.

opzione 1-2-3 riporta

formula la formula contenuta nella cella, sotto forma di etichetta, oppure

ERR se la cella non contiene una formula

valore il risultato della formula se la cella contiene una formula

il valore o l'etichetta se la cella contiene l'uno o l'altra

una stringa

vuota H_GLEEMPTY@SS1N60IT.HLP se la cella è vuota

cella è il nome o l'indirizzo della cella, situata in *zona_versione*, di cui 1-2-3 deve riportare il contenuto.

zona_versione è il nome della zona che contiene la versione desiderata; deve riferirsi a una zona denominata.

nome è una stringa di testo che specifica il nome della versione desiderata. Se esistono più versioni che recano lo stesso *nome*, 1-2-3 usa quella creata per ultima.

autore è una stringa che specifica il nome dell'utente che ha creato la versione. 1-2-3 ricorre all'argomento *autore* per determinare la versione da usare o cancellare. Se uno stesso *autore* ha creato più versioni con lo stesso nome, 1-2-3 usa quella creata per ultima.

Esempi

@DATIVERSIONE("formula";A:B12;VENDITE;"Ottimale") riporta la formula contenuta nella cella A:B12 dell'ultima versione di nome Ottimale nella zona VENDITE

@DATIVERSIONE("valore";A:B12;VENDITE;"Ottimale";"Marco Sidari") riporta il valore o l'etichetta contenuto nella cella A:B12 dell'ultima versione di nome Ottimale creata da Marco Sidari nella zona VENDITE

@MEDIAPOND

@MEDIAPOND(*zona_dati*; *fattori_peso*; [*tipo*]) calcola la media ponderata dei valori contenuti in *zona_dati*.

Argomenti

zona_dati e *fattori_peso* possono essere i nomi o gli indirizzi di zone contenenti valori che abbiano uguale forma e dimensioni.

Se *zona_dati* e *fattori_peso* non hanno forma e dimensioni uguali, @MEDIAPOND produce ERR. *tipo* è un valore che determina come calcolare la media ponderata.

<i>tipo</i>	1-2-3 divide per
0	la somma dei valori della zona <i>fattori_peso</i> ; impostazione predefinita in caso di omissione dell'argomento
1	il numero di valori contenuti in <i>zona_dati</i>

Esempi

Questo esempio, tratto dalla tabella di un database di proprietà immobiliari, usa la funzione @MEDIAPOND per calcolare la media ponderata delle provvigioni dovute a un agente per gli immobili venduti in febbraio.

La zona VENDUTO contiene i prezzi pagati per i tre immobili, mentre PROVV contiene le percentuali sul prezzo di vendita dovute all'agente come provvigione:

VENDUTO PROVV

L. 250.000.000 0,04 L. 340.580.000 0,05 L. 773.250.000 0,04

@MEDIAPOND(VENDUTO;PROVV) = L. 447.846, ossia la media ponderata delle provvigioni dovute all'agente per la vendita dei tre immobili

Funzioni @ affini

@SOMMAPROD somma i prodotti di celle corrispondenti in varie zone.

@GLAVOR

@GLAVOR(*data_inizio*;giorni;[*festività*];[*fine_settimana*]) calcola il valore data corrispondente alla data che cade il numero di *giorni* lavorativi specificato prima o dopo *data_inizio*, escludendo dal calcolo i fine settimana ed eventualmente le festività.

Argomenti

data_inizio è un valore data.

giorni è un numero intero. I numeri interi positivi indicano un numero di giorni posteriore a *data_inizio*, mentre quelli negativi specificano un certo numero di giorni anteriore a *data_inizio*.

festività è un argomento facoltativo che specifica i giorni da escludere dal calcolo di **@GLAVOR**. *festività* può essere il nome o l'indirizzo di una zona contenente valori data.

In caso di omissione dell'argomento *festività*, 1-2-3 non esclude alcun valore dal calcolo di **@GLAVOR**.

fine_settimana è un argomento facoltativo che specifica da quali giorni della settimana è costituito il fine settimana. *fine_settimana* è inoltre una stringa di testo che usa i numeri interi compresi tra 0 (lunedì) e 6 (domenica) a rappresentazione dei giorni della settimana facenti parte del fine settimana.

Per esempio, la stringa "45" indica che venerdì e sabato fanno parte del fine settimana. In caso di omissione dell'argomento *fine_settimana*, 1-2-3 usa per impostazione predefinita la stringa "56", in base alla quale il fine settimana è costituito da sabato e domenica. Per indicare l'assenza del fine settimana, usare il numero 7.

Note

Se si desidera specificare l'argomento *fine_settimana*, mentre si vuole omettere *festività*, basterà specificare una cella vuota per quest'ultimo.

Esempi

Se per ipotesi si volesse determinare in quale data cadrà il trentesimo giorno lavorativo successivo al 1 novembre 1994, si dovrebbe indicare che i giorni 1 novembre e 8 dicembre sono festività. Per fare questo, si immetterebbero i valori data corrispondenti rispettivamente a tali date nella zona denominata FESTIVI. Poiché i fine settimana sono costituiti da sabati e domeniche, l'argomento *fine_settimana* può essere omesso.

@GLAVOR(@DATA(94;11;1);30;FESTIVI) = 34682, cioè mercoledì 14 dicembre 1994

Funzioni @ affini

@360GIORNI e **@D360** calcolano il numero di giorni che intercorrono tra due valori data.

@GIORNILAVNETTI calcola quanti giorni intercorrono tra due date, escludendo i fine settimana e le festività. **@ALTROMESE** calcola la data del giorno che cade il numero specificato di mesi prima o dopo una determinata data.

Esempio: @TESTCHI2 (test per l'indipendenza)

@TESTCHI2(B3..C5) = 0,080809

A	---	A	-----	B	-----	C	--
1			Ruvido			Liscio	
2							
3	Alto		42			51	
4	Medio		25			37	
5	Basso		85			68	

Esempio: @TESTCHI2 (test per la bontà di adattamento)

Con la seguente formula si vuole determinare la bontà di adattamento dei risultati osservati rispetto a quelli attesi:

$$\text{@TESTCHI2}(A2..A9;B2..B9) = 0,996882$$

A	-----	A	-----	B	-----
1	Osservati		Attesi		
2	25		23		
3	34		35		
4	87		91		
5	91		88		
6	34		34		
7	23		22		
8	56		60		
9	70		68		

Esempio: @MEDIADB

Una tabella di database denominata VENDITE contiene l'elenco di vendite immobiliari ad Arezzo, Bologna e Cagliari effettuate nei mesi di aprile e maggio. I prezzi di vendita degli immobili sono riportati nel campo VENDUTO. Si vuole determinare il prezzo medio di una casa venduta a Cagliari:

@MEDIADB(VENDITE;"VENDUTO";CITTA'="Cagliari") = L. 365.666.667

A	-----	A	-----	B	-----	C	----
1	INDIRIZZO			CITTA'		VENDUTO	
2	Corso Roma, 6			Cagliari		720.000.000	
3	Via del Palio, 12			Arezzo		318.000.000	
4	Via Fermi, 80			Bologna		332.000.000	
5	Via Segni, 80			Cagliari		278.000.000	
6	Via S. Isidoro, 7			Cagliari		160.000.000	
7	Via dei Caduti, 1160			Cagliari		227.000.000	
8	Piazza Italia, 13			Arezzo		397.000.000	
9	Largo Alpini, 12			Arezzo		303.000.000	
10	Piazza Tola, 3			Cagliari		669.000.000	
11	Via Compagnoni, 234			Cagliari		140.000.000	

Esempio: @CONTDB

Nella tabella di database denominata VEND_APR vengono elencate le vendite di appartamenti effettuate nel mese di aprile. Nel campo denominato RISCALD viene riportato il tipo di riscaldamento dei vari appartamenti. Si vuole individuare il numero di appartamenti riscaldati a metano:

@CONTDB(VEND_APR;"RISCALD";RISCALD="Metano") = 4

A	----	A	-----	B	-----	C	-----	D	-----
1	INDIRIZZO	LOCALI		RISCALD	PREZZO				
2	Bezzecca 4	3	Metano	L. 198.000.000					
3	Como 2	2	Gasolio	L. 123.000.000					
4	Campana 51	5	Metano	L. 315.000.000					
5	Melloni 36	2	Metano	L. 134.000.000					
6	Slataper 41	3	Gasolio	L. 240.000.000					
7	Panizza 8a	1	Metano	L. 160.000.000					

Esempio: @SELDB

Nella seguente tabella di database denominata VENDITE sono elencati gli appartamenti venduti in tre città nei mesi di aprile e maggio. Le provvigioni sulle vendite sono riportate nel campo denominato PROVV. La seguente formula @SELDB riporta la provvigione dell'agente sulla vendita dell'appartamento di via Filzi 33:

@SELDB(VENDITE;"PROVV";INDIRIZZO="Filzi 33") = L. 12.120.000

A ---	A -----	B -----	C -----
1	INDIRIZZO	AGENTE	PROVV
2	Sarpi 4	Morini	28.800.000
3	Umbria 15	Murri	12.720.000
4	Nordio 54	Morini	19.920.000
5	Mameli 7	Sidoli	11.120.000
6	Delfico 19	Nigro	9.600.000
7	Farini 74	Nigro	13.620.000
8	Bronzino 1	Murri	15.880.000
9	Filzi 33	Camera	12.120.000

Esempio: @MAXDB

Nella seguente tabella di database denominata VENDITE sono elencati gli appartamenti venduti nelle città di Firenze, Siena e Lucca nei mesi di aprile e maggio. Nel campo VENDUTO sono riportati i prezzi di vendita. Per calcolare il prezzo più alto pagato per l'appartamento venduto a Firenze viene utilizzata la seguente formula @MAXDB:

@MAXDB(VENDITE;"VENDUTO";CITTA'"Firenze") = L. 720.000.000

A	--	A	-----	B	-----	C	-----
1		INDIRIZZO		CITTA'		VENDUTO	
2		Sarpi 4		Firenze		720.000.000	
3		Umbria 15		Siena		318.000.000	
4		Nordio 54		Lucca		332.000.000	
5		Mameli 7		Firenze		278.000.000	
6		Delfico 19		Firenze		160.000.000	
7		Farini 74		Firenze		227.000.000	
8		Bronzino 1		Siena		397.000.000	
9		Filzi 33		Siena		303.000.000	
10		Roma 176		Firenze		669.000.000	
11		Crispi 68		Firenze		140.000.000	

Esempio: @MINDB

Nella seguente tabella di database denominata VENDITE sono elencati gli appartamenti venduti nelle città di Firenze, Siena e Lucca nei mesi di aprile e maggio. Nel campo VENDUTO sono riportati i prezzi di vendita. Per calcolare il prezzo più basso pagato per un appartamento venduto a Firenze viene utilizzata la seguente formula @MINDB:

@MINDB(VENDITE;"VENDUTO";CITTA'"Firenze") = L. 140.000.000

A	--	A	-----	B	-----	C	-----
1		INDIRIZZO		CITTA'		VENDUTO	
2		Sarpi 4		Firenze		720.000.000	
3		Umbria 15		Siena		318.000.000	
4		Nordio 54		Lucca		332.000.000	
5		Mameli 7		Firenze		278.000.000	
6		Delfico 19		Firenze		160.000.000	
7		Farini 74		Firenze		227.000.000	
8		Bronzino 1		Siena		397.000.000	
9		Filzi 33		Siena		303.000.000	
10		Roma 176		Firenze		669.000.000	
11		Crispi 68		Firenze		140.000.000	

Esempio: @SCARTODB e @SCARTODBC

La seguente tabella contiene i dati relativi all'altezza e al peso di dieci soggetti selezionati a caso. Si vuole determinare la deviazione standard dell'altezza degli individui che pesano più di 75 kg:

@SCARTODB(A2..A11;"ALTEZZA";PESO>75) = 5,793483

Si supponga ora che i dieci soggetti esaminati siano stati scelti a caso da un più folto gruppo di individui:

@SCARTODBC(A2..A11;"ALTEZZA";PESO>75) = 6,106868

A	-----	A	-----	B	--
1		ALTEZZA (cm)		PESO (kg)	
2		190,50		72,73	
3		187,96		86,36	
4		175,26		68,18	
5		175,26		76,37	
6		180,34		77,27	
7		180,34		72,73	
8		187,96		75,00	
9		172,72		68,18	
10		177,80		70,46	
11		179,07		86,36	

Esempio: @SOMMADB

Nella seguente tabella di database denominata VENDITE sono elencati gli appartamenti venduti in tre città nei mesi di aprile e maggio. Nel campo PROVV sono riportate le provvigioni sulle vendite. La seguente formula @SOMMADB serve a calcolare il totale delle provvigioni incassate dall'agente Nigro:

@SOMMADB(VENDITE;"PROVV";AGENTE="Nigro") = L.23.220.000

A	---	A	-----	B	-----	C	-----
1		INDIRIZZO		AGENTE		PROVV	
2		Sarpi 4		Morini		28.800.000	
3		Umbria 15		Murri		12.720.000	
4		Nordio 54		Morini		19.920.000	
5		Mameli 7		Sidoli		11.120.000	
6		Delfico 19		Nigro		9.600.000	
7		Farini 74		Nigro		13.620.000	
8		Bronzino 1		Murri		15.880.000	
9		Filzi 33		Camera		12.120.000	

Esempio: @VARDB e @VARDBC

La tabella che segue riporta i dati relativi all'altezza e al peso di dieci soggetti selezionati a caso. Si vuole determinare la variazione dei pesi dei soggetti più alti di 180 cm:

$@VARDB(B2..B11;"PESO";ALTEZZA>180) = 38,77462$

Supponendo invece che i soggetti esaminati rappresentino un campione selezionato a caso da un più ampio gruppo, si dovrebbe usare la seguente funzione:

$@VARDBC(B2..B11;"PESO";ALTEZZA>180) = 43,08292$

A	-----	A	-----	B	--
1		ALTEZZA (cm)		PESO (kg)	
2		190,50		72,73	
3		187,96		86,36	
4		175,26		68,18	
5		175,26		76,37	
6		180,34		77,27	
7		180,34		72,73	
8		187,96		75,00	
9		172,72		68,18	
10		177,80		70,46	

Esempio: @TESTF

@TESTF(A2..A13;B2..B15) = 0,157348

A	-----	A	-----	B	-----
1		Campione1		Campione	
2		84,5		1,65	
3		80,7		4,58	
4		34,5		42,6	
5		54,6		4,37	
6		50,5		30,8	
7		33,7		97,7	
8		46,8		87,2	
9		47,6		40,7	
10		22,8		38,4	
11		15,5		10,6	
12		60,6		56,3	
13		80,5		70,5	
14				9,04	
15				97,3	

Esempio: @TOTGENER

La formula @TOTGENER contenuta nella cella A10 somma tutte le celle della zona A1..A8 contenenti @TOTPARZ (A4 e A8).

A	-----	A	-----	B	-----
1		10			
2		15			
3					
4		25		@TOTPARZ (A1..A2)	
5					
6		20			
7		25			
8		45		@TOTPARZ (A6..A7)	
9					
10		70		@TOTGENER (A1..A8)	

@INDX

@INDX(*zona*; *intest_colonna*; *intest_riga*; [*intest_foglio*]) riporta il contenuto di una cella situata all'intersezione identificata da *intest_colonna*, *intest_riga* ed eventualmente *intest_foglio*.

Argomenti

zona è l'indirizzo o il nome di una zona.

intest_colonna è il contenuto di una cella situata nella prima riga di *zona*.

intest_riga è il contenuto di una cella situata nella prima colonna di *zona*.

intest_foglio è un argomento facoltativo che indica il contenuto della prima cella di *zona*. In caso di omissione dell'argomento, **@INDX** esegue la ricerca su tutti i fogli compresi in *zona*.

intest_colonna, *intest_riga* e *intest_foglio* possono essere valori o stringhe di testo.

Esempi

@INDX

Funzioni @ affini

@SCELTA individua un'immissione in una lista. **@SCANSOR** e **@SCANSVER** individuano immissioni nelle tabelle di consultazione orizzontali e verticali. **@CORRISP** riporta la posizione relativa di una cella in una zona. **@SCANSMAX** produce un referimento assoluto alla cella che contiene il valore più alto di una serie di zone. **@SCANSMIN** genera un riferimento assoluto alla cella che contiene il valore inferiore di una serie di zone

Esempio: @CORRISP

Il dosaggio di un farmaco è determinato dal peso corporeo. Il peso di un paziente, immesso nella cella denominata PESO_PAZIENTE, è di 62 kg:

`@IND(B2..C7;2;@CORRISP(PESO_PAZIENTE;B2..C7;1)) = 2`

A	-----	A	-----	B	-----	C	-----
1		Pound		Chilogrammi		Mcg.	
2		50		22,5		1,5	
3		100		45,5		2,0	
4		150		68,0		2,5	
5		200		90,5		3,0	
6		250		113,5		3,5	
7		300		136,0		4,0	

Esempio: @TESTT

@TESTT(A2..A13;B2..B13) = 0,050022

A ----- A ----- B -----

	Campione1	Campione
1		
2	84,5	65,1
3	80,7	85,4
4	34,5	62,4
5	54,6	73,4
6	50,5	80,3
7	33,7	66,7
8	46,8	87,2
9	47,6	70,4
10	22,8	30,2
11	15,5	60,1
12	60,6	56,3
13	80,5	70,5

Esempio: @INDX

Una tabella denominata TARIFFE (A2..E7) contiene l'elenco delle tariffe applicate per la spedizione di un pacco in diverse località:

@INDX(TARIFFE;"New York";1) = L. 9.290, ossia la tariffa applicata per la spedizione di un pacco di tipo 1 a New York

@INDX(TARIFFE;"Parigi";5) = L. 23.880, ossia la tariffa applicata per la spedizione di un pacco di tipo 5 a Parigi

A	----	A	-----	B	-----	C	-----	D	-----	E	----
1						DESTINAZIONE		PACCHI			
2	Tipo pacchi	Londra		Parigi				Francoforte		New York	
3	1	L. 18.360		L. 19.330				L. 20.120		L. 9.290	
4	2	L. 20.320		L. 21.660				L. 22.030		L. 11.250	
5	3	L. 22.440		L. 23.880				L. 24.000		L. 13.250	
6	4	L. 24.140		L. 25.260				L. 25.750		L. 16.850	
7	5	L. 28.320		L. 29.000				L. 29.800		L. 19.540	

@TESTZ

@TESTZ(*zona1*; *media1*; *deviazione_standard1*; [*code*]; [*zona2*]; [*media2*]; [*deviazione_standard2*])
esegue un test *z* su una o due popolazioni, riportando la probabilità associata.

Argomenti

zona1 è una zona che contiene la prima o l'unica serie di dati da esaminare.

media1 è la media nota della popolazione della *zona1*; può assumere qualsiasi valore.

deviazione_standard1 è la deviazione standard nota della popolazione della *zona1*; deve essere un valore maggiore di zero.

code è un argomento facoltativo che specifica la direzione del test *z*.

code	1-2-3 esegue
1	un test <i>z</i> a una coda
2	un test <i>z</i> a due code; impostazione predefinita in caso di omissione dell'argomento

zona2 è una zona contenente la seconda serie di dati da esaminare.

media2 è la media nota della popolazione della *zona2*; può assumere qualsiasi valore. In caso di omissione dell'argomento, 1-2-3 usa il valore 0.

deviazione_standard2 è la deviazione standard nota della popolazione della *zona2*; deve essere un valore maggiore di zero. In caso di omissione dell'argomento, 1-2-3 usa il valore 1.

Non è consentito l'uso di un argomento facoltativo senza specificare quelli che lo precedono.

Nota

@TESTZ calcola la probabilità associata a un test *z* con un'approssimazione di $\pm 7.5 \cdot 10^{-8}$.

Esempi

La zona A1..A8 contiene i seguenti valori: 12, 19, 21, 22, 18, 16, 15, 17. La media della popolazione, costituita da questi valori, è di 16 e la deviazione standard della popolazione è pari a 3,041381:

@TESTZ(A1..A8;16;3,041381) = 0,081512

Funzioni @ affini

@TESTCHI2 esegue un test chi quadrato, @TESTF esegue un test *F*, mentre TESTI esegue un test *t*.

@DATA

@DATA(*anno;mese;giorno*) calcola il valore data per l'*anno*, il *mese* e il *giorno* specificati.

Argomenti

anno può essere un numero intero compreso tra 0 (l'anno 1900) e 199 (l'anno 2099).

mese può essere un numero intero compreso tra 1 e 12.

giorno può essere un numero intero compreso tra 1 e 31. Il valore usato per *giorno* deve essere un giorno valido per il *mese*; ad esempio, non è possibile usare un argomento *giorno* pari a 31 se si utilizza 4 (aprile) per *mese*.

NOTE

Anche se nell'anno 1900 non ci fu il giorno 29 febbraio (in quanto non si trattava di un anno bisestile), 1-2-3 assegna ugualmente un valore data a quel giorno. Ciò non compromette i calcoli relativi alle date a meno che non risultino comprese tra il 1° gennaio 1900 e il 1° marzo 1900. Se occorre usare date relative a tale periodo, sottrarre 1 dai risultati ottenuti.

Se è necessario che i risultati di un calcolo effettuato da @DATA prendano la forma di una data vera e propria, applicare alla cella contenente la funzione @DATA uno dei formati data.

Esempi

@DATA(92;2;21) genera 33655, oppure 21-Feb-92, se la cella ha il formato GG-Mes-AA.

@DATA(91;2;29) genera ERR, poiché 1991 non era un anno bisestile.

Funzioni @ affini

@ORDDATA calcola il valore data di una data immessa sotto forma di etichetta. @ORARIO calcola il valore orario di un orario specificato. @ADESSO calcola il valore data e orario corrispondente all'ora e data correnti.

@DIFDATA

@DIFDATA(*data_inizio*; *data_fine*; *formato*) calcola il numero di anni, mesi o giorni che intercorrono tra due valori data.

Argomenti

data_inizio e *data_fine* sono valori data.

formato è un codice derivato dalla seguente tabella, da immettere come una qualunque stringa di testo, che specifica il formato da applicare al risultato della funzione @DIFDATA.

formato	genera il numero di
a	anni
m	mesi
g	giorni
mg	giorni, ignorando mesi e anni
am	mesi, ignorando gli anni
ag	giorni, ignorando gli anni

Esempi

Nei seguenti esempi vengono usate le date 5 febbraio 1990 e 15 settembre 1993:

@DIFDATA(@DATA(90;2;15);@DATA(93;9;15);"m") genera 43, ossia il numero di mesi che intercorrono tra il 15 febbraio 1990 e 15 settembre 1993.

@DIFDATA(@DATA(90;2;15);@DATA(93;9;15);"mg") genera 0, perché il giorno del mese in *data_inizio* e in *data_fine* è sempre il 15.

@DIFDATA(@DATA(90;2;15);@DATA(93;9;15);"am") genera 7, ossia il numero di mesi che intercorrono tra febbraio e settembre.

Funzioni @ affini

@D360 e @360GIORNI calcolano il numero di giorni che intercorrono tra due date sulla base di un anno di 360 giorni (12 mesi di 30 giorni ognuno). @GIORNI calcola il numero di giorni che intercorrono tra due date sulla base del conteggio dei giorni specificato dall'utente.

@INFODATA

@INFODATA(*data*; *attributo*) fornisce informazioni sul valore data.

Argomenti

data è un valore data.

attributo è un numero intero derivato dalla seguente tabella.

<i>attributo</i>	<i>genera</i>
1	il giorno della settimana sotto forma di etichetta, in formato abbreviato (lun.)
2	il giorno della settimana sotto forma di etichetta, per esteso (lunedì)
3	il giorno della settimana sotto forma di numero intero, da 0 (lunedì) a 6 (domenica)
4	la settimana dell'anno sotto forma di numero intero da 1 a 53
5	il mese dell'anno sotto forma di etichetta, in formato abbreviato (gen.)
6	il mese dell'anno sotto forma di etichetta, per esteso (gennaio)
7	il numero di giorni del mese specificato da <i>data</i>
8	il numero di giorni rimanenti nel mese specificato da <i>data</i>
9	il valore data corrispondente all'ultimo giorno del mese specificato
10	il trimestre in cui si trova la <i>data</i> , sotto forma di numero intero da 1 (T1) a 4 (T4)
11	1 se l'anno specificato da <i>data</i> è un anno bisestile; 0 in caso contrario
12	il giorno dell'anno specificato da <i>data</i> , sotto forma di numero da 1
13	il giorno dell'anno specificato da <i>data</i> , sotto forma di numero

Esempi

@INFODATA(23063;7) = 28, che è il numero di giorni di cui era composto il mese di febbraio 1963.

@INFODATA(@DATA(92;10;5);10) = 4, perché il mese di ottobre cade nel quarto trimestre.

@ORDDATA

@ORDDATA(*stringa*) calcola il valore data della data specificata in *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta. *stringa* deve avere uno dei seguenti formati:

- un formato data compreso nell'elenco "Usati di frequente";
- uno dei cinque formati data ammessi nelle celle con formato Generale: 31-Dic-96, Dic-96, 31-Dic e i formati data esteso e contratto specificati nelle impostazioni regionali (nazionali) del sistema operativo.

NOTA

Se il risultato del calcolo eseguito da **@ORDDATA** deve prendere la forma di una data vera e propria, applicare alla cella contenente la funzione **@ORDDATA** uno dei formati data.

La funzione **@ORDDATA** è utile quando si lavora con dati importati da un altro programma, come ad esempio un programma di elaborazione testi.

Esempi

@ORDDATA("21-Feb-91") genera il valore data 33290

@ORDDATA(Compleanno) genera il valore data 20723 se la cella denominata Compleanno contiene l'etichetta 25-Set-56.

Funzioni @ affini

@DATA calcola il valore data della data specificata. **@STRINGADATA** converte un valore data nella data corrispondente, visualizzandola sotto forma di etichetta.

@GIORNO

@GIORNO(*valore_data*) calcola il giorno del mese (da 1 a 31) corrispondente a *valore_data*.

Argomenti

valore_data può essere un valore compreso tra 1 (1° gennaio 1900) e 73050 (31 dicembre 2099).

NOTA

Per fornire il *valore_data* alla funzione @GIORNO, si può utilizzare un'altra funzione @ di data.

@GIORNO può fornire l'argomento *giorno* ad altre funzioni @ di data che si basano su date calcolate in precedenza.

Esempi

@GIORNO(@ADESSO) = giorno del mese corrente.

@GIORNO(D9) = 12 se la cella D9 contiene il valore data 33250 (12-gen.-91).

Funzioni @ affini

@MESE calcola il mese, @ANNO calcola l'anno e @GSETT calcola il giorno della settimana, utilizzando un valore data.

@GIORNI

@GIORNI(*data_inizio*,*data_fine*,[*base*]) calcola il numero di giorni che intercorrono tra due date, utilizzando la *base* specificata per il conteggio dei giorni.

Argomenti

data_inizio e *data_fine* sono valori data. Se *data_inizio* è anteriore a *data_fine*, il risultato di @GIORNI è un numero positivo; se invece è posteriore, il risultato è negativo. Se *data_inizio* è uguale a *data_fine*, il risultato della funzione @GIORNI è zero.

base è un argomento facoltativo che specifica il tipo di base da utilizzare per il conteggio dei giorni. *base* è un valore derivato dalla tabella sottostante.

base	base per il conteggio dei giorni
0	30/360 (impostazione predefinita in caso di omissione dell'argomento)
1	Effettivo/Effettivo
2	Effettivo/360
3	Effettivo/365
4	Europeo 30/360

Esempi

@GIORNI(@DATA(93;4;16);@DATA(93;9;25)) = 159, che è il numero di giorni che intercorrono tra il 16 aprile 1993 e il 25 settembre 1993, calcolati sulla base di un anno di 360 giorni, ossia 12 mesi di 30 giorni ciascuno.

@GIORNI(@DATA(93;4;16);@DATA(93;9;25);1) = 162, che è il numero di giorni che intercorrono tra il 16 aprile 1993 e il 25 settembre 1993, calcolati sulla base del numero effettivo dei giorni compresi tra le date specificate.

Funzioni @ affini

@DIFDATA calcola il numero di anni, mesi e giorni che intercorrono tra due date. @D360 e @360GIORNI calcolano il numero di giorni che intercorrono tra due date, sulla base di un anno di 360 giorni. @GIORNILAVNETTI calcola il numero di giorni che intercorrono tra due date, escludendo i fine settimana e le festività.

@D360, @360GIORNI

@D360(*data_inizio;data_fine*) calcola il numero di giorni che intercorrono tra due date, sulla base di un anno di 360 giorni (12 mesi di 30 giorni ognuno).

@360GIORNI(*data_inizio;data_fine*) calcola il numero di giorni che intercorrono tra due date, sulla base di un anno di 360 giorni, secondo gli standard del mercato mobiliare statunitense.

Argomenti

data_inizio e *data_fine* sono valori data.

NOTE

La formula usata per il calcolo della funzione **@360GIORNI** è conforme alle modifiche apportate nel 1990 all'edizione 1986 degli *Standard Security Calculation Methods* della Securities Industry Association.

@360GIORNI e **@D360** tendono a dare risposte diverse per gli stessi dati quando *data_inizio* o *data_fine* corrisponde all'ultimo giorno del mese.

Esempi

@360GIORNI(**@DATA**(89;4;16);**@DATA**(89;9;25)) = 159

@D360(33290;33524) = 232, ossia il numero di giorni che intercorrono tra il 21 febbraio 1991 e il 13 ottobre 1991, sulla base di un anno di 360 giorni.

Funzioni @ affini

@DIFDATA calcola il numero di anni, mesi o giorni che intercorrono tra due date. **@GIORNI** calcola il numero di giorni che intercorrono tra due date, usando la base specificata per il conteggio dei giorni. **@GIORNILAVNETTI** calcola il numero di giorni che intercorrono tra due date, escludendo i fine settimana e le festività. **@GLAVOR** calcola la data del giorno che cade il numero specificato di giorni prima o dopo una determinata data, escludendo i fine settimana e le festività. **@ALTROMESE** calcola la data che cade il numero di mesi specificato prima o dopo una determinata data.

@AQP

@AQP(costo;realizzo;vita;periodo) calcola la quota detraibile di un bene con valore iniziale pari a costo, con una certa previsione di vita utile e un valore finale di realizzo per un periodo di tempo specificato, utilizzando il metodo di ammortamento a quote fisse decrescenti.

Argomenti

costo è la somma corrisposta per un bene e consiste in un qualsiasi valore positivo o uguale a zero. Se il costo è uguale a zero, anche il risultato di **@AQP** sarà pari a zero.

realizzo è il valore stimato di un bene al termine della sua vita utile. Questo argomento può essere un qualsiasi valore positivo oppure zero. Se *realizzo* è maggiore di *costo*, il risultato di **@AQP** è un valore negativo.

vita è il numero di periodi di tempo necessari affinché un bene raggiunga il valore di *realizzo*. *vita* può assumere qualsiasi valore maggiore o uguale a uno, purché minore o uguale a *vita*.

periodo è il periodo di tempo per il quale occorre calcolare la quota di ammortamento. Questo argomento deve essere un valore maggiore o uguale a 1.

vita e *periodo* devono essere espressi nella stessa unità di misura, solitamente in anni.

NOTA

Il metodo di ammortamento a quote fisse decrescenti rallenta il processo di ammortamento rispetto al metodo a doppie quote proporzionali ai valori residui; pertanto la quota di ammortamento ammessa in detrazione è maggiore negli ultimi periodi. Il processo di ammortamento termina quando il valore contabile del bene in questione, pari al costo totale del bene meno l'ammortamento totale in tutti i periodi precedenti, raggiunge il valore di realizzo.

Esempi

Per L. 10.000.000 viene acquistata una macchina per ufficio che ha una vita utile di otto anni e un valore di realizzo pari a L. 1.200.000. La funzione che segue ha lo scopo di calcolare la quota di ammortamento relativa al quinto anno.

@AQP(10000000;1200000;8;5) = L. 806.510

Funzioni @ affini

@VDB calcola l'ammortamento secondo il metodo a quote proporzionali con tasso variabile. **@DDB** usa il metodo di ammortamento a doppie quote proporzionali ai valori residui, **@AQC** applica il metodo di ammortamento a quote costanti e **@SYD** calcola l'ammortamento secondo il metodo della somma degli anni di prevista durata.

@DDB

@DDB(*costo;realizzo;vita;periodo*) calcola la quota detraibile di un bene avente un *costo* iniziale, una *vita* utile e un valore finale di *realizzo*, per un *periodo* specificato, secondo il metodo di ammortamento a doppie quote proporzionali ai valori residui.

Argomenti

costo rappresenta la somma pagata per l'acquisto del bene. *costo* può essere un valore qualsiasi, oppure l'indirizzo o il nome di una cella contenente un valore qualsiasi, maggiore o uguale al valore di *realizzo*.

realizzo rappresenta il valore stimato del bene al termine della sua vita utile. *realizzo* può essere un valore qualsiasi oppure l'indirizzo o il nome di una cella contenente un valore qualunque.

vita rappresenta il numero di periodi necessari affinché si compia l'ammortamento del bene fino al raggiungimento del suo valore di *realizzo*. *vita* può essere un valore qualsiasi, oppure l'indirizzo o il nome di una cella contenente un valore qualunque, maggiore di 2.

periodo rappresenta il periodo di tempo per cui si vuole calcolare la quota detraibile. *periodo* può essere un valore qualsiasi, oppure l'indirizzo o il nome di una cella contenente un valore qualsiasi, maggiore o uguale a 1.

vita e *periodo* vanno espressi nelle stesse unità di misura, solitamente in anni.

NOTA

La funzione **@DDB** usa il metodo di ammortamento a doppie quote proporzionali ai valori residui che accelera il processo di ammortamento, applicando (e quindi detraendo) nei primi periodi una quota di ammortamento maggiore rispetto a quella applicata negli ultimi periodi. L'ammortamento termina quando il valore contabile del bene (vale a dire il suo costo totale meno la somma di tutte le quote di ammortamento relative ai periodi precedenti) è pari al valore di *realizzo*.

Se il valore di *realizzo* di un bene è relativamente basso, può succedere che **@DDB** non ammortizzi completamente il bene entro il termine della vita utile stimata. In questo caso è possibile usare **@VDB** che ammortizzerà completamente il bene entro la vita utile stimata.

Esempi

Si supponga di aver appena acquistato una macchina per L. 10.000.000, con una vita utile stimata di 8 anni e un valore di *realizzo* finale di L. 1.200.000. Calcolare la quota di ammortamento per il quinto anno utilizzando il metodo di ammortamento a doppie quote proporzionali:

@DDB(10000000;1200000;8;5) = L. 791.016, ossia l'ammortamento per il quinto anno di vita del bene.

Funzioni @ affini

@VDB calcola l'ammortamento secondo il metodo a quote proporzionali con tasso variabile. **@AQC** usa il metodo a quote costanti, e **@SYD** applica il metodo "americano" (somma degli anni di prevista durata).

@DECIMALE

@DECIMALE(*esadecimale*) converte un valore esadecimale nel suo equivalente decimale con segno.

Argomenti

esadecimale è un valore compreso tra 00000000 e FFFFFFFF, e va immesso come una qualunque stringa di testo. Può essere formato da un massimo di otto caratteri e deve contenere solo numeri da 0 a 9 e lettere (maiuscole o minuscole) da A a F.

NOTE

I valori esadecimali da 00000000 a 7FFFFFFF corrispondono a 0 e a valori decimali positivi.

I valori esadecimali da 80000000 a FFFFFFFF corrispondono a valori decimali negativi.

Esempi

@DECIMALE("1A") = 26

@DECIMALE("FFFFFFFE") = -2

Funzioni @ affini

@ESA converte i numeri decimali in esadecimali.

@GRADIRAD

@GRADIRAD(*gradi*) converte i *gradi* in radianti.

Argomenti

gradi può essere un qualsiasi valore.

Esempi

@GRADIRAD(30) = 0,523599 radianti

@COS(@GRADIRAD(45)) = 0,707107, che è il coseno di un angolo di 45 gradi

Funzioni @ affini

@RADGRAD converte i radianti in gradi.

@DEVQ

@DEVQ(*lista*) calcola la somma delle deviazioni quadratiche dei valori compresi in una *lista*, ricavandola dalla media dei valori stessi.

Argomenti

lista può contenere i seguenti elementi in qualunque combinazione: numeri, formule numeriche e indirizzi o nomi di zone contenenti numeri o formule numeriche. Tali elementi devono essere divisi tra loro dagli appositi separatori di argomenti.

Vedere anche argomenti delle funzioni @ statistiche.

Esempi

@DEVQ(2;3;9;8;15;2;1) = 159,4286

Funzioni @ affini

@SCARTO e @SCARTOPURO calcolano la deviazione standard dei valori compresi in una lista.

@FERR

@FERR(*limite_inferiore*;[*limite_superiore*]) calcola la funzione di errore integrata tra il *limite_inferiore* e il *limite_superiore*.

Argomenti

limite_inferiore è il limite inferiore utilizzabile per l'integrazione di **@FERR** e può assumere qualsiasi valore.

limite_superiore è un argomento facoltativo che specifica il limite superiore utilizzabile per l'integrazione di **@FERR** e può assumere valore maggiore o uguale a *limite_inferiore*. In caso di omissione dell'argomento, **@FERR** esegue l'integrazione tra 0 e *limite_inferiore*.

NOTA

@FERR calcola la funzione d'errore con un'approssimazione di $\pm 1,2 \times 10^{-7}$.

Esempi

@FERR(0,7) = 0,677801

@FERR(0,8) = 0,742101

@FERR(0,7;0,8) = 0,0643, che è la differenza tra i valori degli esempi precedenti.

Funzioni @ affini

@FERRC calcola la funzione d'errore complementare. **@FERRD** calcola la derivata della funzione d'errore.

@FERRC

@FERRC(x) calcola la funzione d'errore complementare integrata tra x e ∞ .

Argomenti

x può assumere qualsiasi valore.

NOTE

@FERRC(x) è uguale a $1-\text{@FERR}(x)$.

@FERRC calcola la funzione d'errore complementare con un'approssimazione di $\pm 3 \cdot 10^{-7}$.

Esempi

@FERRC(0,7) = 0,322199

Funzioni @ affini

@FERR calcola la funzione d'errore integrata tra i limiti inferiore e superiore specificati. @FERRD calcola la derivata della funzione d'errore.

@FERRD

@FERRD(x) calcola la derivata della funzione d'errore.

Argomenti

x può assumere qualunque valore.

Esempi

@FERRD(0,7) = 0,691275

Funzioni @ affini

@FERR calcola la funzione d'errore integrata tra i limiti superiore e inferiore specificati.

@ERR

@ERR genera il valore ERR.

NOTA

La funzione @ERR è utile per segnalare la presenza di errori nei calcoli, ma viene raramente usata da sola. Infatti, se utilizzata come argomento di @SE, @ERR genera il valore ERR in particolari circostanze, come ad esempio nel caso in cui una formula produca un valore inaccettabile (come una rata mensile negativa).

Esempi

@SE(B14>3;@ERR;B14) = ERR se il valore nella cella B14 è maggiore di 3

Funzioni @ affini

@ND genera il valore ND (non disponibile). @DERR verifica la presenza del valore ERR.

@PARI

@PARI(x) arrotonda il valore x al successivo numero pari intero.

Argomenti

x può assumere un qualsiasi valore. Se è un intero pari, la funzione @PARI genera x.

Esempi

@PARI(2,25) = 4

@PARI(2) = 2

@PARI(-2,25) = -4

Funzioni @ affini

@DISPARI arrotonda un valore al successivo intero dispari. @ARROT, @ARROTDIFETTO e @ARROTECESSO arrotondano un valore al numero specificato di cifre decimali. @ARROTM arrotonda un valore al multiplo specificato. @INT tronca un valore, eliminandone la parte decimale. @TRONCADEC tronca un valore a partire dalla cifra decimale specificata.

@ESATTO

@ESATTO(*stringa1*;*stringa2*) confronta due serie di caratteri. Se le due cifre presentano un'equivalenza totale, **@ESATTO** genera 1 (vero); se invece le due serie non sono identiche, **@ESATTO** genera 0 (falso).

Argomenti

stringa1 e *stringa2* possono essere stringhe di testo racchiuse fra virgolette (" "), formule che generano stringhe di testo, oppure gli indirizzi o i nomi di celle contenenti etichette o formule che generano etichette.

NOTA

La funzione **@ESATTO** è più precisa dell'operatore uguale (=) in una formula. A differenza di uguale (=), **@ESATTO** fa distinzione fra lettere maiuscole e minuscole, e tra accentate e non.

La funzione **@ESATTO** viene utilizzata per controllare che i dati immessi coincidano perfettamente con i dati richiesti, come ad esempio nelle macro che confrontano i dati immessi con i dati richiesti prima di procedere. In tal modo è possibile impostare delle password per le macro.

Esempi

@ESATTO("ATENE";"Atene") = 0 (falso).

@ESATTO("Scaduto";B2) = 1 (vero) se la cella B2 contiene l'etichetta Scaduto.

@ESATTO("400";400) = ERR, perché *stringa2* è un valore.

@ESP

@ESP(x) calcola il valore della costante e (approssimativamente 2,718282) elevata alla potenza x .

Argomenti

x può essere un valore compreso all'incirca tra -709,7827 e 709,7827.

NOTA

Se x è maggiore di circa 709,7827 oppure minore di circa -709,7827, il calcolo è troppo grande perché 1-2-3 possa memorizzarlo; di conseguenza @ESP genera ERR.

Esempi

@ESP(0,7) = 2,013753

Funzioni @ affini

@ESP2 calcola il valore di e elevato alla potenza (x^2). @LOGN è l'inverso di @ESP.

@ESP2

@ESP2(x) calcola il valore della costante e (circa 2,718282) elevato alla potenza (x^2).

Argomenti

x è un valore compreso tra circa -106,570 a circa 106,570.

NOTA

Se x è maggiore di circa 106,570 oppure minore di circa -106,570, il calcolo è troppo grande perché 1-2-3 possa memorizzarlo; di conseguenza @ESP2 genera ERR. Se x è minore di circa 15,098 o maggiore di circa 15,098, 1-2-3 è in grado di calcolare e memorizzare il valore di @ESP2, ma non di visualizzarlo (la cella conterrà una serie di asterischi).

Esempi

@ESP2(0,7) = 0,612626

Funzioni @ affini

@ESP calcola il valore di e elevato alla potenza specificata.

@FATTORIALE

@FATTORIALE(n) calcola il fattoriale di n .

Argomenti

n può essere qualsiasi numero intero positivo oppure 0.

NOTE

Il fattoriale di n è uguale al prodotto di tutti i numeri interi positivi compresi tra 1 e n .

Se n è maggiore o uguale a 171, il calcolo diventa troppo grande perché 1-2-3 lo possa memorizzare, e la funzione @FATTORIALE genera ERR.

Esempi

@FATTORIALE(0) = 1

@FATTORIALE(5) = 120, ossia il risultato di $1*2*3*4*5$

Funzioni @ affini

@FATTORIALELOGN calcola il logaritmo naturale del fattoriale di n . @PROD moltiplica i valori indicati nella lista.

@FATTORIALELOGN

@FATTORIALELOGN(n) calcola il logaritmo naturale del fattoriale di n .

Argomenti

n è un qualsiasi numero intero oppure 0.

NOTA

Il fattoriale di n è uguale al prodotto di tutti i numeri interi positivi compresi tra 1 e n .

Esempi

@FATTORIALELOGN(0) = 0, ossia il risultato di @LOGN(1).

@FATTORIALELOGN(5) = 4,787492, ossia il risultato di @LOGN(1*2*3*4*5).

Funzioni @ affini

@FATTORIALE calcola il fattoriale di n . @LOGN calcola il logaritmo naturale di un valore.

@FALSO

@FALSO genera il valore logico 0 (falso).

NOTA

Se un'espressione logica risulta vera (ad esempio, $A1=B1$), ha valore logico 1. Se invece è falsa, ha valore logico 0.

@FALSO equivale a usare il valore 0 nelle formule di elaborazione delle condizioni logiche, con la differenza che, con @FALSO, la formula risulta più comprensibile.

La funzione @FALSO viene usata nelle macro e nelle funzioni @, quali @SE e @SCELTA, che richiedono il valore logico 0 (falso). @FALSO è anche utile come argomento y per @SE, cioè il valore riportato se non viene soddisfatta la condizione.

Esempi

@SE(A6>500;@VERO;@FALSO) = 0, se la cella A6 contiene un valore minore o uguale a 500.

Funzioni @ affini

@VERO genera il valore logico 1.

@DISTRIBF

@DISTRIBF(x ; $gradi_libertà1$; $gradi_libertà2$; [$tipo$]) calcola la distribuzione F .

Argomenti

x è il valore a cui deve essere calcolata la distribuzione F . Il valore di x dipende dal *tipo* immesso:

se tipo è	x è
0	valore critico o limite della distribuzione F cumulativa; valore maggiore o uguale a 0 (impostazione predefinita in caso di omissione dell'argomento)
1	probabilità; valore compreso tra 0 e 1

$gradi_libertà1$ e $gradi_libertà2$ sono i numeri dei gradi di libertà per il primo e il secondo campione. $gradi_libertà1$ e $gradi_libertà2$ sono numeri interi positivi.

tipo è un argomento facoltativo che specifica come 1-2-3 deve calcolare @DISTRIBF.

tipo	1-2-3 calcola
0	il livello di significatività corrispondente al valore critico x (impostazione predefinita in caso di omissione dell'argomento)
1	il valore critico corrispondente al livello di significatività x

NOTE

@DISTRIBF calcola la distribuzione F cumulativa con un'approssimazione di $\pm 3 \cdot 10^{-7}$. Se @DISTRIBF non riesce a eseguire il calcolo con un'approssimazione dello 0,0000001 dopo 100 iterazioni, produce il risultato ERR.

La distribuzione F è una distribuzione continua ottenuta dal rapporto tra due distribuzioni chi-quadrato, ognuna delle quali viene divisa per il relativo numero di gradi di libertà.

@DISTRIBF consente di determinare il grado di variazione di due campioni.

Esempi

@DISTRIBF(3,07;8;10) = 0,05

@DISTRIBF(0,05;8;10) = 0,999865

Funzioni @ affini

@TESTF calcola la probabilità associata a un test F . @DISTRIBCHI2 calcola la distribuzione chi-quadrato.

@DISTRIBT calcola la distribuzione t di Student.

@RICER

@RICER(*stringa_ricerca*; *stringa*; *numero_inizio*) calcola la posizione all'interno della *stringa* in cui 1-2-3 trova la prima occorrenza di *stringa_ricerca*, iniziando dalla posizione indicata da *numero_inizio*.

Argomenti

stringa_ricerca e *stringa* possono essere stringhe di testo racchiuse fra virgolette (" "), una formula che genera una stringa di testo oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta. *numero_inizio* è uno scarto.

NOTE

Se 1-2-3 non trova *stringa_ricerca* in *stringa*, **@RICER** genera il valore ERR. **@RICER** genera ERR anche quando *numero_inizio* è maggiore del numero di caratteri contenuti in *stringa*, oppure se *numero_inizio* è negativo.

@RICER fa distinzione tra maiuscole e minuscole e tra lettere accentate e non; ad esempio, **@RICER** non troverà la *stringa_ricerca* "e" nella *stringa* "FIRENZE".

@RICER può essere usata anche assieme a @MEZZO o @SOSTIT per individuare ed estrarre/sostituire una stringa di testo.

Esempi

@RICER("p";"Fatture in pagamento";0) = 11 perché la *stringa_ricerca* p si trova nella dodicesima posizione della *stringa* Fatture in pagamento

@VALFUT, @VALFUTURO

@VALFUT(*versamenti;interesse;durata*) calcola il valore futuro di un investimento sulla base di una serie di *versamenti* di uguale importo, di un tasso d'*interesse* periodico e del numero di periodi specificati da *durata*.

@VALFUTURO(*versamenti;interesse;durata,[tipo];[valore_attuale]*) calcola il valore futuro di un investimento del *valore_attuale* sulla base di una serie di *versamenti* di uguale importo, di un tasso d'*interesse* periodico e del numero di periodi specificati da *durata*. @VALFUTURO esegue i calcoli per una rendita immediata o una rendita anticipata, a seconda del valore specificato per *tipo*.

Argomenti

versamenti e *durata* possono assumere qualsiasi valore.

Nella funzione @VALFUTURO, l'argomento *durata* deve avere un valore positivo.

interesse è un valore maggiore di -1.

tipo è un argomento facoltativo che specifica se deve essere calcolata una rendita immediata o anticipata.

tipo 1-2-3 calcola

- | | |
|---|--|
| 0 | una rendita immediata (versamenti da effettuare alla fine di un periodo); impostazione predefinita in caso di omissione dell'argomento |
| 1 | una rendita anticipata (versamento da effettuare all'inizio di un periodo) |

valore_attuale è un argomento facoltativo che specifica il valore attuale di una serie di versamenti futuri. Questo argomento può assumere qualsiasi valore. In caso di omissione, 1-2-3 usa un *valore_attuale* pari a zero.

Non è consentito l'uso di un argomento facoltativo senza aver specificato quelli che lo precedono.

NOTA

Il periodo usato per calcolare l'*interesse* deve essere uguale a quello utilizzato per *durata*; ad esempio, se si intende calcolare una rata mensile, occorre immettere l'interesse e la durata in incrementi mensili. Solitamente ciò significa che il tasso di interesse deve essere diviso per 12 e che il numero di anni di *durata* deve essere moltiplicato per 12.

Esempi

Si supponga di voler depositare L. 2.000.000 ogni anno per i prossimi vent'anni in un conto risparmio. Questo genere di conto rende un interesse composto del 7,5%, corrisposto l'ultimo giorno dell'anno. A questo punto si vuol calcolare il valore che il conto assumerà tra vent'anni. Il versamento annuale viene effettuato l'ultimo giorno dell'anno.

@VALFUT(2000000;0,075;20) produce il risultato di L. 86.609.363, ossia l'importo del conto alla fine dei vent'anni.

Se invece il versamento annuale viene effettuato il primo giorno dell'anno, si dovrà utilizzare la seguente funzione:

@VALFUTURO(2000000;0,075;20;1) = L. 93.105.064

Funzioni @ affini

@VALFUT2 calcola il valore futuro di un investimento sulla base di una serie di *versamenti* di uguale importo, di un tasso di *interesse* periodico e del numero di periodi specificati da *durata*, presupponendo una rendita anticipata.

@FVAMOUNT genera il valore futuro di una somma complessiva investita a un certo tasso di interesse per un certo numero di periodi.

@VALATT e @VALATTUALE calcolano il valore attuale di un investimento. @VAN calcola il valore attuale netto di un investimento, scontando il valore futuro al valore presente.

@STRINGADATA

@STRINGADATA(*data*) converte un valore data nella data corrispondente sotto forma di etichetta con il formato data internazionale standard.

Argomenti

data deve essere un valore data.

NOTE

Per modificare il formato data internazionale standard, usare le impostazioni internazionali del sistema operativo.

Esempi

Quando il formato data internazionale standard è mm/gg/aa, **@STRINGADATA**(34635) produce l'etichetta 10/28/94.

Funzioni @ affini

@ORDDATA calcola il valore data della data specificata sotto forma di etichetta. **@DATA** calcola il valore data corrispondente alla data specificata.

@DURATA, @DURATAM

@DURATA(*liquidazione;scadenza;cedola;rendimento;[frequenza];[base]*) calcola la durata annua di un titolo con pagamento periodico degli interessi.

@DURATAM(*liquidazione;scadenza;cedola;rendimento;[frequenza];[base]*) calcola la durata annua modificata di un titolo con pagamento periodico degli interessi.

Argomenti

liquidazione è il giorno dei compensi di un titolo e consiste in un valore data.

scadenza è il giorno in cui viene rimborsato un titolo; deve essere un valore data. Se *scadenza* è inferiore o uguale a *liquidazione*, le funzioni **@DURATA** e **@DURATAM** generano **ERR**.

cedola è il tasso nominale annuo di un titolo, e consiste in un valore positivo o uguale a zero.

rendimento rappresenta il rendimento annuale, e può essere un valore positivo o uguale a 0.

frequenza è un argomento facoltativo che indica il numero di pagamenti delle cedole previsti in un anno. *frequenza* può assumere i valori illustrati nella tabella che segue:

frequenza	frequenza dei pagamenti delle cedole
1	annuale
2	semestrale; impostazione predefinita in caso di omissione dell'argomento
4	trimestrale
12	mensile

base è un argomento facoltativo che indica il tipo di base per il conteggio dei giorni da usare. Questo argomento può assumere i valori illustrati dalla seguente tabella:

base	base per il conteggio dei giorni
0	30/360; impostazione predefinita in caso di omissione dell'argomento
1	effettivo/effettivo
2	effettivo/360
3	effettivo/365
4	europeo 30/360

Non è consentito usare un argomento facoltativo senza specificare quelli che lo precedono.

NOTE

La durata è il periodo di tempo medio ponderato dei movimenti di cassa di un titolo. I pesi sono il valore attuale di ogni movimento di cassa espresso come frazione del valore attuale di tutti i movimenti di cassa.

Esempi

Un titolo prevede come data di liquidazione il 1 luglio 1993 e come data di scadenza il 1 dicembre 1998. Il tasso nominale d'interesse semestrale è del 5,50% e il rendimento annuo del 5,61%. La base per il conteggio dei giorni usata per il titolo è 30/360.

La seguente funzione calcola la durata annua del titolo:

@DURATA(**@DATA**(93;7;1);**@DATA**(98;12;1);0,055;0,0561;2;0) = 4,734591

Quest'altra funzione, invece, riporta la durata annua modificata del titolo:

@DURATAM(**@DATA**(93;7;1);**@DATA**(98;12;1);0,055;0,0561;2;0) = 4,605410

Funzioni @ affini

@MATURATO calcola gli interessi maturati sui titoli che prevedono pagamenti periodici degli interessi. **@PREZZO** calcola il prezzo dei titoli con pagamento periodico degli interessi per un valore nominale pari a \$100.

@RENDIMENTO calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi.

@DVUOTA

@DVUOTA(*posizione*) verifica se in *posizione* si trova una cella vuota. Se trova una cella vuota in *posizione*, la funzione produce 1 (vero); in caso contrario, produce il valore 0 (falso).

Argomenti

posizione è il nome o l'indirizzo di una sola cella. Se si specifica una zona per *posizione*, @DVUOTA genera 0 (falso).

Esempi

@DVUOTA(A1) = 1 se la cella A1 è una cella vuota.

@DVUOTA(A1) = 0 se la cella A1 contiene il valore 1.963.

@DVUOTA(A1) = 0 se la cella A1 contiene l'etichetta Incassi.

@DVUOTA(A1) = 0 se la cella A1 contiene un prefisso d'etichetta.

@SCANSMAX, @SCANSMIN

@SCANSMAX(*lista_zone*) produce un riferimento assoluto, comprendente anche il nome del file della cartella di lavoro, alla cella che contiene il valore più alto di una serie di zone.

@SCANSMIN(*lista_zone*) genera un riferimento assoluto, comprendente anche il nome del file della cartella di lavoro, alla cella che contiene il valore inferiore di una serie di zone.

Argomenti

lista_zone può essere costituito da una qualsiasi combinazione di zone. I nomi o gli indirizzi delle zone devono essere divisi tra loro dagli appositi separatori di argomenti.

Per includere in *lista_zone* una zona composta da una sola cella, occorre rappresentarla con un indirizzo di zona: ad esempio, per usare la cella A1, occorre immettere A1..A1.

1-2-3 ignora le etichette e le celle vuote comprese in *lista_zone*.

Se nessuna delle celle di *lista_zone* contiene dei valori, le funzioni **@SCANSMAX** e **@SCANSMIN** producono ND.

Esempi

La directory di 1-2-3 contiene i file delle cartelle di lavoro OFFERTA1.WK4, OFFERTA2.WK4 e OFFERTA3.WK4, ognuno dei quali contiene un'offerta proposta da un fornitore diverso. Tutte le cartelle di lavoro sono state create sulla base dello stesso modello e perciò in ognuno di esse il totale dell'offerta è contenuto in una cella denominata TOTALE.

La formula che segue produce la posizione, comprendente anche il nome del file della cartella di lavoro, dell'offerta più alta:

@SCANSMAX(<<OFFERTA1.WK4>>TOTALE;<<OFFERTA2.WK4>>TOTALE;<<OFFERTA3.WK4>>TOTALE)

Quest'altra formula, invece, dà la posizione, nonché il nome del file della cartella di lavoro, dell'offerta inferiore:

@SCANSMIN(<<OFFERTA1.WK4>>TOTALE;<<OFFERTA2.WK4>>TOTALE;<<OFFERTA3.WK4>>TOTALE)

Funzioni @ affini

@SCANSOR e **@SCANSVER** individuano le immissioni nelle tabelle di consultazione orizzontali e verticali. **@IND** individua il contenuto di una cella quando si specifica lo scarto della colonna e della riga. **@SCELTA** sostituisce una tabella di consultazione che occupa una sola riga. **@CORRISP** trova la posizione relativa della cella di cui si è indicato il contenuto. **@INDX** riporta il contenuto di una cella, la cui posizione è specificata con le intestazioni di colonna, riga e foglio.

@GIORNILAVNETTI

@GIORNILAVNETTI(*data_inizio*; *data_fine*; [*festività*]; [*fine_settimana*]) calcola il numero di giorni che intercorrono tra *data_inizio* e *data_fine*, escludendo i fine settimana e le festività.

Argomenti

data_inizio e *data_fine* sono valori data.

festività è un argomento facoltativo che specifica i giorni da escludere dal calcolo di **@GIORNILAVNETTI**, ed è costituito dal nome o dall'indirizzo di una zona contenente valori data.

fine_settimana è un argomento facoltativo che specifica da quali giorni della settimana è costituito il fine settimana. *fine_settimana* è inoltre una stringa di testo che usa i numeri interi compresi tra 0 (lunedì) e 6 (domenica) a rappresentazione dei giorni facenti parte del fine settimana.

Per esempio, la stringa "45" indica che venerdì e sabato fanno parte del fine settimana. In caso di omissione dell'argomento *fine_settimana*, 1-2-3 usa la stringa "56", in base alla quale il fine settimana è costituito da sabato e domenica. Per indicare l'assenza del fine settimana, usare il numero 7.

Non è consentito l'uso di un argomento facoltativo in assenza di quelli che lo precedono.

NOTE

Il risultato di **@GIORNILAVNETTI** comprende sia la *data_inizio* che la *data_fine*.

Se si desidera specificare l'argomento *fine_settimana* mentre si vuole omettere *festività*, basterà specificare una cella vuota per quest'ultimo.

Esempi

In questo esempio si vuole determinare quanti giorni lavorativi intercorrono tra martedì 1 novembre 1994 e martedì 1 dicembre 1994. Poiché il 1 novembre è un giorno di festa, occorre specificare il corrispondente valore data in una zona denominata FESTIVI. Inoltre, dato che i fine settimana sono costituiti da sabati e domeniche, l'argomento *fine_settimana* può essere omissivo.

@GIORNILAVNETTI(**@DATA**(94;11;1);**@DATA**(94;12;1);FESTIVI) = 21

Funzioni @ affini

@360GIORNI e **@D360** calcolano entrambe il numero di giorni che intercorrono tra due valori data. **@GIORNI** calcola il numero di giorni che intercorrono tra due date utilizzando la base per il conteggio dei giorni specificata.

@GLAVOR calcola la data del giorno che cade il numero specificato di giorni prima o dopo una determinata data, escludendo i fine settimana e le festività. **@ALTROMESE** calcola la data che cade un certo numero di mesi prima o dopo la data specificata.

@ALTROMESE

@ALTROMESE(*data_inizio*; *mesi*; [*giorno_del_mese*]; [*base*]) calcola il valore data della data che cade un dato numero di *mesi* prima o dopo la *data_inizio*.

Argomenti

data_inizio deve essere un valore data.

mesi deve essere un numero intero, positivo se si intende specificare un numero di mesi successivi alla *data_inizio*, oppure negativo se i mesi precedono.

giorno_del_mese è un argomento facoltativo che indica il giorno del mese in cui deve cadere il risultato di @ALTROMESE. Questo argomento può assumere i valori illustrati nella tabella che segue.

<i>giorno_del_mese</i>	@ALTROMESE riporta
0	data che cade nello stesso giorno del mese di <i>data_inizio</i> . Se però <i>data_inizio</i> cade in un giorno del mese che non esiste in quel mese (per esempio, se <i>data_inizio</i> corrisponde a 30 gennaio 1994 e il mese risultante è febbraio che ha 28 giorni), @ALTROMESE riporta la data dell'ultimo giorno del mese. Questa è inoltre l'impostazione predefinita dell'argomento <i>giorno_del_mese</i> in caso di omissione.
1	data del primo giorno del mese
2	data dell'ultimo giorno del mese

base è un argomento facoltativo che indica il tipo di base per il conteggio dei giorni che si intende usare; può assumere i valori illustrati dalla seguente tabella.

<i>base</i>	base per il conteggio dei giorni
0	30/360
1	effettivo/effettivo; impostazione predefinita in caso di omissione dell'argomento
2	effettivo/360
3	effettivo/365
4	europeo 30/360

Non è consentito usare un argomento facoltativo senza specificare quelli che lo precedono.

Esempi

Per determinare la data dell'ultimo giorno del mese, un mese dopo martedì 7 aprile 1994, si userebbe la seguente funzione:

@ALTROMESE(@DATA(94;4;7);1;2) = 34485, ovvero martedì 31 maggio 1994

Funzioni @ affini

@360GIORNI e @D360 calcolano entrambe il numero di giorni che intercorrono tra due valori data. @GLAVOR calcola la data del giorno che cade il numero specificato di giorni prima o dopo una determinata data, escludendo i fine settimana e le festività. @GIORNILAVNETTI calcola il numero di giorni che intercorrono tra due date, escludendo i fine settimana e le festività.

@FISSALUNGH

@FISSALUNGH(*stringa*; *lunghezza*; [*allineamento*]) produce un'etichetta della *lunghezza* indicata. L'etichetta consiste in una *stringa* e tanti spazi vuoti quanti sono sufficienti per posizionare la *stringa* in base all'*allineamento*.

Argomenti

stringa può essere una qualsiasi stringa di testo.

lunghezza può consistere in un qualsiasi numero intero compreso tra 1 e 512. Se *lunghezza* è inferiore al numero di caratteri compresi nella *stringa*, @FISSALUNGH produce la *stringa*.

allineamento è un argomento facoltativo che indica come deve essere allineata la *stringa*. Questo argomento può assumere i valori indicati nella tabella che segue:

<i>allineamento</i>	1-2-3 allinea la <i>stringa</i>
0	a sinistra degli spazi vuoti determinati dalla <i>lunghezza</i> ; impostazione predefinita in caso di omissione dell'argomento
1	al centro degli spazi vuoti determinati dalla <i>lunghezza</i> ; se il numero di spazi vuoti è dispari, 1-2-3 aggiunge lo spazio rimasto singolo a sinistra della <i>stringa</i>
2	a destra degli spazi vuoti determinati dalla <i>lunghezza</i>

La maggior parte dei caratteri usati in Windows sono di tipo proporzionale. Nei caratteri a spaziatura proporzionale, gli spazi vuoti occupano uno spazio inferiore a quello dedicato alle lettere.

Esempi

Negli esempi che seguono, ogni • rappresenta uno spazio vuoto.

@FISSALUNGH("Element Nine, Inc.";24) = Element Nine, Inc.●●●●●

@FISSALUNGH("Element Nine, Inc.";24;1) = ●●●Element Nine, Inc.●●●

@FISSALUNGH("Element Nine, Inc.";24;2) = ●●●●●Element Nine, Inc.

Funzioni @ affini

@TRONCA elimina gli spazi precedenti, seguenti e consecutivi da una stringa di testo.

@SOMMANEGATIVI, @SOMMAPOSITIVI

@SOMMANEGATIVI(*lista*) esegue la somma dei soli valori negativi contenuti in *lista*.

@SOMMAPOSITIVI(*lista*) esegue la somma dei soli valori positivi contenuti in *lista*.

Argomenti

lista può contenere i seguenti elementi in qualunque combinazione: numeri, formule numeriche e indirizzi o nomi di zone contenenti numeri o formule numeriche. Tali elementi devono essere divisi tra loro dagli appositi separatori di argomenti.

Esempi

@SOMMANEGATIVI(-2;21;5;12;-2;-7) = -11

@SOMMAPOSITIVI(-2;21;5;12;-2;-7) = 38

Funzioni @ affini

@SOMMA somma i valori di una lista. **@SOMMADB** somma i valori contenuti in una tabella di database che soddisfano le condizioni specificate. **@TOTPARZ** somma i valori contenuti in una lista e indica a **@TOTGENER** quali valori dovrà sommare.

Esempio: @IND

La tabella denominata AUMENTO (A3..E3) riporta gli aumenti salariali concessi in base al rendimento dei dipendenti.

@IND(AUMENTO;2;3), immessa in una cella avente il formato Percentuale, genera il valore 5%, ossia la percentuale di aumento salariale concessa a un impiegato con un rendimento pari a 3 e un livello retributivo pari a 2.

@IND(AUMENTO;1;2), immessa in una cella avente il formato Percentuale, genera il valore 7%, ossia l'aumento salariale di un dipendente con un rendimento pari a 2 e un livello retributivo pari a 1.

```
A ----- A ----- B -- C -- D -- E ----
1
2
3          Valutazione          -- LIVELLO RETRIBUTIVO --
4          1          10% 9%  8%  7%
5          2          7%  6%  5%  4%
6          3          6%  5%  4%  3%
7          4          3%  2%  1%  0%
8          5          0%  0%  0%  0%
```

@GAMMA

@GAMMA(x) calcola la funzione gamma.

Argomenti

x può assumere qualsiasi valore eccetto lo zero e i numeri interi negativi.

NOTA

La funzione @GAMMA calcola la distribuzione gamma, approssimandola fino alla sesta cifra significativa.

Esempi

@GAMMA(0,5) = 1,772454

@GAMMA(5) = 24

Funzioni @ affini

@BETA calcola la funzione beta. @GAMMAI calcola la funzione gamma incompleta. @GAMMALOGN calcola il logaritmo naturale della funzione gamma.

@GAMMAI

@GAMMAI(*a*; *x*; [*complemento*]) calcola la funzione gamma incompleta.

Argomenti

a deve essere un valore positivo.

x può essere un valore positivo oppure zero.

complemento è un argomento facoltativo che specifica come 1-2-3 deve calcolare la funzione @GAMMAI.

complemento	1-2-3 calcola
0	P(<i>a</i> , <i>x</i>); impostazione predefinita in caso di omissione dell'argomento
1	Q(<i>a</i> , <i>x</i>), che equivale a 1-P(<i>a</i> , <i>x</i>)

NOTA

La funzione @GAMMAI calcola la funzione gamma incompleta, approssimandola fino alla sesta cifra significativa.

Esempi

@GAMMAI(7,5;12,497;1) = 0,050024

Funzioni @ affini

@GAMMA calcola la funzione di distribuzione gamma. @GAMMALOGN calcola il logaritmo naturale della funzione gamma.

@GAMMALOGN

@GAMMALOGN(x) calcola il logaritmo naturale della funzione gamma.

Argomenti

x può assumere qualsiasi valore maggiore di zero.

NOTA

La funzione @GAMMALOGN calcola il logaritmo naturale della funzione gamma, approssimandolo fino alla sesta cifra significativa.

Esempi

@GAMMALOGN(0,5) = 0,572365

@GAMMALOGN(5) = 3,178054

Funzioni @ affini

@GAMMA calcola la funzione di distribuzione gamma. @GAMMAI calcola la funzione gamma incompleta.

@MEDIAGEOM

@MEDIAGEOM(*lista*) calcola la media geometrica dei valori compresi in una *lista*.

Argomenti

lista può essere composta da qualsiasi combinazione dei seguenti elementi: numeri, formule numeriche che generano numeri e indirizzi o nomi di zona contenenti numeri o formule numeriche. Gli elementi della *lista* devono essere divisi tra loro dagli appositi separatori di argomenti.

Tutti i valori della *lista* devono essere maggiori di zero.

Vedere anche Argomenti delle funzioni @ statistiche.

NOTE

La media geometrica di una *lista* di n valori è uguale alla n -esima radice del prodotto dei valori stessi.

A parità di *lista*, il risultato di @MEDIAGEOM è inferiore a quello di @MEDIA a meno che tutti i valori della *lista* siano uguali; in questo caso, @MEDIAGEOM e @MEDIA producono lo stesso risultato.

Esempi

@MEDIAGEOM(A1..A4) = 239,1886, dove A1..A4 contiene i valori 160, 227, 397 e 227

Funzioni @ affini

@MEDIAARMON calcola la media armonica dei valori compresi in una lista. @MEDIA e @MEDIAPURA calcolano la media aritmetica dei valori compresi in una lista. @MEDIANA calcola il valore mediano di una lista di valori.

@MEDIAARMON

@MEDIAARMON(*lista*) calcola la media armonica dei valori compresi in una *lista*.

Argomenti

lista può essere composta da qualsiasi combinazione dei seguenti elementi: numeri, formule numeriche che generano numeri, e indirizzi o nomi di zona contenenti numeri o formule numeriche. Gli elementi della *lista* devono essere divisi tra loro dagli appositi separatori di argomenti.

Tutti i valori della *lista* devono essere maggiori di zero.

Vedere anche Argomenti delle funzioni @ statistiche.

NOTE

La media armonica dei valori compresi in *lista* è il reciproco della media aritmetica dei reciproci dei valori della *lista* stessa. Per esempio, la media di 1/2, 1/3 e 1/4 è 13/36; pertanto, la media armonica di 2, 3 e 4 è 36/13.

A parità di *lista*, il risultato di **@MEDIAARMON** è sempre inferiore a quello di **@MEDIAGEOM**.

Esempi

@MEDIAARMON(25;50;75) = 40,90909

Funzioni @ affini

@MEDIAGEOM calcola la media geometrica dei valori compresi in una lista. **@MEDIA** e **@MEDIAPURA** calcolano la media aritmetica dei valori compresi in una lista. **@MEDIANA** calcola il valore mediano di una lista di valori.

@ESA

@ESA(x) converte un numero decimale nell'equivalente esadecimale.

Argomenti

x deve essere un numero intero compreso tra -2.147.483.648 e 2.147.483.647. Se x non è un numero intero, 1-2-3 lo tronca per renderlo tale.

NOTE

I valori esadecimali compresi tra 00000000 e 7FFFFFFF corrispondono allo 0 e a valori decimali positivi.

I valori esadecimali compresi tra 80000000 e FFFFFFFF corrispondono a valori decimali negativi.

Esempi

@ESA(162) = A2

Funzioni @ affini

@DECIMALE converte i numeri esadecimali in decimali.

@SCANSOR

@SCANSOR(*x*;*zona*;*scarto_riga*) riporta il contenuto di una cella situata in una riga di una tabella di consultazione orizzontale, ossia una zona la cui prima riga contiene informazioni numeriche in ordine crescente oppure etichette.

Argomenti

x può essere un valore o una stringa, a seconda del contenuto della prima riga della tabella di consultazione orizzontale.

Prima riga	<i>x</i>
Valori	<i>x</i> può essere un qualsiasi valore maggiore o uguale al primo valore di <i>zona</i> . Se <i>x</i> è minore del primo valore specificato in <i>zona</i> , @SCANSOR genera ERR. Se invece <i>x</i> è maggiore dell'ultimo valore situato nella prima riga di <i>zona</i> , @SCANSOR si ferma all'ultima cella della riga specificata da <i>scarto_riga</i> , riportandone il contenuto come risposta.
Etichette	Stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta. Se <i>x</i> non corrisponde esattamente al contenuto di una cella della prima riga di <i>zona</i> , @SCANSOR genera ERR.

zona rappresenta la posizione occupata dalla tabella di consultazione orizzontale, e può essere un indirizzo o un nome di zona. Se la *zona* è tridimensionale, 1-2-3 usa solamente il primo foglio di lavoro della *zona*.

scarto_riga rappresenta lo scarto corrispondente alla posizione occupata dalla riga nella *zona*.

NOTA

@SCANSOR confronta *x* con ogni cella della prima riga di una tabella. Quando 1-2-3 individua una cella nella prima riga contenente *x* (oppure, se *x* è un valore, il valore più vicino ma non superiore a *x*), scende lungo la colonna per il numero di righe specificato da *scarto_riga*, riportandone il contenuto.

Esempi

@HLOOKUP

Funzioni @ affini

@SCANSVER ricerca il contenuto di una cella in una tabella di consultazione verticale. @IND ricerca il contenuto di una cella quando vengono specificati gli scarti sia della riga che della colonna. @SCELTA sostituisce una tabella di consultazione che occupa solo una riga. CORRISP riporta la posizione relativa di una cella con il contenuto specificato. INDX riporta il contenuto di una cella specificata dai titoli di colonna, riga e foglio di lavoro. @SCANSMAX produce un riferimento assoluto alla cella che contiene il valore più alto di una serie di zone. @SCANSMIN genera un riferimento assoluto alla cella che contiene il valore più basso di una serie di zone.

@ORA

@ORA(*valore_orario*) estrae l'ora, ossia un valore compreso tra 0 (mezzanotte) e 23 (23:00 o 11:00 PM), da un *valore_orario*.

Argomenti

valore_orario può essere un valore compreso tra 0,000000 (mezzanotte) e 0,999988 (11:59:59 PM). Solitamente il *valore_orario* viene fornito da un'altra funzione @ di ora.

NOTA

La parte relativa all'ora è utile nei calcoli basati su ore intere -- ad esempio, il computo del compenso orario o delle ore trascorse dal momento in cui si è iniziato a lavorare su un determinato progetto, o a stampare un foglio di lavoro.

Esempi

@ORA(0,51565) = 12, poiché 0,51565 equivale al valore orario per 12:22:32 PM.

@ORA(**@ORARIO**(13;45;18)) = 13 (1:00 PM), poiché 13 è l'argomento *ora* della funzione **@ORARIO**(13;45;18).

Funzioni @ affini

@MINUTI estrae i minuti, mentre **@SECONDI** estrae i secondi da un valore orario.

@SE

@SE(*condizione*;x;y) valuta la *condizione*, riportando uno di due valori a seconda dell'esito della valutazione. Se la *condizione* è vera, **@SE** genera x; se la *condizione* è falsa, genera y.

Argomenti

condizione è solitamente una formula logica, oppure l'indirizzo o il nome di una cella contenente una formula logica. Tuttavia, come *condizione* si può usare qualsiasi formula, numero, stringa di testo racchiusa fra virgolette (" "), oppure il nome o l'indirizzo di una cella. 1-2-3 elabora come vera qualsiasi *condizione* diversa da zero e come falsa qualsiasi *condizione* che corrisponde a zero. Se usate come *condizione*, le celle vuote, le stringhe di testo, i valori **ERR** e **ND** equivalgono tutti a zero.

x e y possono essere valori, stringhe di testo racchiuse fra virgolette (" "), oppure indirizzi o nomi di celle contenenti valori o etichette.

NOTE

@SE viene usata in combinazione con **@ERR** e **@ND** per documentare errori o dati mancanti nelle formule. Inoltre, può servire a evitare ERR, ND o un errore di calcolo qualora manchino dei dati o non siano precisi -- ad esempio, per evitare le divisioni per zero.

Le funzioni **@SE** possono essere nidificate allo scopo di creare una condizione complessa. Ad esempio:

@SE(TOT>10000000;TOT*0,15,**@SE**(TOT>5000000;TOT*0,10;TOT*0,02))

contiene due funzioni **@SE** nidificate che servono a determinare il tasso di provvigione in base a tre livelli di vendite: maggiore di 10.000.000, maggiore di 5.000.000 e uguale o minore di 5.000.000.

Esempi

@SE(SALDO>=0;SALDO;"Scoperto") riporta il valore della cella SALDO qualora sia pari a 0 o positivo, oppure riporta l'etichetta Scoperto qualora il valore in SALDO sia negativo.

@IND

@IND(*zona*; *colonna*; *riga*; [*foglio*]) riporta il contenuto della cella situata nel punto di intersezione della *colonna*, *riga* e *foglio* (facoltativo) di una zona.

Argomenti

zona può essere un indirizzo o un nome di zona.

colonna è lo scarto della colonna utilizzata da **@IND**.

riga può essere lo scarto della riga utilizzata da **@IND**, oppure l'indirizzo o il nome di una cella contenente il valore 0 o un numero intero positivo.

foglio è un argomento facoltativo ed è lo scarto del foglio utilizzato da **@IND**. Se non si specifica l'argomento *foglio*, **@IND** usa solo il primo foglio della *zona*.

Esempi

@IND

Funzioni @ affini

@SCANSOR e **@SCANSVER** individuano i dati immessi in tabelle di consultazione orizzontali o verticali. **@SCELTA** individua un'immissione in una lista. **CORRISP** riporta la posizione relativa di una cella con il contenuto specificato.

INDX riporta il contenuto di una cella specificata dai titoli di colonna, riga e foglio. **@SCANSMAX** produce un riferimento assoluto alla cella che contiene il valore più alto di una serie di zone. **@SCANSMIN** genera un riferimento assoluto alla cella che contiene il valore inferiore di una serie di zone.

@INFO

@INFO(*attributo*) riporta informazioni riguardanti la sessione di 1-2-3 in corso.

Argomenti

attributo può essere uno degli elementi elencati nella tabella seguente, da immettere sotto forma di stringhe di testo.

attributo	risultato
codritornodb	ND
msgdriverdb	ND
contrecorddb	ND
macro_scansione	sì, se è attivo il modo Scansione; No, in caso contrario
macro_traccia	sì, se è aperta la finestra Traccia macro; No, in caso contrario
memdisp	quantità di memoria disponibile
modo	modo attivo: 0 ATTENDERE 1 PRONTO 2 ETICHETTA 3 MENU 4 VALORE 5 PUNTA 6 MODIFICA 7 ERRORE 8 CERCA 9 ARCHIVI 10 GUIDA 11 PARAMETRI 13 NOMI 99 Tutti gli altri modi (ad esempio, i modi definiti dall'utente con {INDICA})
numarch	numero di cartelle di lavoro aperte
origine	indirizzo assoluto della prima cella a sinistra del foglio attivo
codritornosist	ND
versionesist	ND
ricalc	modo di ricalcolo attivo, espresso sotto forma di una delle due stringhe "automatico" o "manuale"
versione	numero della versione di 1-2-3 per Windows
altezza_videata	altezza della videata (in pixel)
larghezza_videata	larghezza della videata (in pixel)
selezione	indirizzo della zona attualmente selezionata oppure nome del grafico, oggetto disegnato o tabella di interrogazione attualmente selezionato
porzione_selezione	ND
tipo_selezione	tipo della selezione attiva: Zona, Disegno, Interrogazione o Grafico

sistema	nome del sistema operativo
memtot	memoria totale disponibile (sia la quantità disponibile che la quantità in uso)
dirwin	percorso della directory che contiene Windows, comprensivo della lettera dell'unità
autore	il nome dell'utente che per primo ha salvato la cartella di lavoro in uso
data_creazione	il <u>valore data</u> corrispondente alla data in cui la cartella di lavoro attiva è stata salvata per la prima volta
tempo_modifica	il <u>valore orario</u> corrispondente al numero totale di ore e minuti durante i quali è rimasta aperta la cartella di lavoro in uso
directory	il <u>percorso</u> attivo, comprensivo della lettera dell'unità
ultima_revisione_di	il nome dell'utente che ha salvato la cartella di lavoro in uso per ultimo
data_ultima_revisione	il valore data corrispondente alla data in cui la cartella di lavoro corrente è stata salvata l'ultima volta
numero_fogli	il numero dei fogli contenuti nella cartella di lavoro in uso
dimensioni_cartella	le dimensioni in kilobyte (Kb) della cartella di lavoro in uso

NOTE

Oltre agli attributi elencati qui sopra, *attributo* può essere anche un qualsiasi Elemento di sessione.

La funzione @INFO è utile nelle macro quando sono necessarie informazioni relative allo stato di 1-2-3 (ad esempio, per indicare all'utente il percorso usato in una macro che automatizza la memorizzazione delle cartelle di lavoro, oppure per avvertirlo della scarsa disponibilità di memoria).

Può essere usata assieme a @SE per controllare lo stato di 1-2-3 e per indicare alla macro cosa fare in determinate situazioni; ad esempio, se cambiare percorso, cancellare dati superflui o chiudere le cartelle di lavoro aperte se scarseggia la memoria disponibile.

Esempi

@INFO("numarch") = 2, se sono aperte due cartelle di lavoro

@INFO(B4) = 3, se B4 contiene l'etichetta modo e 1-2-3 si trova in modo MENU

Funzioni @ affini

@CELLA fornisce informazioni sulla prima cella di una zona. @CURSORE fornisce informazioni sulla cella attiva.

@INT

@INT(x) riporta la parte intera di x.

Argomenti

x è un valore.

NOTA

Per visualizzare solo i valori che hanno un dato numero di cifre decimali e ottenere allo stesso tempo la massima precisione di calcolo, usare il formato Fisso. Non usare @INT.

Esempi

@INT(35;67) = 35

@INT(@ADESSO) = valore data corrispondente alla data e all'ora correnti, poiché la parte dell'ora è un valore decimale.

Funzioni @ affini

@ARROTONDA, @ARROTONDAECESSO, e @ARROTONDADIFETTO arrotondano un valore al più vicino multiplo della potenza di 10 specificata. @TRONCADEC tronca un valore alla cifra decimale indicata.

@INTRATA, @CAPITRATA

@INTRATA(*capitale;interesse;durata;data_inizio;[data_fine];[tipo];[valore_futuro]*) calcola la porzione costituita dagli interessi cumulativi della rata periodica da pagare per un prestito (*capitale*), dato il tasso di *interesse* per un certo numero di periodi di pagamento (*durata*).

@CAPITRATA(*capitale;interesse;durata;data_inizio;[data_fine];[tipo];[valore_futuro]*) calcola la porzione costituita dal capitale della rata periodica da pagare per un prestito (*capitale*), dato il tasso di *interesse* per un certo numero di periodi di pagamento (*durata*).

Argomenti

capitale e *durata* sono valori. *durata* può assumere qualsiasi valore eccetto lo zero.

interesse deve essere un numero decimale o una percentuale maggiore di -1.

data_inizio è la data, all'interno della durata del prestito, da cui si vuole partire per il calcolo dell'interesse o del capitale. Questo argomento può assumere valori maggiori o uguali a zero, ma minori di *durata*.

data_fine è la data, all'interno della durata del prestito, in cui deve terminare il calcolo dell'interesse o del capitale. Questo argomento può assumere qualsiasi valore maggiore di *data_inizio*. In caso di omissione, *data_fine* è uguale a *data_inizio*.

tipo è un argomento facoltativo che specifica se il calcolo deve essere eseguito per una rendita immediata o anticipata.

<i>tipo</i>	1-2-3 esegue il calcolo per una rendita
0	immediata (pagamenti da effettuare alla fine di un periodo); impostazione predefinita in caso di omissione dell'argomento
1	anticipata (pagamenti da effettuare all'inizio di un periodo)

valore_futuro è un argomento facoltativo che specifica il valore futuro di una serie di rate. Questo argomento può assumere qualsiasi valore. In caso di omissione, 1-2-3 usa un *valore_futuro* uguale a zero.

Non è consentito l'uso di un argomento facoltativo senza quelli che lo precedono.

NOTA

Il periodo usato per il calcolo dell'*interesse* deve essere uguale a quello di *durata*. Per esempio, supponendo di calcolare una rata mensile, si dovranno immettere gli interessi e la durata in incrementi mensili. Solitamente ciò significa che occorre dividere il tasso d'interesse per 12 e moltiplicare il numero di anni della *durata* per 12.

Esempi

È stato concesso un prestito di L. 8.000.000 per tre anni a un tasso d'interesse annuo del 10,5%, capitalizzato mensilmente. Le rate mensili ammontano a L. 260.000. Per stabilire la porzione costituita dagli interessi delle rate dell'ultimo anno, si ricorre alla seguente funzione:

@INTRATA(8000000;0,105/12;36;25;36) = L. 170.448

Per determinare la porzione costituita dal capitale delle rate dell'ultimo anno, si dovrebbe invece usare la seguente funzione:

@CAPITRATA(8000000;0,105/12;36;25;36) = L. 2.949.786

Funzioni @ affini

@PAGAM calcola la rata da pagare per un prestito.

@TASSOINT

@TASSOINT(*durata*; *rata*; *valore_attuale*; [*tipo*]; [*valore_futuro*]; [*stima*]) calcola il tasso di interesse periodico necessario affinché un investimento (*valore_attuale*) raggiunga il *valore_futuro* nel corso del periodo di capitalizzazione indicato da *durata*.

Argomenti

durata deve essere un numero intero positivo.

rata e *valore_attuale* possono assumere qualsiasi valore.

tipo è un argomento facoltativo che specifica se il calcolo deve essere eseguito per una rendita immediata o anticipata.

<i>tipo</i>	1-2-3 esegue il calcolo per una rendita
0	immediata (pagamenti da effettuare alla fine di un periodo); impostazione predefinita in caso di omissione dell'argomento
1	anticipata (pagamenti da effettuare all'inizio di un periodo)

valore_futuro è un argomento facoltativo che specifica il valore futuro di una serie di rate. Questo argomento può assumere qualsiasi valore. In caso di omissione, 1-2-3 usa un *valore_futuro* uguale a zero.

stima è un argomento facoltativo che esprime il tasso d'interesse previsto. Può essere un valore compreso tra 0 e 1. In caso di omissione, 1-2-3 usa una *stima* pari a 0,10 (10%).

Non è consentito l'uso di un argomento facoltativo senza quelli che lo precedono.

NOTE

Per il calcolo del tasso d'interesse, @TASSOINT si serve di una serie di approssimazioni a partire da quella del valore di *stima*. Le soluzioni di questa funzione sono molteplici: al primo tentativo conviene usare una *stima* realistica del tasso d'interesse; qualora il risultato sia zero o un valore minore di 1, è bene tentare di attribuire a *stima* un valore diverso.

Se @TASSOINT non riesce a eseguire il calcolo con un'approssimazione dello 0,0000001 dopo trenta iterazioni, produrrà il risultato ERR. Se, anche cambiando il valore di *stima*, la funzione continua a generare ERR, conviene ricorrere a @VAN per calcolare una stima più consona. Se la funzione @VAN dà valore positivo, è segno che la stima utilizzata è troppo bassa; al contrario, se il risultato di @VAN è un valore negativo, vuol dire che è troppo alta. Se la stima è corretta, @VAN genera il valore zero.

Il periodo usato per il calcolo della *stima* deve essere uguale a quello della *durata*. Per esempio, supponendo di calcolare una rata mensile, si dovranno immettere gli interessi e la durata in incrementi mensili. Solitamente ciò significa che occorre dividere il tasso d'interesse per 12 e moltiplicare il numero di anni della *durata* per 12.

Esempi

Sono stati depositati L. 6.000.000 in un conto da cui si vogliono prelevare L. 100.000 al mese per otto anni. Per stabilire il tasso d'interesse da ottenere allo scopo di effettuare tali prelievi, si può usare la seguente funzione:

@TASSOINT(96,100;6000;0;0;0,01) = 0,010623, ossia 1,06% al mese

Funzioni @ affini

@VAN calcola il valore attuale netto di una serie di flussi di cassa futuri. @VALATT e @VALATTUALE calcolano il valore attuale di una rendita sulla base di una serie di versamenti di uguale importo. @VALFUT e @VALFUTURO calcolano il valore futuro di una rendita. @TASSO calcola il tasso d'interesse periodico necessario affinché un investimento raggiunga un dato valore futuro.

@AMMORT

@AMMORT(*stima*;zona) calcola il tasso di rendimento interno (profitto) per una serie di movimenti di cassa generati da un investimento. Il tasso di rendimento interno è il tasso percentuale che rende il valore attuale di una serie di movimenti di cassa futuri uguale all'investimento iniziale.

Argomenti

stima può essere un valore decimale o percentuale, oppure l'indirizzo o il nome di una cella contenente un valore decimale o percentuale che rappresenta il tasso di rendimento interno ipotizzato. Nella maggior parte dei casi, *stima* dovrebbe essere un valore percentuale compreso tra 0 (0%) e 1 (100%). In presenza di movimenti di cassa molto consistenti, occorre rendere il più accurato possibile l'argomento *stima*.

zona può essere l'indirizzo o il nome di una zona contenente i movimenti di cassa. 1-2-3 considera i numeri negativi come uscite e i numeri positivi come entrate di cassa. Solitamente, il primo movimento di cassa di una zona è un numero negativo (uscita) che rappresenta l'investimento iniziale. 1-2-3 assegna il valore 0 a tutte le celle vuote e a tutte le etichette situate in *zona*, includendole nei calcoli.

NOTE

La funzione **@AMMORT** può essere utilizzata per determinare la redditività di un investimento. Per valutare un investimento, si può abbinare **@AMMORT** ad altre funzioni finanziarie quali **@VAN**.

1-2-3 parte dal presupposto che i movimenti di cassa avvengano a intervalli equidistanti.

@AMMORT esegue una serie di approssimazioni a partire dal valore di *stima* per calcolare il tasso di rendimento interno. È pertanto necessario partire con una *stima* ragionevole del tasso di rendimento interno. Inoltre, dal momento che sono possibili più soluzioni, se il risultato è minore di 0 o maggiore di 1 si può provare a immettere un'altra *stima*; se, dopo 30 iterazioni, **@AMMORT** non riesce ad approssimare il risultato fino a raggiungere 0,0000001, la formula produrrà il risultato ERR. Se i valori stimati continuano a generare il valore ERR, utilizzare la funzione **@VAN** per determinare una stima più precisa. Se **@VAN** genera un valore negativo, significa che la stima è troppo alta; se invece genera il valore 0, vuol dire che la stima è corretta. Per determinare il tasso di rendimento interno in presenza di tassi diversi, si consiglia di usare la funzione **@MEDIA**.

Esempi

Il piano seguente calcola il tasso di rendimento interno di un investimento iniziale di L. 10.000.000, seguito da 12 versamenti mensili di 1.500.000. *stima* (12%) si trova nella cella STIMA, mentre i versamenti sono elencati nella zona denominata MOVIM_CASSA. La funzione:

@AMMORT(STIMA;MOVIM_CASSA)

genera un tasso di rendimento interno pari a 10,45%.

Funzioni @ affini

@VAN calcola il valore attuale netto di una serie di movimenti di cassa futuri. **@VALATT** calcola il valore attuale di una rendita sulla base di una serie di versamenti di uguale importo. **@VALFUT** calcola il valore futuro di una rendita. **@TASSO** riporta il tasso d'interesse periodico necessario affinché un investimento raggiunga un determinato valore futuro.

@AMMORTMODIF calcola tasso di rendimento interno modificato. **@XIRR** calcola il tasso di rendimento interno per una serie di movimenti di cassa in entrata e in uscita.

@DFCAGG, @DAGG, @DMACRO

@DFCAGG(*nome*) controlla il *nome* per verificare se si tratta di una funzione LotusScript globale aggiuntiva definita. Se *nome* è una funzione di script aggiuntiva definita, **@DFCAGG** genera il valore 1 (vero); in caso contrario, **@DFCAGG** genera il valore 0 (falso).

@DAGG(*nome*) controlla il *nome* per verificare se si tratta di un programma aggiuntivo caricato in memoria. In caso affermativo, **@DAGG** genera il valore 1 (vero); in caso contrario, **@DAGG** genera il valore 0 (falso).

@DMACRO(*nome*) controlla il *nome* per verificare se si tratta di una subroutine LotusScript globale aggiuntiva definita. In caso affermativo, la funzione genera il valore 1 (vero); in caso contrario, genera il valore 0 (falso).

Argomenti

nome è il nome del file della cartella di lavoro del programma aggiuntivo (a esclusione dell'estensione .12A) o della funzione o della subroutine da controllare e viene immesso sotto forma di stringa di testo.

NOTA

@DAGG genera 1 (vero) solo per i programmi aggiuntivi caricati con File - Programmi aggiuntivi - Gestione, anche se questi contengono solo funzioni @.

@DFCAGG e **@DMACRO** generano il valore vero se si specifica una funzione usandone solo il nome e tale funzione si trova nella cartella di lavoro in uso o in un programma aggiuntivo caricato. Per verificare la presenza di una funzione in un'altra cartella di lavoro, specificare il file e il nome della funzione nel seguente modo:
"<<miofile>>miafunzione".

1-2-3 riconosce funzioni e subroutine solo se i loro argomenti e i tipi di ritorno a capo sono quelli ammessi. Per i numeri a virgola mobile, 1-2-3 non riconosce il tipo Single: tutte le funzioni e subroutine devono usare il tipo Double.

Le zone sono argomenti ammessi, ma devono essere specificati come argomenti Variant. I nomi di oggetti di 1-2-3 sono tipi di argomenti non ammessi.

Esempi

@DFCAGG("gradi") = 1, se **@GRADI** è una funzione script aggiuntiva definita.

@DFCAGG("sommadb") = 0, perché **@SOMMADB** è una funzione @ di 1-2-3 e non una funzione di script aggiuntiva.

@DAGG("bilancio") = 1, se è caricato in memoria il programma aggiuntivo **BILANCIO**.

@DMACRO("ruolopaga") = 1, se {**RUOLOPAGA**} è una subroutine di script aggiuntiva definita.

@DERR

@DERR(x) verifica se x ha valore ERR. Se x corrisponde a ERR, @DERR genera 1 (vero); in caso contrario, @DERR genera 0 (falso).

Argomenti

x può essere qualsiasi valore, posizione, stringa di testo o condizione.

NOTA

La funzione @DERR può essere utilizzata per bloccare errori derivanti dall'uso di 0 come divisore. Ad esempio, la formula @SE(@DERR(A1/A2);0;A1/A2) verifica il risultato della divisione A1/A2 (ossia il contenuto della cella A1 diviso per il contenuto della cella A2): se il risultato è il valore ERR, la formula produrrà il valore 0; se invece il risultato corrisponde a un qualsiasi altro valore, sarà quello a venire fornito come risultato.

Esempi

Il sottoprogramma CTRLQTA' è costituito da tre brevi sottoprogrammi che verificano i dati immessi nelle celle denominate QTA' e PREZZO. CTRLQTA' verifica se l'immissione in QTA' è un valore; in caso affermativo, l'elaborazione passa al sottoprogramma CTRERRND; se invece QTA' non contiene un valore, NUOVAQTA' richiede una nuova immissione e trasferisce il controllo a CTRLQTA'. CTRERRND usa @DERR per determinare se QTA' contiene il valore ERR; se @DERR produce il risultato 1 (vero), viene richiesto un nuovo valore; se invece QTA' non contiene ERR e PREZZO non contiene ND, il sottoprogramma moltiplica i valori delle due celle, immettendo il risultato nella cella denominata TOTALE.

...

```
CTRLQTA'      {SE @DNUMERO(QTA')}{SALTO CTRERRND}
NUOVAQTA'     {ACCETTAN "Valore quantità: ";QTA'}
              {SALTO CTRLQTA'}
CTRERRND      {SE @DERR(QTA')}{SALTO NUOVAQTA'}
              {SE @DND(PREZZO)}{ACCETTAN "Nuovo prezzo: ";PREZZO}{SALTO
CTRLQTA'}
              {POSIZIONA}TOTALE~+QTA*Prezzo~
```

...

Funzioni @ affini

@DND verifica la presenza del valore ND.

@DND

@DND(x) verifica se x ha il valore ND. Se x corrisponde a ND, @DND genera 1 (vero); in caso contrario, @DND genera 0 (falso).

Argomenti

x può essere qualsiasi valore, posizione, stringa di testo o condizione.

Esempi

Il sottoprogramma CTRLQTA' è costituito da tre brevi sottoprogrammi che verificano i dati immessi nelle celle denominate QTA' e PREZZO. CTRLQTA' verifica se l'immissione in QTA' è un valore; in caso affermativo, l'elaborazione passa al sottoprogramma CTRERRND; se invece QTA' non contiene un valore, NUOVAQTA' richiede una nuova immissione e trasferisce il controllo a CTRLQTA'. CTRERRND usa @DND per determinare se PREZZO contiene il valore ND; se @DND genera il valore 1 (vero), viene richiesta l'immissione di un nuovo valore. Se PREZZO non contiene ND e QTA' non contiene ERR, il sottoprogramma moltiplica i valori delle due celle, immettendo il risultato nella cella denominata TOTALE.

...

```
CTRLQTA'      {SE @DNUMERO(QTA')}{SALTO CTRERRND}
NUOVAQTA'     {ACCETTAN "Valore quantità: ";QTA'}
              {SALTO CTRLQTA'}
CTRERRND      {SE @DERR(QTA')}{SALTO NUOVAQTA'}
              {SE @DND(PREZZO)}{ACCETTAN "Nuovo prezzo: ";PREZZO}{SALTO
              CTRLQTA'}
              {POSIZIONA}TOTALE~+QTA*Prezzo~
```

...

Funzioni @ affini

@DERR verifica la presenza del valore ERR.

@DNUMERO

@DNUMERO(x) verifica se x contiene un valore. Se x è un valore, ND, ERR o una cella vuota, @DNUMERO genera 1 (vero); se invece x è una stringa di testo o una cella contenente un'etichetta o una formula che genera un'etichetta, @DNUMERO produce il valore 0 (falso).

Argomenti

x può essere qualsiasi valore, posizione, stringa di testo o condizione. Se x è una zona costituita da più celle, @DNUMERO genera 0 (falso) anche se la prima cella della zona contiene un valore.

Esempi

Il sottoprogramma CTRLQTA' è costituito da tre brevi sottoprogrammi che verificano le immissioni delle celle denominate QTA' e PREZZO. CTRLQTA' usa la funzione @DNUMERO per determinare se l'immissione di QTA' è un valore; in caso affermativo, l'elaborazione viene trasferita al sottoprogramma CTRERRND. Se QTA' non contiene un valore, il sottoprogramma NUOVAQTA' richiede una nuova immissione.

...

```
CTRLQTA'      {SE @DNUMERO(QTA')}{SALTO CTRERRND}
NUOVAQTA'     {ACCETTAN "Valore quantità: ";QTA'}
               {SALTO CTRLQTA'}
CTRERRND      {SE @DERR(QTA')}{SALTO NUOVAQTA'}
               {SE @DND(PREZZO)}{ACCETTAN "Nuovo prezzo: ";PREZZO}{SALTO
               CTRLQTA'}
               {POSIZIONA}TOTALE~+QTA*Prezzo~
```

...

Funzioni @ affini

@DSTRINGA verifica la presenza di un'etichetta. Per stabilire se una cella contiene un valore o un'etichetta, si possono usare anche le funzioni @CELLA e @CURSORE.

@DZONA

@DZONA(*zona*) verifica se *zona* è un nome di zona definito oppure un indirizzo di zona valido (ossia un indirizzo di zona costituito dalle lettere di foglio e colonna comprese tra A e IV, e dai numeri di riga compresi tra 1 e 8192). Se *zona* è un nome di zona definito o un indirizzo di zona valido, @DZONA genera 1 (vero); se *zona* non è un nome di zona definito né un indirizzo di zona valido, @DZONA genera 0 (falso).

Argomenti

zona può essere qualsiasi stringa di testo o indirizzo di zona.

NOTE

La funzione @DZONA è utile in combinazione con @SE per determinare se l'immissione è un nome di zona valido per le chiamate di sottoprogrammi e per l'esecuzione di salti con {INDIRIZZO}.

La funzione @DZONA può essere utilizzata solo nelle cartelle di lavoro caricate in memoria.

Esempi

@DZONA(A1) = 1 (vero).

@DZONA(A1..C3) = 1(vero).

@DZONA(VENDITE) = 1 (vero), se VENDITE è un nome di zona definito.

@DZONA(PREZZO) = 0 (falso), se PREZZO è un nome di zona indefinito.

@DZONA(3) = 0 (falso).

@DZONA("PROVVIGIONE") = 0 (falso), perché l'argomento è racchiuso fra virgolette (" ").

@DSTRINGA

@DSTRINGA(x) valuta x per verificare se è una stringa di testo o un'etichetta. Se x è una stringa di testo, una cella contenente un'etichetta o una formula che genera un'etichetta, @DSTRINGA genera il valore 1 (vero); se x è un valore, ERR, ND o una cella vuota, @DSTRINGA genera 0 (falso).

Argomenti

x può essere qualsiasi valore, posizione, stringa di testo o condizione.

Esempi

Il sottoprogramma CTRLSTR verifica il contenuto della cella denominata CLIENTE. Se CLIENTE contiene un'etichetta (@DSTRINGA(CLIENTE) = 1), il sottoprogramma salta al nuovo sottoprogramma denominato ORDINE. Se CLIENTE non contiene un'etichetta, il sottoprogramma richiede una nuova immissione.

...

```
CTRLSTR      {SE @DSTRINGA(CLIENTE)}{SALTO ORDINE}  
             {ACCETTAE "Nome CLIENTE: ";CLIENTE}  
             {CTRLSTR}
```

...

Funzioni @ affini

@DNUMERO verifica la presenza di un valore. Per determinare se una cella contiene un'etichetta, si può usare anche @CELLA e @CURSORE.

@CURTOSI

@CURTOSI(*zona*;*[tipo]*) calcola la curtosi dei valori compresi nella *zona*.

Argomenti

zona è il nome o l'indirizzo di una zona contenente valori. Se *zona* contiene meno di quattro valori, @CURTOSI genera il valore ERR.

tipo è un argomento facoltativo che specifica se il calcolo deve essere effettuato su una popolazione o su un campione.

<i>tipo</i>	1-2-3 calcola
0	la curtosi della popolazione; impostazione predefinita in caso di omissione dell'argomento
1	la curtosi campionaria

NOTA

La curtosi misura la concentrazione di una distribuzione intorno alla media. Una curtosi positiva indica che la curva di distribuzione è piuttosto ripida, mentre una curtosi negativa è segno di una curva di distribuzione relativamente piatta.

Esempi

La zona DATI contiene i seguenti valori: 2, 5, 5, 9, 1, 2, 4.

@CURTOSI(DATI;1) = 1,021488

@CURTOSI(DATI) = -0,32438

Funzioni @ affini

@ASIMMETRIA calcola l'asimmetria dei valori di una zona.

@GRANDE

@GRANDE(*zona*; *n*) individua l'*n*-esimo valore più alto della *zona*.

Argomenti

zona è il nome o l'indirizzo di una zona contenente valori.

n è un numero intero positivo. Se *n* è maggiore del numero di valori contenuti in *zona*, @GRANDE produce il valore ERR.

Esempi

Una zona denominata PUNTEGGIO contiene i seguenti valori corrispondenti agli esiti di un test: 87, 85, 90, 80, 82, 92, 79, 85, 95, 86.

@GRANDE(PUNTEGGIO;3) produce 90, ossia il terzo punteggio più alto

Funzioni @ affini

@PICCOLO individua l'*n*-esimo valore più basso di una zona. @MAX e @MAXPURO produce il valore più elevato di una zona. @MIN e @MINPURO trova il valore inferiore di una zona.

@SINISTRA

@SINISTRA(*stringa*; *n*) fornisce i primi *n* caratteri di *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

n può essere un numero intero positivo o 0. Se *n* è 0, il risultato è una stringa vuota. Se *n* è maggiore della lunghezza della *stringa*, **@SINISTRA** riporta l'intera stringa.

NOTA

@SINISTRA conta come caratteri anche i segni di interpunzione e gli spazi.

La funzione **@SINISTRA** è utile quando si vuole copiare solo una parte di un'etichetta in un'altra cella a partire dall'inizio dell'etichetta (ad esempio, per separare i titoli come Dott. e Sig. dai relativi nomi).

In una macro, **@SINISTRA** può estrarre parti di etichette immesse dall'utente per memorizzarle in un database, per usarle nelle chiamate di sottoprogrammi o per modificare la macro stessa.

La funzione **@SINISTRA** va usata assieme a **@RICER** quando non si conosce l'esatto valore di *n* oppure quando *n* può variare.

Esempi

@SINISTRA(TEL;3) = prefisso del numero telefonico situato nella cella TEL

@SINISTRA(A1;**@RICER**("*";A1;0)) = primo nome della cella A1 (ad esempio, Riccardo, se la cella A1 contiene il nome Riccardo Salvini.). L'asterisco (*) rappresenta uno spazio.

Funzioni @ affini

@MEZZO estrae i caratteri dall'interno di *stringa*. **@DESTRA** riporta gli ultimi *n* caratteri di *stringa*.

@LUNGH

@LUNGH(*stringa*) conta i caratteri in *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

NOTA

@LUNGH conta come caratteri anche i segni di interpunzione e gli spazi.

La funzione @LUNGH può essere usata assieme a @TRONCA per determinare la lunghezza di *stringa* senza contare eventuali spazi iniziali, finali o consecutivi.

@LUNGH è anche utile nelle applicazioni in cui le etichette devono avere una determinata lunghezza, come i codici di avviamento postale e i codici degli ordini di acquisto.

Esempi

@LUNGH("frigorifero") = 11

@LUNGH(A5&G12) = numero totale dei caratteri situati nelle celle A5 e G12.

@LUNGH(@TRONCA(" Sig. Bernardi")) = 13

@LOGN

@LOGN(x) calcola il logaritmo naturale (in base e) di x.

Argomenti

x può essere un valore oppure l'indirizzo o il nome di una cella contenente un valore maggiore di 0.

NOTE

Un logaritmo si dice naturale quando usa come base il valore e (circa 2,718282).

Esempi

@LOGN(2) = 0,693147

@LOGN(@ESP(1)) = 1, perché @ESP(1) = 2,718282

Funzioni @ affini

@ESP è la funzione inversa di @LOGN. @LOG calcola il logaritmo decimale (in base 10) di x.

@LOG

@LOG(x) calcola il logaritmo decimale (in base 10) di x.

Argomenti

x può essere un valore oppure l'indirizzo o il nome di una cella contenente un valore maggiore di 0.

Esempi

$10^{(@LOG(8)/3)} = 2$, cioè la radice cubica di 8

@LOG(B3) = 0,60206, se la cella B3 contiene il valore 4

Funzioni @ affini

@LOGN calcola il logaritmo naturale (in base e) di un valore.

@MINUSC

@MINUSC(*stringa*) converte in minuscole tutte le lettere di *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

NOTA

Se si usa un tipo di ordinamento, impostato con File - Preferenze - 1-2-3, sotto "Ordinamento per lingua" nella linguetta Generali, l'alternanza di maiuscole e minuscole va a incidere sull'ordinamento delle etichette; due etichette identiche potrebbero quindi risultare diverse qualora avessero una diversa combinazione di maiuscole e minuscole.

Esempi

@MINUSC("LOTUS") = lotus

Funzioni @ affini

@MAIUSC converte in maiuscole tutte le lettere di una stringa. @NOMEPRO converte in maiuscola la prima lettera di ogni parola di una stringa e le rimanenti in minuscole.

@MAX, @MAXPURO

@MAX(*lista*) determina il valore più alto nella *lista*.

@MAXPURO(*lista*) determina il valore più alto nella *lista*, ignorando le celle contenenti etichette.

Argomenti

lista può comprendere una delle seguenti combinazioni di elementi: numeri, formule numeriche e indirizzi o nomi di zone contenenti numeri o formule numeriche. Separare i vari elementi di *lista* con gli appositi separatori di argomenti.

Vedere anche Argomenti delle funzioni @ statistiche.

Esempi

Una zona denominata TEST contiene i seguenti dati: -5, -7, -9, -11, Gennaio.

@MAX(TEST) genera 0, ossia il valore dell'etichetta Gennaio, il più alto nella zona TEST

@MAXPURO(TEST) ignora l'etichetta Gennaio e segnala -5 come valore più alto della zona TEST

Funzioni @ affini

@MIN e @MINPURO individuano il valore più basso in una lista. @MAXDB segnala il valore più alto contenuto nel campo di una tabella di database che soddisfa le condizioni specificate. @GRANDE produce l'*n*-esimo valore più alto di una lista.

@MEDIANA

@MEDIANA(*lista*) calcola il valore mediano di una *lista*.

Argomenti

lista può contenere una qualsiasi combinazione dei seguenti elementi: numeri, formule numeriche e indirizzi o nomi di zona contenenti numeri o formule numeriche. Gli elementi della *lista* vanno divisi tra loro dagli appositi separatori di argomenti.

Vedere anche Argomenti delle funzioni @ statistiche.

NOTE

Se la *lista* contiene un numero dispari di valori, @MEDIANA produce il valore mediano. Se invece il numero di valori è pari, la funzione produce la media aritmetica dei due valori mediani.

Esempi

@MEDIANA(5;12;65;82;9) = 12

@MEDIANA(5;12;65;82;9;78) = 38,50

Funzioni @ affini

@PUREMEDIAN calcola il valore mediano di una *lista*, ignorando le celle vuote, le etichette e le formule che producono etichette.

@MEDIAGEOM calcola la media geometrica dei valori compresi in una lista. @MEDIAARMON calcola la media armonica dei valori compresi in una lista. @MEDIA e @MEDIAPURA calcolano la media dei valori di una lista.

@MEZZO

@MEZZO(*stringa*;numero_inizio;*n*) riporta *n* caratteri di *stringa* a partire dal carattere situato nella posizione specificata da *numero_inizio*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

numero_inizio può essere uno scarto, oppure l'indirizzo o il nome di una cella contenente un numero intero positivo o uno 0. Se *numero_inizio* è maggiore della lunghezza di *stringa*, **@MEZZO** genera una stringa vuota.

n può essere qualsiasi numero intero positivo o uno 0, oppure l'indirizzo o il nome di una cella contenente un numero intero positivo o uno 0. Se *n* è 0, **@MEZZO** genera una stringa vuota; se *n* è maggiore della lunghezza di *stringa*, 1-2-3 riporta tutti i caratteri a partire da *numero_inizio* sino alla fine di *stringa*.

NOTA

@MEZZO conta come caratteri anche i segni di interpunzione e gli spazi.

Se non si conosce la lunghezza di *stringa*, si consiglia di usare un numero alto per *n*; 1-2-3 ignora gli spazi in eccesso, riportando l'intera *stringa* a partire da *numero_inizio*.

La funzione **@MEZZO** viene usata per estrarre una parte di un'etichetta. Per estrarre parte di un'etichetta di cui non si conosce il *numero_inizio*, si deve usare **@MEZZO** in combinazione con @RICER.

@MEZZO può essere usato nelle macro per memorizzare parti di etichette immesse dall'utente allo scopo di creare chiamate di sottoprogrammi o di modificare la macro stessa.

Esempi

@MEZZO("Ognuno sta solo sul cuor della terra trafitto da un raggio di sole: ed è subito sera";37;29) = trafitto da un raggio di sole.

Funzioni @ affini

@SINISTRA riporta i primi *n* caratteri di *stringa*, mentre @DESTRA riporta gli ultimi *n* caratteri di *stringa*.

@MIN, @MINPURO

@MIN(*lista*) determina il valore più basso in una *lista*.

@MINPURO(*lista*) determina il valore più basso in una *lista*, ignorando tutte le celle contenenti etichette.

Argomenti

lista può contenere una combinazione qualunque dei seguenti elementi: numeri, formule numeriche e indirizzi o nomi di zone contenenti numeri o formule numeriche. Gli elementi della *lista* vanno divisi fra loro con gli appositi separatori di argomenti.

Vedere anche Argomenti delle funzioni @ statistiche.

Esempi

Una zona denominata TEST contiene le seguenti immissioni: 5, 7, 9, 11, Gennaio.

@MIN(TEST) produce 0, ossia il valore dell'etichetta Gennaio, il valore più basso nella zona TEST.

@MINPURO(TEST) ignora l'etichetta Gennaio e segnala 5 come valore più basso di TEST.

Funzioni @ affini

@MAX e @MAXPURO individuano il valore più alto di una *lista*. @MINDB segnala il valore più basso contenuto nel campo di una tabella di database che soddisfa le condizioni specificate.

@MINUTI

@MINUTI(*valore_orario*) estrae i minuti, ossia un valore compreso tra 0 e 59, da *valore_orario*.

Argomenti

valore_orario può essere un valore, oppure l'indirizzo o il nome di una cella contenente un valore compreso tra 0,000000 (mezzanotte) e 0,999988 (11:59:59 PM). Solitamente l'argomento *valore_orario* viene fornito da un'altra funzione @ di ora.

NOTE

La parte relativa ai minuti è utile nei calcoli che riguardano solo i minuti, ad esempio il tempo trascorso dall'avvio di un'applicazione.

Esempi

@MINUTI(0,333) = 59, perché 0,333 è il valore orario per le 7:59:31

@MINUTI(**@ORARIO**(11;15;45)) = 15, perché 15 è l'argomento *minuti* della funzione **@ORARIO**(11;15;45)

Funzioni @ affini

@ORA estrae l'ora, mentre **@SECONDI** estrae i secondi da un valore orario.

@AMMORTMODIF

@AMMORTMODIF(*zona*; *tasso_finanziario*; *tasso_reinvestimento*; [*tipo*]) calcola il tasso di rendimento interno modificato (profitto) per una serie di movimenti di cassa generati da un investimento.

Il tasso di rendimento interno è il tasso percentuale che rende il valore attuale di una serie prevista di futuri movimenti di cassa uguale all'investimento iniziale.

Argomenti

zona è l'indirizzo o il nome di una zona contenente i movimenti di cassa. 1-2-3 considera i numeri negativi come uscite e i numeri positivi come entrate di cassa. La *zona* deve contenere almeno un valore positivo e uno negativo.

Solitamente, il primo movimento di cassa nella *zona* è un numero negativo (uscita) che rappresenta l'investimento iniziale. 1-2-3 assegna il valore 0 a tutte le celle vuote e a tutte le etichette situate in *zona*, includendole nei calcoli.

tasso_finanziario è il tasso d'interesse pagato sulle somme di denaro utilizzate per i movimenti di cassa.

tasso_reinvestimento è il tasso d'interesse che si riceve sui movimenti di cassa man mano che vengono reinvestiti.

tasso_finanziario e *tasso_reinvestimento* possono assumere qualsiasi valore.

tipo specifica il momento in cui vengono effettuati i movimenti di cassa. *tipo* è un numero intero, il cui significato è illustrato dalla seguente tabella.

<i>tipo</i>	i movimenti di cassa avvengono
0	all'inizio di ogni periodo; impostazione predefinita in caso di omissione dell'argomento
1	alla fine di ogni periodo

NOTE

La funzione @AMMORTMODIF consente di determinare la redditività di un investimento. Per valutare un investimento, si può abbinare @AMMORT ad altre funzioni finanziarie quali @VAN.

1-2-3 parte dal presupposto che i movimenti di cassa avvengano a intervalli equidistanti.

Si noti che l'argomento facoltativo *tipo* della funzione @AMMORTMODIF è supportato solo dalla versione 5 e successive: pertanto, qualora si salvi una formula che utilizzi la funzione @AMMORTMODIF corredata dell'argomento *tipo* e si apra la cartella di lavoro in una versione precedente, tale formula produrrà il valore ERR.

Esempi

Si supponga di possedere un condominio. Cinque e sei anni fa sono stati presi in prestito L. 100.000.000 al tasso d'interesse annuo del 9,5% per acquistare lo stabile. La lista che segue, memorizzata nella zona REDDITO, contiene gli importi degli investimenti iniziali e il successivo reddito di locazione:

L. -100.000.000

L. -100.000.000

L. 45.500.000

L. 47.000.000

L. 48.500.000

L. 50.000.000

L. 50.000.000

Nel corso di questi anni, i profitti reinvestiti hanno reso l'11% annuo.

@AMMORTMODIF(REDDITO;0,095;0,11) = 7,95%

Il tasso di rendimento diventa leggermente inferiore se gli investimenti vengono realizzati alla fine dell'anno:

@AMMORTMODIF(REDDITO;0,095;0,11;1) = 7,70%

Funzioni @ affini

@AMMORT calcola il tasso di rendimento interno.

@XIRR calcola il tasso di rendimento interno per una serie di movimenti di cassa in entrata e in uscita.

@MOD, @MODULO

@MOD(x;y) e @MODULO(x;y) calcolano il resto (modulo) di x/y.

Argomenti

x è un valore. Se x è 0, @MOD e @MODULO generano 0.

y è un valore diverso da zero.

NOTE

Il risultato di @MOD è $x - y * @INT(x/y)$. Il segno del risultato (+ o -) è sempre uguale al segno di x.

Il risultato di @MODULO è $x - y * @ARROTDIFETTO(x/y)$. Il segno del risultato (+ o -) è sempre uguale al segno di x.

Esempi

@MOD(9;4) = 1

@MODULO(9;4) = 1

@MOD(-14;3) = -2

@MODULO(-14;3) = 1

Funzioni @ affini

@QUOZIENTE calcola il risultato di x/y, troncandolo per renderlo intero.

@MESE

@MESE(*valore_data*) estrae il mese, ossia un valore compreso tra 1 e 12, da *valore_data*.

Argomenti

valore_data può essere un valore, oppure l'indirizzo o il nome di una cella contenente un valore compreso tra 1 (1 gennaio 1900) e 73050 (31 dicembre 2099).

NOTE

Per fornire il valore dell'argomento *valore_data*, si può usare anche un'altra funzione @ di data.

Esempi

@MESE(**@DATA**(91;3;27)) = 3, perché 3 è l'argomento *mese* della funzione **@DATA**(91;3;27).

@MESE(20181) = 4, perché il valore data 20181 equivale alla data 02-Apr-55.

@MESE(**@ADESSO**) = mese in corso.

Funzioni @ affini

@GIORNO calcola il giorno usando un valore data. **@ANNO** calcola l'anno usando un valore data.

Esempio: @SCANSOR

La tabella di consultazione orizzontale denominata COSTI (A2..E7) elenca i costi per spedire uno stesso pacco in città diverse. La funzione:

@SCANSOR("Francoforte";COSTI;3)

immessa in una cella avente il formato Valuta con 0 cifre decimali, genera il valore L. 24.000 che corrisponde al costo della spedizione di un pacco di tipo 3 a Francoforte.

A	----	A	-----	B	-----	C	-----	D	-----	E	-----
1						DESTINAZIONE	PACCO				
2	Tipo di pacco	Londra		Parigi		Francoforte		New York			
3	1	L. 18.360		L. 19.300		L. 20.100		L. 90.200			
4	2	L. 20.300		L. 21.600		L. 22.300		L. 110.200			
5	3	L. 22.400		L. 23.800		L. 24.000		L. 113.250			
6	4	L. 24.100		L. 25.200		L. 25.750		L. 116.850			
7	5	L. 28.320		L. 29.000		L. 29.800		L. 119.540			

@N

@N(*zona*) riporta il valore contenuto nella prima cella di *zona*, oppure il valore 0 se la cella contiene un'etichetta.

Argomenti

zona può essere una cella oppure un indirizzo o un nome di zona.

NOTE

La funzione @N può essere usata in combinazione con qualsiasi funzione @ o formula nel caso in cui l'immissione debba essere un valore ma vi sia la possibilità che la cella contenga un'etichetta. @N può essere quindi utilizzata per evitare che le formule generino un valore ERR.

@N è inoltre utile nelle macro per verificare le immissioni dell'utente.

Esempi

+100+@N(B5..F5) = 885, se la cella B5 contiene il valore 785.

@N(A5)+@N(B5) genera 785 se A5 contiene un'etichetta e B5 contiene il valore 785.

Funzioni @ affini

@S riporta l'etichetta contenuta nella prima cella di una zona. @DNUMERO può determinare se una cella contiene un valore.

@ND

@ND riporta il valore ND (non disponibile).

NOTE

La funzione @ND può essere usata durante la creazione di un foglio di lavoro per contrassegnare le celle in cui saranno immessi dati di tipo da determinarsi. Le formule che fanno riferimento a tali celle produrranno il risultato ND fino a quando non verranno immessi i dati appropriati.

@ND può anche essere usata per determinare quali formule dipendono da una data cella.

Esempi

@SE(@CELLA("tipo";B14)="v";@ND;B14) genera il valore ND se la cella B14 è vuota.

Funzioni @ affini

@ERR genera il valore ERR. @DND verifica la presenza del valore ND.

@ADESSO

@ADESSO calcola il valore che corrisponde alla data e all'ora correnti in base all'orologio del computer. Tale valore comprende sia un valore data (parte intera) che un valore orario (parte decimale).

NOTE

@ADESSO può essere usata con F2 (MODIFICA) e con F9 (CALCOLA) allo scopo di creare una registrazione permanente della data e dell'ora -- ad esempio, per stampare fogli di lavoro a una determinata ora oppure per calcolare il tempo trascorso da un dato evento.

Al valore generato da @ADESSO può essere assegnato uno qualsiasi dei formati Ora o Data. Se si assegna un formato Data alla funzione @ADESSO, 1-2-3 visualizza soltanto la parte del valore corrispondente alla data (parte intera). Se invece viene assegnato un formato Ora, 1-2-3 visualizza solo la parte corrispondente al valore dell'ora (parte decimale). In entrambi i casi, 1-2-3 memorizza ed esegue i calcoli con l'intero valore temporale.

1-2-3 aggiorna la funzione @ADESSO ogni volta che si esegue un ricalcolo.

Esempi

@ADESSO = 31050,5 a mezzogiorno del 3 gennaio 1985.

@ADESSO = 33418,395 alle 9:28 di mattina del 19 giugno 1991.

Funzioni @ affini

@OGGI calcola il valore data corrispondente alla data corrente.

@VAN

@VAN(*interesse*; *zona*; [*tipo*]) calcola il valore attuale netto di una serie di movimenti di cassa futuri (*zona*), scontati a un tasso d'*interesse* periodico costante.

Argomenti

interesse può essere un valore decimale o percentuale, oppure l'indirizzo o il nome di una cella contenente un valore decimale o percentuale.

zona è la zona che contiene i movimenti di cassa.

tipo specifica il momento in cui vengono effettuati i movimenti di cassa. *tipo* è un numero intero, il cui significato è illustrato dalla seguente tabella.

tipo	i movimenti di cassa avvengono
0	all'inizio di ogni periodo; impostazione predefinita in caso di omissione dell'argomento
1	alla fine di ogni periodo

NOTE

La funzione @VAN viene utilizzata per valutare un investimento o per confrontarlo con altri investimenti. @VAN calcola l'investimento iniziale necessario per realizzare una determinata uscita di cassa a un dato tasso.

@VAN genera ERR se *zona* contiene più di una riga o più di una colonna. Ad esempio, @VAN genera ERR se *zona* è A1..D25, mentre non produce il valore ERR se *zona* è A1..D1 (ossia una sola riga) oppure A1..A25 (una sola colonna).

Si noti che l'argomento facoltativo *tipo* della funzione @VAN è supportato solo dalla versione 5 e successive: pertanto, qualora si salvi una formula che utilizzi la funzione @VAN corredata dell'argomento *tipo* e si apra la cartella di lavoro che la contiene con una versione precedente di 1-2-3, tale formula produrrà il valore ERR.

Esempi

In questo esempio la funzione @VAN viene usata per scontare al loro valore attuale in lire una serie di versamenti irregolari, investiti a un tasso d'interesse annuo del 11,5%.

zona è costituita da una lista di movimenti di cassa con cadenza mensile per 12 mesi, situati in una zona denominata VERSAMENTI:

L.	0
L.	0
L.	2.500.000
L.	2.500.000
L.	3.000.000
L.	5.000.000
L.	6.000.000
L.	9.000.000
L.	3.000.000
L.	2.500.000
L.	0
L.	7.500.000
L.	41.000.000 TOTALE

Per garantire che a @VAN venga fornito l'esatto numero di periodi, occorre includere nella *zona* i mesi nei quali non è stato effettuato alcun versamento. Poiché i versamenti hanno cadenza mensile, @VAN richiede che l'*interesse* (tasso di sconto), situato in una cella denominata SCONTO, venga espresso sotto forma di percentuale mensile:

$$0,115/12 = 0,96$$

$$@VAN(SCONTO;VERSAMENTI) = L. 38.084,128$$

Il risultato cambia se le uscite di cassa avvengono all'inizio di ogni periodo.

$$@VAN(SCONTO;VERSAMENTI;1) = \$38.449,10$$

Funzioni @ affini

@VALATT calcola il valore attuale di una rendita costituita da una serie di versamenti di uguale importo. @VALFUT calcola il valore futuro di una rendita.

@XNPV calcola il valore attuale netto di una serie di flussi di cassa in uscita e in entrata.

@DISPARI

@DISPARI(x) arrotonda il valore x al successivo numero dispari intero.

Argomenti

x può assumere qualsiasi valore. Se è un numero intero dispari, @DISPARI genera x .

Esempi

@DISPARI(3,25) = 5

@DISPARI(3) = 3

@DISPARI(-3,25) = -5

Funzioni @ affini

@PARI arrotonda un valore al successivo numero pari intero. @ARROT, @ARROTDIFETTO e @ARROTECESSO arrotondano un valore a un numero avente le cifre decimali specificate. @ARROTM arrotonda un valore al multiplo specificato. @INT tronca un valore, scartando la parte decimale. @TRONCADEC tronca un valore alla cifra decimale specificata.

@PERCENTILE

@PERCENTILE(*x*; *zona*) calcola l'*x*-esimo percentile campionario tra i valori di una *zona*.

Argomenti

x è il percentile da individuare; deve essere un numero compreso tra 0 e 1.

zona è il nome o l'indirizzo della zona contenente i valori.

1-2-3 assegna il valore 0 a tutte le etichette comprese nella *zona*, includendole nel calcolo di **@PERCENTILE**, mentre ignora tutte le celle vuote presenti nella *zona*.

Esempi

Una zona denominata PUNTEGGIO contiene i seguenti esiti di un test: 87, 85, 90, 80, 82, 92, 79, 85, 95, 86. Si desidera trovare il punteggio corrispondente al 90° percentile:

@PERCENTILE(0,9;PUNTEGGIO) = 92,3

Funzioni @ affini

@LIVELLOPERC individua il percentile di un valore all'interno di una zona.

@DECILE calcola l'*x*-esimo decile e **@QUARTILE** calcola un dato quartile.

@PERMUT

@PERMUT($n;r$) calcola il numero di sequenze ordinate (permutazioni) di r oggetti selezionabili dall'insieme totale di n oggetti.

Argomenti

n può essere qualsiasi numero intero positivo o zero.

r può essere qualsiasi numero intero positivo o zero, purché maggiore di n .

Se n e r non sono numeri interi, 1-2-3 li tronca per renderli tali.

Esempi

I test programmati per le ore 9:00, 10:00 e 11:00 del mattino saranno monitorati da tre membri del reparto su cinque. La formula sottostante calcola tutte le diverse possibilità di assegnazione dei compiti:

@PERMUT(5;3) = 60

Funzioni @ affini

@COMBIN calcola tutte le possibili estrazioni di r da n , senza considerazioni di ordine.

@PI

@PI riporta il valore π (calcolato a 3,14159265358979). Il valore π è il rapporto tra la circonferenza e il diametro di un cerchio.

Esempi

@PI = 3,1415926536

@PI*4^2 = 50,26548, ossia l'area di una circonferenza con raggio 4.

@PAGAM, @PAGAMENTO, @PAGAMC

@PAGAM(*capitale;interesse;durata*) calcola le rate per il rimborso di un prestito (*capitale*) a un dato tasso d'*interesse* per un numero specificato di periodi di pagamento (*durata*).

@PAGAMENTO(*capitale;interesse;durata;[tipo];[valore_futuro]*) calcola le rate per il rimborso di un prestito (*capitale*) a un dato tasso d'*interesse* per un numero specificato di periodi di pagamento (*durata*). @PAGAMENTO esegue i calcoli per una rendita immediata o anticipata, a seconda del valore specificato per *tipo*.

@PAGAMC(*capitale;interesse;durata*) è un caso particolare della funzione @PAGAM per le convenzioni canadesi dei vincoli ipotecari.

Argomenti

capitale e *durata* possono assumere qualsiasi valore.

interesse può essere un numero decimale o una percentuale maggiore di -1.

tipo è un argomento facoltativo che specifica se il calcolo deve essere eseguito per una rendita immediata o anticipata.

<i>tipo</i>	1-2-3 esegue il calcolo per una rendita
0	immediata (pagamenti da effettuare alla fine del periodo); impostazione predefinita in caso di omissione dell'argomento
1	anticipata (pagamenti da effettuare all'inizio del periodo)

valore_futuro è un argomento facoltativo che specifica il valore futuro di una serie di rate; può assumere qualsiasi valore. In caso di omissione, 1-2-3 usa un *valore_futuro* uguale a zero.

Non è consentito l'uso di un argomento facoltativo senza quelli che lo precedono.

NOTA

Per @PAGAM e @PAGAMENTO, il periodo usato per il calcolo dell'*interesse* deve essere uguale a quello di *durata*. Per esempio, supponendo di calcolare una rata mensile, si dovranno immettere gli interessi e la durata in incrementi mensili. Solitamente ciò significa che occorre dividere il tasso d'interesse per 12 e moltiplicare per 12 il numero di anni della *durata*.

Per @PAGAMC, il periodo usato per calcolare l'*interesse* è costituito da anni, mentre quello per la *durata* è costituito da mesi.

Esempi

È stato concesso un prestito di L. 8.000.000 per tre anni a un tasso d'interesse annuo del 10,5%, capitalizzato mensilmente. Le rate mensili vanno versate l'ultimo giorno di ogni mese. La seguente funzione consente di calcolare l'ammontare della rata mensile:

@PAGAM(8000000;0,105/12;36) = L. 260.019

Se i pagamenti fossero dovuti al primo giorno del mese, si userebbe la seguente funzione:

@PAGAMENTO(8000000;0,105/12;36;1;-2500000) = L. 198.898

Per calcolare le rate mensili con @PAGAMC, si userebbe la seguente funzione:

@PAGAMC(8000000;0,105;36) = L. 259.180,7

Funzioni @ affini

@PAGAM2 calcola le rate di rimborso di un prestito a un dato tasso di *interesse* per un numero specificato di periodi di pagamento, presupponendo una rendita anticipata. @IPAGAM calcola la porzione costituita dagli interessi di una rata costante. @ISP calcola la porzione costituita dagli interessi di una rata, per cui la parte di capitale rimane invariata in ogni periodo.

@INTRATA calcola la porzione costituita dagli interessi cumulativi della rata periodica da pagare per un investimento.

@CAPITRATA calcola la porzione costituita dal capitale della rata periodica da pagare per un investimento. @TERM calcola il numero di versamenti da effettuare per un investimento.

@POISSON

@POISSON(x ; $media_arrivi$; [$cumulativa$]) calcola la distribuzione di Poisson.

Argomenti

x è il numero di "arrivi" (osservazioni di un certo fenomeno) e può essere un numero intero positivo oppure 0.

$media_arrivi$ è il numero di ripetizioni del fenomeno attese e può essere un numero intero positivo oppure 0.

Se x e $media_arrivi$ non sono numeri interi, 1-2-3 li tronca per renderli tali.

$cumulativa$ è un argomento facoltativo che specifica come 1-2-3 deve calcolare la funzione @POISSON.

<i>cumulativa</i>	1-2-3 calcola
0	la probabilità che si verifichino esattamente x eventi; impostazione predefinita in caso di omissione dell'argomento
1	la probabilità che si verifichino al massimo x eventi

NOTE

@POISSON calcola la distribuzione di Poisson con un'approssimazione di $\pm 3 \cdot 10^{-7}$.

@POISSON può essere utile per prevedere il numero di eventi che si verificheranno in un certo periodo di tempo; per esempio, il numero di visitatori che varcheranno i cancelli di un luna park nell'arco di un'ora.

Esempi

Si prevede che da un casello autostradale passino sei automobili all'ora. La funzione sottostante consente di calcolare la probabilità che in un'ora passino al massimo quattro automobili:

@POISSON(4;6;1) = 0,285057, ossia il 28,51%

La funzione sottostante serve invece a calcolare la probabilità che in un'ora passino esattamente quattro automobili:

@POISSON(4;6) = 0,133853, ossia il 13,39%

@LIVELLOPERC

`@LIVELLOPERC(x;zona;[cifre_decimali])` individua il percentile di x tra i valori compresi in una *zona*.

Argomenti

x può assumere qualsiasi valore.

zona è il nome o l'indirizzo di una zona contenente valori.

cifre_decimali è un argomento facoltativo che specifica il numero di cifre decimali a cui deve essere arrotondato il risultato della funzione; può avere un valore compreso tra 0 e 100. In caso di omissione, 1-2-3 usa 2 cifre decimali.

NOTA

Se x non è un valore compreso nella *zona*, 1-2-3 assegna il livello di 0° percentile al valore più basso della *zona* e il livello di 100° percentile al valore più elevato, eseguendo un'interpolazione.

Esempi

Una zona denominata PUNTEGGIO contiene i seguenti esiti di test: 87, 85, 90, 80, 82, 92, 79, 85, 95, 86. Si vuole determinare il percentile del punteggio pari a 90:

`@LIVELLOPERC(90;PUNTEGGIO)` = 0,78 o 78%.

Funzioni @ affini

`@PERCENTILE` calcola il percentile campionario dei valori compresi in una lista.

@PREZZO

@PREZZO(*liquidazione*; *scadenza*; *cedola*; *rendimento*; [*rimborso*]; *frequenza*); [*base*]) calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a \$100.

Argomenti

liquidazione è il giorno in cui avviene il pagamento dei compensi e consiste in un valore data.

scadenza è il giorno in cui viene rimborsato un titolo e deve essere un valore data. Se *scadenza* è inferiore o uguale a *liquidazione*, la funzione @PREZZO genera ERR.

cedola è il tasso annuo nominale di un titolo e consiste in un valore positivo o uguale a zero.

rendimento rappresenta il rendimento annuale e può essere qualsiasi valore positivo.

rimborso è un argomento facoltativo che specifica il valore di rimborso di un titolo con valore nominale pari a \$100. Questo argomento deve essere costituito da un valore positivo o da zero. In caso di omissione, 1-2-3 usa un rimborso pari a 100.

frequenza è un argomento facoltativo che indica il numero di pagamenti delle cedole previsti in un anno. *frequenza* è espresso con un valore ricavato dalla tabella sottostante.

<i>frequenza</i>	frequenza dei pagamenti delle cedole
1	annuale
2	semestrale; impostazione predefinita in caso di omissione dell'argomento
4	trimestrale
12	mensile

base è un argomento facoltativo che indica il tipo di base per il conteggio dei giorni da usare; viene espresso con un valore ricavato dalla tabella sottostante.

<i>base</i>	base per il conteggio dei giorni
0	30/360; impostazione predefinita in caso di omissione dell'argomento
1	effettivo/effettivo
2	effettivo/360
3	effettivo/365
4	europeo 30/360

Non è consentito usare un argomento facoltativo senza specificare quelli che lo precedono.

Esempi

Un'obbligazione ha come data di liquidazione il 1° luglio 1993 e come data di scadenza il 1° dicembre 1998. Il tasso nominale d'interesse semestrale è pari al 5,50% e il rendimento annuale è del 5,61%. L'obbligazione ha una base per il conteggio dei giorni di 30/360.

Per calcolare il prezzo dell'obbligazione, si userebbe la seguente funzione:

@PREZZO(@DATA(93;7;1);@DATA(98;12;1);0,055;0,0561;100;2;0) = \$99,49

Funzioni @ affini

@MATURATO calcola gli interessi maturati sui titoli che prevedono pagamenti periodici degli interessi.

@RENDIMENTO calcola il rendimento dei titoli con pagamento periodico degli interessi. @DURATA calcola la durata annua e @DURATAM calcola la durata annua modificata di un titolo con pagamento periodico degli interessi.

@PREZZO2 calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a ¥100, in base alle convenzioni giapponesi.

@NOMEPRO

@NOMEPRO(*stringa*) converte in maiuscola la prima lettera di ogni parola della *stringa*, trasformando le restanti in minuscole.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

NOTA

La funzione **@NOMEPRO** è utile per combinare stringhe di provenienza diversa e per uniformare la struttura di maiuscole/minuscole delle etichette in tutta la cartella di lavoro. **@NOMEPRO** può essere usata anche in un database per garantire un uso uniforme delle maiuscole nei nomi prima di riordinare i nomi stessi o di utilizzarli per creare etichette di indirizzi.

Se si usa un tipo di ordinamento ASCII, impostato con File - Preferenze - 1-2-3 sotto "Ordinamento per paese" nella linguetta Generali, la presenza di maiuscole e minuscole va a incidere sull'ordinamento delle etichette; due etichette identiche potrebbero quindi risultare diverse qualora avessero una diversa combinazione di maiuscole e minuscole.

Esempi

@NOMEPRO(A7&";"&G7) riporta la stringa Michele Santini; Milano, Italia se la cella A7 contiene l'etichetta MICHELE SANTINI e la cella G7 contiene l'etichetta Milano, Italia. Va notato che il punto e virgola (;) è racchiuso fra virgolette in modo da essere considerato una stringa di testo anziché un separatore di argomenti.

Funzioni @ affini

@MINUSC converte in minuscole tutte le lettere di una stringa, mentre **@MAIUSC** converte in maiuscole tutte le lettere di una stringa.

@VALATT, @VALATTUALE

@VALATT(*versamenti;interesse;durata*) calcola il valore attuale di un investimento, sulla base di una serie di *versamenti* di uguale importo, attualizzati a un tasso d'*interesse* periodico per la *durata* specificata.

@VALATTUALE(*versamenti;interesse;durata;[tipo];[valore_futuro]*) determina il valore attuale di un investimento dal *valore_futuro* specificato, sulla base di una serie di *versamenti* uguali, attualizzati a un tasso d'*interesse* periodico per la *durata* specificata. @VALATTUALE esegue i calcoli per una rendita immediata o anticipata, a seconda del valore specificato per *tipo*.

Argomenti

versamenti e *durata* possono assumere qualsiasi valore.

interesse è un numero decimale o una percentuale maggiore di -1.

tipo è un argomento facoltativo che specifica se deve essere calcolata una rendita immediata o anticipata.

<i>tipo</i>	1-2-3 calcola una rendita
0	immediata (versamento da effettuare alla fine del periodo); impostazione predefinita in caso di omissione dell'argomento
1	anticipata (versamento da effettuare all'inizio del periodo)

valore_futuro è un argomento facoltativo che specifica il valore futuro di una serie di versamenti; può assumere qualsiasi valore. In caso di omissione, 1-2-3 usa un *valore_futuro* pari a zero.

Non è consentito usare un argomento facoltativo senza aver specificato quelli che lo precedono.

NOTE

Il periodo usato per calcolare l'*interesse* deve essere uguale a quello utilizzato per *durata*; per esempio, se si intende calcolare un versamento mensile, occorre immettere l'*interesse* e la *durata* in incrementi mensili. Solitamente, ciò significa che il tasso di *interesse* deve essere diviso per 12 e che il numero di anni di *durata* deve essere moltiplicato per 12.

La funzione @VALATT viene utilizzata per valutare un investimento o per confrontarlo con altri investimenti.

@VALATT è utile anche per confrontare tra loro diversi tipi di investimenti -- ad esempio per valutare se, giunti allo scadere di una polizza pensionistica, sia meglio riscuotere l'intero capitale maturato oppure optare per una rendita periodica. Inoltre, si può usare @VALATT assieme a @PAGAM per creare una tabella di ammortamento.

@VALATT è complementare alla funzione @PAGAM: @VALATT indica l'importo massimo che si può prendere a prestito, dato un tetto massimo per la rata mensile. Viceversa, @PAGAM indica l'importo delle rate mensili da versare, dato l'importo del prestito che si vuole richiedere.

Esempi

Si supponga di aver vinto un premio di L. 10.000.000 che può essere riscosso in 20 rate di L. 500.000 con scadenza a fine anno oppure in un'unica soluzione di L. 4.000.000. A questo punto si vuole scoprire quale delle due possibilità conviene di più.

Se viene accettata la rateizzazione annuale di L. 500.000, si prevede di investire questa somma a un *interesse* del 8% capitalizzato annualmente.

@VALATT(500000;0,08;20) produce il risultato di L. 4.909.070; ciò significa che l'importo di 10.000.000 riscosso nell'arco di 20 anni ha un valore attuale di 4.909.070 lire.

La funzione sottostante parte dal presupposto che le riscossioni avvengano all'inizio di ogni anno:

@VALATTUALE(500000;0,08;20;1) = L. 5.301.800

Funzioni @ affini

@VALFUT e @VALFUTURO calcolano il valore futuro di un investimento sulla base di una serie di versamenti di uguale importo. @VAN calcola il valore attuale netto di un investimento, attualizzando il valore futuro a quello presente. @PAGAM e @PAGAMENTO calcolano il rimborso di un prestito a un dato tasso d'*interesse* per un numero specificato di periodi di pagamento.

@VALFUT2 calcola il valore futuro di un investimento sulla base di una serie di *versamenti* di uguale importo, di un tasso di *interesse* periodico e del numero di periodi specificati da *durata*, presupponendo una rendita anticipata.

@PVAMOUNT genera il valore attuale di una somma complessiva dariceversi in un dato numero di periodi futuri e

attualizzata a un dato tasso di interesse.

@QUOZIENTE

@QUOZIENTE(x;y) calcola il risultato di x/y , troncandolo all'intero.

Argomenti

x può essere qualsiasi valore. Se è uguale a 0, @QUOZIENTE genera 0.
 y deve essere un valore diverso da 0.

Esempi

@QUOZIENTE(7;3) = 2

@QUOZIENTE(12,25;3,5) = 3

@QUOZIENTE(-7;3) = -2

Funzioni @ affini

@MOD calcola il resto (modulo) di x/y .

@RADGRADI

@RADGRADI(*radianti*) converte i radianti in gradi.

Argomenti

radianti deve essere un valore.

Esempi

@RADGRADI(0,523599) = 30 gradi

Funzioni @ affini

@GRADIRAD converte i gradi in radianti.

@CAS

@CAS riporta un valore casuale compreso tra 0 e 1. 1-2-3 calcola @CAS fino a 15 cifre decimali. Tutte le volte che viene ricalcolato il foglio di lavoro, @CAS riporta un diverso valore casuale.

NOTE

Per convertire in un valore fisso il valore generato da @CAS, premere F2 (MODIFICA) e poi F9 (CALCOLA).

Per generare valori casuali con diversi intervalli numerici, moltiplicare @CAS per l'intervallo stesso. Per generare valori casuali interi, applicare la funzione @ARROT o @INT al risultato.

Esempi

@CAS = 0,419501 oppure qualsiasi valore compreso tra 0 e 1.

@CAS*10 = 6,933674 oppure qualsiasi valore compreso tra 0 e 10.

@INT(@CAS*50)+1 = 49 oppure qualsiasi intero compreso tra 1 e 50.

@LIVELLO

@LIVELLO(*elemento*; *zona*; [*ordine*]) calcola la grandezza o il livello di un valore di una zona in relazione agli altri valori contenuti nella stessa zona.

Argomenti

elemento è il valore di cui si vuole determinare il livello.

zona è l'indirizzo o il nome di una zona contenente valori. La *zona* deve comprendere l'*elemento*.

ordine è un argomento facoltativo che specifica come stabilire il livello dell'*elemento*; viene derivato dalla tabella sottostante.

ordine	1-2-3 tratta i valori della zona come se fossero disposti in ordine
0	decrescente (da 9 a 1) prima di determinare il livello dell' <i>elemento</i> ; impostazione predefinita in caso di omissione dell'argomento
1	crescente (da 1 a 9) prima di determinare il livello dell' <i>elemento</i>

NOTA

A parità di numero, 1-2-3 assegna uguale livello nella *zona*. La presenza di più numeri uguali incide tuttavia sul livello dei successivi numeri nella *zona*. Per esempio, nella zona contenente i valori 2, 4, 6, 8, 8, 10, 12, il numero 8 appare due volte e ha livello 4 in base all'ordine crescente; il numero 10 ha livello 6, mentre non ci sono numeri con livello 5.

Esempi

La zona denominata VENDITE (A1..A5) contiene i seguenti valori espressi in migliaia:

L. 5.000

L. 4.900

L. 5.150

L. 4.800

L. 4.900

@LIVELLO(4900;VENDITE) = 3; L. 4.900 è il terzo valore più alto della zona VENDITE. Nessun valore della zona assume il livello 4.

@LIVELLO(4900;VENDITE;1) = 2; perché VENDITE ha un ordine crescente e L. 4.900 è il secondo più basso nella zona VENDITE. Nessun valore assume il livello 3.

@TASSO

@TASSO(*valore_futuro;valore_attuale;durata*) calcola il tasso d'interesse periodico necessario affinché un investimento (*valore_attuale*) raggiunga un *valore_futuro* nel corso dei periodi di capitalizzazione specificati in *durata*.

Argomenti

valore_futuro, *valore_attuale* e *durata* possono essere valori, oppure indirizzi o nomi di celle contenenti valori.

Esempi

Si supponga di aver investito l'importo di 10.000.000 di lire in BOT per un periodo di 5 anni, alla fine del quale si riscuoterà la somma di 18.000.000 di lire, e di voler determinare il tasso d'interesse periodico di tale investimento.

@TASSO(18000000;10000000;60) genera 9,84%, ossia il tasso d'interesse periodico (mensile). Per determinare il tasso d'interesse annuo, usare la formula $((1+\text{@TASSO}(18000000;10000000;60))^{12})-1$. Il risultato è un interesse annuo del 12,47%.

Funzioni @ affini

@IRATE calcola il tasso di interesse periodico necessario affinché un investimento raggiunga un valore futuro.

@CONVERTIREF

@CONVERTIREF(*referimento*) converte le lettere delle colonne o dei fogli di 1-2-3 (A - IV) in numeri (1 - 256) e viceversa.

Argomenti

referimento specifica una colonna o un foglio di 1-2-3; può essere una lettera da A a IV immessa sotto forma di stringa di testo oppure un numero intero compreso tra 1 e 256.

La funzione **@CONVERTIREF** non fa distinzione tra maiuscole e minuscole, perciò il *referimento* può essere immesso indifferentemente con le une o con le altre.

Esempi

@CONVERTIREF(10) = J

@CONVERTIREF("J") = 10

Funzioni @ affini

@COLONNE conta il numero di colonne comprese in una zona, mentre **@FOGLI** conta il numero di fogli contenuti in una zona. **@COORD** crea un indirizzo di cella usando i valori specificati.

@REGRESSIONE

@REGRESSIONE(*zona_x*; *zona_y*; *attributo*; [*calcolo*]) esegue una regressione lineare multipla, generando la statistica specificata.

Argomenti

zona_x contiene le variabili indipendenti. Questo argomento consiste nel nome o nell'indirizzo di una zona che può contenere fino a 75 colonne e 8192 righe.

zona_y contiene l'insieme di valori che determinano la variabile dipendente. *zona_y* è il nome o l'indirizzo di una zona formata da una sola colonna e da un numero di righe pari a quello della *zona_x*.

attributo specifica il tipo di valore che si intende calcolare con la regressione ed è un valore derivato dalla seguente tabella:

attributo	1-2-3 calcola
1	la costante
2	l'errore standard della stima Y
3	R quadrato
4	il numero di osservazioni
5	i gradi di libertà
da 101 a 175	il coefficiente X (curvatura) della variabile indipendente specificata
da 201 a 275	l'errore standard del coefficiente per la variabile indipendente specificata da <i>attributo</i>

Nel caso degli ultimi due attributi, 1-2-3 numera le variabili indipendenti contenute nella *zona_x* a partire da 1, procedendo dall'alto verso il basso e da sinistra verso destra.

Per esempio, se *zona_x* è B2..D7, l'attributo 201 individua l'errore standard del coefficiente per la variabile indipendente contenuta nella colonna B; l'attributo 102 individua il coefficiente X della variabile indipendente contenuta nella colonna C.

calcolo è un argomento facoltativo che specifica l'intercetta Y.

calcolo	1-2-3
0	usa 0 come intercetta Y
1	calcola l'intercetta Y; impostazione predefinita in caso di omissione dell'argomento

NOTA

A parità di dati, la funzione @REGRESSIONE e il comando Zona - Risolvi - Regressione producono lo stesso risultato.

Esempi

@REGRESSIONE

Funzioni @ affini

@FORECAST dà il valore previsto di x sulla base di una tendenza lineare tra i valori contenuti nella *zona_y* e nella *zona_x*.

@RIPETI

@RIPETI(*stringa*; *n*) duplica la *stringa* per il numero di volte specificato da *n*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

n può essere un numero intero positivo.

NOTA

@RIPETI duplica la stringa per il numero di volte specificato, a prescindere dalla larghezza della colonna attiva. Per questa particolarità, @RIPETI si differenzia dal prefisso di ripetizione etichetta (barra inversa) che invece si limita a ripetere l'etichetta fino ad aver riempito la colonna attiva.

Esempi

@RIPETI("Ciao ";3) genera la stringa Ciao Ciao Ciao.

@RIPETI("-";10) riporta una serie di dieci trattini (-----).

@SOSTIT

@SOSTIT(*stringa_originale*;numero_inizio;n;stringa_sostit) sostituisce *n* caratteri di *stringa_originale* con *stringa_sostit*, a partire da *numero_inizio*.

Argomenti

stringa_originale e *stringa_sostit* possono essere stringhe di testo racchiuse fra virgolette (" "), formule che generano stringhe di testo, oppure gli indirizzi o i nomi di celle contenenti etichette o formule che generano etichette.

numero_inizio è lo scarto di un carattere in *stringa_originale*; può essere qualsiasi valore positivo o 0, oppure l'indirizzo o il nome di una cella contenente un valore positivo o 0. Se *numero_inizio* è maggiore della lunghezza di *stringa_originale*, **@SOSTIT** aggiunge *stringa_sostit* in fondo a *stringa_originale*.

n può essere qualsiasi numero intero positivo o 0, oppure l'indirizzo o il nome di una cella contenente un numero intero positivo o 0. Se *n* è 0, **@SOSTIT** aggiunge *stringa_sostit* da *numero_inizio* senza cancellare i caratteri di *stringa_originale*.

NOTA

@SOSTIT conta come caratteri anche gli spazi e i segni di interpunzione. Se si usa **@SOSTIT** per aggiungere o inserire del testo, non dimenticare di includere gli spazi necessari.

Si consiglia di usare **@RICER** assieme a **@SOSTIT** per cercare e sostituire un'etichetta o per calcolare un *numero_inizio* sconosciuto.

@SOSTIT è utile per sostituire una serie di caratteri con un'altra serie (ad esempio, per cambiare il prefisso in un database di numeri telefonici).

Esempi

@SOSTIT(CELLA;**@RICER**("-"CELLA;0);1;"") copia l'etichetta di CELLA (4-24) sotto forma di 4/24.

@DESTRA

@DESTRA(*stringa*; *n*) riporta gli ultimi *n* caratteri di *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

n può essere un numero intero positivo o 0, oppure l'indirizzo o il nome di una cella contenente un numero intero positivo o 0. Se *n* è pari a 0, il risultato sarà una stringa vuota. Se *n* è maggiore della lunghezza di *stringa*, **@DESTRA** riporta l'intera stringa.

NOTE

@DESTRA conta come caratteri anche i segni di interpunzione e gli spazi.

La funzione **@DESTRA** è utile per copiare solo una parte di un'etichetta in un'altra cella (ad esempio, per estrarre solo il cognome da etichette che contengono nome e cognome).

In una macro, **@DESTRA** può estrarre parti di etichette immesse dall'utente per memorizzarle in un database, per usarle nelle chiamate di sottoprogrammi o per modificare la macro stessa.

Si consiglia di usare **@DESTRA** assieme a **@RICER** qualora non si conosca l'esatto valore di *n* o qualora *n* sia variabile.

Esempi

@DESTRA(B3;10) = giocattoli, se B3 contiene l'etichetta Vendite giocattoli

Funzioni @ affini

@SINISTRA riporta i primi *n* caratteri di *stringa*. **@MEZZO** riporta un certo numero di caratteri di *stringa*.

@ARROT, @ARROTDIFETTO, @ARROTECESSO

@ARROT($x;n$) arrotonda il valore x al più vicino multiplo della potenza di 10 specificata da n .

@ARROTDIFETTO($x;n$;*direzione*) arrotonda per difetto il valore x al più vicino multiplo della potenza di 10 specificata da n .

@ARROTECESSO($x;n$;*direzione*) arrotonda per eccesso il valore x al più vicino multiplo della potenza di 10 specificata da n .

Argomenti

x può assumere qualsiasi valore.

n deve essere un valore compreso tra -100 e 100. Se si omette questo argomento nelle funzioni @ARROTDIFETTO e @ARROTECESSO, 1-2-3 usa il valore 0.

Se n è	@ARROT
positivo	modifica la parte decimale del numero (a destra della virgola); ad esempio, se n è 2, 1-2-3 arrotonda x al centinaio più vicino
negativo	modifica la parte intera del numero (a sinistra della virgola); ad esempio, se n è -2, 1-2-3 arrotonda x al centinaio più vicino
0	esegue l'arrotondamento al numero intero più vicino

direzione è un argomento facoltativo che specifica come arrotondare i numeri negativi; può essere 0 o 1.

- Per @ARROTECESSO: se la *direzione* è 0, 1-2-3 arrotonda i valori negativi per eccesso; se invece è 1, esegue l'arrotondamento per difetto.
- Per @ARROTDIFETTO: se la *direzione* è 0, 1-2-3 arrotonda i valori negativi per difetto; se invece è 1, esegue l'arrotondamento per eccesso.

In caso di omissione della *direzione*, 1-2-3 usa 0. Se x è un valore positivo, *direzione* non ha alcun effetto.

NOTA

Per visualizzare i valori con il numero specificato di cifre decimali ottenendo il massimo della precisione di calcolo, usare il formato Fisso; non usare @ARROT.

Esempi

@ARROT(134,578;2) = 134,58

@ARROT(134,578;0) = 135

@ARROT(134,578;-2) = 100.

@ARROTDIFETTO(134,578;2) = 134,57

@ARROTDIFETTO(134,578;0) = 134

@ARROTDIFETTO(134,578;-2) = 100

@ARROTECESSO(134,578;2) = 134,58

@ARROTECESSO(134,578;0) = 135

@ARROTECESSO(134,578;-2) = 200

Funzioni @ affini

@ARROTM arrotonda un valore al multiplo specificato. @PARI arrotonda un valore al successivo numero pari intero.

@DISPARI arrotonda un valore al successivo numero dispari intero. @INT tronca un valore, scartandone la parte

decimale. @TRONCADEC tronca un valore alla cifra decimale indicata.

@ARROTM

@ARROTM(*x*; *multiplo*; [*direzione*]) arrotonda il valore *x* al *multiplo* più vicino.

Argomenti

x e *multiplo* possono assumere qualsiasi valore, purché abbiano lo stesso segno.

direzione è un argomento facoltativo che specifica se *x* deve essere arrotondato per eccesso o per difetto.

<i>direzione</i>	1-2-3 arrotonda <i>x</i>
0	per eccesso; impostazione predefinita in caso di omissione dell'argomento
1	per difetto

Esempi

@ARROTM(25,37;0,05) = 25,40

@ARROTM(15,24;0,05;1) = 15,20

Funzioni @ affini

@INT tronca un valore, scartandone la parte decimale. **@ARROT**, **@ARROTDIFETTO** e **@ARROTECESSO** eseguono un arrotondamento alla cifra decimale specificata. **@PARI** arrotonda un valore al successivo numero pari intero. **@DISPARI** arrotonda un valore al successivo numero dispari intero. **@TRONCADEC** tronca un valore alla cifra decimale indicata.

@RIGHE

@RIGHE(*zona*) conta il numero di righe in *zona*.

Argomenti

zona può essere un indirizzo o un nome di zona.

NOTE

@RIGHE può essere usata insieme a {PER} in una macro che ripete la stessa operazione su una serie di righe per stabilire quando deve interrompersi.

Esempi

@RIGHE(A3..B7) = 5 (righe da 3 a 7)

@RIGHE(PUNTEGGIO) = 43, se PUNTEGGIO è la zona B3..B45

Funzioni @ affini

@COLONNE conta le colonne, mentre @FOGLI conta i fogli di lavoro in una zona.

Esempio: @REGRESSIONE

Il proprietario di una gelateria in una località turistica vuole calcolare quanti chili di gelato venderà il giorno successivo. Si ritiene che le vendite dipendano principalmente da tre fattori: il numero di ore di insolazione, la temperatura a mezzogiorno e il numero di torpedoni parcheggiati in prossimità della gelateria. Si vuole determinare la correlazione esistente tra questi fattori e le vendite. Dopo aver raccolto i dati per sei giorni consecutivi, si inseriscono le relative informazioni in un foglio di lavoro.

	A -----	A -----	B -----	C-----	D-----
1	Vendite di gelati		Sole	Gradi	Torpedoni parcheggiati
2		250	3	28	10
3		545	5	32	7
4		550	5	31	8
5		450	6	29	10
6		605	6	32	11
7		615	7	31	9

`@REGRESSIONE(B2..D7;A2..A7;3) = 0,979872`

La grande prossimità di R quadrato a 1 indica che esiste una forte correlazione tra le vendite di gelato, le condizioni atmosferiche e il numero di torpedoni.

@S

@S(*zona*) riporta l'etichetta contenuta nella prima cella della *zona*, oppure una stringa vuota se la cella è vuota o contiene un valore.

Argomenti

zona è l'indirizzo di cella o il nome di zona.

NOTE

La funzione @S può essere usata in combinazione con qualsiasi funzione @ di testo o formula di testo nel caso in cui l'immissione debba essere un'etichetta (ad esempio, un codice di avviamento postale) ma vi sia la possibilità che la cella contenga un valore. @S può essere quindi utilizzata per evitare che le formule di testo producano il risultato ERR (ad esempio, +A1&A2 genererebbe ERR se una delle due celle contenesse un valore).

Esempi

Nelle istruzioni macro

```
{SE @S(B6)=""}{BIP}{INDICA "L'IMMISSIONE DEVE ESSERE UN'ETICHETTA"}
```

@S genera una stringa vuota se la cella B6 contiene un valore o una cella vuota; in tal caso 1-2-3 emette un segnale acustico e cambia l'indicatore di modo in L'IMMISSIONE DEVE ESSERE UN'ETICHETTA.

Funzioni @ affini

@N riporta il valore contenuto nella prima cella di una *zona*. @DSTRINGA può determinare se una cella contiene un valore.

@SEC

@SEC(x) calcola la secante dell'angolo x . La secante è data dal rapporto tra l'ipotenusa e il lato adiacente a un angolo acuto di un triangolo rettangolo ed è il reciproco del coseno.

Argomenti

x è un angolo espresso in radianti; può assumere un valore compreso tra $-2^{\wedge}63$ e $2^{\wedge}63$.

Esempi

@SEC(@DEGTORAD(30)) = 1,154701, ossia la secante di un angolo di 30 gradi

Funzioni @ affini

@ASEC calcola l'arcosecante di un valore. @ASECH calcola l'arcosecante iperbolica di un valore. @COS calcola il coseno di un angolo. @SECH calcola la secante iperbolica di un valore.

@SECH

@SECH(x) calcola la secante iperbolica dell'angolo x. La secante iperbolica è il reciproco del coseno iperbolico. Il risultato della funzione @SECH è un valore maggiore di 0 oppure minore o uguale a 1.

Argomenti

x è un valore compreso all'incirca tra -709,7827 e 709,7827.

Esempi

@SECH(@DEGTORAD(30)) = 0,87701

Funzioni @ affini

@ASECH calcola l'arcocosecante iperbolico di un valore. @SEC calcola la secante di un valore.

@SECONDI

@SECONDI(*valore_orario*) estrae i secondi, ossia un numero intero compreso fra 0 e 59, da *valore_orario*.

Argomenti

valore_orario può essere un valore oppure l'indirizzo o il nome di una cella contenente un valore compreso tra 0,000000 (mezzanotte) e 0,999988 (11:59:59 PM).

Esempi

@SECONDI(0,333) = 31

@SECONDI(**@ORARIO**(11;15;45)) = 45, poiché 45 è l'argomento *secondi* della funzione. **@ORARIO**(11;15;45)

Funzioni @ affini

@ORA estrae l'ora, mentre **@MINUTI** estrae i minuti da un valore orario.

@SOMMASERIE

@SOMMASERIE($x;n;m;coefficienti$) calcola la somma di una serie di potenze.

Argomenti

x è la base della serie di potenze.

n è l'esponente iniziale a cui deve essere elevato x .

m è l'incremento di cui deve aumentare n per ogni termine della serie.

x , n e m possono assumere qualsiasi valore.

coefficienti è una zona contenente i coefficienti per cui 1-2-3 deve moltiplicare ogni successiva potenza di x . Il numero di celle comprese in *coefficienti* determina il numero di termini della serie; per esempio, se *coefficienti* contiene dieci celle, la serie di potenze sarà composta da dieci termini.

Esempi

La zona DATI contiene i seguenti coefficienti: 0,2, 0,7, 1, .3.

@SOMMASERIE(3,5;2;1;DATI) = 227,5438

@FOGLI

@FOGLI(*zona*) conta il numero di fogli in *zona*.

Argomenti

zona può essere un indirizzo o un nome di zona.

NOTA

@FOGLI può essere usata assieme a {PER} in una macro che ripete un'operazione su una serie di fogli per stabilire quando deve interrompersi.

Esempi

@FOGLI(T_2) = 4, se T_2 è la zona B:B3..E:C45 (fogli B, C, D ed E)

Funzioni @ affini

@COLONNE conta le colonne, mentre @RIGHE conta le righe di una zona. @CONVERTIREF converte le lettere delle colonne o dei fogli di 1-2-3 (A - IV) in numeri (1 - 256) e viceversa.

@SEGNO

@SEGNO(x) produce 1 se x è un valore positivo, 0 se x è 0, e -1 se x è un valore negativo.

Argomenti

x può assumere qualsiasi valore.

Esempi

$$\text{@SEGNO}(15) = 1$$

$$\text{@SEGNO}(15*0) = 0$$

$$\text{@SEGNO}(-15) = -1$$

@SEN

@SEN(x) calcola il seno dell'angolo x . Il seno è dato dal rapporto tra il cateto opposto all'angolo acuto di un triangolo rettangolo e l'ipotenusa:

$\text{seno}(\text{angolo}) = \text{cateto opposto}/\text{ipotenusa}$

Argomenti

x è un angolo espresso in radianti. x può essere un valore oppure l'indirizzo o il nome di una cella contenente un valore compreso tra -2^{63} e 2^{63} .

Esempi

@SEN(@GRADIRAD(30)) = 0,5, ossia il seno di un angolo di 30 gradi.

Funzioni @ affini

@ASEN calcola l'arcoseno di un valore per trovare la misura di un angolo. **@SENH** calcola il seno iperbolico di un angolo.

@SENH

@SENH(x) calcola il seno iperbolico dell'angolo x.

Argomenti

x può assumere qualsiasi valore.

Esempi

@SENH(@GRADIRAD(30)) = 0,547853

Funzioni @ affini

@ASENH calcola l'arcoseno iperbolico di un valore. @SEN calcola il seno di un angolo.

@ASIMMETRIA

@ASIMMETRIA(zona;*[tipo]*) calcola l'asimmetria dei valori di una *zona*.

Argomenti

zona è il nome o l'indirizzo di una zona contenente valori. Se *zona* contiene meno di tre valori, la funzione **@ASIMMETRIA** produce **ERR**.

tipo è un argomento facoltativo che specifica se deve essere calcolata l'asimmetria della popolazione o quella campionaria.

<i>tipo</i>	1-2-3 calcola
0	l'asimmetria della popolazione; impostazione predefinita in caso di omissione dell'argomento
1	l'asimmetria campionaria

NOTA

L'asimmetria misura la simmetria di una distribuzione rispetto alla media. Un'asimmetria positiva indica la presenza di una coda a sinistra, mentre un'asimmetria negativa indica una coda a destra.

Esempi

La zona DATI contiene i seguenti valori: 2, 5, 6, 9, 1, 2, 4.

@ASIMMETRIA(DATI) = 0,584816

Funzioni @ affini

@CURTOSI calcola la curtosi dei valori compresi in una lista. **@SCARTO** e **@SCARTOPURO** calcolano la deviazione standard di una popolazione. **@VAR** e **@VARPURA** calcolano la varianza della popolazione.

@AQC

@AQC(*costo;realizzo;vita*) calcola la quota di deprezzamento di un bene secondo il metodo di ammortamento a quote costanti. Il bene ha un *costo* iniziale, una *vita* utile prevista e un valore finale di *realizzo*.

Argomenti

costo rappresenta l'importo pagato per il bene; può essere un valore qualsiasi oppure l'indirizzo o il nome di una cella contenente un valore qualsiasi.

realizzo rappresenta il valore del bene alla fine della sua vita utile; può essere un valore qualsiasi oppure l'indirizzo o il nome di una cella contenente un valore qualsiasi.

vita rappresenta il numero di periodi necessari affinché si compia l'ammortamento del bene fino al raggiungimento del valore di realizzo; può essere un valore qualsiasi oppure l'indirizzo o il nome di una cella contenente un valore qualsiasi (zero escluso).

NOTA

Il metodo di ammortamento a quote costanti suddivide il costo ammortizzabile (costo di acquisizione meno valore di realizzo) in parti uguali per ogni periodo della vita utile del bene. La vita utile è il numero di periodi (solitamente anni) necessari per raggiungere il completo ammortamento del bene.

Esempi

Si supponga di aver acquistato una macchina per ufficio al prezzo di 10.000.000 di lire, con una vita utile prevista di 10 anni e un valore di realizzo di 1.200.000 di lire, e di voler calcolare le spese di ammortamento annuali secondo il metodo di ammortamento a quote costanti. La formula **@ACQ**(10000000;1200000;10) riporta il valore L.880.000, ossia la quota di deprezzamento annuale.

Funzioni @ affini

@AQP calcola l'ammortamento con il metodo a quote fisse decrescenti. **@DDB** calcola l'ammortamento di un bene secondo il metodo a quote proporzionali; **@VDB** calcola l'ammortamento di un bene secondo il metodo a quote proporzionali con tasso variabile, mentre **@SYD** usa il metodo "americano" (somma degli anni di prevista durata).

@PICCOLO

@PICCOLO(*zona*;n) individua l'*n*-esimo valore più basso nella *zona*.

Argomenti

zona è il nome o l'indirizzo di una zona contenente valori.

n è un numero intero positivo. Se *n* è maggiore del numero di valori compresi nella *zona*, @PICCOLO produce ND.

Esempi

Una zona denominata PUNTEGGIO contiene i seguenti esiti di test: 87, 85, 90, 80, 82, 92, 79, 85, 95, 86:

@PICCOLO(PUNTEGGIO;3) produce 82, ossia il terzo valore più basso

Funzioni @ affini

@GRANDE individua l'*n*-esimo valore più alto in una zona. @MAX e @MAXPURO individuano il valore più alto in una zona. @MIN e @MINPURO individuano il valore più basso in una zona.

@RADQ

@RADQ(x) calcola la radice quadrata positiva di x.

Argomenti

x può essere un valore positivo.

Esempi

@RADQ(B3) = 10, se B3 contiene il valore 100.

@RADQ(-2) = ERR, poiché x è negativo.

@RADQPI

@RADQPI(x) calcola la radice quadrata di x*.

Argomenti

x può essere qualsiasi valore oppure 0.

Esempi

@RADQPI(0,5) = 1,253314

@RADQPI(2) = 2,506628

Funzioni @ affini

@RADQ calcola la radice quadrata positiva di un valore. @PI produce il valore.

@SCARTO, @SCARTOC, @SCARTOPURO, @SCARTOCPURO

@SCARTO(*lista*) calcola la deviazione standard di una popolazione di valori situati in *lista*.

@SCARTOC(*lista*) calcola la deviazione standard campionaria di una *lista* di valori.

@SCARTOPURO(*lista*) calcola la deviazione standard della popolazione di valori compresi in *lista*, ignorando le celle contenenti etichette.

@SCARTOCPURO(*lista*) calcola la deviazione standard campionaria dei valori di una *lista*, ignorando le celle contenenti etichette.

Argomenti

lista può comprendere una combinazione qualsiasi dei seguenti elementi: numeri, formule numeriche e indirizzi o nomi di zona contenenti numeri o formule. Ricordarsi di separare tra loro i vari elementi di *lista* con gli appositi **separatori di argomenti**.

Vedere anche [Argomenti delle funzioni @ statistiche](#).

NOTE

@SCARTO e **@SCARTOPURO** usano il metodo n (o della popolazione) per calcolare la deviazione standard di una popolazione di dati. Il metodo n presuppone che i valori selezionati costituiscano l'intera popolazione. Se invece sono solo un campione della popolazione, la deviazione standard risulta falsata dagli errori introdotti con il campionamento.

@SCARTOC e **@SCARTOCPURO** usano il metodo n-1 (o del campione) per calcolare la deviazione standard di un popolazione di dati campione. Il metodo n-1 produce una deviazione standard leggermente maggiore di quella calcolata con il metodo n in modo da compensare gli errori di campionamento. Non essendo viziata da errori di campionamento, una deviazione standard maggiore risulta più precisa.

La deviazione standard è uguale alla radice quadrata della varianza di tutti i singoli valori rispetto alla media aritmetica dei valori stessi.

Esempi

[@SCARTO](#) e [@SCARTOC](#)

Funzioni @ affini

[@SCARTODB](#) e [@SCARTODBC](#) calcolano la deviazione standard dei valori di un campo di una tabella di database.

[@VAR](#) e [@VARPURA](#) calcolano la varianza della popolazione dei valori compresi in una lista. [@VARC](#) e

[@VARCPURA](#) calcolano la varianza campionaria dei valori compresi in una lista.

@STRINGA

@STRINGA($x;n$) converte il valore x in un'etichetta applicandole il formato specificato da n .

Argomenti

x è un valore.

n è un numero intero, il cui significato è illustrato dalla seguente tabella.

Valore di n	Formato generato
da 0 a 116	<u>Fisso</u> , con n cifre decimali
da 1000 a 1116	<u>Punto</u> , con $n-1000$ cifre decimali
da -18 a -1	<u>Scientifico</u> , con un numero di cifre pari a @ASS(n)
da 10001 a 10512	<u>Generico</u> , con un massimo di $n-10000$ caratteri

NOTA

@STRINGA ignora tutti i caratteri di formattazione usati da 1-2-3 per visualizzare il valore x , ivi inclusi tutti i simboli di valuta e gli altri simboli di formattazione numerica immessi dall'utente o creati da 1-2-3 dopo aver selezionato un formato numerico. Per esempio, se la cella A7 contiene il valore L. 45,23, @STRINGA (A7;2) produce l'etichetta 45,23.

Esempi

@STRINGA(203;3) = 203,000 (sotto forma di etichetta)

@STRINGA(1.23587;0) = 1 (sotto forma di etichetta)

@STRINGA(20500;1002) = 20.500,00 (sotto forma di etichetta)

@STRINGA(@PI;-5) = 3,1416E+000 (sotto forma di etichetta)

@STRINGA(123456.789;10008) = 1,2E+005 (sotto forma di etichetta)

Funzioni @ affini

@VALORE converte un numero immesso sotto forma di testo nel rispettivo valore.

@TOTPARZ

@TOTPARZ(lista) somma i valori di una *lista*. Questa funzione può servire per indicare quali celle dovrà sommare @TOTGENER.

Argomenti

lista può comprendere una combinazione qualsiasi dei seguenti elementi: numeri, formule numeriche e indirizzi o nomi di zona contenenti numeri o formule. Ricordarsi di separare tra loro i vari elementi di *lista* con gli appositi separatori di argomenti.

Vedere anche Argomenti delle funzioni @ statistiche.

Esempi

@TOTPARZ(B5..B9) somma i valori contenuti in B5..B9.

@TOTPARZ(VENDITE;M25..R25) somma i valori contenuti nella zona VENDITE e quelli della zona M25..R25.

Funzioni @ affini

@SOMMA somma i valori compresi in una lista. @SOMMAPOSITIVI esegue la somma dei soli valori positivi contenuti in una lista. @SOMMANEGATIVI esegue la somma dei soli valori negativi contenuti in una lista.

@SOMMA

@SOMMA(*lista*) aggiunge i valori di *lista*.

Argomenti

lista può comprendere una combinazione qualsiasi dei seguenti elementi: numeri, formule numeriche e indirizzi o nomi di zona contenenti numeri o formule. Ricordarsi di separare tra loro i vari elementi di *lista* con gli appositi separatori di argomenti.

Vedere anche [Argomenti delle funzioni @ statistiche](#).

Esempi

@SOMMA(B5..B9) riporta la somma dei valori situati nella zona B5..B9

@SOMMA(VENDITE;M25..R25) riporta la somma dei valori situati nelle zone VENDITE e M25..R25

SmartIcons correlate



Somma i valori contenuti nella zona evidenziata o adiacente, a seconda che siano incluse delle celle vuote sotto o a destra della zona.

Funzioni @ affini

[@SOMMADB](#) calcola la somma dei valori che soddisfano le condizioni specificate.

[@NSUM](#) somma ogni *n*-esimo valore di *lista*, partendo da *scarto*. [@TOTPARZ](#) somma i valori compresi in una lista e indica quali valori [@TOTGENER](#) deve sommare. [@SOMMAPOSITIVI](#) esegue la somma dei soli valori positivi contenuti in una lista. [@SOMMANEGATIVI](#) esegue la somma dei soli valori negativi contenuti in una lista.

@SOMMAPROD

@SOMMAPROD(*lista*) moltiplica i valori situati nelle celle corrispondenti di più zone, sommandone i prodotti.

Argomenti

lista può essere una qualsiasi combinazione di zone contenenti valori e aventi le stesse dimensioni e forma. Se le zone della *lista* non hanno dimensioni e forma uguali, @SOMMAPROD genera il valore ERR.

Vedere anche Argomenti delle funzioni @ statistiche.

NOTA

Se le zone della *lista* sono colonne, @SOMMAPROD esegue la moltiplicazione riga per riga; se invece le zone in *lista* sono righe, @SOMMAPROD esegue la moltiplicazione colonna per colonna. Se ciascuna zona della *lista* comprende più di una colonna, @SOMMAPROD moltiplica riga per riga.

Esempi

Nel seguente esempio preso da una tabella di database di un'agenzia immobiliare, la funzione @SOMMAPROD viene usata per calcolare il totale delle provvigioni dovute agli agenti per la vendita di appartamenti nel mese di febbraio 1990.

lista comprende due zone: VENDUTO (D4..D6), contenente i prezzi pagati per la vendita di tre appartamenti, e PROVVIS (E4..E6) contenente la percentuale per le provvigioni dovute agli agenti sul prezzo di vendita:

VENDUTO	PROVV
L. 250.000.000	4%
L. 340.580.000	5%
L. 770.325.000	4%

@SOMMAPROD(VENDUTO;PROVV) = L. 57.841.000, ossia il totale delle provvigioni (L. 10.000.000 + L. 17.029.000 + L. 30.812.000) dovute agli agenti per la vendita di tre appartamenti.

Funzioni @ affini

@SOMMAQUADRATI calcola la somma dei quadrati dei valori compresi in una lista. @SOMMAXMY2 calcola la somma dei quadrati delle differenze tra i valori situati in celle corrispondenti di due zone diverse.

@SOMMAQUADRATI

@SOMMAQUADRATI(*lista*) calcola la somma dei quadrati dei valori compresi nella *lista*.

Argomenti

lista può comprendere una combinazione qualsiasi dei seguenti elementi: numeri, formule numeriche e indirizzi o nomi di zona contenenti numeri o formule. Ricordarsi di separare tra loro i vari elementi di *lista* con gli appositi separatori di argomenti.

Vedere anche [Argomenti delle funzioni @ statistiche](#).

Esempi

@SOMMAQUADRATI(2;4;6) = 56

Funzioni @ affini

[@SOMMA](#) somma i valori compresi in una lista. [@SOMMAPROD](#) moltiplica i valori contenuti in celle corrispondenti di zone diverse, sommandone i prodotti.

@SOMMAXMY2

@SOMMAXMY2(*zona1*; *zona2*) sottrae i valori contenuti nella *zona2* dalle celle corrispondenti della *zona1*, eleva le differenze al quadrato per poi sommare i risultati.

Argomenti

zona1 e *zona2* sono zone contenenti valori e devono avere uguali forma e dimensioni, altrimenti la funzione @SOMMAXMY2 produce ERR.

NOTA

Se *zona1* e *zona2* sono formate da una sola colonna, 1-2-3 esegue la sottrazione riga per riga. Se le due zone sono formate da più colonne, 1-2-3 procede per colonna.

Esempi

Nell'esempio sottostante la *zona1* è denominata MARTEDI e la *zona2* è denominata MERCOLEDI:

MARTEDI	MERCOLEDI
---------	-----------

5	3
---	---

4	4
---	---

7	8
---	---

@SOMMAXMY2(MARTEDI;MERCOLEDI) = 5

Funzioni @ affini

@SOMMAPROD calcola la somma dei prodotti dei valori situati in celle corrispondenti di diverse zone.

@SOMMAQUADRATI calcola la somma dei quadrati dei valori compresi in una lista.

@SYD

@SYD(costo;realizzo;vita;periodo) calcola la quota detraibile per il deprezzamento di un bene, per un *periodo* specificato, secondo il metodo "americano" (somma degli anni di prevista durata). Il bene ha un *costo* iniziale, una *vita* utile prevista e un valore finale di *realizzo*.

Argomenti

costo rappresenta la somma pagata per il bene; può essere un valore qualsiasi.

realizzo rappresenta il valore di un bene alla fine della sua vita utile; può essere un valore qualsiasi.

vita rappresenta il numero di periodi necessari affinché il bene si deprezzi fino a raggiungere il valore di *realizzo*. *vita* può essere un valore maggiore o uguale a 1.

periodo rappresenta il periodo per cui si vuole calcolare la quota di deprezzamento; può essere un valore maggiore o uguale a 1.

NOTE

Il metodo di ammortamento basato sulla somma degli anni di prevista durata accelera il tasso di deprezzamento, consentendo di detrarre una quota di ammortamento maggiore nei periodi iniziali rispetto a quelli finali (anche se non tanto quanto **@DDB**). Il costo ammortizzabile è uguale al costo effettivo meno il valore di *realizzo*.

La funzione **@SYD** viene utilizzata quando si vuole detrarre una quota di ammortamento maggiore all'inizio della vita utile di un bene.

Esempi

Si supponga di aver acquistato una macchina per ufficio al prezzo iniziale di L. 10.000.000, con una vita utile prevista di 10 anni e un valore di *realizzo* dopo 10 anni di L. 1.200.000. Si vuole calcolare la quota di ammortamento per il quinto anno, usando il metodo della somma degli anni di prevista durata. **@SYD**(10000000;1200000;10;5) genera il valore L. 960.000, ossia la quota di ammortamento per il quinto anno.

Funzioni @ affini

@DDB calcola l'ammortamento di un bene secondo il metodo a doppie quote proporzionali ai valori residui. **@VDB** calcola l'ammortamento con il metodo a quote proporzionali con tasso variabile, mentre **@AQP** di usa il metodo a quote fisse decrescenti e **@AQC** il metodo a quote costanti.

Esempio: @SCARTO e @SCARTOC

La seguente tabella contiene i dati relativi all'altezza e al peso di dieci soggetti selezionati a caso. Si vuole determinare la deviazione standard delle altezze:

$$\text{@SCARTO}(A2..A11) = 5,793483$$

Si supponga ora che i soggetti esaminati rappresentino un campione selezionato casualmente da un più ampio gruppo di individui.

A	-----	A	-----	B	--
1		ALTEZZA (cm)		PESO (kg)	
2		190,50		72,73	
3		187,96		86,36	
4		175,26		68,18	
5		175,26		76,37	
6		180,34		77,27	
7		180,34		72,73	
8		187,96		75,00	
9		172,72		68,18	
10		177,80		70,46	
11		179,07		86,36	

@TAN

@TAN(x) calcola la tangente dell'angolo x . La tangente è data dal rapporto tra il cateto opposto all'angolo acuto di un triangolo rettangolo e il cateto adiacente allo stesso angolo:

$\tan(\text{angolo}) = \text{cateto opposto}/\text{cateto adiacente}$

Argomenti

x è un angolo espresso in radianti e può assumere un valore compreso tra -2^63 e 2^63 .

Esempi

@TAN(@GRADIRAD(35)) = 0,700208, ossia la tangente di un angolo di 35 gradi.

Funzioni @ affini

@ATAN calcola l'arcotangente utilizzando la tangente di un angolo.

@TANH calcola la tangente iperbolica di un angolo.

@TANH

@TANH(x) calcola la tangente iperbolica dell'angolo x. La tangente iperbolica è data dal rapporto tra il seno iperbolico e il coseno iperbolico. Il risultato della funzione @TANH è sempre un valore compreso tra -1 e 1.

Argomenti

x è un valore compreso all'incirca tra circa -709,7827 e 709,7827.

Esempi

@TANH(@GRADIRAD(30)) = 0,480473

Funzioni @ affini

@ATANH calcola l'arcotangente iperbolica di un valore. @TAN calcola la tangente di un angolo.

@DISTRIBT

@DISTRIBT(*x*;*gradi_libertà*;*[tipo]*;*[code]*) calcola la distribuzione *t* di Student.

Argomenti

Il valore di *x* dipende dal valore immesso per *tipo*.

tipo	x
0	valore critico o limite massimo della variabile casuale della distribuzione <i>t</i> cumulativa; può assumere qualsiasi valore; impostazione predefinita in caso di omissione dell'argomento
1	probabilità; deve essere un valore compreso tra 0 e 1

gradi_libertà è il numero dei gradi di libertà del campione; deve essere un numero intero positivo.

tipo è un argomento facoltativo che specifica come 1-2-3 deve calcolare la funzione @DISTRIBT.

tipo	1-2-3 calcola
0	il livello di significatività corrispondente al valore critico <i>x</i> ; impostazione predefinita in caso di omissione dell'argomento
1	il valore critico corrispondente al livello di significatività <i>x</i>

code è un argomento facoltativo che specifica la direzione del test *t*.

code	1-2-3 esegue
1	un test <i>t</i> a una coda
2	un test <i>t</i> a due code; impostazione predefinita in caso di omissione dell'argomento

Non è consentito l'uso di un argomento facoltativo senza quelli che lo precedono.

NOTE

@DISTRIBT calcola la distribuzione *t* cumulativa con un'approssimazione del $\pm 3 \cdot 10^{-7}$. Se @DISTRIBT non riesce a ottenere un'approssimazione dello 0,0000001 dopo 100 iterazioni, produce ERR.

La distribuzione *t* di Student è la distribuzione del rapporto esistente tra una distribuzione normale standardizzata e la radice quadrata del quoziente di una distribuzione chi-quadrato divisa per il numero di gradi di libertà.

Esempi

@DISTRIBT(2,228;10) = 0,05

@DISTRIBT(0,05;10;1) = 2,228

Funzioni @ affini

@DISTRIBCHI2 calcola la distribuzione chi-quadrato. @DISTRIBF calcola la distribuzione *F*. @TTEST calcola la probabilità associata a un test *t* di Student.

@TERM, @NUMPERIODI

@TERM(*versamenti;interesse;valore_futuro*) calcola il tempo necessario per accumulare un capitale di *valore_futuro*, effettuando *versamenti* periodici di uguale importo, dato un tasso d'*interesse* periodico.

@NUMPERIODI(*versamenti;interesse;valore_futuro;[tipo];[valore_attuale]*) calcola il tempo necessario per effettuare una serie di *versamenti* di uguale importo del *valore_attuale* specificato fino ad accumulare un capitale di *valore_futuro*, dato un tasso d'*interesse* periodico. **@NUMPERIODI** può eseguire il calcolo per una rendita immediata o anticipata, a seconda del valore specificato per *tipo*.

Argomenti

versamenti rappresenta il valore degli investimenti di uguale importo; può assumere qualunque valore ad eccezione di 0.

interesse rappresenta il tasso d'interesse periodico; può essere un qualsiasi valore maggiore di -1.

valore_futuro rappresenta l'importo che si vuole raggiungere; può essere un qualsiasi valore.

tipo è un argomento facoltativo che specifica se il calcolo deve essere eseguito per una rendita immediata o anticipata.

<i>tipo</i>	1-2-3 calcola una rendita
0	immediata (versamenti da effettuare alla fine del periodo); impostazione predefinita in caso di omissione dell'argomento
1	anticipata (versamento da effettuare all'inizio del periodo)

valore_attuale è un argomento facoltativo che specifica il valore attuale di una serie di versamenti futuri. Questo argomento può assumere qualsiasi valore. In caso di omissione, 1-2-3 usa un *valore_attuale* pari a zero.

Non è consentito l'uso di un argomento facoltativo senza aver specificato quelli che lo precedono.

NOTE

Per calcolare il tempo necessario per rimborsare un prestito, usare un argomento *valore_futuro* negativo con la funzione **@TERM**. Ad esempio, per sapere quanto tempo si impiegherà per restituire un prestito di 10.000.000 di lire a un tasso d'interesse annuo del 10%, versando 1.174.000 all'anno, si usi la seguente formula:

@ASS(@TERM(1174000;0,1;-10000000)), che genera il valore 20,03: ci vorranno quindi 20 anni per rimborsare l'importo del prestito.

Esempi

Si supponga di voler depositare l'importo di 2.000.000 di lire alla fine di ogni anno su un conto di risparmio che rende un tasso d'interesse di 7,5% capitalizzato annualmente e di voler stabilire quanto tempo si impiegherà per raggiungere l'importo di 100.000.000 di lire:

@TERM(2000000;0,075;100000000) = 21,5 anni

Se i versamenti vengono effettuati all'inizio dell'anno, si deve usare quest'altra funzione:

@NUMPERIODI(2000000;0,075;100000000;1) = 20,76 anni

Funzioni @ affini

@CTERM calcola il numero di periodi di capitalizzazione per un investimento effettuato con una sola operazione di versamento.

@TERM2 calcola il tempo necessario per accumulare un capitale di *valore_futuro*, effettuando *versamenti* periodici di uguale importo, dato un tasso d'*interesse* periodico, secondo la convenzione della rendita anticipata.

@ORARIO

@ORARIO(*ora;minuti;secondi*) calcola il valore orario per l'*ora*, i *minuti* e i *secondi* specificati.

Argomenti

ora può essere un numero intero compreso tra 0 (mezzanotte) e 23 (11:00 PM).

minuti può essere un numero intero compreso tra 0 e 59.

secondi può essere un numero intero compreso tra 0 e 59.

NOTA

Per assegnare all'orario il formato desiderato, usare un formato ora.

Esempi

La formula (**@ORARIO**(13;0;0)-**@ORARIO**(9;15;0))*9500*24 calcola l'importo dovuto a un collaboratore esterno per una giornata di lavoro, sottraendo l'orario di inizio da quello di fine e moltiplicando il risultato per una tariffa oraria di L. 9.500.

Funzioni @ affini

@ORDORA converte le etichette in valori orari.

@ORDORA

@ORDORA(*stringa*) calcola il valore orario dell'ora specificata nella *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa tra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta. *stringa* deve avere uno dei quattro formati ora di 1-2-3.

NOTA

La funzione @ORDORA è utile per convertire gli orari immessi sotto forma di etichette in valori orari e per poterli quindi utilizzare nei calcoli. Questa funzione è particolarmente indicata per i dati importati da un altro programma, ad esempio di elaborazione testi.

Per assegnare al valore orario il formato ora desiderato, usare un formato ora.

Esempi

@ORDORA("08:19:27 AM") genera 0,34684

@ORDORA("08:19:27 AM") riporta il valore 0,34684 nella forma 08:19:27 AM, se la cella presenta il formato HH:MM:SS AM/PM.

Funzioni @ affini

@ORARIO calcola il valore orario quando vengono specificati ora, minuti e secondi.

@OGGI

@OGGI calcola il valore data che corrisponde alla data corrente del computer.

NOTA

Al valore di @OGGI può essere attribuito qualsiasi formato data. 1-2-3 ricalcola @OGGI ogni volta che si esegue un ricalcolo.

Esempi

@OGGI = 31048, se la data del giorno è 1 gennaio 1985

@OGGI = 33418, se la data del giorno è 29 giugno 1991

@TRONCA

@TRONCA(*stringa*) elimina eventuali spazi iniziali, finali e consecutivi dalla *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

NOTA

La funzione @TRONCA impedisce che le immissioni di un database contengano spazi superflui che potrebbero incidere sull'ordinamento di una zona.

Esempi

@TRONCA(" 45 3/8") = 45 3/8, eliminando lo spazio che precede 45 e uno o due spazi situati tra 45 e 3/8

@TRONCA(" Via Milano 60") = Via Milano 60, eliminando lo spazio prima di Via e uno o due spazi situati tra Milano e 60

Funzioni @ affini

@FISSALUNGH produce un testo allineato entro il numero di spazi specificati.

@VERO

@VERO genera il valore logico 1 (vero).

NOTE

Se un'istruzione logica risulta vera (ad esempio, A1=B1), ha valore logico 1. Se invece è falsa, ha valore logico 0.

@VERO equivale a usare il valore 1 nelle formule di elaborazione delle condizioni logiche, con la differenza che, con @VERO, la formula risulta più comprensibile.

Esempi

@SE(A6>500;@VERO;@FALSO) = 1, se la cella A6 contiene un valore maggiore di 500

Funzioni @ affini

@FALSO genera il valore logico 0.

@TRONCADEC

@TRONCADEC(x;[n]) tronca il valore x alla cifra decimale specificata da n.

Argomenti

x può essere un valore qualsiasi.

n è un argomento facoltativo compreso tra -100 e 100.

Se n è	@TRONCADEC
positivo	agisce sulla parte decimale del numero (a destra della virgola); per esempio, dando a n il valore 2, 1-2-3 tronca x al centesimo più vicino
negativo	modifica la parte intera del numero (a sinistra della virgola); per esempio, dando a n il valore -2, x viene troncato al centinaio più vicino
0	tronca x al numero intero più vicino; impostazione predefinita in caso di omissione dell'argomento

NOTA

Per visualizzare i valori con il numero specificato di cifre decimali ottenendo il massimo della precisione di calcolo, scegliere il formato numerico Fisso; non usare @TRONCADEC.

Esempi

@TRONCADEC(123,45) = 123

@TRONCADEC(-123,45) = -123

@TRONCADEC(123,45;-2) = 100

@TRONCADEC(123,45;1) = 123,4

@TRONCADEC(-123,45;-2) = -100

@TRONCADEC(-123,45;1) = -123,4

Funzioni @ affini

@ARROT, @ARROTDIFETTO e @ARROTECCESSO arrotondano un valore alla cifra decimale specificata.

@ARROTM arrotonda un valore al multiplo specificato. @PARI arrotonda un valore al successivo numero pari intero.

@DISPARI arrotonda un valore al successivo numero dispari intero. @INT tronca un valore, scartandone la parte decimale.

@MAIUSC

@MAIUSC(*stringa*) converte in maiuscole tutte le lettere di *stringa*.

Argomenti

stringa può essere una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta.

NOTA

L'alternanza di maiuscole e minuscole va a incidere sull'ordinamento delle etichette di una zona; due etichette identiche potrebbero quindi risultare diverse qualora avessero una diversa combinazione di maiuscole e minuscole.

Esempi

@MAIUSC("Giuseppe Verdi") = GIUSEPPE VERDI

@MAIUSC(B2) = ATTENZIONE, se B2 contiene l'etichetta attenzione

Funzioni @ affini

@MINUSC converte in minuscole tutte le lettere di *stringa*. @NOMEPRO converte in maiuscola solo la prima lettera di ogni parola di *stringa*.

@VALORE

@VALORE(*stringa*) converte un numero immesso come *stringa* nel valore corrispondente.

Argomenti

stringa può essere una stringa di testo racchiusa tra virgolette (" ") oppure un'etichetta contenente solo numeri. *stringa* può avere l'aspetto di un numero normale (456,7), di un numero in formato scientifico (4.567E2), di un numero misto (45 7/8) o di un numero cui è stato assegnato un formato (L. 45.600).

NOTE

@VALORE ignora gli spazi iniziali e finali; se però la *stringa* contiene degli spazi che separano i simboli dai numeri (ad esempio, L. 1.500), questa funzione produce **ERR**.

@VALORE genera 0 quando la *stringa* è costituita da una cella vuota o da una stringa vuota, mentre produce **ERR** se la *stringa* contiene caratteri non numerici.

Per sostituire la funzione **@VALORE** con il rispettivo valore, premere F2 (MODIFICA) e successivamente F9 (CALCOLA).

Non si possono eseguire calcoli nell'argomento *stringa* di **@VALORE** benché sia possibile creare una formula con varie funzioni **@VALORE**; per esempio, **@VALORE**("22"+"20") = 0, ma **@VALORE**("22")+**@VALORE**("20") = 42.

Esempi

@VALORE("543") = il valore 543.

@VALORE(B3) = il valore 49,75 se la cella B3 contiene l'etichetta 49 3/4.

@VALORE("85%") = il valore 0,85.

Funzioni @ affini

@STRINGA converte un valore in un'etichetta.

@VAR, @VARC, @VARPURA, @VARCPURA

@VAR(*lista*) calcola la varianza della popolazione di una *lista* di valori.

@VARC(*lista*) calcola la varianza campionaria di una *lista* di valori.

@VARPURA(*lista*) calcola la varianza della popolazione di una *lista* di valori, ignorando le celle contenenti etichette.

@VARCPURA(*lista*) calcola la varianza campionaria di una *lista* di valori, ignorando le celle contenenti etichette.

Argomenti

La *lista* può comprendere una combinazione qualsiasi dei seguenti elementi: numeri, formule numeriche e indirizzi o nomi di zone contenenti numeri o formule. Ricordarsi di separare tra loro i vari elementi di *lista* con gli appositi separatori di argomenti.

Vedere anche [Argomenti delle funzioni @ statistiche](#).

NOTE

Le funzioni @ di varianza possono essere utili in molti test statistici di analisi della varianza.

@VAR e **@VARPURA** usano il metodo n (o della popolazione) per calcolare la varianza. Il metodo n presuppone che i valori selezionati costituiscano l'intera popolazione. Se invece i valori rappresentano solo un campione della popolazione, la varianza risulta falsata dagli errori introdotti con il campionamento.

@VARC e **@VARCPURA** usano il metodo n-1 (o del campione) per calcolare la varianza. Il metodo n-1 produce una varianza leggermente maggiore di quella calcolata con il metodo n in modo da compensare gli errori di campionamento. Non essendo viziata da errori di campionamento, la varianza maggiore risulta più precisa.

Esempi

[@VAR](#) e [@VARC](#)

Funzioni @ affini

[@VARDB](#) e [@VARDBC](#) calcolano la varianza di una popolazione di valori che soddisfano le condizioni specificate.

@VDB

@VDB(costo;realizzo;vita;periodo_iniz;periodo_fin;[tasso_ammortamento];[commutatore]) calcola la detrazione per l'ammortamento di un bene con valore iniziale pari a *costo*, una *vita* utile prevista e un valore finale di *realizzo* per il periodo di tempo specificato da *periodo_iniz* e *periodo_fin*, utilizzando il metodo a quote proporzionali con tasso variabile.

Argomenti

costo rappresenta la somma pagata per l'acquisto di un bene; può essere qualsiasi valore maggiore di quello di *realizzo*.

realizzo rappresenta il valore del bene al termine della sua vita utile; può essere un valore qualsiasi.

vita rappresenta il numero di periodi necessari affinché il bene si ammortizzi fino a raggiungere il valore di *realizzo*; può essere qualsiasi valore maggiore di 0.

periodo_iniz rappresenta il momento nella vita di un bene a partire dal quale si vuole iniziare a calcolare l'ammortamento; può essere qualsiasi valore maggiore o uguale a 0, ma minore di quello specificato da *vita*.

periodo_fin rappresenta il momento nella vita di un bene in cui si vuole terminare il calcolo dell'ammortamento; può essere qualsiasi valore maggiore di quello di *periodo_iniz*.

periodo_iniz e *periodo_fin* corrispondono alla vita utile del bene relativa al periodo fiscale. Ad esempio, volendo calcolare l'ammortamento per il primo anno di vita di un bene acquistato all'inizio del secondo trimestre dell'anno fiscale, *periodo_iniz* sarebbe uguale a 0 e *periodo_fin* avrebbe un valore pari a 0,75 (1 anno meno un trimestre).

@VDB può essere usata per i calcoli di ammortamento anche su più periodi.

@VDB permette l'uso dell'opzione di periodo iniziale al fine di calcolare l'ammortamento per il periodo in cui il bene è in servizio. **@VDB** usa la parte decimale di *periodo_iniz* e *periodo_fin* per determinare l'opzione di periodo iniziale. Se *periodo_iniz* e *periodo_fin* hanno entrambi una parte decimale, **@VDB** userà la parte decimale di *periodo_iniz*.

tasso_ammortamento è un argomento facoltativo che specifica la percentuale di ammortamento a quote costanti da usare come tasso di ammortamento. Se si omette questo argomento, 1-2-3 usa il 200%, ossia il tasso di ammortamento a doppie quote proporzionali. *tasso_ammortamento* può essere qualsiasi valore maggiore o uguale a 0, anche se i tassi normalmente usati sono: 0,25, 1,50, 1,75 e 2.

commutatore è un argomento facoltativo da includere per evitare che **@VDB** passi all'ammortamento a quote costanti per la vita utile residua. Generalmente, il passaggio al calcolo a quote costanti avviene quando l'ammortamento a quote costanti è maggiore di quello a quote proporzionali.

commutatore	@VDB
0	passa automaticamente all'ammortamento a quote costanti quando questo è maggiore di quello a quote proporzionali
1	non passa mai all'ammortamento a quote costanti

Non è consentito l'uso di un argomento facoltativo in assenza di quelli che lo precedono.

NOTE

Il metodo a quote proporzionali con tasso variabile mantiene un tasso costante di ammortamento finché il valore di *realizzo* di un bene non scende al di sotto del risultato della seguente equazione:

$(\text{valore contabile} * ((1 - (\text{tasso} / \text{vita})) \text{vita}))$

dove valore contabile = costo - realizzo - ammortamento precedente.

A questo punto 1-2-3 passa all'ammortamento a quote costanti per la vita utile residua del bene allo scopo di evitare una sopravvalutazione del valore di *realizzo*. Passando all'ammortamento a quote costanti, 1-2-3 ritocca, quando necessario, il risultato di **@VDB** per garantire che l'ammortamento totale per la vita del bene sia uguale al costo del bene stesso meno il suo valore di *realizzo*.

Esempi

Nel seguente esempio si calcola l'ammortamento di una macchina per ufficio acquistata a metà del primo trimestre dell'anno fiscale al prezzo di 10.000.000 di lire. La vita utile del bene è di 10 anni e il valore di *realizzo* è pari a 600.000 lire. Le seguenti formule calcolano le quote di ammortamento per ciascuno dei 10 anni, utilizzando il metodo a quote proporzionali con tasso variabile e presupponendo un tasso di ammortamento del 150%. Va notato che il passaggio all'ammortamento a quote costanti avviene nel sesto anno di vita.

@VDB(10000000;600000;10;0;0,875;1,5)	= L. 1.312.500
@VDB(10000000;600000;10;0,875;1,875;1,5)	= L. 1.303.130
@VDB(10000000;600000;10;1,875;2,875;1,5)	= L. 1.107.660
@VDB(10000000;600000;10;2,875;3,875;1,5)	= L. 941.510
@VDB(10000000;600000;10;3,875;4,875;1,5)	= L. 800.280
@VDB(10000000;600000;10;4,875;5,875;1,5)	= L. 767.790
@VDB(10000000;600000;10;5,875;6,875;1,5)	= L. 767.790
@VDB(10000000;600000;10;6,875;7,875;1,5)	= L. 767.790
@VDB(10000000;600000;10;7,875;8,875;1,5)	= L. 767.790
@VDB(10000000;600000;10;8,875;9,875;1,5)	= L. 767.790
@VDB(10000000;600000;10;9,875;10;1,5)	= L. <u>95.970</u>
Ammortamento totale (costo meno valore di <i>realizzo</i>)	L. 9.400.000

Funzioni @ affini

@DDB calcola l'ammortamento di un bene secondo il metodo a doppie quote proporzionali ai valori residui. @AQC usa il metodo a quote costanti, mentre @SYD utilizza il metodo della somma degli anni di prevista durata (metodo "americano").

@SCANSVER

@SCANSVER(*x*; *zona*; *scarto_colonna*) determina il contenuto della cella situata nella colonna specificata di una tabella di consultazione verticale.

Argomenti

x può essere un valore o una stringa, a seconda del contenuto della prima colonna della tabella di consultazione verticale.

Se la prima colonna contiene	<i>x</i> può essere
valori	qualsiasi valore maggiore o uguale al primo valore di <i>zona</i> . Se <i>x</i> è minore del primo valore della <i>zona</i> , @SCANSVER genera ERR. Se <i>x</i> è maggiore dell'ultimo valore della prima colonna della <i>zona</i> , @SCANSVER si ferma all'ultima cella della colonna specificata da <i>scarto_colonna</i> , riportandone il contenuto.
etichette	una stringa di testo racchiusa fra virgolette (" "), una formula che genera una stringa di testo, oppure l'indirizzo o il nome di una cella contenente un'etichetta o una formula che genera un'etichetta. Se <i>x</i> non corrisponde esattamente al contenuto di una cella nella prima colonna di <i>zona</i> , @SCANSVER genera ERR.

zona rappresenta la posizione della tabella di consultazione verticale; può essere un indirizzo o un nome di zona qualsiasi. Se è una zona tridimensionale, 1-2-3 usa solo il primo foglio di lavoro della zona.

scarto_colonna rappresenta lo scarto corrispondente alla posizione in cui si trova la colonna nella *zona*.

NOTA

@SCANSVER confronta *x* con ciascuna cella della prima colonna di una tabella. Quando viene localizzata una cella nella prima colonna che contiene *x* (oppure, se *x* è un valore, il valore più vicino purché non maggiore di *x*), 1-2-3 si sposta lungo la riga per il numero di colonne specificato da *scarto_colonna*, riportandone il contenuto.

Esempi

@SCANSVER

Funzioni @ affini

@SCANSOR ricerca il contenuto di una cella in una tabella di consultazione orizzontale. @IND individua il contenuto di una cella quando si specifica lo scarto della colonna e della riga. @SCELTA sostituisce una tabella di consultazione che occupa una sola riga. CORRISP riporta la posizione relativa di una cella con il contenuto specificato. INDX riporta il contenuto di una cella specificata dai titoli di colonna, riga e foglio di lavoro. @SCANSMAX produce un referimento assoluto alla cella che contiene il valore più alto di una serie di zone. @SCANSMIN genera un referimento assoluto alla cella che contiene il valore inferiore di una serie di zone.

@GSETT

@GSETT(*data*) estrae il giorno della settimana dalla *data*, visualizzandolo sotto forma di numero intero compreso tra 0 (lunedì) e 6 (domenica).

Argomenti

data è un valore data.

Esempi

@GSETT(@DATA(91;7;3)) = 2, mercoledì

Funzioni @ affini

@MESE calcola il mese estraendolo da un valore data. @ANNO calcola l'anno estraendolo da un valore data.

Esempio: @VAR e @VARC

La tabella sottostante riporta i dati relativi all'altezza e al peso di dieci soggetti selezionati a caso. Si vuole determinare le variazioni di peso:

$$@\text{VAR}(B2..B11) = 38,77462$$

Supponendo invece che i soggetti esaminati rappresentino un campione selezionato a caso da un più ampio gruppo, si userebbe la seguente funzione:

$$@\text{VARC}(B2..B11) = 43,08292$$

A	-----	A	-----	B	--
1		ALTEZZA (cm)		PESO (kg)	
2		190,50		72,73	
3		187,96		86,36	
4		175,26		68,18	
5		175,26		76,37	
6		180,34		77,27	
7		180,34		72,73	
8		187,96		75,00	
9		172,72		68,18	
10		177,80		70,46	
11		179,07		86,36	

Esempio: @SCANSVER

Nella seguente tabella di consultazione verticale denominata TABELLA_RETTA (in A3..E11) viene elencata la retta da pagare mensilmente in funzione del reddito e della categoria in cui rientra la famiglia dell'alunno.

@SCANSVER(52200000;TABELLA_RETTA;1), immessa in una cella avente il formato Valuta con 0 cifre decimali, genera il valore L.170.000, ossia la retta corrispondente al reddito più vicino ma non superiore a L. 52.200.000

A	-----	A	-----	B	-----	C	-----	D	-----	E	--
1				----- Categoria per determinazione retta-----							
2	Reddito<=	1		2		3		4			
3	L. 18.700.000	L. 60.000		L. 65.000		L. 75.000		L. 85.000			
4	L. 37.000.000	L. 90.000		L. 95.000		L. 105.000		L. 115.000			
5	L. 43.600.000	L. 125.000		L. 135.000		L. 140.000		L. 150.000			
6	L. 52.200.000	L. 170.000		L. 185.000		L. 195.000		L. 210.000			
7	L. 61.000.000	L. 230.000		L. 250.000		L. 270.000		L. 285.000			

@ANNO

@ANNO(*valore_data*:[*tipo*]) estrae l'anno, ossia un numero intero compreso tra 0 (1900) e 199 (2099), da un *valore_data*.

Argomenti

valore_data può essere un numero intero oppure l'indirizzo o il nome di una cella contenente un numero intero compreso tra 1 (1 gennaio 1900) e 73050 (31 dicembre 2099).

tipo è il numero 0 o 1. Se *tipo* è 1, **@ANNO** genera l'anno nel formato a quattro cifre. Se *tipo* è uguale a 0 o viene omissso, la funzione **@ANNO** genera lo scarto dell'anno rispetto all'anno 1900 (ad esempio, **@ANNO** genera 123 per rappresentare l'anno 2023).

NOTA

@ANNO può anche costituire l'argomento *anno* di altre funzioni **@** di data basate su date calcolate in precedenza.

Esempi

@ANNO(20181) = 55, poiché il valore data 20181 corrisponde alla data 02-Apr-55

@ANNO(**@ADESSO**) = l'anno corrente

@ANNO(**@ORDDATA**("14-Feb-91")) = 91

Funzioni @ affini

@GIORNO estrae il giorno del mese (da 1 a 31), mentre **@MESE** estrae il mese (da 1 a 12) da un valore data.

@RENDIMENTO

@RENDIMENTO(*liquidazione*; *scadenza*; *cedola*; *prezzo*; [*rimborso*]; [*frequenza*]; [*base*]) calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi.

Argomenti

liquidazione è il giorno dei compensi di un titolo e consiste in un valore data.

scadenza è il giorno in cui viene rimborsato un titolo; deve essere un valore data. Se *scadenza* è inferiore o uguale a *liquidazione*, la funzione @PREZZO genera ERR.

cedola è il tasso nominale annuale di un titolo e consiste in un valore positivo o uguale a zero.

prezzo è il prezzo del titolo per un valore nominale di \$100 e può essere qualsiasi valore positivo.

rimborso è un argomento facoltativo che specifica il valore di rimborso di un titolo con valore nominale pari a \$100. Questo argomento deve essere costituito da un valore positivo o zero. In caso di omissione, 1-2-3 usa un rimborso pari a 100.

frequenza è un argomento facoltativo che indica il numero di pagamenti delle cedole previsti in un anno.

<i>frequenza</i>	frequenza dei pagamenti delle cedole
1	annuale
2	semestrale; impostazione predefinita in caso di omissione dell'argomento
4	trimestrale
12	mensile

base è un argomento facoltativo che indica il tipo di base per il conteggio dei giorni da usare.

<i>base</i>	base per il conteggio dei giorni
0	30/360; impostazione predefinita in caso di omissione dell'argomento
1	effettivo/effettivo
2	effettivo/360
3	effettivo/365
4	europeo 30/360

Non è consentito usare un argomento facoltativo senza specificare quelli che lo precedono.

Esempi

Un'obbligazione ha come data di liquidazione il 1° luglio 1993 e come data di scadenza il 1° dicembre 1998. Il tasso nominale d'interesse semestrale è pari al 5,50%. Il costo è di \$99,50, il valore di rimborso è di \$100 e la base per il conteggio dei giorni è 30/360.

@RENDIMENTO(@DATA(93;7;1);@DATA(98;12;1);0,055;99,5;100;2;0) = 0,056072

Funzioni @ affini

@MATURATO calcola gli interessi maturati sui titoli che prevedono pagamenti periodici degli interessi. @PREZZO calcola il prezzo dei titoli con pagamento periodico degli interessi e valore nominale pari a \$100. @DURATA calcola la durata annua e @DURATAM calcola la durata annua modificata di un titolo con pagamento periodico degli interessi.

@RENDIMENTO2 calcola il rendimento di titoli che prevedono il pagamento periodico degli interessi, secondo le convenzioni giapponesi.

