What Xess functions are available?

What Xess functions are available?

A list of functions is available in Xess and information on how to use them are available by clicking on "Help" within Xess. A list is also included here.

```
@@(R)           Reference to the cell or range R

@ABS(X)         Returns the absolute value of X
                X - a numeric value

@ACCRINT(I,Ft,S,R,P,F[,B])
                Returns the accrued interest for a security that pays
                periodic interest.  Accrued interest is the amount the
                buyer must compensate the seller for the portion of
                the next coupon interest payment the seller has earned
                but will not receive from the issuer.
                I - issue date
                Ft - first coupon date of the security.
                S - settlement date
                R - annual coupon rate
                P - par value of the security
                F - number of coupon payments per year
                B - day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365
                Example:
                A security has the following terms: 4/1/93 settlement date,
                6/1/93 maturity date, 10% semiannual coupon, $1000 par
                value, 30/360 bais
                @accrint(34060,34121,10%,1000,2,0) = 33.333

@ACCRINTM(I,S,R,P,[B])
                Returns the accrued interest for a security that pays
                interest at maturity
                I - the issue date
                S - the settlement date
                R - the annual coupon rate
                P - the par value of the security
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@ACOS(X)        Returns the arc cosine of X
                X - a numeric value between -1 and 1

@ACOSH(X)       Returns the hyperbolic arc cosine of X
                X - a numeric value greater than or equal to 1

@ALARM(X,S)     If X evaluates non-zero, then the string S is evaluated
                as an expression, and the terminal beeps. The return value
                is set to the result of evaluating the S expression. If
                X evaluates to zero, S is not evaluated and the return
                value is zero. See also @SALARM function below.
                e.g. @ALARM(pressure > 500, "@REMOTE_FN(Pressure)")
```

```
@AND(...)       Returns 1 if all arguments are 1; 0 if any arguments are 0;
                -1 if any arguments are neither 0 nor 1.
                argument list - any combination of numbers, cells or ranges

@ANNOTATE(S,X,Y[,H[,V]])
                Generates a string suitable for use as an Xess graph
                annotation, in which case it will cause the string S
                to be displayed at position (X,Y) within the graph.  The
                optional parameters H and V specify how the string S
                will be justified with respect to the point (X,Y).
                H and V represnet horizontal and vertical justication values.
                (H,V)  ->   X,Y Placement
                (0,0)       bottom left corner of S (the default).
                (0,1)       top left corner of S.
                (1,0)       bottom right corner of S.
                (1,1)       top right corner of S.

                S - a string value
                X,Y - graph coordinates
                H,V - optional text justification parameters
                Example:
                @ANNOTATE("Average",5/2,12,1,0.5) = "(2.5,12,1,0.5)Average"

@ASIN(X)        Returns the arc sine of X
                X - a numeric value between -1 and 1

@ASINH(X)       Returns the hyperbolic arc sine of X
                X - a numeric value

@ATAN(X)        Returns the 2-quadrant arc tangent of X
                X - a numeric value

@ATANH(X)       Returns the hyperbolic arc tangent of X
                X - a numeric value greater than -1 and less
                than 1.

@ATAN2(X,Y)     Returns the 4-quadrant arc tangent of Y/X
                X, Y - numeric values, one of them must be non-zero

@AVG(...)       Returns the average (mean) of its arguments
                argumentlist - any combination of numbers, cells or ranges

@CAVG(...,C)    Returns the conditional average (mean) of its arguments
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@CCOUNT(...,C)  Returns a conditional count of its non-blank arguments
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@CEIL(X)        Returns the smallest integer greater than or equal to X
                X - a numeric value

@CELLREF(C,R)   Returns the value contained in cell C,R.
                C - Column, an integer value
                R - Row, an integer value

@CHAR(N)        Returns the character represented by N in the ASCII
                character set.
                N - an integer value between 1 and 255.

@CHOOSE(N,...)  Returns the value at the Nth place in the argument
                list.
                N - an integer.
                @CHOOSE(2,"mary","sue","beth") ==> "sue"

@CLEAN(S)       Returns a "printable" representation of string S by
                stripping out unprintable characters.

@CMAX(...,C)    Returns the conditional maximum of its arguments
```

```
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@CMIN(...,C)    Returns the conditional minimum of its arguments
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@CODE(S)        Returns the ASCII code for the first character in string S.
                S - a string value

@COL(C)         Returns the column associated with cell C, or the
                left-most column if C is a range.  Returns the column
                of the current cell if C is omitted.

@COLS(R)        Returns the number of columns in range R.

@CORR(R1,R2)    Returns Pearson's product-moment correlation coefficient for
                the paired data in ranges R1 and R2
                R1, R2 must have the same dimensions

@COS(X)         Returns the cosine of X
                X - a numeric value

@COSH(X)        Returns the hyperbolic cosine of X
                X - a numeric value

@COUNT(...)     A count of its non-blank arguments
                argumentlist - any combination of numbers, cells or ranges

@CCOUNT(...,C)  Returns a conditional count of its non-blank arguments
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@COUPDAYBS(S,M,F,[B])
                Returns the number of days between the beginning of the
                coupon period to the settlement date.
                S - the settlement date
                M - the maturity date
                F - the number of coupon payments per year
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@COUPDAYS(S,M,F,[B])
                Returns the number of days in the coupon period that
                the settlement date is in.
                S - the settlement date
                M - the maturity date
                F - the number of coupon payments per year
                B - the day count basis to be used
                    0   30/360
                    1   actual/actual
                    2   actual/360
                    3   actual/365

@COUPDAYSNC(S,M,F,[B])
                Returns the number of days between the settlement date
                and the next coupon date.
                S - the settlement date
                M - the maturity date
                F - the number of coupon payments per year
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@COUPNCD(S,M,F,[B])
```

```
                Returns the next coupon date after the settlement date.
                S - the settlement date.
                M - the maturity date
                F - the number of coupon payments per year
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@COUPNUM(S,M,F,[B])
                Returns the number of coupon payments between the
                settlement date and munurity date.
                S - settlement date
                M - maturity date
                F - the number of coupon payments per year
                B - day count basis
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@COUPPCD(S,M,F,[B])
                Returns the previous (most recent) coupon date before
                S - the settlement date
                M - the maturity date
                F - the number of coupon payments per year
                B - day count basis
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@CSTD(...,C)    Returns the conditional standard deviation (N weighting) of
                its arguments
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@CSTDS(...,C)   Returns the conditional sample deviation (N-1 weighting)
                of its arguments.
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@CSUM(...,C)    Returns the conditional sum of its arguments
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@CTERM(R,FV,PV) Returns the number of compounding periods for an investment
                R - Rate of return ( e.g. .07 for 7 %)
                FV - future value of the investment
                PV - present value of the investment

@CUMIPMT(R,N,P,S,E,T)
                Returns the cumulative interest paid between S
                (start) and E (end) on a loan.
                R - Rate, a number
                N - Number of payments, an integer > 0
                P - Present Value
                S - Start of loan
                E - End of loan
                T - Type is the timing of the loan: 0 means payment
                at the end of each period, and 1 means payment at
                the beginning of each period.

@CUMPRINC(R,N,P,S,E,T)
                Returns the cumulative principle paid between S
                (start) and E (end) on a loan.
                R - Rate, a number
                N - Number of payments, an integer > 0
                P - Present Value
                S - Start of loan
                E - End of loan
                T - Type is the timing of the loan: 0 means payment
                at the end of each period, and 1 means payment at
                the beginning of each period.

@CVAR(...,C)    Returns the conditional population variance (N weighting)
                of its arguments
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@CVARS(...,C)   Returns the conditional sample variance (N-1 weighting)
                of its arguments
                argumentlist - any combination of numbers, cells, or ranges
                C - a constraint expression

@DATE(Y,M,D)    returns the date value for year Y, month M, and date D.
                Y - year
                M - month
                D - day

@DATEVALUE(S)   Returns the corresponding date value for a given string S.
                S - string value of a date.
                Examples:
                @DATEVALUE("September-91")
                @DATEVALUE("3/24/54")

@DAY(DT)        Returns the day number in the date/time value DT.
                DT - a date/time value

@DAYS360(S,E)   Returns the number of days between two dates, based
                on a 30-month/360-day year.
                S,E - date/time values
                @DAYS360(@DATEVALUE("16-NOV-93"),@DATEVALUE("17-NOV-93"))

@DB(C,S,L,N,M)  Returns the fixed-declining real depreciation of an
                asset for a specified period.
                C - initial Cost of asset
                S - Salvage value of asset
                L - Number of periods over which the asset is
                depreciating (Life)
                N - Number of periods over which to calculate the
                depreciation.
                M - (optional) Number of Months in the first year,
                which is assumed to be 12 if omitted.

@DDB(C,S,L,N)   Returns the double-declining depreciation allowance
                C - cost
                S - salvage value
                L - allowable value
                N - number of depreciation periods

@DEGREES(X)     Returns 180/pi times X, converting radians to degrees.
                X - a numeric value

@DET(M)         Returns the determinant of the matrix range M, which
                must be a square matrix.
                M - a range, which must contain an equal number of
                rows and columns.

@DFT(R)         The Discrete Fourier Transform of the range R.
                R must represent a real vector (either its row or column
                dimension must be 1) or a complex vector (either its row
                or column dimension must be 2). The result is always a
                complex vector.
                R - a range representing the vector to be transformed.

@DISC(S,M,P,R,B)
                Returns the discount rate of a security.
                S - Settlement date
                M - Maturity
                P - Price per $100
                R - Redemption value per $100
                B - Basis is the standard day count basis

@DOLLARDE(D,F)  Returns the amount in Dollar Decimal form of an amount
                expressed in fractional form.
                D - Dollars (in decimal form)
                F - Fraction (denominator of fraction)

@DOLLARFR(D,F)  Returns the amount in Dollar Fractional form of an
                amount expressed in decimal form.
                D - Dollars (in fractional form)
                F - Fraction (denominator of fraction)

@DOT(R1,R2)     Returns the dot product of the vectors R1 and R2
                R1, R2 - each a range, which must contain an equal
                number of rows and columns, or must be a dimension
                1 x n (row vector) or n x 1 (column vector).

@DURATION(S,M,C,Y,F,B)
                Returns the Macauley duration for an assumed par
                value of $100.
                S - Settlement date, a date/time value
                M - Maturity date, a date/time value
                C - Annual Coupon rate
                Y - Annual Yield
                F - Frequency -- number of payments per year.
                B - Basis is the day count basis

@EDATE(S,M)     Returns a date/time value representing the date M
                months after S, if M is positive, and M months before
                S, if M is negative.  The date returned will be on the
                same day of the month as S.
                S - Start date, a date/time value
                M - Months, an integer

@EFFECT(R,P)    Returns the effective annual interest rate.
                R - Nominal Rate
                P - Number of compounding Periods per year

@EIGEN(M)       Generates the eigenvalues of the matrix M, which must be square
                and symmetric.
                M - a matrix which must be square and symmetric.

@EOMONTH(S,M)   Returns a date/time value representing the date M
                months after S, if M is positive, and M months before
                S, if M is negative.  The date returned will be on the
                last day of the month.
                S - Start date, a date/time value
                M - Months, an integer

@ERF(L[,U])     Returns the error function integrated between L (lower
                limit) and U (upper limit).  If U is omitted, @ERF
                integrates between 0 and L.
                L, R - a non-negative numeric value

@ERFC(L)        Returns the comlementary error function integrated
                between L and infinity.  @ERFC(L)=1-@ERF(L).
                L - a non-negative numeric value

@EXACT(S1,S2)   Returns 1 if S1 exactly matches string S2; otherwise, 0
                S1, S2 - string values

@EXP(X)         Returns e raised to the X power
                X - a numeric value representing exponentiation

@F(M,N,F)       Returns the integral of Snedecor's F-distribution with M and N
                degrees of freedom from minus infinity to F
                M,N,F - numeric values
```

```
@FACT(X)        Returns X * (X-1) * ... * 1 (X factorial, or X!)
                X is a numeric value.

@FALSE          Returns the logical value 0

@FFT(R)         Generates the Discrete Fourier Transform of the range R
                using a Fast Fourier Transform algorithm.  The length of
                the vector must be a power of 2. @FFT will produce the
                same results as @DFT, but is much faster (N*log(N)) times
                as opposed to N**2 times.
                R - a range representing the vector to be transformed.

@FIND(S1,S2,N)  Returns the index of the first occurrence of S1 in S2
                S1, S2 - String values
                N - a numeric value

@FILEEXISTS(S)  Returns 1 if file S can be opened to read; otherwise 0
                S - a string value

@FIND(I,T,N)    Returns the index of the first occurrence of string
                I in string T, starting at search position N in
                string T.  Note that the index of a string starts
                with 0.
                @FIND("bc","abcdefg",0)=1

@FLOOR(X)       Returns the largest integer less than or equal to X
                X - a numeric value

@FORECAST(X,R1,R2)
                Returns a predicted Y value for X when R1 and R2 are
                known x values and y values respectively

@FORMAT(F,N,X)  Returns the string formed by formatting the value X using
                the Xess format code F and precision N
                F - Xess format code
                N - the precision level, from 0 to 15
                X - a numeric value

@FRAC(X)        Returns the fractional portion of X
                X - a numeric value

@FREQUENCY(R, B)
                Returns a frequency distribution for a set of values R
                with a set of intervals B.
                R - a range of values on which frequencies will be counted
                B - a range of intervals used to group values in R
                @FREQUENCY(A1..A8, B1..B2)={3,2,2}
                where A1..A8 = {70,79,80,61,83,93,88,97} and
                      B1..B2 ={80,90}

@FTEST(R1,R2)   Returns the significance level (alpha) of the two-sided
                F-test on the variances of the data specified by
                ranges R1 and R2

@FV(P,R,N)      Returns the future value of an annuity
                P - periodic payment
                R - interest rate
                N - number of periods

@FVSCHEDULE(P,R)
                Returns the future value of an initial principle
                after compounding by a series of interest rates.
                P - Principal, or Present value
                R - An array of interest rates by which to compound

@GAMMA(X)       Returns the value of the gamma function evaluated at X
                X - a numeric value which must be non-negative

@GMEAN(...)     Returns the geometric mean of its arguments
```

```
                argumentlist -- any combination of numbers, cells,
                or ranges.

@GRAND          Returns a 12th-degree binomial approximation to a Gaussian
                random number with zero mean and unit variance

@HEXTONUM(S)    Returns the numeric value for the hexadecimal
                interpretation of S.
                S - a string value

@HLOOKUP(X,R,N) Searches the first row (known as the index row)
                in range R for the numeric or string value which "matches"
                X, and returs the value N rows below in the column
                where the match is found.  The index row is searched from
                left to right.  If X is a string value, an exact match must
                be found or an error is returned.  If X is a numeric value,
                the matching column is determined by the following rules:
                1) Strings and blank cells in the index row are ignored.
                2) If the first value in the index row is greater than X,
                   an error is returned.
                3) Searching stops when a numeric value which is greater than
                   or equal to X is found in the index row.  If the value
                   found is greater than X, the preceding column is the
                   matching column.
                4) If there are no numeric values in the index row which are
                   greater than or equal to X, the last column in R is
                   considered to be the matching column.

@HMEAN(...)     Returns the harmonic mean of its arguments
                argumentlist - any combination of numbers, cells, or ranges.

@HOUR(DT)       Returns the hour value (0-23) of date/time value DT.
                DT - a date/time value

@IF(X,T,F)      Returns the value of T if X evaluates to non-zero,
                or F if X evaluates to zero
                X - numeric value
                T,F - numeric or string values, cell or range references

@INIT(X1,X2)    Returns its first argument on the first recalculation pass
                and its second argument on all subsequent recalculation
                passes when Xess is performing iterative calculations.
                X1 - initial value
                X2 - iterative or subsequent value

@INT(X)         Returns the integer portion of X
                X - a numeric value

@INTERP2D(R1,R2,N)
                Returns interporlation value for a 2 dimensional
                vector.
                R1 - a range reference points to  x values of the vector;
                R2 - a range reference points to  y values of the
                     vector.  XR and YR have to be same size.
                N  - a numerical value or range for which the
                     interporlation value will be based on.

@INTERP3D(R,X,Y)
                Returns interporlation value for a 3 dimensional
                vector.  X and Y ranges must be the same size.
                R - a range reference points to x, y and z values of
                    the vector;
                X, Y  - numerical values or ranges for which the
                    interporlation value will be based on.

@INTRATE(S,M,I,R,B)
                Returns the interest rate of a fully invested security
                S - Settlement date
                M - Maturity
                I - Investment amount
                R - Redemption amount
                B - Basis

@INVDFT(R)      Generates the inverse of the Discrete Fourier Transform
                of the range R. R must represent a real vector (either
                its row or column dimension must be 1) or a complex
                vector (either its row or column dimension must be 2).
                The result is always a complex vector.
                R - a range representing the vector to be inverse transformed.

@INVERT(M)      Generates the inverse of matrix M, which must be square.
                M - a square matrix.

@INVFFT(R)      Generates the inverse of the Discrete Fourier
                Transform of the range R using a Fast Fourier
                Transform algorithm. The length of the vector
                must be a power of 2.  @INVFFT will produce the
                same results as @INVDFT, but is much faster
                (N*log(N)) time as opposed to N**2 time.
                R - a range representing the vector to be inverse
                transformed.

@IPMT(R,P,NP,PV,FV[,T])
                Returns the interest payment for a specific period for
                an investment based on periodic, constant payments and
                a constant interest rate.
                R - interest rate per period
                P - the period for which the interest will be calculated
                NP - the total number of payment periods.  Make sure
                     the units used be R and NP are consistent.  For
                     example, for a 5-year loan with 12% annual
                     interest, if you make payment monthly, use 12%/12
                     for (monthly) R and 5*12 for NP. If you mkae
                     annual payment on the same loan, use 12% for R
                     and 5 for NP.
                PV - present value of the investment.
                FV - future value or a cash balance you would like to
                     attain at the end of the last period.
                T - timing of the payment
                    0   payment is made at the end of the period.
                    1   payment is made at the beginning of the period.

@IRR(G,F)       Returns the internal rate of return on an investment
                Related functions: @MIRR and @XIRR
                G - a numeric value (an initial "guess" G)
                F - cash flow, represented by a range. It must contain
                at least two cash flow values

@ISERROR(X)     Returns 1 if X "contains" an error, otherwise 0.
                X is an expression, which may include references to
                other cells.  X "contains" an error if the expression
                causes an error, or if it refers to any cells which
                have errors.

@ISNUMBER(X)    Returns 1 if X is a numeric value; otherwise 0
                X - a numeric or a string value

@ISSTRING(X)    Returns 1 if X is a string value; otherwise 0
                X - a numeric or a string value

@LARGE(R,N)     Returns the Nth largest datum in range R.
                R - Range of data
                N - Integer specifying which datum to choose.
                @LARGE(A1..A10,3) = 7
                where A1..A10={38,3,9,45,7,90,17,2,75,12}

@LEFT(S,N)      Returns the string composed of the leftmost N characters of S
                If N is greater than or equal to the length of S, then the
                entire string S is returned
                S - a string value
                N - a numeric value

@LENGTH(S)      Returns the number of characters in S
                S - a string value

@LINCOEF(X,Y)   Generates the straight line least squares fit.
                X - a range representing a column vector of independent
                variable values
                Y - a range representing a column vector of dependent variable
                variable values

@LINFIT(X,Y)    Returns the straight line least squares fit.
                X - a column vector of values for the independent variable.
                Y - a column vector of values for the dependent variable.
                This function is equivalent to @POLYFIT(X,Y,1).

@LLS(X,Y)       (for all version prior to Xess 3.1.2b)
                The linear least squares solution X to the overdetermined
                system of equations AX=Y.
                X - a range representing the coefficient matrix which
                must intercept the origin (rightmost column of X must
                be all 1's)
                Y - a column vector of values for the dependent variable

@LLS(X,Y,flag)  (Xess 3.1.2b and later)
                The linear least squares solution X to the overdetermined
                system of equations AX=Y.
                X - a range representing the coefficient matrix
                Y - a column vector of values for the dependent variable
                flag =   1   (X data intercepts origins)
                         0   (X data does not intercept the origin)

@LN(X)          Returns the log base e of X
                X - a numeric value greater than 0

@LNGAMMA(X)     Returns the log base e of the gamma function evaluated at X
                X - a numeric value greater than 0

@LOG(X)         Returns the log of X
                X - a numeric value greater than 0

@LOG10(X)       Returns the log base 10 of X
                X - a numeric value greater than 0

@LOG2(X)        Returns the log base 2 of X
                X - a numeric value greater than 0

@LOWER(S)       The string S is converted to lower case
                S - a string value

@MATCH(V,R[,T]) Returns the relative position in range R of
                value V in a specific way.
                V - the value to be matched.  V can be a string,  a
                    numeric value.  If V is a string, T must be 0.  MATCH
                    allows regular expression type "wildcard".
                T - match type
                    T > 0 means find largest value in R <= V
                    T = 0 or omitted means find first value in R = V
                    T < 0 means find smallest value in R >= V

@MAX(...)       Returns the maximum of its arguments
                argumentlist - any combination of numbers, cells, or ranges.

@MDURATION(S,M,R,Y,F[,B])
                Returns the modified Macauley duration of a security
                assuming $100 face value.
                S - the settlement date
                M - the maturity date
                R - the security's annual coupon rate
                Y - the security's annual yield
                F - the number of coupon payments per year
                B - day count basis
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@MEDIAN(...)    Returns the median (middle value) of the range R1.

@MESSAGE(X,S)   If X evaluates non-zero, the string S is displayed on
                the status line and the terminal beeps. The return
                value of the function is set to the value of X.
                e.g. @MESSAGE(@HOUR(@NOW) > 17, "Time to go home!")
                Tells you that it is time to go home after 5:00 PM.

@MID(S,N1,N2)   Returns the string of length N2 that starts at position N1 in S.
                S - a string value

@MIN(...)       Returns the minimum of its arguments
                argumentlist - any combination of numbers, cells, or ranges.

@MINUTE(DT)     Returns the minute value (0-59) of date/time value DT.
                DT - a date/time value

@MIRR(V,F,R)    Returns the Modified Internal Return Rate for a range
                of periodic cash flows.
                V - A range of values representing periodic income
                (positive values) and payments (negative values)
                F - Finance rate, the interest paid on the payments
                R - Reinvestment rate, the rate of return on the income

@MMUL(M1,M2)    Generates the product of multiplying matrix M2 by matrix M1.
                M1 and M2 can be multiplied only if number of columns in
                M1 is equal to the number of rows in M2.
                M1, M2 - matrices.

@MOD(X,Y)       Returns the remainder of X/Y, with the same sign as X.
                This is NOT the modulus, see @MODULUS below for that.
                X, Y - numeric values

@MODE(...)      Returns the mode, or most frequently occurring datum,
                of all the arguments.  Empty cells and cells containing
                text are ignored.

@MODULUS(X,Y)   Returns the modulus of X/Y, which is always positive.
                X, Y - numeric values

@MONTH(DT)      Returns the number of the month in date/time value DT.
                DT - a date/time value

@MSQ(...)       Returns the mean of the squares of its arguments
                argumentlist - any combination of numbers, cells, or ranges.

@N(R)           Returns the numeric value of the top left cell in range R
                R - a range

@NAND(...)      0 if all arguments are 1; 1 if any arguments are 0;
                -1 if any arguments are neither 0 nor 1.
                argumentlist - any combination of numbers, cells or ranges

@NETWORKDAYS(S,E[,H])
                Returns the number of whole working days, starting
                at S and going to E, excluding days in H.
                S - Starting date, a date/time value
                E - Ending date, a date/time value
                H - A range of dates to exclude, such as holidays.

@NOMINAL(R,C)   Returns the nominal annual interest rate for an
                effective interest rate and the number of compounding
                periods each year.
                R - Effective interest rate
                C - Compounding periods per year

@NOR(...)       0 if any arguments are 1; 1 if all arguments are 0;
                -1 if any arguments are neither 0 nor 1.
                argumentlist - any combination of numbers, cells or ranges

NOT(X)          Returns 1 if X=0; 0 if X=1; otherwise -1
                argumentlist - any combination of numbers, cells or ranges
                X - a numeric value

@NOW            Returns the date/time value of the current system date
                and time.

@NPV(R,CF)      Returns the present value of a series of future cash
                flows at a given rate. Related functions: @MNPV, @XNPV.
                R - periodic interest rate
                CF- future cash flow series (must be a range)

@NUMTOHEX(X)    Returns the hexadecimal representation of the integer
                portion of X, for -2147483648 <= X <= 2147483647.
                X - a numeric value

@ODDFPRICE(S,M,FC,R,Y,RD,F[,B])
                Returns the price per $100 face value of a security
                with an odd (shor or long) first period.
                S - settlement date
                M - maturity date
                FC - first coupon date of the security
                R - annual coupon rate of the security
                Y - annual yield of the security
                RD - redemption value of the security at maturity per
                    $100 face value
                F - the number of coupon payments per year
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@ODDFYIELD(S,M,I,FC,R,PR,RD,F[,B])
                Returns the yield per of a security with an odd (short
                or long) first period.
                S - settlement date
                M - maturity date
                I - issue date
                FC - first coupon date of the security
                R - annual coupon rate of the security
                PR - the price of the security per $100 face value
                RD - redemption value of the security at maturity per
                     $100 face value
                F - the number of coupon payments per year
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@ODDLPRICE(S,M,LC,R,Y,RD,F,[B])
                Returns the price per $100 face value of a security
                with an odd (short or long) last period.
                S - settlement date
                M - maturity date
                LC - first coupon date of the security
                R - annual coupon rate of the security
                Y - annual yield of the security
                RD - redemption value of the security at maturity per
                     $100 face value
                F - the number of coupon payments per year
                B - the day count basis to be used
```

```
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@ODDLYIELD(S,M,LC,R,PR,RD,F,[B])
                Returns the yield per of a security with an odd (short
                or long) first period.
                S - settlement date
                M - maturity date
                LC - Last coupon date of the security
                R - annual coupon rate of the security
                PR - the price of the security per $100 face value
                RD - redemption value of the security at maturity per
                     $100 face value
                F - the number of coupon payments per year
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@OR(...)        Logical OR function.  Returns 1 if any argument in
                the list equals 1, 0 if not, and -1 if there is any
                argument in the list not equal to 0 or 1.

@PERCENTILE(R,N)
                Returns the datum from the range R which is at the Nth
                percentile in R.  Only numeric data in R are considered.
                The result will be interpolated if N is not a multiple
                of 1/(s-1), where s is the size of R.
                R - A range of data
                N - A numeric value between 0 and 1, inclusive.

@PERCENTRANK(R,D[,S])
                Returns the percentile rank of the datum D in range R,
                to significance S.
                R - Range of numeric data
                D - Datum to find
                S = Number of significant digits

@PERMUT(S,T)    Returns "S choose T", or the number of T objects that
                can be chosen from the set S, where order is significant.
                S - Number of objects to choose from
                T - Number of objects to be chosen

@PI             Returns a numeric approximation of PI.

@PLS(X,Y,d)     (for all version prior to Xess 3.1.2b)
                analyzes the least squares polynomial model. The output is
                identical to that of @LLS, with the polynomial coefficients
                listed in the order of decreasing degree.
                X - a range representing a row or column vector of independent
                    variable values which must intercept the origin
                Y - a range representing a row or column vector of dependent
                    variable values.
                d - polynomial degree in range 1 - 10

@PLS(X,Y,d,[,flag])  (Xess 3.1.2b and later)
                analyzes the least squares polynomial model. The output is
                identical to that of @LLS, with the polynomial coefficients
                listed in the order of decreasing degree.
                X - a range representing a row or column vector of independent
                    variable values.
                Y - a range representing a row or column vector of dependent
                    variable values.
                d - polynomial degree in range 1 - 10
                flag - omitted/1     X data intercepts the origin
                       0             X data does not intercept the origin
```

```
@PMT(P,R,N)     Returns the periodic payment for a loan, given the
                present value P and interest rate R, and the number
                of periods N.
                P,R,N - Numeric values

@POLY(X,...)    Returns the value of the Nth degree polynomial
                in X, where N is the number of data in the argument
                list.  The argument list may contain numbers and
                cell references, including ranges.  Any non-numeric
                argument causes an error.
                @POLY(2,3,4,5) = (3*(2**2)+4*2+5) = 25

@POLYCOEF(X,Y,d)
                Returns the least squares coefficients for the polynomial fit.
                X - a range representing a row or column vector of independent
                variable values.
                Y - a range representing a row or column vector of dependent
                variable values.
                d - polynomial degree in the range 1 to 10.

@POLYFIT(X,Y,d) Returns the least squares polynomial fit.
                X - a column vector of values for the independent variable.
                Y - a column vector of values for the dependent variable.
                d - the polynomial degree in the range of 1 to 10.

@PPMT(R,P,NP,PV,FV,T)
                Returns the payment on the principal for a specific
                period for an investment based on periodic, constant
                payments and a constant interest rate.
                R - interest rate per period
                P - the period for which the interest will be calculated
                NP - the total number of payment periods.  Make sure
                     the units used be R and NP are consistent.  For
                     example, for a 5-year loan with 12% annual
                     interest, if you make payment monthly, use 12%/12
                     for (monthly) R and 5*12 for NP. If you mkae
                     annual payment on the same loan, use 12% for R
                     and 5 for NP.
                PV - present value of the investment.
                FV - future value or a cash balance you would like to
                     attain at the end of the last period.
                T - timing of the payment
                    0   payment is made at the end of the period.
                    1   payment is made at the beginning of the period.

@PRICE(S,M,R,Y,RD,F[,B])
                Returns the price per $100 face value of a security
                that pays periodic interest
                S - settlement date
                M - maturity date
                R - annual coupon rate of the security
                Y - annual yield of the security
                RD - redemption value of the security at maturity per
                     $100 face value
                F - the number of coupon payments per year
                B - the day count basis to be used
                    0 or omitted       30/360
                    1                  actual/actual
                    2                  actual/360
                    3                  actual/365

@PRICEDISC(S,M,D,RD[,B])
                Returns the price per $100 face value of a discounted
                security.
                S - settlement date
                M - maturity date
                D - discount rate of the security
                RD - redemption value of the security at maturity per
                     $100 face value
                B - the day count basis to be used
```

```
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@PRICEMAT(S,M,I,R,Y[,B])
                Returns the price per $100 face value of a security
                that pays interest at maturity
                S - settlement date
                M - maturity date
                I - issue date
                R - annual coupon rate of the security
                Y - annual yield of the security
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@PRODUCT(...)   Multiplies all the numeric elements of the argument
                list together, ignoring empty and non-numeric values,
                and returns the product.

@PROPER(S)      Returns a copy of string S with the first letter of
                every word capitalized.

@PTTEST(R1,R2)  Returns the significance level (alpha) of the two-sided T-test
                for the paired samples contained in ranges R1 and R2
                R1 and R2 must have the same size.

@PV(P,R,N)      Returns the present value of an annuity given
                periodic payment P, interest rate R, and N
                periods.

@QUARTILE(R,Q)  Finds the quartile Q of the data in range R.
                Equivalent to @PERCENTILE(R, Q/4)
                R = Range of cells
                Q = Quartile as follows:
                         0      Minimum value
                         1      First quartile (25th percentile)
                         2      Second quartile (50th percentile)
                         3      Third quartile (75th percentile)
                         4      Maximum value

                @QUARTILE(A1..B4, 2) = 7.5
                where A1..B4 = {1,2,4,7,8,9,10,12}

@RADIANS(D)     Returns PI/180 * D, converting degrees to radians.
                D is a numeric value.

@RAND           Returns a uniformly distributed random number on
                the interval [0,1].  The number will change every
                time the spreadsheet is recalculated.

@RANGEREF(UC,UR,LC,LR)
                Returns a reference to the range described by
                the corner co-ordinates UC, UR, LC, LR.
                UC = Upper left column of the range
                UR = Upper left row of the range
                LC = Lower right column of the range
                LR = Lower right row of the range

@RANK(E,R[,O])  Returns the rank of a numeric argument E in the
                range R.  The rank of a number is its size relative to
                other values in the list, which is equivalent to it's
                position in the list after the list is sorted.  RANK
                gives duplicate numbers the same rank. The presence of
                duplicate numbers will affect the ranks of subsequent
                numbers.  For examples, if there are two rank of 3,
                the next rank will be 5.
                E - a numeric value whose rank you want to find
                R - a reference to a range of values.  Non-numeric
                    values will be ignored.
                O - a numeric value specifying the way the numbers to
                    be ranked. RANK ranks E as if R is in descending order
                    if O is 0 or omitted. Otherwise, it ranks E as if
                    R is in ascending order.
                @RANK(5, C2..C6) = 2
                @RANK(5, C2..C6, 1) = 3
                @RANK(7, C2..C6, 1) = 5
                where C2..C6 = {2,7,5,1,5}

@RATE(F,P,N)    Returns the interest rate required to go from present
                value P to future value F in N compounding periods.

@RECEIVED(S,M,I,D[,B])
                Returns the value at maturity of a fully invested
                security. Dates must be entered as a serial date value.
                S = Settlement date, date of purchase
                M = Maturity date.
                I = Investment amount
                D = Discount rate
                B = Day count basis as follows:
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@REGEX(R,S)     Returns True(1) if the strings R and S match, and
                and False (0) otherwise. This function is a regular
                expression pattern matcher and is similar to @EXACT(S1,S2)
                function, except @REGEX allows "wildcard" comparisons.
                R - A string representing a regular expression
                S - A string
                @REGEX("t.*e", "table") = True
                @REGEX("F[0-9]", "F3") = True

@REPEAT(S,N)    Returns a string containing N copies of string S.

@REPLACE(S1,N1,N2,S2)
                Returns the string formed by replacing the N2
                characters starting at position N1 in string
                S1 with string S2.
                @REPLACE("Expense Report",8,3,"Log") = "Expense Logort"
                @REPLACE("Expense Report",8,6,"Log") = "Expense Log"
                @REPLACE("ABC",8,3,"XYZ") = "XYZABC"

@RIGHT(S,N)     Returns the string composed of the rightmost N
                characters of S.  If N is greater than or
                equal to the length of S, then the entire string
                S is returned.

@RMS(...)       Returns the square root of the mean of squares of its
                arguments, which may be any combination of numbers, cells,
                and ranges.  Blank cells and cells containing strings
                are not counted.

@ROUND(X,N)     Returns the number X rounded to the number of decimal
                places specified by the integer N, for N between 15
                and -15, inclusive.
                @ROUND(@PI,2) = 3.14
                @ROUND(1234.5678,-2) = 1200

@ROW(C)         Returns the row index of the cell referenced by C,
                the top row of C if C is a range, or the current
                row if C is omitted.

@ROWS(R)        Returns the number of rows in the specified range R.

@S(R)           Returns the string value of the top left cell in range R.
```

```
                Returns a null string if the cell is blank or contains a
                numeric value.
                R - a range

@SALARM(X,S)    If X evaluates non-zero, then the string S is evaluated
                as an expression. The return value is set to the result
                of evaluating the S expression. If X evaluates to zero,
                S is not evaluated and the return value is zero. Does
                not beep like the @ALARM function above.
                e.g. @SALARM(pressure > 500, "@REMOTE_FN(Pressure)")

@SECOND(T)      Returns the seconds (0 - 59) component of the
                date/time value T.

@SIGMOID(X)     Returns the value of the sigmoid function 1/(1 + exp(-X))
                X - a numeric value

@SIN(X)         Returns the sine of X
                X - a numeric value

@SINH(X)        Returns the hyperbolic sine of X
                X - a numeric value

@SLN(C,S,L)     Returns the straight-line depreciation allowance
                given cost C, salvage value S, and allowable life L.

@SMALL(R,N)     Returns the Nth smallest number in range R.

@SQRT(X)        Returns the positive square root of X
                X - a positive numeric value

@SSE(...)       Returns the sum squared error of its arguments. This function
                equivalent to @VAR(...)/@COUNT(...)
                argumentlist - any combination of numbers, cells, or ranges

@SSQ(...)       Returns the sum of squares of its arguments
                argumentlist - any combination of numbers, cells, or ranges

@STD(...)       Returns the population standard deviation (N weighting) of its
                arguments
                argumentlist - any combination of numbers, cells, or ranges

@STDS(...)      Returns the population standard deviation (N-1 weighting) of
                its arguments
                argumentlist - any combination of numbers, cells, or ranges

@STRCAT(...)    Returns the concatenation of all its arguments.
                argumentlist - any combination of numbers, cells, or ranges

@STRING(X,N)    Returns the string representing the numeric value of X,
                to N decimal places
                X - a numeric value
                N - a numeric value between 0 and 15

@STRLEN(...)    Returns the total length of all strings in its arguments
                argumentlist - any combination of numbers, cells, or ranges

@SUM(...)       Returns the sum of its arguments
                argumentlist - any combination of numbers, cells, or ranges

@SUMPRODUCT(R1,R2)
                Returns the dot product of the vectors R1 and R2
                R1, R2 - each a range, which must contain an equal number of
                rows and columns, or must be a dimension 1 x n (row vector) or
                n x 1 (column vector).

@SYD(C,S,L,N)   Returns the "sum-of-years-digits" depreciation allowance
                given cost C, salvage value S, allowable life L, and
                N depreciation periods.
```

```
@T(N,T)         Returns the integral of Student's T-distribution with
                N degrees of freedom from minus infinity to T.
                N, T - numeric values

@TAN(X)         Returns the tangent of X
                X - a numeric value

@TANH(X)        Returns the hyperbolic tangent of X
                X - a numeric value

@TBILLEQ(S,M,D) Returns the bond-equivalent yield (BEY) for a Treasury Bill.
                If the term is one half-year or less, BEY is
                equivalent to a actural/365 simple interest rate.  If
                the term of the security is more than one-half year,
                BEY is equivalent to a semiannually compounded
                Treasury bond yield.
                S = Settlement date
                M = Maturity date
                D = Discount rate of Treasury Bill

@TBILLPRICE(S,M,D)
                Returns the price per $100 face value for a Tresury
                bill.
                S - settlement date
                M - maturity date
                D - discount rate of the Tresury bill

@TBILLYIELD(S,M,D)
                Returns the yield on a treasury bill, given settlement
                date S, maturity date M, and discount rate D.
                S = Settlement date
                M = Maturity date
                D = Discount rate of Treasury Bill

@TERM(P,R,FV)   Returns the number of payment periods for an investment
                P - amount of periodic payments
                R - interest rate
                FV - future value of the investment

@TIME(H,M,S)    Returns the time value represented as a fraction of a day,
                starting at midnight.  H (hours) must be between 0 and 23,
                M (minutes) and S (seconds) must be between 0 and 59.

@TIMEVALUE(S)   Returns the corresponding time value for a given string
                value S.
                Example:
                @TIMEVALUE("12:55:00 AM")
                @TIMEVALUE("0:55.00")

@TODAY          Returns the date value corresponding to the current
                system date.

@TRANSPOSE(M)   Generates the transpose of matrix M.

@TREND(NX, KX, KY)
                TREND fits a straight line to KX (known x's) and KY
                (known y's) using least square method, and then
                returns the y values along the line for NX (new x's)
                NX - new x values for which TREND will return the y-values
                KX - known x values
                KY - known y values

@TRIM(S)        Returns the string formed by removing leading, trailing,
                and consecutive embedded spaces from string S.

@TRUE           Returns the logical value 1 (true).

@TTEST(R,X)     Returns the significance level (alpha) of the two-sided single
                population T-test for the population samples contained
                in range R
                R - a range
                X - a numeric value

@TTEST2EV(R1,R2)
                Returns the significance level (alpha) of the two-sided dual
                population T-test for ranges R1 and R2, where their
                variances are equal

@TTEST2UV(R1,R2)
                Returns the significance level (alpha) of the two-sided single
                population T-test for ranges R1 and R2, where their
                variances are not equal

@UPPER(S)       Returns the string S converted to upper case.
                S - a string value

@VALUE(S)       Returns the numeric value represented by the string S
                or 0 if S does not represent a number.
                S - a string value

@VAR(...)       Returns the population variance (N weighting) of its arguments
                argumentlist - any combination of numbers, cells, or ranges

@VARS(...)      Returns the population variance (N-1 weighting) of its arguments
                argumentlist - any combination of numbers, cells, or ranges

@VDB(C,S,L,S,E)
                Returns the depreciation of an asset between two specific
                period using the fixed-declining balance method.
                C - the initial cost of the asset.
                S - salvage , the value at the end of the depreciation.
                L - life, the total number of periods over which the asset
                    is being depreciated.
                S - Starting period for the calculation.
                E - Ending period for the calculation.

@VECLEN(...)    Returns the square root of the sum of squares of its arguments
                argumentlist - any combination of numbers, cells or ranges

@VLOOKUP(X,R,N) Searches the first column (known as the index column)
                in range R for the numeric or string value which "matches"
                X, and returs the value N columns to the right in the row
                where the match is found.  The index column is searched from
                top to bottom.  If X is a string value, an exact match must
                be found or an error is returned.  If X is a numeric value,
                the matching row is determined by the following rules:
                1) Strings and blank cells in the index column are ignored.
                2) If the first value in the index column is greater than X,
                   an error is returned.
                3) Searching stops when a numeric value which is greater than
                   or equal to X is found in the index column.  If the value
                   found is greater than X, the preceding row is the matching
                   row.
                4) If there are no numeric values in the index column which are
                   greater than or equal to X, the last row in R is considered
                   to be the matching row.

@VSUM(...)      Returns the "visual sum" of its arguments, using precision and
                rounding of formatted cell values
                argumentlist - any combination of numbers, cells, or ranges

@WEEKDAY(D)     Returns an integer representing the day of the week
                on which the day D falls.  1 is Sunday, 7 is Saturday.

@WORKDAY(S,D[,H]) Returns the day that is D working days after day S,
                excluding weekends and all holidays specified as dates
                in range H.

@XIRR(G,V,D)    Returns the internal rate of return for a series of
                cash flows with variable intervals
                G - a guessed value of the result
                V - a range, contains a series of cash flows
                D - a range, contains the schedule of cash flows in V
                V and D must be one-dimentional ranges and have the
                same size.

@XNPV(R,V,D)    Returns the net present value for a series of cahs
                flows with variable intervals.
                R - discount rate to apply to the cash flows
                V - a range, contains a series of cash flows
                D - a range, contains the schedule of cash flows in V
                V and D must be one-dimentional ranges and have the
                same size.

@XOR(...)       Returns -1 if any arguments are UNKNOWN;
                1 if the total number of arguments with the value 1 is odd;
                0 if the total number of arguments with the value 1 is even.
                argumentlist - any combination of numbers, cells or ranges

@XVALUE(N,C)    Returns the value of cell C in sheet N.
                N - a string containing the pathname of another
                spreadsheet.
                C - a valid cell reference or a string containing a cell
                reference; you can specify a cell reference from the other
                sheet as well by using a cell name enclosed in quotes.

@XHLOOKUP(F,X,R,N)
                Searches the first row (known as the index row) in range R
                in file F for the numeric or string value which "matches"
                X, and returs the value N rows below in the column
                where the match is found.  The index row is searched from
                left to right.  If X is a string value, an exact match must
                be found or an error is returned.  If X is a numeric value,
                the matching column is determined by the following rules:
                1) Strings and blank cells in the index row are ignored.
                2) If the first value in the index row is greater than X,
                   an error is returned.
                3) Searching stops when a numeric value which is greater than
                   or equal to X is found in the index row.  If the value
                   found is greater than X, the preceding column is the
                   matching column.
                4) If there are no numeric values in the index row which are
                   greater than or equal to X, the last column in R is
                   considered to be the matching column.

@XVLOOKUP(F,X,R,N) Searches the first column (known as the index column)
                in range R in spreadsheet file F for the numeric or string
                value which "matches" X, and returns the value N columns to
                the right in the row where the match is found.

                The index column is searched from top to bottom.  If X is
                a string value, an exact match must be found or an error
                is returned.  If X is a numeric value, the matching row is
                determined by the following rules:
                1) Strings and blank cells in the index column are ignored.
                2) If the first value in the index column is greater than X,
                   an error is returned.
                3) Searching stops when a numeric value which is greater than
                   or equal to X is found in the index column.  If the value
                   found is greater than X, the preceding row is the matching
                   row.
                4) If there are no numeric values in the index column which are
                   greater than or equal to X, the last row in R is considered
                   to be the matching row.

@YEAR(DT)       Returns the four-digit year value of a date/time value DT.
                DT - a date/time value

@YEARFRAC(S,E[,B])
                Returns the year fraction representing the number of
                whole days between S (start date) and E (end date)
                S - a date value that represents the start date
                E - a date value that represents the end date
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@YIELD(S,M,R,PR,RD,F[,B])
                Returns the yield of a security that pays periodic
                interest
                Returns the yield per of a security with an odd (short
                or long) first period.
                S - settlement date
                M - maturity date
                R - annual coupon rate of the security
                PR - the price of the security per $100 face value
                RD - redemption value of the security at maturity per
                     $100 face value
                F - the number of coupon payments per year
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@YIELDDISC(S,M,PR,RD[,B])
                Returns the annual yield for a discounted security
                S - settlement date
                M - maturity date
                PR - the price of the security per $100 face value
                RD - redemption value of the security at maturity per
                     $100 face value
                B - the day count basis to be used
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
                    3                   actual/365

@YIELDMAT(S,M,I,R,PR[,B])
                Returns the annual yield of a security which pays interest at
                maturity.  All dates must be expressed as serial date values.
                S = settlement date
                M = maturity date
                I = issue date
                R = interest rate at date of issue
                PR = the price of the security per $100 face value
                B = the day count basis to be used:
                    0 or omitted        30/360
                    1                   actual/actual
                    2                   actual/360
```

```
            3                    actual/365
```