Ages!1.5 Technical Reference

- Draft April 30th 2009 -

Table of Contents
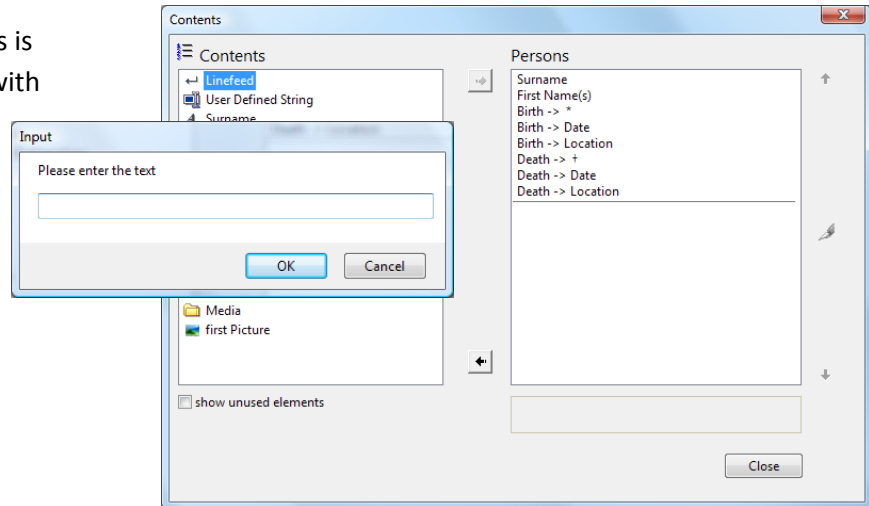
Using Calculations in Ages! Layouts

The Ages! layout engine can be used to calculate certain elements of a report or chart. To use this feature, add a "User Defined String" to your list of fields. If you enter any simple text, it will be treated as static text.

If the first character of your input is an equal sign (=), everything that follows it will be treated as a formula.

The syntax of these formulas is designed to be compatible with Microsoft Excel.

Very basic example:

=3+4

will return

7

see section A for a list of operators

Function example:

=MID("Text",2,2)

will return

ex

see section B for a complete list of supported functions

Data access example

=BIRT.YEAR

will retrieve the current invididual's birth year: BIRT retrieves the event object, which in turn has the property YEAR

1965

see section C for a list of data objects and properties

Array element example

=OCCU[1].TYPE

will retrieve the current invididual's first occupation.

Builder

The functions available are a subset of the Excel 2007 functions as described below. For details, please refer to the Microsoft Excel 2007 function reference.
http://office.microsoft.com/en-us/excel/CH100645021033.aspx (English)

Localized versions of above document exist, such as:
http://office.microsoft.com/de-de/excel/CH100645021031.aspx (German)

Please note: Ages! will **not** understand the localized (German or otherwise) function names, numbers have to use the English format (pi=3.14 , not 3,14)

A - List of operators

+	add
-	subtract
*	multiply
/	divide
^	exponential
&	concatenate two strings
-	unary minus

B - List of supported functions

ABS(*number*)
ACOS(*number*)
ACOSH(*number*)
AND(*logical1, logical2 [,logicalN]*)
ASIN(*number*)
ASINH(*number*)
ATAN(*number*)
ATAN2(*number_dx, number_dy*)
ATANH(*number*)
CEILING(*number, significance*)
CHAR(*number*)
CLEAN(*text*)
CODE(*text*)
COMBIN(*number,number_chosen*)
CONCATENATE(*text1,text2 [,textN]*)
COS(*number*)
COSH(*number*)
DATE(*year,month,day*)
DAY(*datevalue*)
DEGREES(*angle*)
EOMONTH(*datevalue*)
EVEN(*number*)
EXACT(*text1, text2*)
EXP(*number*)
FACT(*number*)
FACTDOUBLE(*number*)
FALSE()
FIND(*find_text, within_text [,start_number]*)
FIXED(*number*)
FLOOR(*number1, number2 [,numberN]*)
GCD(*number1, number2 [,numberN]*)
HOUR(*datevalue*)
IF(*logical,true_expression,false_expression*)
IIF(*logical,true_expression,false_expression*)
INFO(*text*)
INT(*number*)
ISERROR(*value*)
ISEVEN(*number*)
ISLOGICAL(*value*)
ISODD(*number*)
ISNUMBER(*value*)
ISTEXT(*value*)
LCM(*number1, number2 [,numberN]*)

LEFT(*text [,number]*)
LEN(*text*)
LN(*number*)
LOG(*number [,base]*)
LOG10(*number*)
LOWER(*text*)
MID(*text, number_start, number_count*)
MINUTE(*datevalue*)
MONTH(*datevalue*)
MOD (*number, divisor*)
N(*value*)
NOT(*logical*)
NOW()
ODD(*number*)
OR(*logical1, logical2 [,logicalN]*)
PI()
PRODUCT(*number1, number2 [,numberN]*)
PROPER(*text*)
POWER(*base, exponent*)
QUOTIENT(*number, divisor*)
RADIANS(*degrees*)
RAND()
RANDBETWEEN(lower_boundary,upper_boundary)
REPLACE(*text, start_number, length_number, replace_with_text*)
REPT(*text, number*)
RIGHT(*text, number*)
ROMAN(*number [,romantype]*)
ROUND(*number, significance*)
ROUNDDOWN(*number, significance*)
ROUNDUP(*number, significance*)
SEARCH(find_*text, within_text [,start_number]*)
SECOND(*datevalue*)
SIGN(*number*)
SIN(*number*)
SINH(*number*)
SQRT(*number*)
SQRTPI(*number*)
SUBSTITUTE(*find_text, within_text, replace_with_text [,number_of_repetitions]*)
SUM(*number1 [,numberN]*)
T(*value*)
TAN(*number*)
TANH(*number*)
TIME(*datevalue*)
TEXT(*number,format_text*)
TODAY()
TRIM(*text*)

TRUE()
TRUNC(*number*)
TYPE(*value*)
UPPER(*text*)
VALUE(*numeric*)
WEEKDAY(*datevalue [,weektype]*)
WEEKNUM(*datevalue [,weektype]*)
YEAR(*datevalue*)

C - List of Ages! Data Objects and Properties

Simple Data Type

Object Type

Age object

DAYS	Integer Age in full days (max 31)
MONTHS	Integer Age in full months (max 12)
YEARS	Integer Age in full years

Event object

ADDRESS	String postal address
AGE	Age object, for invididual events only
AGENCY	String agency
ATTENDANTS[index].PERSON	Individual Object of the attending person
ATTENDANTS[index].ROLE	String Role of the attending person in the event
CALCAGE	Age object, for invididual events only *new 1.53*
CAUSE	String only available for deaths, retirements and anulments
DATE	String localized full date
DAY	Integer day part of the event's date
HUSB_AGE	Age object - for family events only
LOCATION	Location object, place of the event
MONTH	Integer month part of the event's date
NAME	String event name (e.g. "Birth")
NOTES[*index*]	Note objects
PICTURES[*index*]	Media objects
SERIALDATE	Integer or empty String number of days since 31.12.1899, may be negative. To be used with the Excel™-compatible date functions
SORTDATE	String
SOURCES[*index*]	Source Citation objects
TYPE	String event type (e.g. "Carpenter")
WIFE_AGE	Age object - for family events only
YEAR	Integer

Family object

ANUL or ANUL[*index*]	`Event`	object for primary (or n-th) annulment
CENS or CENS[*index*]	`Event`	object for primary (or n-th) census
DIV or DIV[*index*]	`Event`	object for primary (or n-th) divorce
DIVF or DIVF[*index*]	`Event`	object for primary (or n-th) divorce filing
ENGA or ENGA[*index*]	`Event`	object for primary (or n-th) engagement
EVENTS[*index*]	`Event`	objects
HUSB	`Individual`	object
HUSB_PARTNER_NO	`Integer`	
MARB or MARB[*index*]	`Event`	object for primary (or n-th) marriage bans
MARC or MARC[*index*]	`Event`	object for primary (or n-th) marriage contract
MARL or MARL[*index*]	`Event`	object for primary (or n-th) marriage license
MARR or MARR[*index*]	`Event`	object for primary (or n-th) wedding
MARS or MARS[*index*]	`Event`	object for primary (or n-th) marriage settlement
NOTES[*index*]	`Note`	objects
PICTURES[*index*]	`Media`	objects
SLGS or SLGS[*index*]	`Event`	object for primary (or n-th) sealing to spouse (LDS)
SOURCES[*index*]	`Source Citation`	objects
STATUS	`String`	marriage status
WIFE	`Individual`	object
WIFE_PARTNER_NO	`Integer`	

Individual object

ADOP or ADOP[*index*]	Event	object for primary (or n-th) adoption
BAPL or BAPL[*index*]	Event	object for primary (or n-th) LDS
BAPM or BAPM[*index*]	Event	object for primary (or n-th) baptism
BARM or BARM[*index*]	Event	object for primary (or n-th) bar-mitzvah
BASM or BASM[*index*]	Event	object for primary (or n-th) bath-mitzvah
BIRT or BIRT[*index*]	Event	object for primary (or n-th) birth event
BLES or BLES[*index*]	Event	object for primary (or n-th) blessing
BURI or BURI[*index*]	Event	object for primary (or n-th) burial
CAST or CAST[*index*]	Event	object for primary (or n-th) caste
CENS or CENS[*index*]	Event	object for primary (or n-th) census
CHR or CHR[*index*]	Event	object for primary (or n-th) christening
CHRA or CHRA[*index*]	Event	object for primary (or n-th) adult christening
CONF or CONF[*index*]	Event	object for primary (or n-th) confirmation
CONL or CONL[*index*]	Event	object for primary (or n-th) LDS confirmation
CREM or CREM[*index*]	Event	object for primary (or n-th) cremation
CURRENTAGE	Age	object (calculated as difference between today and birth)
DABOVILLE	String	D'Aboville no.-available on descendant reports only
DEAT or DEAT[*index*]	Event	object for primary (or n-th) death
DECEASED	Integer	1, if confirmed dead, or assumed to be dead
DSCR or DSCR[*index*]	Event	object for primary (or n-th) description
EDUC or EDUC[*index*]	Event	object for primary (or n-th) education
EMIG or EMIG[*index*]	Event	object for primary (or n-th) emigration
ENDL or ENDL[*index*]	Event	object for primary (or n-th) LDS endowment
ESTIMATED_BIRTH_YEAR	Integer	calculates an estimated birth year
EVENTS[*index*]	Event	objects
FATHER	Individual	object (available on some reports only)
FATHERS_CHILD_NO	Integer	
FCOM or FCOM[*index*]	Event	object for primary (or n-th) first communion
FIRSTNAMES	String	all given names in full length (John Fitzgerald)
FIRSTNAMESSHORT	String	shortened variant of first names (John F.)
GENERATION	String	(only available on some reports)
GENERATION_DIRECT	String	(only available on some reports)
GRAD or GRAD[*index*]	Event	object for primary (or n-th) graduation
HENRY_CLASSIC	String	Henry no. (123456789XABC…)- available on descendant reports only
HENRY_MODERN	String	Henry no., modern (123456789ABCD…)- available on descendant reports only
HENRY_MODIFIED	String	Henry no., modern (123456789(10)(11)(12)(13)…)- available on descendant reports only

IDNO or IDNO[*index*]	`Event` object for primary (or n-th) ID-Number
IMMI or IMMI[*index*]	`Event` object for primary (or n-th) immigration
INDI_NO	`Integer` (only available on some reports)
KEKULE	`Integer` (only available on some reports)
LASTNAME	`String`
MARRIAGES[*index*]	`Family` objects
MORENAMES[*index*].FIRSTNAME	`String`
MORENAMES[*index*].LASTNAME	`String`
MORENAMES[*index*].TYPE	`String`
MOTHER	`Individual` object (only available on some reports)
NATI or NATI[*index*]	`Event` object for primary (or n-th) nationality
NATU or NATU[*index*]	`Event` object for primary (or n-th) naturalization
NCHI or NCHI[*index*]	`Event` object for primary (or n-th) number of children
NMR or NMR[*index*]	`Event` object for primary (or n-th) number of marriages
NOTES[*index*]	`Note` objects
NUMBER_OF_CHILDREN	`Integer` number of children (calculated total over all partnerships)
NUMBER_OF_PARENTS	`Integer` number of parent family objects (PARENTS[])
NUMBER_OF_PARTNERS	`Integer` number of partnership family objects (MARRIAGES[])
NUMBER_OF_UNKNOWN_PARTNERS	`Integer` number of partnership family objects with unknown partner
OCCU or OCCU[*index*]	`Event` object for primary (or n-th) occupation
ORDN or ORDN[*index*]	`Event` object for primary (or n-th) ordinance
OTHERFAMS[*index*]	`Family` objects (only available on some reports)
PARENTS	`Family` object (only available on some reports)
PARENTS[*index*]	`Family` objects
PARTNERSHIP	`Family` object (only available on some reports)
PICTURES[*index*]	`Media` objects
PROB or PROB[*index*]	`Event` object for primary (or n-th) probate
PROP or PROP[*index*]	`Event` object for primary (or n-th) property
RELI or RELI[*index*]	`Event` object for primary (or n-th) religion
RESI or RESI[*index*]	`Event` object for primary (or n-th) residence
RETI or RETI[*index*]	`Event` object for primary (or n-th) retirement
SARAGOSSA	`String` - available on descendent reports only
SEX	`String` "M", "F" or "?"
SLGC or SLGC[*index*]	`Event` object for primary (or n-th) LDS sealing to child
SOURCES[*index*]	`Source Citation` objects
SSN or SSN[*index*]	`Event` object for primary (or n-th) social security number
TITL or TITL[*index*]	`Event` object for primary (or n-th) title
WILL or WILL[*index*]	`Event` object for primary (or n-th) will

Location object

LATITUDE	String (e.g.: 10° 5' 20" N)
LONGITUDE	String (e.g.: 12° 1' 12" E)
NAME	String
NOTES[*index*]	Note objects
PICTURES[*index*]	Media objects
POSTALCODE	String
SOURCES[*index*]	Source Citation objects

Media object

PICTURE	String @filename@

Note object

TEXT	String

Repository object

ADDRESS	String
NAME	String
NOTES[*index*]	Note objects

Source object

ABBR	String
AGENCY	String
AUTHOR	String
CALLNUMBERS[*index*].MEDIATYPE	String
CALLNUMBERS[*index*].NUMBER	String
DATAEVENTS[index].EVENT	String event type (e.g. "Birth")
DATAEVENTS[index].LOCATION	String
DATAEVENTS[index].TIMERANGE	String localized time range
NOTES[*index*]	Note objects
PICTURES[*index*]	Media objects
PUBLICATION	String
REPOSITORY	Repository object
TEXT	String
TITLE	String

Source Citation object

ABBR	String
NOTES[*index*]	Note objects
PAGE	String
PICTURES[*index*]	Media objects
RECDATE	String
RECEVENT	String
ROLE	String
TEXT	String
TITLE	String

Submitter object

NAME	String

Task object

DATE	String localized full date
EXPENSES	String
NOTES[*index*]	Note object
PRIORITY	Integer
PROGRESS	Integer
REPOSITORY	Repository object
TEXT	String
TITLE	String

Report object

DATE	String localized printing date
FILENAME	String
PAGE_NO	Integer
SUBM	Submitter object
TITLE	String