Expressions:

Expressions are used in several places in *DataExplorer* you will find that you can use them to create new Columns of data in the Document Inspector, enter values in any of the tools that look like paste_2.tiff ¬, and give expressions to Columns.

In any expression you use with the Active Document you may reference any of its Columns by their name of any of the Document's Variables that you added from the Document Inspector. Expressions are case sensitive, so be careful to type what you mean. If you reference a Column when giving an expression for one of the above tools it will use the first value of that Column to evaluate the Expression. If you reference a Column in the Fill Column panel the Expression will do its best to match the corresponding values in the Column.

Please read through the following definition of expressions.

Built-in-operators:

Operator	Description
+	binary addition
-	binary subtraction, unary negation
/	binary division
*	binary multiplication
%	binary modulus
^	binary exponential
<	less than
>	greater than
<=	less than or equal
>=	greater than or equal
==	equal
!=	not equal
!	(unary) negation
||	logical or
&&	logical and

Keep in mind that the operators `<`, `>`, `<=`, `>=`, `==`, `!=`, `!`, `||`, and `&&`, evaluate to either '`1`' or '`0`'. '`1`'means **True** and `0` means **False** in a conditional expression.

The conditional operator is `?:` and can be used as follows:

`expression1 : expresssion2 ? expression3.`

If `expression1` is **True** then `expression2` is the resulting value otherwise `expression3` is the resulting value. You may use the conditional function if you do not like or understand this operator. The conditional function looks like:

`if(expression1, epression2, expression3, expression4` [optional]`)`

If `expression1` is **True** then `expression2` is the resulting value otherwise `expression3` is the resulting value. If `expression1` is undetermined (NaN, +Infinity, etc.) `expression4` is the resulting value.

Associativity and Order of Operations:

Operators are listed in order of precedence with operators of equal precedence on each line.

Operator	**Description**
`^`	right to left
`- !`	right to left
`* / %`	left to right
`+ -`	left to right

`<  >  <=  >=`	left to right
`==  !=`	left to right
`&&`	left to right
`||`	left to right
`?:`	right to left

Constants:

Constant	Value
E	2.7182818284590452354
LOG2E	1.4426950408889634074
LOG10E	0.43429448190325182765
LN2	0.69314718055994530942
LN10	2.30258509299404568402
PI	3.14159265358979323846
PI_2	1.57079632679489661923
PI_4	0.78539816339744830962
1_PI	0.31830988618379067154
2_PI	0.63661977236758134308
2_SQRTPI	1.12837916709551257390
SQRT2	1.41421356237309504880
SQRT1_2	0.70710678118654752440
MAXDOUBLE	1.7976931348623157e308

MAXINT	2147483647

Built-in-functions:

The following is a list of functions that *DataExplorer* understands. If you do not know what a particular function is you may want to reference the Unix manual page for the corresponding operator listed under **C -Function Called**. This can be done through the application *Digital Librarian* or through the command line using the Unix command *man*.

Function	**No of Parameters**	**C-Function Called**
abs	1	abs
acos	1	acos
acosh	1	acosh
asin	1	asin
asinh	1	asinh
atan	1	atan
atanh	1	atanh
atan2	2	atan2
cbrt	1	cbrt
ceil	1	ceil
copysign	2	copysign
cos	1	cos
cosh	1	cosh

drem	2	drem
erf	1	erf
erfc	1	erfc
exp	1	exp
expm1	1	expm1
fabs	1	fabs
finite	1	finite
floor	1	floor
gammap	2	internal
gammaq	2	internal
fmod	2	fmod
hypot	2	hypot
if	3, 4	internal
jn	2	jn
j0	1	j0
j1	1	j1
ln	1	log
log10	1	log10
log1p	1	log1p
pow	2	pow
rand	0	random (is called then divided by MAXINT)
rint	1	rint
scalb	2	scalb
sin	1	sin
sinh	1	sinh
sqrt	1	sqrt

tan	1	tan
tanh	1	tanh
yn	2	yn
y0	1	y0
y1	1	y1