

# Expression Language

---

## Intro

Expressions are used in various places in xReporter: to calculate default values, to do validation, to insert calculated values inside SQL statements, ...

The expressions are very similar to those found in spreadsheets.

## Basics

In expressions there are four basic types: strings, numbers, dates and booleans.

A string is presented between double quotes, e.g. "hallo".

For numbers a dot should be used as decimal separator.

The following mathematical operations can be applied on numbers: +, -, \* and /. All these operations are done with high precision. Multiplication and division take precedence over addition and subtraction. The addition operation cannot be used to concatenate strings, for this you can use the Concat function (see further on).

The following comparison operators are available: <, >, <=, >=, =, !=. These work for strings, numbers and dates. The result of such a comparison is always either TRUE or FALSE.

It is also possible to use variables, the available variables depend on the context in which the expression is evaluated.

## User-defined functions

It is possible to write additional (e.g. domain-specific) functions which can then be used within the expression language. An example of this can be found in the class `sample.IsOddFunction` (used in the "bigtable" sample report). These new functions should then be registered in xReporter's `config.xml`.

## Description of the functions

### Date and time functions

#### TextToDate(string)

Converts a string representing a date, to a date. The date should be given in US format, thus month/day/year.

Example: `TextToDate("7/15/2002")` returns the date 15 juli 2002.

#### Date(number,number,number)

Creates a date based on three parameters: year, month, day.

Example: `Date(2002, 7, 15)` returns the date 15 juli 2002.

#### Now()

Returns the date representing the current date/time.

### **DayOfWeek(date)**

Returns the day of the week according to US conventions, thus sunday is 1, monday is 2, and so on.

### **DayOfMonth(date)**

Returns the day of the month.

### **DayOfYear(date)**

Returns the day of the year.

### **BeginOfMonth(date)**

Returns the first day of the month (as a date) of the given date.

### **EndOfMonth(date)**

Returns the last day of the month (as a date) of the given date.

### **BeginOfQuarter(date)**

Returns the first day of the quarter (as a date) of the given date.

### **EndOfQuarter(date)**

Returns the last day of the quarter (as a date) of the given date.

### **BeginOfYear(date)**

Returns the first day of the year (as a date) of the given date.

### **EndOfYear(date)**

Returns the last day of the year (as a date) of the given date.

### **DaysInMonth(date)**

Returns the number of days in month of the given date.

### **Month(date)**

Returns the month of the given date. January is 1, Februari is 2, and so on.

### **Year(date)**

Returns the year of the given date.

### **AddDays(date, number)**

Returns the date result of adding a number of days to a given date. By specifying a negative number of days one will effectively 'subtract' days from the original.

## SetTime(date, number, number, number)

Returns the date with it's 'time-part' set to the hour, minutes and seconds specified in the function's arguments.

## Mathematical functions

### Random()

Returns a random number between 0 and 1.

### Remainder(number1,number2)

Returns the remainder when dividing number1 by number2

### Power(number1,number2)

Raises number1 to the power of number2.

### Abs(number)

Returns the absolute value of the number.

### Floor(number)

Returns the largest number that is not larger than the parameter and is whole number.

### Ceiling(number)

Returns the smallest number not smaller than the parameter and is a whole number.

## Logical functions

### True()

Returns the boolean value TRUE.

### False()

Returns the boolean value FALSE.

### And(boolean1,boolean2,boolean3,...,booleanN)

Returns TRUE if all parameters are TRUE.

### Or(boolean1,boolean2,boolean3,...,booleanN)

Returns TRUE if at least one of the parameters is TRUE.

### Not(boolean)

Returns TRUE if the parameter is FALSE, and FALSE if the parameter is TRUE.

If(boolean,expr1,expr2)

If the first parameter is TRUE then expr1 will be evaluated, otherwise expr2.

## String functions

Concat(string1,string2,...,stringN)

Combines multiple strings to one string.

UpperCase(string)

Converts a string to uppercase.

LowerCase(string)

Converts a string to lower case.

Trim(string)

Removes all spaces before and after the string.

Substring(string, beginposition, endposition)

Returns a string that is a substring of the given string. The position of the first character is 0.

Length(string)

Returns the length of the given string.

## Extra functions

NVL(any, any)

Returns the second argument if the first evaluates to null, or when evaluation of the first yields an exception.

## XReporter-specific functions

UserId() : string

Returns the id of the current user.

CustomerId() : string

Returns the id of the customer to which the current user belongs.

UserProperty(string) : string

Retrieves a property of the current user. The parameter is the name of the property.

DataSourceId() : string

Returns the id of the current data source.

## DataSourceParam(string) : string

Returns the value of one of the connection properties of the current data source. As parameter the name of the connection property should be supplied.

## ReportId() : string

Returns the id of the current report.