Calculator (Variables) Component

Use this component to perform mathematical operations on variables. In the configuration wizard, create new (calculated) variables or change existing variable values according to specified expressions.

> **Important:** The system calculates expression values following their order in the expressions list (starting with the top). An expression formula can only use expressions located above it in the list. An incorrect position in the list may result in an error.

Ports

Input

- **Input variables**: Provide a list of variables to use within the node.

Output

- **Output variables**: This port displays the input variables and new variables calculated by the node.

Configuration wizard

The configuration window contains the following panes:

1. Expression list
2. Expression code pane
3. Variables
4. Function list

Expression list

Use this pane to enter *expressions*: calculated variables that supplement the input dataset after processing or modifying its source variables. For each dataset row, the system calculates the expression value using the formula in the expression code pane.

Create a new expression using the toolbar or context menu. Use these toolbar actions:

- **Add expression**: Add a new expression with default parameters.
- **Clone expression**: Add a new expression with the same data type, description, and formula as the current expression.
- **Edit expression parameters...** (`F2`): Configure the expression parameters.
- **Replace variable**: Enable the mode that replaces the source variable value with the value calculated in the expression.
- **Move up** (`Ctrl` + `Up`): Move the expression up one position in the list.
- **Move down** (`Ctrl` + `Down`): Move the expression down one position in the list.
- **Delete expression** (`Delete`): Delete the current expression.
- **Delete all expressions...** (`Shift` + `Delete`): Delete all existing expressions.

In addition, use the following actions in the expression list context menu:

- **Move to expression** (`Enter`): Select the expression and show its code.
- **Insert relative name into expression**: Insert the expression name into the code pane.

When you add or edit an expression, the system displays the parameter editing dialog. Use it to configure these expression parameters:

- **Name**: Enter the variable's name to use in the output dataset.
- **Caption**: Enter the variable's caption to use in the output dataset.
- **Data type**: Specify the variable's data type in the output dataset.
- **Intermediate**: Allow the system to use the expression in calculations without including it in the output dataset variable list.
- **Description**: Add any comment or explanatory information.

> **Important:** Start the unique name with an uppercase or lowercase Latin letter or an underscore. In subsequent positions, you can also use digits.

When you first open the configuration wizard, the expression list contains one `real` element named `Expr1`. By default, the system assigns a new expression the caption `ExpressionN` and the name `ExprN`, where *N* is an index that ensures uniqueness.

After configuring the expression parameters, enter the calculation formula in the code pane.

Expression code pane

Enter the formula for calculating the expression in the code pane. Insert references to input variables and function syntax into the expression code by entering them manually, double-clicking them in the corresponding panes, or dragging them from the list. Insert references to expressions created in the Calculator by entering them manually or dragging them. You can also use the context menu to add references to expressions or variables to the code pane.

> **Important:** From the current expression, reference only expressions located above it in the list.

If an input variable and an expression have the same name, the system uses the variable in the expression code by default. Use the `Expr.` prefix to reference the expression.

An expression formula can contain:

- References to other dataset variables (through variable names) or to previously calculated expressions
- References to input port variables
- Parentheses that define the order of operations
- Mathematical and relational operators
- Logical operators (`AND` , `OR` , `NOT` , and `XOR`) and values (`TRUE` or `1` ; `FALSE` or `0`)
- Functions that follow the syntax description (see Function list)
- String expressions in quotation marks, for example, `"string expression"`
- Integer and real numbers
- Single-line and multiline comments

> **Note:** The **Calculator (Variables)** node represents all numeric values as real double-precision numbers under the IEEE 754-2008 standard. Integer values outside the `[-(2^53 - 1), 2^53 - 1]` range are subject to the loss of accuracy.
> Example: The result of calculating `Val("6 405 472 121 753 949 359")` using the `integer` data type is `6 405 472 121 753 949 184` .

Press Esc to leave the code pane and return to the previously active pane.

Code pane toolbar

Preview: Check the calculations by viewing up to the first *100 rows* of the result table. Press F3 to open the window.

- At the bottom of the **Preview** window, use the **Console** button to view expression calculation errors. Expect the console to remain empty when no errors occur.
- Use the **Expand** and **Collapse** buttons to adjust the **Console** size.

In addition to the button that opens **Preview**, use the quick access toolbar buttons to insert a preset or template into the expression code pane.

- Logical operators:
  - **AND**: logical `AND`
  - **OR**: logical `OR`
  - **NOT**: logical `NOT`
  - **XOR**: logical exclusive `OR`
- Comparison operators:

    - **=**: equality
    - **<>**: inequality
    - **<**: check whether a value is less than another value
    - **>**: check whether a value is greater than another value
    - **<=**: check whether a value is less than or equal to another value
    - **>=**: check whether a value is greater than or equal to another value
- Templates:
    - **Number**: `0.0` for a real number
    - **String**: `""` for a string value
    - **Date**: `StrToDate("____-__-__", "YYYY-MM-DD")` for a date value
    - **Time**: `StrToDate("__:__:__", "HH:NN:SS")` for a time value
- Logical values:
    - **FALSE**: a `false` value
    - **TRUE**: a `true` value

Comments

Start a single-line comment with `//` (two slashes). The comment continues to the end of the line. Enclose a multiline comment between `/*` (slash-asterisk) and `*/` (asterisk-slash).

Press `Ctrl` + `/` in the code pane to comment or uncomment code.

Comment example:

```
// Single-line comment example.

IF(IsNull(var_one),var_two,var_one) // The single-line comment continues
to the end of the line.

/* All characters between
(slash-asterisk) and
(asterisk-slash) form
a multiline comment. */
```

Variables

This pane contains the list of variables that the input port passes to the node. Configure the list and variable parameters in the node input ports.

Double-click a list item to insert the variable name into the expression code pane. You can also drag the item or use its context menu.

The context menu contains these commands:

- **Replace with calculated expression**: Add a new expression for the selected variable and enable the mode that replaces the source variable value with the value calculated in the expression.
- **Insert to expression text**: Insert the variable name into the code pane.
- **Move to expression**: Use this command after applying **Replace with calculated expression** to the variable. Select the expression that replaces this variable; then, activate the code pane.

> **Important:** After applying **Replace with calculated expression** to a variable, click the variable caption link to select the expression that replaces this variable and activate the code pane. This action works like the **Move to expression** context menu command.

Function list

View names, input arguments, and descriptions of the available functions.

Filter functions by category and name.

Double-click a function to insert its syntax into the expression code pane. You can also drag the function.

> **Note:** The **Calculator (Variables)** node uses lazy evaluation. This means, the system performs a calculation only when another workflow node needs to display or use the expression value.

Read on: Variables to Table Component