

Conditional Expression

The form of a conditional expression is as follows:

$$\{Condition\} \text{ ? } [expression1] : [expression2]$$

Where:

Condition is:

- an object that acquires a value of *Boolean* type
- a logical expression enclosed in any type of brackets - { }, () or []

If the value of the condition is TRUE, then *expression1* is evaluated. Otherwise, *expression2* is evaluated.

Expression1 and Expression2 is:

- arithmetical expression enclosed in any brackets - { }, () or []

The result value of an expression that is evaluated is the result of the conditional expression. Just the expression, for which the Condition is met, is evaluated.

Warning:

If a function, that uses its previous internal state (e.g. *%PrevV*, *%Integral*,...) for evaluation, is included in the expression that has not been evaluated, the internal state is not correct!

Example of conditional expression

The task is to indicate the state when the water level (the object *Water_level*) in the cistern falls under the minimum (the object *Minimum*).

We define a new eval tag called *Low_level* of logical type, which value is determined by the expression:

$$\{Water_level < Minimum\} \text{ ? } [TRUE] : [FALSE]$$

If the arithmetical expression $\{Water_level < Minimum\}$ is true, then the value of the object *Low_level* is TRUE, otherwise, the value is FALSE. The object *Low_level* is to be evaluated if a change of the input objects occurs (i.e. when *Water_level* or *Minimum* has changed).



Related pages:

[Mathematical apparatus for Events and Eval tags](#)