mathOnIssueList()

This function returns a **number list** with results of the given calculation performed for each issue in the specified list.

Syntax
`mathOnIssueList(issueList, mathExpression) #Output: Number list`

Examples

Parser expression	Description
`mathOnIssueList(linkedIssues("is blocked by"), ({function.issue.dueDate} != null ? {function.issue.dueDate} - {function.issue.created} : 0) / {HOUR})`	This example returns a **number list** with the number of days from issue creation to the due date for all issues linked using **" is blocked by"** issue link type. To achieve this, the following functions are used: • linkedIssues() **{HOUR}** is a time macro used to convert the output to hours.
`mathOnIssueList(subtasks(), round(({system.currentDateTime} - {function.issue.created}) / {DAY}))`	This example returns a **number list** with days of when each sub-task was created. To achieve this, the following functions are used: • round() • subtasks() **{DAY}** is a time macro used to convert the output to days.

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
`issueList`	ISSUE LIST	Any given issue list. Usually this value is retrieved from a function (e.g. linkedIssues() or subtasks()).
`mathExpression`	TEXT	A valid math expression. The field values of the function issue is retrieved by using the **function origin** in front of the field codes.

Output

This function returns a NUMBER LIST

If you only want to retrieve a specific field value from the given issues without any calculations, you might want to use the function **fieldValue()**.

Use cases and examples

Use case	JWT feature	Workflow function	Field type	Automated action	Parser functions
Operate with field values from sub-tasks			Number		