List operators

On this page

General Information

There are **three** different data types that return **lists**. i.e., types that are based on lists, or ordered collections of elements.

These data types are:

- **Issue lists** ISSUE []
- **Number lists** NUMBER []
- **String lists** STRING []

List Operators

There are **four available operators** for working on **list-based** data types:

Operator	Behavior	Examples

l `APPEND` m	Returns a list with elements in **l** followed by elements in **m**, therefore the number of elements is the sum of the number of elements in **l** and **m**.
	Order is respected. It may contain repeated elements.
		`[1, 2, 3] APPEND [3, 4, 4] = [1, 2, 3, 3, 4, 4]`
		`["blue", "red", "red"] APPEND ["red", "green"] = ["blue", "red", "red", "red", "green"]`
		`subtasks() UNION subtasks()` returns a list containing twice all the sub-tasks of current issue.

l `UNION` m	Returns a list with elements in **l** and elements **m** without repetitions.
	Order is respected.
		`[1, 2, 3] UNION [3, 4, 4] = [1, 2, 3, 4]`
		`["blue", "red", "red"] UNION ["red", "green"] = ["blue", "red", "green"]`
		`linkedIssues() UNION subtasks()` returns a list with linked issues and sub-tasks of current issue without repetitions.

l `INTERSECT` m	Returns a list with the elements present in both lists simultaneously. Returned list doesn't contain element repetitions.
	Order is respected.
		`[1, 1, 2, 3] INTERSECT [1, 3, 5] = [1, 3]`
		`["red", "blue", "blue"] INTERSECT ["blue", "yellow", "yellow"] = ["blue"]`
		`linkedIssues() INTERSECT subtasks()` returns a list with those sub-tasks which are also linked to current issue.

l `EXCEPT` m	Returns a list with elements in **l** which are not present in list **m**. Returned list doesn't contain element repetitions.
	Order is respected.
		`[1, 2, 2, 3, 3] EXCEPT [2, 5, 6] = [1, 3]`
		`["red", "red", "blue", "blue", "green"] EXCEPT ["blue", "yellow"] = ["red", "green"]`
		`linkedIssues() EXCEPT subtasks()` returns a list with linked issues which are not sub-tasks of current issue.

- **l** and **m** are both lists of the same data type: **number**, **string** or **issues**.
- All operators are case insensitive, i.e., they can also be written in lower case: **`append`**, **`union`**, **`intersect`** and **`except`** .
- There are 4 equivalent and homonym functions available for each type of list, and its behavior is exactly equivalent to that of its corresponding operator. This way, you can choose to use operators or functions according to your preference. Although operators yield shorter expressions and with fewer parentheses, the usage of functions produces a more functional consistent syntax

Precedence Order and Associativity

OPERATORS	PRECEDENCE	ASSOCIATIVITY
l **`INTERSECT`** m	1 (highest)	Left-to-Right
l **`UNION`** m, l **`EXCEPT`** m, l **`APPEND`** m	2 (lowest)	Left-to-Right