

JMESPath Query Language Cheatsheet

Basic Expressions

Concept	Usage/Description	Example
Identifier	Used to access a key within a JSON object.	Name = "foo"
Subexpression	Used to access nested values.	State.Code = "200"
null	The value returned when a nonexistent key is referenced.	null
Index Expressions	Used to access an item in a list. Zero based.	test.list[1] = "b"

Slicing

General form: [start:stop:step]

Given: test = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

Syntax	Slice Returned
test[1:5]	[1, 2, 3, 4]
test[:5]	[0, 1, 2, 3, 4]
test[::2]	[0, 2, 4, 6 8]
test[::-1]	[9, 8, 7, 6, 5, 4 3, 2, 1, 0]

Projections apply expressions to collections of elements. There are 5 types.

Type	What do	example
List/Slice	Used to get a given keyval from all items in a list/slice of JSON objects	testdict[*].key
Object	Used like list/slice projections, but within a dict of dicts	testdict.*.attr
Flatten	Used to restructure a JSON object in order to get a list of bucketed lists of an attribute	testdict[*].subdict[*].attr
Filter	Used to get data from any subdict matching the query expression	testdict[?attr1=='1'].attr2

BLAH BLAH BLAH BLAH