

# 1 Get

The *get* Operation accepts the following arguments

- A Key or Collection of Key(s)
- A Collection of Scalar(s) or Key Value pair(s)

and will return the collection member identified by the key.

- When passed a Collection of Scalar(s), the Key should be the integer index of the target member
- When passed a Collection of Key Value pair(s), the Key should be mapped to a target Value
  - A collection of Key(s) indicates navigation into a nested Collection of Key Value pair(s)
- when passed a Statement, the Key can be a [JSONPath](#) string

## 1.1 Formal Definition

The *get* operation performs tree traversal of the provided collection *data* and returns the value *v* located at path *path*.

$$get(data, path) \rightarrow v$$

An example of this functionality with respect to a Collection of Scalar(s) *X* defined as

$$X = \langle x_0, x_1, x_2 \rangle$$

demonstrates that an index of a collection can be used for lookup

$$get(X, 0) = x_0 \wedge get(X, 1) = x_1 \wedge get(X, 2) = x_2$$

When *get* is passed a Collection of Key Value pair(s) *KV* defined as

$$KV = \langle k_0v_{k_0}, k_1v_{k_1}, k_2v_{k_2} \rangle$$

a Key can be passed as *path* to return the mapped Value

$$get(KV, k_1) = v_{k_1} \wedge get(KV, k_2) = v_{k_2} \wedge get(KV, k_0) = v_{k_0}$$

And when *get* is passed a Statement *s*

```
s = < id : 12345
    actor : < account :
        < homePage : https : //www.example.com, name : example >>
    verb < id : https : //www.example.com/verbs/example >
    object < id : https : //www.example.com/activity/example >>
```

*path* can be a JSONPath string, a Key or Collection of Key(s)

- $path = \text{JSONPath}$

$get(s, \$.object.id) = https://www.example.com/activity/example$

- $path = \text{Collection of Key(s)}$

$get(s, < object, id >) = https://www.example.com/activity/example$

- $path = \text{Key}$

$get(s, object) = < id : https://www.example.com/activity/example >$

$get(s, id) = 12345$

Which establishes how *get* navigates into flat and nested collections to return some value *v*.