Nested JSON OBJECTS

To see how to examine a JSON object, please read the following article first: Accessing JSON OBJECTS in database schemas.

In some cases the JSON objects can also hold JSON objects themselves (nested objects). These can be identified because the text that appears on running an <OBJECT>::TEXT will have the following structure:

```
{"<NESTED OBJECT NAME1>":{"COLUMN NAME 1": <value>, "COLUMN NAME 2": <value>, ... etc}, {"<NESTED OBJECT
NAME2>":{"COLUMN NAME 1": <value>, "COLUMN NAME 2": <value>, ... etc}, ... etc}
```

Example:

Or:

To access elements of the nested elements you first need to get to the nested object itself and identify what information you want to get.

- To access an element in an object you use: ->>
- To access an object within an object you use: ->

In the above screenprint we see for instance that "Communities" is a nested object from the "applications" object in the "cqx_data.community_activity_applications" table that contains a field called "user_count". To get the specific user_count of community activity I therefor first need to access the "Applications" object to get to the "Communities" object to then retrieve the specific "user_count" field value.

applications **1**

{"Wikis": {"user_count": 2, "events_read": 7, "events_total": 7}, "Bookmarks": {"user_count": 1, "events_read": 0, "events_total": 17}, "Communities": **2** {"user_count": 5, **3** "events_read": 12, "event...

{"Activities": {"user_count": 1, "events_read": 1, "events_total": 7}, "Communities": **2** {"user_count": 1, **3** "events_read": 1, "events_total": 1}}

{"Files": {"user_count": 2, "events_read": 0, "events_total": 4}, "Activities": {"user_count": 1, "events_read": 5, "events_total": 5}, "Communities": **2** {"user_count": 1, **3** "events_read": 2, "events_tota...

{"Wikis": {"user_count": 4, "events_read": 22, "events_total": 23}, "Communities": **2** {"user_count": 4, **3** "events_read": 6, "events_total": 6}}

This can be done in two ways by either using a sub select:

```
SELECT
split_communities_object ->> 'user_count' as user_count_communities,
*
FROM
(SELECT
*,
applications -> 'Communities' AS split_communities_object
FROM
cqx_data.community_activity_applications
) splitObject
```

Or by splitting it all in one go (more direct option):

```
SELECT
applications -> 'Communities' ->> 'user_count' as user_count_communities,
*
FROM
cqx_data.community_activity_applications
```

The result in both cases is:

user_count_communiti...	activity_date	community_uuid	docdate	applications
5	2017-06-20	120a2a7d-899c-40c6-a4...	2017-06-21	[object Object]
1	2017-06-20	26a17848-00da-4d6f-8f...	2017-06-21	[object Object]
1	2017-06-20	42172513-39e4-4001-a7...	2017-06-21	[object Object]
4	2017-06-20	4f6fd81c-b693-49f7-83c...	2017-06-21	[object Object]

⚠ **Note! If the Object doesn't contain a nested object called "Communities" the value will simply be empty.**