

API Lezen

Met de KampAdmin API kun je gegevens uit specifieke tabellen opvragen via **GET-verzoeken**. Dit type verzoek haalt data op en geeft een gestructureerde **JSON-response** terug, die de inhoud van de opgevraagde records bevat. Alleen tabellen en kolommen die zijn **geactiveerd**, zullen zichtbaar zijn in de response.

Basis API-request

Een basis API-request bestaat dus uit het versturen van een GET-verzoek naar een specifieke URL, met verschillende parameters die gaan uitmaken welke data je juist gaat terugkrijgen en met het token voor authenticatie

Een request voor zo'n standaard data-opvraging ziet er als volgt uit:

```
[GET] /api/v3/{tenant_id}/{table_name}/records?token={api_token}
```

Voorbeeld Parameters

- `tenant_id` : Jouw organisatie-ID.
- `table_name` : De naam van de tabel waar je de data van wil oproepen, bijvoorbeeld `activities`.
- `api_token` : Het **unieke token** dat aan de tabel is gekoppeld.



Opgelet:

Als de tabel niet geactiveerd is voor API gebruik, zal je geen toegang krijgen tot de gegevens. Daarboven zal je alleen de informatie te zien krijgen van de kolommen die geactiveerd zijn voor hun data te lezen.

Een voorbeeld:

De volgende call:

```
[GET] /api/v3/{tenant_id}/avo_playgrounds/records?token={playground_api_token}
```

Geeft, als de kolommen 'id', 'text_column' and 'number_column' geactiveerd zijn om te lezen, de response:

```
{
  "count": 500,
  "rows": [
    {
      "id": "de0e01df-3989-41de-a081-2576847460e2",
      "text_column": "Action Ninja",
      "number_column": 42
    },
    {
      "id": "a3bf2ca9-e81e-4af8-9d5c-e180763361fa",
      "text_column": "I Married a Forbidden Wolf",
      "number_column": 74
    },
    {
      "id": "c654f640-100b-4a51-bbcb-1e3a9a60ae2e",
      "text_column": "Curse of the Death Dreams",
      "number_column": 77
    },
    ...
  ]
}
```

Limiet & offset

Een response zal standaard maximaal 100 rijen bevatten. De count geeft het totaal van het aantal rijen/records op dat voldoet aan de opgegeven filters.

Je kan deze limiet verder ophogen en de volgende batch aan resultaten opvragen via deze parameters:

- **limit:** Het aantal resultaten je maximum zal krijgen per verzoek (Het absolute maximum dat je hier kan instellen = 1000). Voorbeeld: `&limit=10`
- **offset:** Skipt de eerste N resultaten, handig voor paginering. Voorbeeld: `&offset=10`

Sorteren

Via **sort_by** en **sort_order** kan je de resultaten sorteren.

Voorbeeld: `&sort_by=number_column&sort_order=asc`

Filters

Via extra parameters in je request, kan je meer specifieke data opvragen. Elke filterer werkt op basis van een specifieke kolom en een filterconditie, zoals 'bevat' of 'is gelijk aan'.

Een filterparameter heeft het volgende formaat:

```
filters[column][operator][]=waarde
```

- **column:** de kolom waarop je wilt filteren.
- **operator:** de filterconditie (bijvoorbeeld `contains`, `is`, `gte`).
- **waarde:** de waarde waarop je wilt filteren.

Je kunt meerdere filters combineren binnen één request via '&'. Hieronder volgen de beschikbare kolomtypes en bijbehorende filteropties.

Tekstvelden (Text, TextArea)

- **Bevat:** `filters[text_column][contains][]=waarde`
- **Bevat niet:** `filters[text_column][not_contains][]=waarde`
- **Exact overeenkomend:** `filters[text_column][is][]=waarde`
- **Komt niet overeen:** `filters[text_column][is_not][]=waarde`
- **Begin met:** `filters[text_column][starts_with][]=waarde`
- **Eindigt met:** `filters[text_column][ends_with][]=waarde`

'waarde' wordt hierbij opgegeven als tekst, spaties worden daarbij '%20' zoals in standaard url-codering. Hoofdletters spelen geen rol.

Numerieke velden (Number, Money, Date)

- **Gelijk aan:** `filters[number_column][is][]=waarde`
- **Groter dan of gelijk aan:** `filters[number_column][gte][]=waarde`
- **Kleiner dan of gelijk aan:** `filters[number_column][lte][]=waarde`
- **Groter dan:** `filters[number_column][gt][]=waarde`
- **Kleiner dan:** `filters[number_column][lt][]=waarde`
- **Binnen bereik:** `filters[number_column][is_within][]=begin_waarde to eind_waarde`

'waarde' voor number en money velden is daarbij een integer of float.

'waarde' voor date velden wordt verwacht in 'yyyy-mm-dd' formaat.

Booleaanse velden (Boolean)

- **Ja:** `filters[boolean_column][is_true][]=`
- **Nee:** `filters[boolean_column][is_false][]=`
- **Is leeg:** `filters[boolean_column][is_null][]=`
- **Is niet leeg:** `filters[boolean_column][is_not_null][]=`

Hierbij is geen 'waarde' nodig.

Selectievelden (Select)

- **Is gelijk aan:** `filters[select_column][is][]=waarde`
- **Is niet gelijk aan:** `filters[select_column][is_not][]=waarde`

'waarde' is hierbij de codenaam van de waarde (vaak in het Engels).

Meerder opties kunnen worden voorzien door ze te schrijven via '%2C'

Bijvoorbeeld: option-a%2Coption-b

Relationele velden (Relation)

Voor een filter op een lijst van relaties:

- **Is gelijk aan:** `filters[relation_column][is][]=uuid1,uuid2`
- **Is niet gelijk aan:** `filters[relation_column][is_not][]=uuid1,uuid2`

De 'waarde' bestaat uit uuid's van de relatie.

Voorbeeld

Hier is een voorbeeld van een URL met verschillende filters toegepast in één API-verzoek:

```
[GET] /api/v3/{tenant_id}/users/records?token={api_token}
&filters[name][contains][]=John
&filters[age][is][]=30
&filters[created_at][gte][]=2024-11-01
&filters[salary][is_within][]=2000 to 5000
&filters[address][is_not_null][]=
&filters[status][is][]=active
&filters[status][is][]=pending
```

Response

De response is aan JSON met alle items die aan je filters voldoen, met de kolommen/velden die geactiveerd zijn.

Voor afbeeldingen/bestanden wordt een ID teruggegeven (UUID), via die ID kan je de afbeelding zelf opvragen in verschillende formaten: [lees hier hoe dat verloopt](#).

Datums en tijden worden steeds in UTC-formaat meegegeven.

Voor meer informatie, kan je steeds mailen naar info@kampadmin.be.
