

Query API

Tutorial

Inhaltsverzeichnis

1	Allgemeines.....	3
1.1	Voraussetzungen und Sicherheitsaspekte.....	3
1.2	Query API Key.....	4
2	Aufruf und Authentifizierung.....	5
3	Echtzeitwerte lesen und schreiben.....	6
3.1	Abfrage der verfügbaren Ressourcen.....	6
3.2	Abfrage des aktuellen Wertes.....	6
3.3	Schreiben eines Wertes.....	7
4	Trendwerte lesen.....	9
4.1	Abfrage der verfügbaren Trends.....	9
4.2	Lesen eines Trends.....	9
5	Listen lesen.....	10
5.1	Abfrage der verfügbaren Listen.....	10
5.2	Lesen einer Liste.....	10
6	Fehlermeldungen und Status.....	11
7	Notizen.....	12

1 Allgemeines

Mit der myGEKKO Query API kann ein Drittsystem auf die an myGEKKO angeschlossenen Systeme und Funktionen zugreifen und dessen Zustände abfragen, sowie steuern. Die Query API wird über GET-Anfragen über http/https aufgerufen. Somit ist eine Abfrage auch über einen herkömmlichen Browser möglich. Die Query API bietet nicht nur die Möglichkeit Echtzeitwerte von System abzufragen und zu steuern, sondern kann auch für die Abfrage von historischen Trendaufzeichnungen und Listen verwendet werden.

Die Query API ist lokal auf Ihrem myGEKKO und zusätzlich über unsere myGEKKO Plus Dienste zugänglich.

Diese zwei Varianten unterscheiden sich in den unterschiedlichen Authentifizierungsmechanismen, welche in den weiteren Kapiteln durch "LOKAL" und "PLUS" aufgezeigt werden.

1.1 Voraussetzungen und Sicherheitsaspekte

LOKAL Um die lokale Query API nutzen zu können müssen folgende Voraussetzungen erfüllt werden:

- Ihr myGEKKO Gebäuderegler ist im Netzwerk angeschlossen
- "Lokaler Webserver/Query API HTTP" in den Einstellungen "Lokale Dienste"(Zahnrad > Einstellungen > 3. Tab > Lokale Dienste) ist aktiviert.
- "Lokaler Webserver" in den Einstellungen "Lokaler Webserver" (Zahnrad > Einstellungen > 3. Tab > Lokaler Webserver) ist aktiviert und Query API Benutzer wurden angelegt.



VORSICHT

Die lokale Query API ist über HTTP (Port 80) erreichbar, deshalb sollte der lokale Webzugang nur in gesicherten Netzwerken verwendet werden. Den Port 80 nie ohne Schutz öffentlich zugänglich machen.

PLUS Um die Query API über die myGEKKO Plus Services zu nutzen, muss der myGEKKO Gebäuderegler auf der myGEKKO Webseite angemeldet sein und der Dienst myGEKKO Query API abonniert sein. Weitere Informationen entnehmen Sie aus "myGEKKO OS Das Buch" im Kapitel "myGEKKO Plus Services".

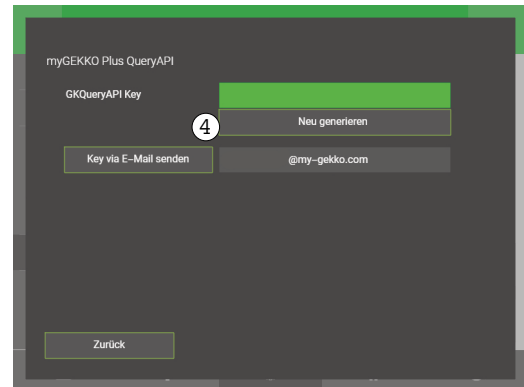


HINWEIS

Aus Sicherheitsgründen ist der Zugang über HTTP nicht erlaubt, sondern nur über HTTPS erreichbar.

1.2 Query API Key

1. Melden Sie sich am myGEKKO Gebäuderegler als Konfigurator an und wechseln in die Einstellungen (Zahnrad > Einstellungen)
2. Klicken Sie auf den 3. Tab (Globus) und schließlich auf "Plus Erweitert"
3. Tippen Sie unter dem Punkt "Plus Query API" auf den Button "Verwaltung >"
4. Um einen neuen Query API Key zu generieren auf "Neu generieren" klicken



Nachdem der Query API Key generiert wurde, erscheint ein Feld "key via E-Mail senden", mit welchem Sie sich den Query API Key zusenden lassen können.



HINWEIS

Der Button erscheint nur, falls im System Mail/SMS als Anschlussstyp "E-Mail" und als Versandtyp "myGEKKO Plus Mail" eingetragen wurde. Hierzu muss der Plus Service Mail aktiv sein.

2 Aufruf und Authentifizierung

Unten erhalten Sie einen Einblick in die verschiedenen Authentifizierungsmöglichkeiten in der lokalen Query API und der Plus Query API.

LOKAL *https://IP_IHRES_GEKKOS/api/v1/var/?
username=mustermann&password=test278!254azZM*

username: Benutzername eines Benutzers der lokalen Query API
password: Passwort des Benutzers

ODER

https://IP_IHRES_GEKKOS/api/v1/login

Es öffnet sich ein Popup-Fenster, welches Ihnen die Möglichkeit gibt den Benutzernamen und das Passwort einzugeben.

PLUS *https://live.my-gekkko.com/api/v1/var/
?username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFzc9D
&gekkoid=W96Y-WZDZ-8YTE-8FT3*

username: myGEKKO Plus Benutzername
key: Der Key wird über "Erweiterte Einstellungen" im Menü myGEKKO Plus generiert
gekkoid: Ihr myGEKKO ID wird im Menü Systeminfo (Zahnrad > Systeminfo > myGEKKO ID) angezeigt

3 Echtzeitwerte lesen und schreiben

- siehe Kapitel 3.1
- siehe Kapitel 3.2
- siehe Kapitel 3.3

3.1 Abfrage der verfügbaren Ressourcen

Die Abfrage der verfügbaren Ressourcen erfolgt über das Keyword "**var**". Folgende ROOT-Abfrage liefert alle im myGEKKO OS verfügbaren Systeme und dazugehörigen Objekte als JSON-Objekt zurück.

Über Live-Web:

<https://live.my-gekko.com/api/v1/var/>?

username=<mustermann@email.com>&key=<password>&gekkoid=<XXXX-XXXX-XXXX-XXXX>

<mustermann@email.com> = Benutzername/Email des Benutzers

<password> = Passwort des Accounts

<XXXX-XXXX-XXXX-XXXX> = myGEKKO-ID des Controllers

ODER

Lokaler Zugriff:

<https://<xxx.xxx.xxx.xxx>/api/v1/trend/?username=<admin>&password=<password>>

<xxx.xxx.xxx.xxx> = IP-Adresse der Controllers im Netzwerk

<admin> = Benutzername des Nutzers im myGEKKO OS

<mygekko> = Passwort des Nutzers im myGEKKO OS

Ein Beispiel für das Discovery der Variablen finden Sie im Anhang oder im Wiki.

Um nur einen Teil des Ressourcenbaums abzufragen, kann man auch in die Hierarchie navigieren und z. B. ...**v1/var/lights/item96**... abfragen.

Platzhalter

Sollte nicht der gesamte Name bzw. die gesamte Bezeichnung bekannt sein, kann das *-Zeichen als Platzhalter verwendet werden.

Abfrage aller Lichtgruppen

Bsp.: ...v1/var/lights/group*/name...

```
{
  "group0": {
    "name": "Grp1"
  },
  "group15": {
    "name": "Grp15"
  }
}
```

3.2 Abfrage des aktuellen Wertes

Um den aktuellen Wert abzufragen, reicht es aus die Funktion "**status**" hinter dem Ressourcenbaum dazuzuschreiben:

[https://live.my-gekko.com/api/v1/var/lights/status?
username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFzc9D&gek-
koid=W96Y-WZDZ-8YTE-8FT3](https://live.my-gekko.com/api/v1/var/lights/status?username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFzc9D&gek-koid=W96Y-WZDZ-8YTE-8FT3)

```
"lights": {  
  "item96": {  
    "sumstate": {  
      "value": "0;60;0;"  
    }  
  },  
}
```

Die Interpretation des Wertes kann aus dem Tag "value" der Ressourcenabfrage entnommen werden. Anhand des obigen Beispiels erfolgt die Interpretation folgendermaßen:

- 0; = Off
- 60; = 60% gedimmt
- 0 = nicht gesperrt
- ; = kein RGB



HINWEIS

Da die Ressourcen bei der Abfrage gleichzeitig abonniert werden und im Hintergrund vom myGEKKO OS geholt werden, kann es sein, dass die Abfrage nach ca. 2 Sekunden wiederholt werden muss, um die Werte zu erhalten. Sollte die Abfrage nicht zyklisch aufgerufen werden, deabonniert der Server diese Werte wieder.

Sollten Sie mehrere Ressourcen abfragen müssen, dann empfiehlt es sich, nicht jeden einzelnen Wert sondern immer den ganzen Zweig abzufragen.

Sollten Sie jedoch nur die Werte der Lichter benötigen, ist es ratsam den Zweig der Lichter und keine Root-Abfrage zu starten, da ansonsten unnötig Daten transportiert werden.

Array-Abfrage Wird die Rückgabe der Werte in Form eines Arrays benötigt, kann dies als Parameter "**format=array**" mitgegeben werden:

[https://live.my-gekko.com/api/v1/var/lights/
status?username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFz-
c9D&gek-koid=W96Y-WZDZ-8YTE-8FT3&format=array](https://live.my-gekko.com/api/v1/var/lights/status?username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFzc9D&gek-koid=W96Y-WZDZ-8YTE-8FT3&format=array)

```
"lights": {  
  "item96": {  
    "sumstate": {  
      "value": [0, 60, " ", 0]  
    }  
  },  
}
```

3.3 Schreiben eines Wertes

Um den aktuellen Wert zu schreiben, wird am Ende des Ressourcenzweigs die Funktion "**set**" und der Wert als Parameter "**value**" hinzugefügt.

Beispiel <https://live.my-gekkko.com/api/v1/var/lights/item96/scmd/set?value=D60&username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFzc9D&gekkoid=W96Y-WZDZ-8YTE-8FT3>

Sobald der Befehl erfolgreich ausgeführt wurde, erscheint die Meldung *OK*.

Ob eine Ressource beschreibbar ist und wie der Schreibbefehl aussieht entnehmen Sie der Beschreibung "**value**" aus der Ressourcenabfrage.

```
Bsp.: {  
    "value": "1|0|D100|T|C44782 (On|Off|Dim%|Toggle|RGB24Bit)",  
    "type": "STRING",  
    "permission": "WRITE",  
    "index": 100001  
}
```

```
1:      On  
0:      Off  
D100:   Dimmen 100 %  
C44782: Farbe laut RGB24Bit-Wert
```


4 Trendwerte lesen

- siehe Kapitel 4.1
- siehe Kapitel 4.2

4.1 Abfrage der verfügbaren Trends

Die Abfrage der verfügbaren Trendwerte erfolgt über das Keyword "**trend**". Folgende Root-Abfrage liefert alle im myGEKKO OS verfügbaren Trends und dazugehörigen Objekte als JSON-Objekte zurück.

Beispiel <https://live.my-gekko.com/api/v1/trend/?username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFzc9D&gekkoid=W96Y-WZDZ-8YTE-8FT3>

```
{
  "globals": {
    "meteo": {
      "trend0": {
        "description": "Aussen",
        "type": "REAL[]",
        "permission": "READ",
        "unit": "°C",
        "samplerate": 900
      },
    },
  },
}
```

4.2 Lesen eines Trends

Um den Trend abzufragen, reicht es aus die Funktion "**status**" hinter dem Ressourcenzweig dazuschreiben, sowie die Parameter "**tstart**", "**tend**" und "**datacount**".

Beispiel <https://live.my-gekko.com/api/v1/trend/meteo/trend0/status?tstart=2017-02-19T15:00:00+01:00&tend=2017-03-19T15:00:00+01:00&datacount=96&username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFzc9D&gekkoid=W96Y-WZDZ-8YTE-8FT3>

Die Zeit "**tstart**" und "**tend**" muss im genormten Format inkl. Zeitzone angegeben werden: 2017-02-19T15:00:00+01:00



HINWEIS

Es kann immer nur ein trend abgefragt werden, nicht ein ganzer Zweig von Trends. Des Weiteren ist die maximale Anzahl der Datenobjekte ("**datacount**") auf 500 begrenzt. Aufgrund dessen empfiehlt sich bei der Trendabfrage eine Abfrage je Tag zu machen.

5 Listen lesen

- siehe Kapitel 5.1
- siehe Kapitel 5.2

5.1 Abfrage der verfügbaren Listen

Die Abfrage der verfügbaren Listen erfolgt über das Keyword **"list"**. Folgende Root-Abfrage liefert alle im myGEKKO OS verfügbaren Trends und dazugehörigen Objekte als JSON-Objekt zurück.

Beispiel <https://live.my-gekko.com/api/v1/list/?username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFzc9D&gekkoid=W96Y-WZDZ-8YTE-8FT3>

```
"globals": {
  "meteo": {
    "lists": {
      "list0": {
        "description": "Meteolist",
        "type": "TABLE",
        "permission": "READ",
        "time-domain": "year"
      }
    }
  }
},
```

5.2 Lesen einer Liste

Um eine Liste abzufragen, reicht es aus die Funktion **"status"** hinter dem Ressourcenweig dazuschreiben sowie die Parameter **"startrow"**, **"rowcount"** und **"year"**.

Ob **"year"** angegeben werden muss, finden Sie in Kapitel "Abfrage der verfügbaren Listen" unter **"time-domain"**. Steht bei dieser Liste "year", dann gibt es für jedes Jahr eine separate Liste und das Jahr muss angegeben werden, ansonsten nicht.

```
https://live.my-gekko.com/api/v1/list/costs/item0/list0/status
?startrow=0&rowcount=100&year=2017
&username=mustermann@email.com&key=y767ZV6n8k4B7iyvYFzc9D
&gekkoid=W96Y-WZDZ-8YTE-8FT3
```



HINWEIS

Es kann immer nur eine Liste abgefragt werden, nicht ein ganzer Zweig von Listen.

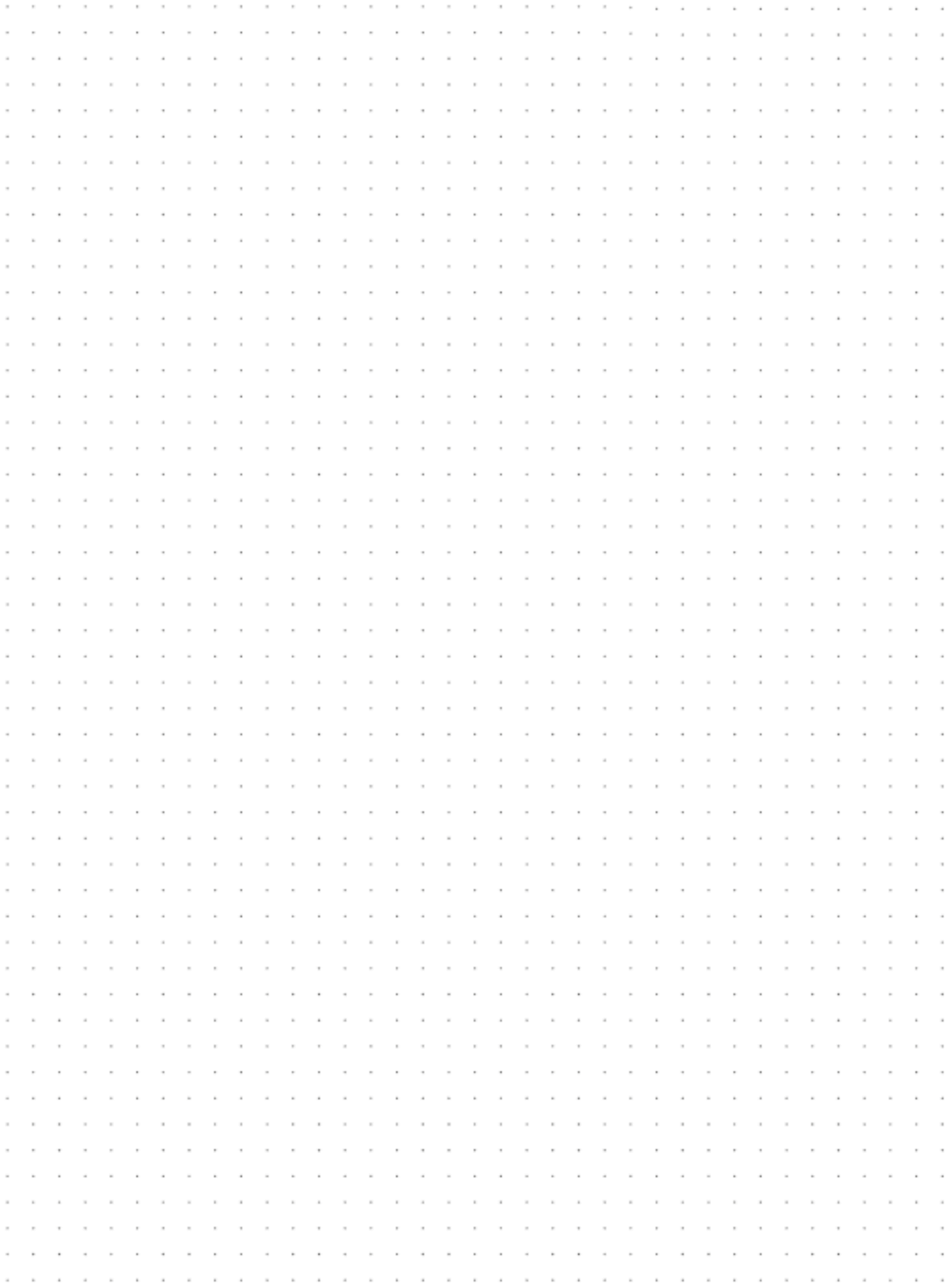
6 Fehlermeldungen und Status

Sämtliche unten genannten Fehlermeldungen erscheinen als gewohntes Alarmpopup im myGEKKO OS und werden in der Alarmhistory geloggt.

Die Query API liefert verschiedene Fehler- und Statusmeldungen:

OK (200)	Abfrage war erfolgreich
Bad Request (400)	Allgemeiner Fehler z. B. falsche Syntax
Forbidden (403)	Benutzername, Passwort oder Key sind falsch
Not Found (404)	Zu lesende bzw. schreibende Ressource konnte nicht gefunden werden
Method not allowed (405)	Falscher Befehl, z. B. Ressource ist nur lesbar, nicht schreibbar
Gone (410)	myGEKKO offline oder falsche myGEKKO ID
Too Many Requests (429)	Bei Verwendung der lokalen Query API wurde die maximale Anzahl der Sessions erreicht. Bei Verwendung der Plus Query API wurde die maximale Anzahl der Rate/Burstlimits erreicht.
No Response (444)	Die Anfrage konnte nicht im myGEKKO OS ausgeführt werden

7 Notizen





Query API

Tutorial

myGEKKO | Ekon GmbH

St. Lorenznerstr. 2
I-39031 Bruneck (BZ)
Tel. +039 0474 551820
info@my.gekko.com

www.my-gekko.com

Ekon Vertriebs GmbH
Fürstenrieder Str. 279a, D-81377 München

Vertriebsbüro Eislingen
Schillerstr. 21, D-73054 Eislingen