



Ereignisorientiertes Heizen mit der Homematic via Google Kalender

Exposé

YouTube: youtube.com/LukasvandeHaag
youtube@helduser.eu

Inhalt

1.1 Vorwort:.....	2
1.2 Ausgangslage:	2
1.3 Die Funktionsweise:.....	4
1.4 Bedingungen.....	6
2.1 Google Kalender anlegen und berechtigter Personenkreis.....	7
2.2 Systemvariablen.....	17
2.2.1 Hinweise zum Daten setzen in der CCU3.....	25
2.3 Kalenderskript.....	26
2.4 Schaltskript	27
3.1 Setzen von Daten in Systemvariablen (EQ-3 CCU).....	29
3.2 Homematic Klassik oder IP Aktoren	30
3.3 Daten eines Temperaturfühler in eine Skriptvariable setzen:	31
3.4 Wetterdienst:.....	32
3.3 Videolinks:	33

1.1 Vorwort:

In diesem Projekt geht es darum, mit Hilfe des Smarthome-Systems Homematic Klassik / Homematic IP, eine ereignisgesteuerte Beheizung von Räumen via Google Kalender zu realisieren.

Bei der Entwicklung des Skriptes wurden möglichst alle Fälle eingetragen. Auf Grund der Funktionsweise der CCU-Firmware ist es manchmal nicht ganz möglich eine Programmierung durchzuführen, wie man sie eigentlich in einer Hochsprache durchführen würde. Daher sind die Skripte an mancher Stelle eigenwillig wirkend. Dies ist der Terminologie der CCU geschuldet.

Zum Beispiel können keine Arrays angelegt, Parameter oder Daten von Programm zu Programm übergeben werden. Daher müssen alle Daten über Systemvariablen ausgetauscht werden.

Das Parsen der JSON-Daten erfolgt ohne viel Aufwand auf eigener Basis.

Vorbedingung ist, dass auf der CCU CUX-Daemon installiert ist. Hierzu bitte zum Beispiel dieser Anleitung folgen:

<https://www.homematic-inside.de/software/cuxd>

1.2 Ausgangslage:

Die Projektidee stammt von dem Zuschauer Helmut Diedrichs aus dem Jahr 2022. Seitdem steht Europa vor dem Beginn des Klimawandels sowie der kompletten Einstellung von Gaslieferungen durch die Pipelines Nordstream 1 und 2. Wobei letztere nie in Betrieb ging. Es existieren also zwei sehr gute Gründe Gas sowie CO² einzusparen.

In Gemeindehäusern oder anderen Mehrzweckräumen ist meist ein Zettel aufgehängt, auf dem zum Umweltschutz hingewiesen wird.



Zu Beginn der Entwicklung der Skripte gab es die folgenden Eckpunkte:

- Um die Möglichkeit zu schaffen, dass die Idee auch von Vereinen sowie Kirchengemeinden verwendet werden kann, war ein wichtiger Eckpunkt die Lösung so zu gestalten, dass sie rein auf Basis der CCU3 oder eine Raspberry Matic lauffähig ist. Ohne dass weitere Bausteine wie ein IO-Broker oder dergleichen benötigt wird.
- Die Skripte sollten so gestaltet sein, dass sie smarthomeübergreifend ohne Umprogrammierung benutzt werden können.
- Das Schalten der Heizungen soll über einen Kalendereintrag des Google Kalender möglich sein, weil das Erstellen und Benutzen eines solchen kostenlos und auf somit auf jedem Smartphone nutzbar ist. Weiter bringt ein Google Kalender einige Punkte wie Zugriffsschutz,

Berechtigungen und der Betrieb auf einem Server mit sich, sodass man sich hiermit nicht weiter beschäftigen muss.

- Der Kalender sollte möglichst auch noch zur Planung von Raumbelagungen nutzbar bleiben.
- Es sollten hausbedingte Begebenheiten einstellbar sein und berücksichtigt werden.

-Es soll eine Raumgrundtemperatur einstellbar sein.

1.3 Die Funktionsweise:

Zum Betrieb wird eine CCU3 mit Heizungsaktoren sowie der Cux-Daemon benötigt.

Grundsätzlich kann mit diesen Skripten auch Licht oder Lüfter gesteuert werden.

Grundsätzlich ist es möglich alle „binären“ Verbraucher zu Schalten. Also solche die entweder Ein oder ausgeschaltet sind.

Aktoren die Ein- oder ausgeschaltet werden, werden Terminscharf geschaltet. Das heißt es gibt für diese Aktoren keine Offsetzeiten.

Zu Beginn muss eine Vielzahl an Systemvariablen angelegt und mit Daten gefüllt werden. Weiter sind zwei Programme anzulegen, die jeweils ein Skript ausführen.

Über Offsetwerte ist es möglich, je nach Hausbedingungen, das Heizen vor einem Termin zu beginnen und vor dem Ende selbigen zu beenden. Hierbei wird die Außentemperatur beachtet und mit einbezogen. Diese kann über einen externen Homematic-Sensor gemessen oder von einem Wetterdienst der REST-API* abgefragt werden.

*Hierzu existiert ein anderes Projekt des Autors, in dem dieses Thema behandelt und umgesetzt wurde.

Ein Funktionsbeispiel:

In einem Kalender existiert ein Termin:



Der Termin, der zeitgleich die Raumbellegung darstellt, stellt im Titel die Raumbezeichnung dar. In diesem Beispiel der Raum EG-01. In der CCU ist ein Steuerzeichen festzulegen, welches auch beim Erstellen des Termins mit eingestellt werden muss. In diesem Fall sagt der Titel „EG-02_“ aus, dass eben der Raum zu diesem Zeitpunkt belegt ist und mit dem Steuerzeichen „-“ wird festgelegt, dass geheizt werden soll. Ebenso sagt der Eintrag im Titel „EG-01%“ aus das zu dieser Zeit die Verbraucher in dem Raum geschaltet werden. Es ist theoretisch möglich durch entsprechendes Anlegen der Variablen in der CCU in einem Termin einem Raum zu Heizung und zum Beispiel eine ganze Etage zu beleuchten.

Eine Mischung der Einträge im Titel des Termins ist nicht möglich. Wird zum Beispiel der Titel „EG-01_&“ gewählt, weil mal Heizen und Schalten will, so wird das nicht funktionieren. Der Termin wird von der CCU ignoriert!!

Es muss direkt OHNE Leerzeichen hinter der Raumbezeichnung stehen. Ein freistehendes Steuerzeichen wird nicht beachtet!

In diesem Beispiel geben wir eine Außentemperatur von 11.4 °C, sowie eine Grundtemperatur von 18.5 °C an. In der CCU ist ebenso ein Offsetwert für die Außentemperatur im Bereich von 6°C - 12°C von 15 Minuten festgelegt.

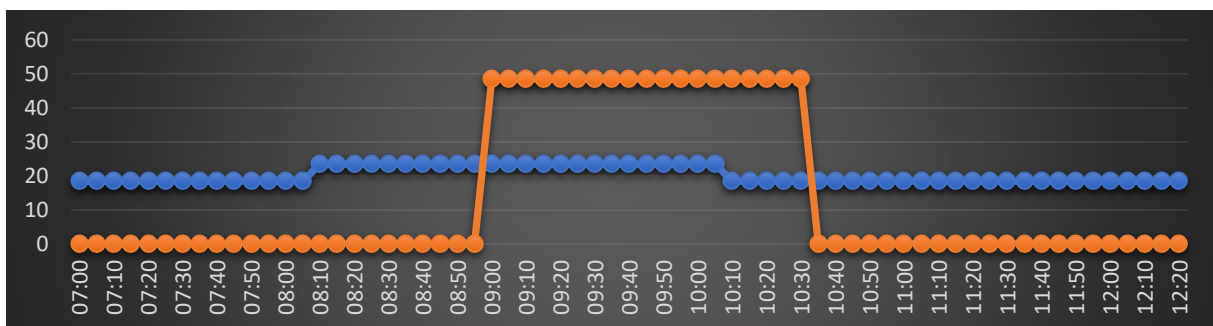
Der grundsätzliche Offset zum Einschalten ist auf 35 Minuten gesetzt.

Der Offset zum Ausschalten ist auf 25 Minuten gesetzt.
Somit kommt man im Beispiel auf einen realen Heizhorizont von
08:10 Uhr - 10:05 Uhr

Die folgende Grafik zeigt den Termin in orange, wie er im Kalender eingestellt ist, sprich von 09:00 Uhr - 10:30 Uhr.

In blau dargestellt die Soll-Raumtemperatur.

Es ist zu sehen, wie die Heizung des Raums vorgelagert ein- und ausgeschaltet wird. Die Ein/Aus Linie für ein Aktor z.B. für ein Lüfter oder Licht wird genau an der Gelben verlaufen.



1.4 Bedingungen

Zum Betrieb der Skripte gibt es folgende Punkte zu beachten.

- Überlappende Termine für einen Raum sind möglichst zu unterbinden. Findet daher eine Veranstaltung von 09:00 Uhr -11:00 Uhr und direkt danach von 11:00 Uhr -14:00 Uhr statt, so ist im Kalender ein Termin von 09:00 Uhr -14:00 Uhr einzustellen.
- Ein Termin sollte mindesten 2 Stunden vor Beginn der Veranstaltung im Kalender eingestellt sein.
- Sollte ein Termin kurzfristig abgesagt werden, so ist dieser möglichst zeitnah zu entfernen. Er muss spätestens 62 Minuten vor dem

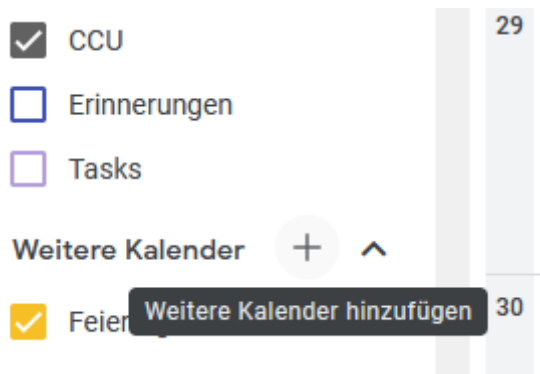
geplanten Beginn gelöscht oder das Steuerzeichen im Titel entfernt sein.

2.1 Google Kalender anlegen und berechtigter Personenkreis

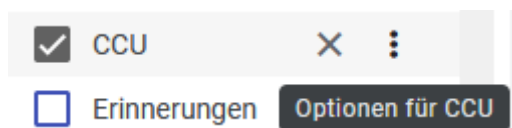
Anleitung zum Google Kalender:

Zur Benutzung des Skripts muss ein Google Kalender vorhanden sein. Dieser Kalender muss in einem Google-Profil freigegeben werden.

1. Im Browser auf das Google Profil klicken, dann den Kalender öffnen. Links im Bild mit dem Pluszeichen einen Kalender anlegen.



2. Im Kalender auf die drei Punkte klicken



dort auf „Einstellungen und Freigabe“ klicken

3.

Für bestimmte Personen oder Gruppen freigeben

The screenshot shows the 'Freigabe für bestimmte Personen oder Gruppen' (Share with specific people or groups) interface. It features three rows, each with a placeholder icon for a person or group. The first two rows have a 'Termine ändern' (Change events) button and a close icon (X). The third row has a dropdown menu labeled 'Änderungen vornehmen und Freigabe verwalten' (Make changes and manage sharing). Below these rows is a button labeled '+ Personen und Gruppen hinzufügen' (Add people and groups).

Weitere Informationen zum [Freigeben Ihres Kalenders für andere](#)

„Für bestimmte Personen oder Gruppen freigeben“ die Nutzer mit ihrer Gmail-Adresse für diesen Kalender eintragen. Sie benötigen die Rechte „Termine ändern“.

4.

Zugriffsberechtigungen für Termine

The screenshot shows the 'Zugriffsberechtigungen für Termine' (Event access permissions) interface. It has a checkbox labeled 'Öffentlich freigeben' (Publicly share) which is checked. To the right is a dropdown menu labeled 'Alle Termindetails anzeigen' (Show all event details). Below these is a button labeled 'Link zum Freigeben abrufen' (Get sharing link).

Weitere Informationen zur [Freigabe Ihres Kalenders](#)

„Zugriffsberechtigung für Termine“ öffentlich freigeben auswählen. Dies wird benötigt, damit die CCU auf den Kalender zugreifen kann.

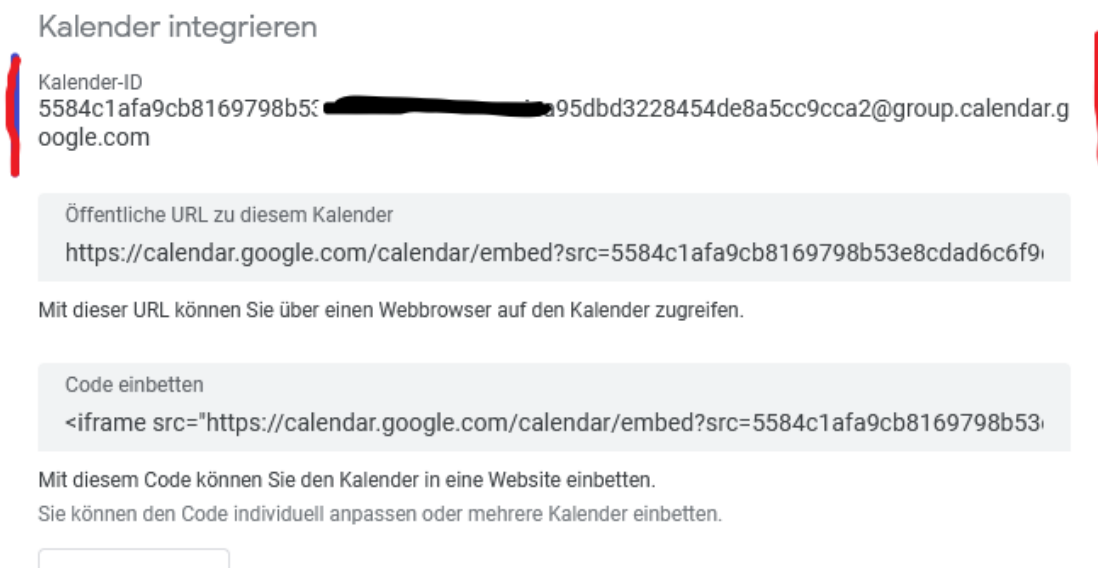
5.

The screenshot shows a dialog box titled 'Freigabelink für Ihren Kalender' (Sharing link for your calendar). It contains the text 'Mit diesem Link können nur Nutzer, die Sie zulassen, auf Ihren Kalender zugreifen.' (Only users you allow can access your calendar with this link). Below the text is a large rectangular input field for the link. At the bottom right are two buttons: 'Abbrechen' (Cancel) and 'Link kopieren' (Copy link).

Auf den Button „Link zum Freigeben abrufen“ klicken und den darin enthalten Link kopieren. Mit diesem Link können andere Google Nutzer auf den Kalender zugreifen.

Scrollen sie auf dieser Seite weiter nach unten bis zum Bereich „Kalender integrieren“, dort befindet sich der Eintrag „Kalender-ID“. Kopieren sie den dort stehenden Link. Dieser muss im Kalenderskript in der Variable „Kalender“ kopiert werden.

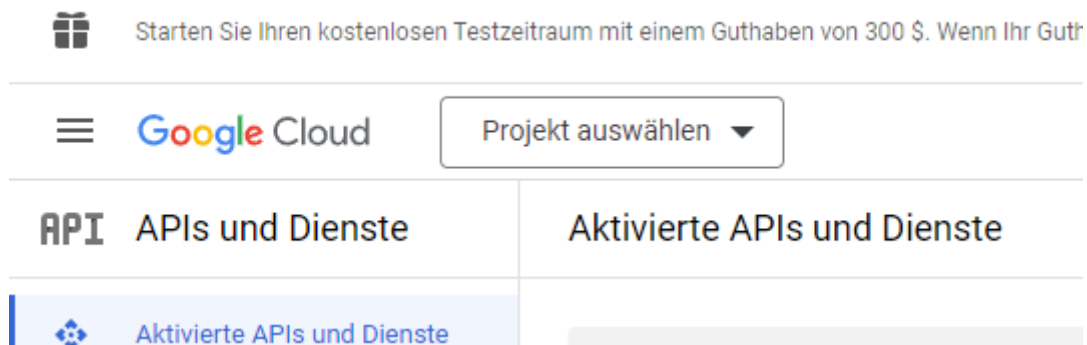
Der Link im Kalender-Skript muss aussehen wie der hier im Bild dargestellten Link. Ein https Link kann im CCU-Skript nicht verarbeitet werden.



6. Im Browser muss der Nutzer mit dem Google-Konto angemeldet sein. Danach die folgende Website eingeben:

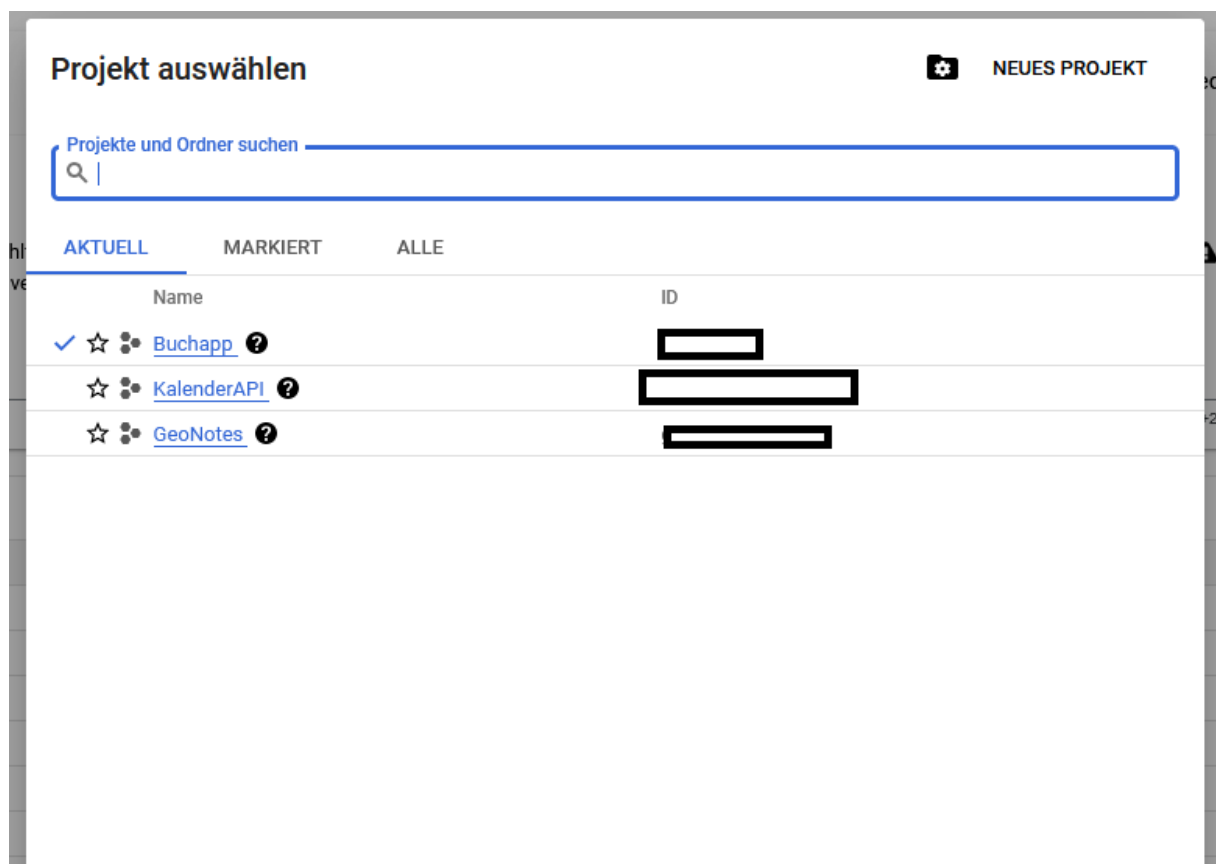
<https://console.cloud.google.com/apis/dashboard?>

7.



Im linken oberen Rand auf „Projekt auswählen“ klicken. Ggf. befindet sich hier schon ein vorhandenes Projekt, das angezeigt wird.

8.



Auf „Neues Projekt“ klicken und ein Projekt anlegen:
Dazu einen entsprechenden Namen vergeben.

Neues Projekt



In Ihrem Kontingent sind noch 22 projects offen. Fordern Sie eine Erhöhung an oder löschen Sie Projekte. [Weitere Informationen](#)

[MANAGE QUOTAS](#)

Projektname *

My Project 70445



Project ID: sunny-inn-391715. Sie kann später nicht mehr geändert werden.

[BEARBEITEN](#)

Speicherort *

Keine Organisation

[DURCHSUCHEN](#)

Übergeordnete Organisation oder übergeordneter Ordner

ERSTELLEN

ABBRECHEN

Auf „Erstellen“ klicken.

Bei mehreren Projekten in den Profilen das angelegte auswählen.

Projekt auswählen



NEUES PROJEKT

🔍 Projekte und Ordner suchen

AKTUELL

MARKIERT

ALLE

Name

ID

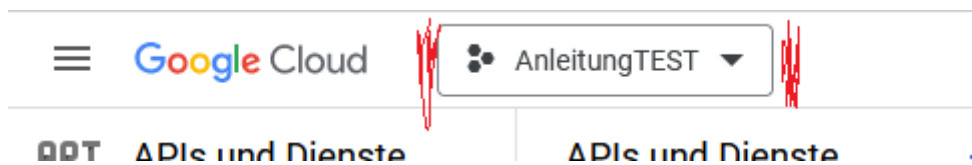
☆ [AnleitungTEST](#) ?

anleitungtest

✓ ☆ [Buchapp](#) ?

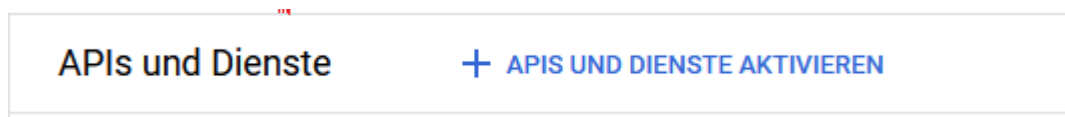
☆ [KalenderAPI](#) ?

☆ [GeoNotes](#) ?



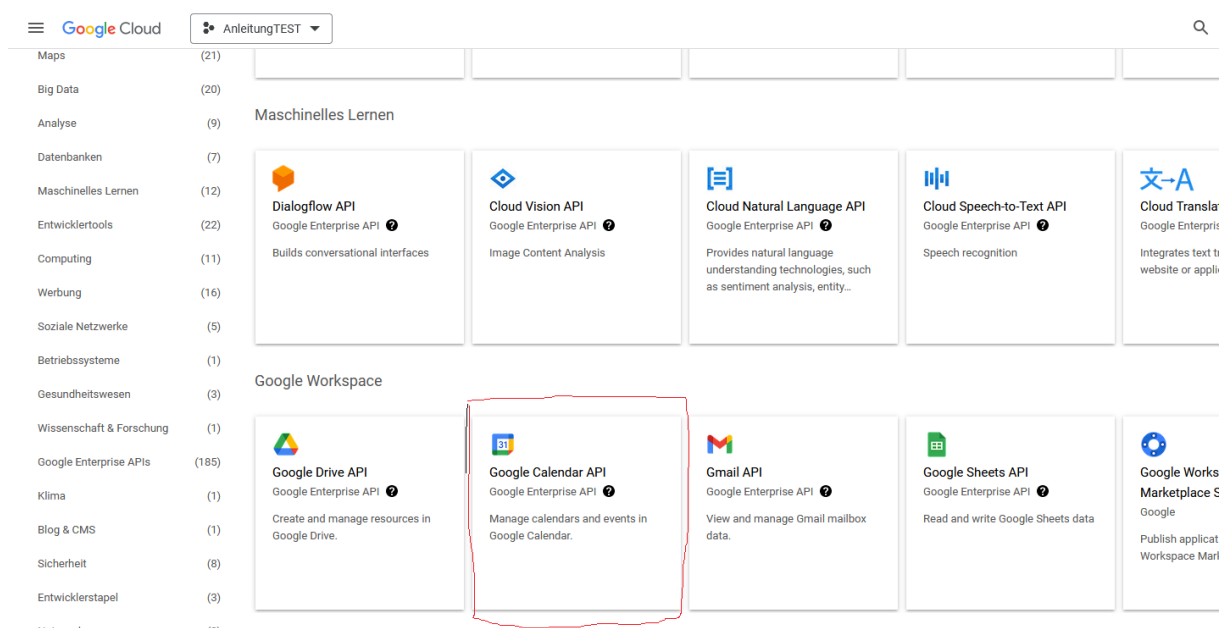
Name des Projekts wird oben angezeigt.

9.



Auf „APIS und Dienste aktivieren“ klicken.

10.




Dort die API für „Google Calendar API“ auswählen.

← → ↻ https://console.cloud.google.com/apis/library/calendar-json.googleapis.com?proj...

☰ Google Cloud AnleitungTEST ▾

← Produktdetails



Google Calendar API

[Google Enterprise API](#)

Manage calendars and events in Google Calendar.

AKTIVIEREN **API TESTEN** ↗

ÜBERSICHT DOKUMENTATION SUPPORT ÄHNLICHE PRODUKTE

Übersicht

With the Calendar API, you can display, create and modify calendar events as well as work with many other calendar-related objects, such as calendars or events.


Weitere Information

Typ: SaaS & API

Dann auf AKTIVIEREN klicken.

11.

Zur Verwendung dieser API sind möglicherweise Anmeldedaten erforderlich. **ANMELDEDATEN ERSTELLEN**



Google Calendar API

The Google Calendar API lets you manage your calendars and events.

Von Google Enterprise API ⓘ

Dienstname	Typ	Status	OVERVIEW ↗	QUICKSTARTS ↗	API REFERENCE ↗
calendar-json.googleapis.com	Öffentliche API	Aktiviert			

MESSWERTE KONTINGENTE ANMELDEDATEN

Auf „Anmeldedaten erstellen“ klicken.

12.

Anmeldedaten erstellen

1 Art der Qualifikation

Welche API verwenden Sie?

Verschiedene APIs verwenden unterschiedliche Authentifizierungsplattformen. Einige Anmeldedaten können eingeschränkt werden, um nur bestimmte APIs aufzurufen.

API auswählen *

Google Calendar API ▼

Auf welche Daten wird zugegriffen? *

Für die Autorisierung des Zugriffs sind je nach Art der angeforderten Daten unterschiedliche Anmeldedaten erforderlich. [Weitere Informationen](#)

☒ Nutzerdaten ?

Zu einem Google-Nutzer gehörige Daten, z. B. die E-Mail-Adresse oder das Alter. Nutzereinzwilligung ist erforderlich. Dadurch wird ein OAuth-Client erstellt.

☐ Anwendungsdaten

Daten Ihrer eigenen Anwendung, z. B. das Cloud Firestore-Back-End Ihrer Anwendung. Dadurch wird ein Dienstkonto erstellt.

WEITER

2 Meine Anmeldedaten

FERTIG

ABBRECHEN

Anmeldedaten erstellen.

Bei der API-Auswahl muss „Google Calendar API“ eingestellt sein. Danach auf „Fertig“ klicken.

13.



Im Menü auf „Anmeldedaten“ klicken. Im Fenster dann oben auf „+ ANMELDEDATEN ERSTELLEN“ klicken.
Danach auf „API-Schlüssel erstellen“ klicken. Es wird nun ein Schlüssel erstellt.

14.



Den Schlüssel in „Mein API-Schlüssel“ herauskopieren. Wird für das Skript benötigt.

15.

Dann auf „Bearbeiten Sie den API-Schlüssel“ klicken.

16.

Schlüsseleinschränkungen



Dieser Schlüssel ist nicht eingeschränkt. Damit eine nicht autorisierte Verwendung verhindert wird, sollten Sie einschränken, wo und für welche APIs er verwendet werden kann. [Weitere Informationen](#)

Überg
um ih

Erstel

Anwendungseinschränkung festlegen

Anwendungseinschränkungen begrenzen die Nutzung eines API-Schlüssels auf bestimmte Websites, IP-Adressen, Android- oder iOS-Apps. Sie können pro Schlüssel eine Anwendungseinschränkung festlegen.

- ☒ Nicht vorhanden
- ☐ Websites
- ☐ IP-Adressen
- ☐ Android-Apps
- ☐ iOS-Apps

API-Einschränkungen

API-Einschränkungen geben die aktivierten APIs an, die von diesem Schlüssel aufgerufen werden können.

- ☐ Schlüssel nicht einschränken
Dieser Schlüssel kann jede API aufrufen.
- ☒ Schlüssel einschränken

1 API

Ausgewählte APIs:

Google Calendar API

Hinweis: Es kann bis zu 5 Minuten dauern, bis Einstellungen wirksam werden

SPEICHERN

ABBRECHEN

Hier „Schlüssel einschränken“ anklicken und die Calendar API auswählen und speichern.

Nun ist Google so weit konfiguriert...

2.2 Systemvariablen

Thermostate den Räumen zuordnen. Raumvariable

ZZ_EG01Heizen
ZZ_EG01Schalten
ZZ_EG02Heizen
ZZ_EG02Schalten
ZZ_EG03Heizen
ZZ_EG03Schalten

1;Heizung_Essbereich
0;Licht_Essbereich;Licht_Küche;Licht_Decke
0;Heizung_Garderobe
1;Licht_Garderobe
0;Heizung_Wohnzimmer
0;Licht_Wohnzimmer1

Art: Zeichenkette

Für jeden Raum oder Stockwerk ist eine Variable als Zeichenkette anzulegen.

Der Inhalt der Variable muss die Namen (einer oder mehrere) der Kanäle der Heizungsthermostate enthalten, sowie eine durch Semikolon vorangestellte 0.

Mehrere Aktoren sind auch durch ein Semikolon zu trennen. Es ist dabei unerheblich, ob Namen des Kanals oder Seriennummer mit Kanalnummer. Jeder Kanalname ist auch mit einem Semikolon zu trennen.

Es ist möglich, Thermostate mehrfach in Raumzuordnungen einzugeben. Zum Beispiel für einen „Raum“, der ein komplettes Stockwerk darstellt.

Es ist unerheblich, ob es sich um einen Raum oder ein Stockwerk handelt, beide werden gleichbehandelt.

Das Beispiel enthält den Raum EG0.12, dieser besitzt zwei Thermostate:

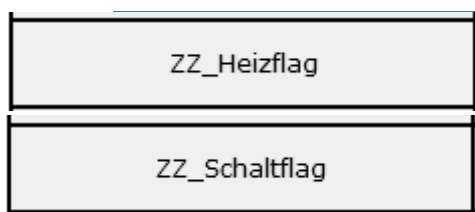
Sodann muss die Variable lauten: **0;012Heizung1;012Heizung2**

Es ist zwingend nötig die 0 voranzustellen! Sie ist ein Steuerzeichen.

Grundsätzlich gilt für ALLE Variablen, dass die hier dargestellten Namen also Vorschlag anzusehen sind. Es ist möglich die Variablen völlig anders zu benennen.

Beachten sie zwingend dass für jeden Raum zwischen den Aktoren zum Heizen und zum Schalten getrennt eine Variablen mit entsprechendem Inhalt anzulegen ist. Eine Vermischung der Aktoren in einer Variablen führt dazu dass das Schaltskript nicht korrekt ausgeführt wird und von der CCU abgebrochen wird. Benennen sie jede Variable entsprechend so dass sie sie gut auseinanderhalten können.

Schaltflagzeichen für Kalender



Art: Zeichenkette

Der Inhalt der Variable ist auf das Zeichen zu setzen, welches später im Kalender als Steuerzeichen verwendet werden soll

Beispiel:

In der Variable ist ein _ (Unterstrich) eingegeben. So müssen alle Kalendereinträge, an denen geheizt werden soll, dieses Zeichen am Ende besitzen.

Eintrag „EG-05_“ ergibt, dass Raum 05 im EG geheizt wird.

Eintrag „EG-05“ ergibt, dass Raum 05 im EG NICHT geheizt wird.

Dieser Eintrag kann als reiner Belegungstermin benutzt werden.

Dies ist analog für das Schaltzeichen durchzuführen.

Grundtemperatur

ZZ_Grundtemp

Art: Integer

Der Inhalt der Variable ist auf die Grundtemperatur zu setzen.

Alle Heizungen, die ausgeschaltet werden, werden auf den Wert der Variable gesetzt.

Offset-Werte für Ein/Aus Zeit

In der Variable werden Zeiten im Wert von Minuten einzugeben.

OffsetAus ergibt, dass die Heizungen in diesem Haus / der Anlage um den Wert der Variable früher ausgeschaltet werden.

OffsetEin ergibt, dass die Heizungen in diesem Haus / der Anlage um den Wert der Variable früher eingeschaltet werden.
Diese Einstellung ist im Skript Schalten vorzunehmen.

Beispiel:

Terminiert wird ein Raum von 15:00 Uhr - 17:00 Uhr und die Variablen sind jeweils auf 25 Minuten gesetzt. Das bedeutet, dass die Heizung um 14:35 Uhr ein- und um 16:35 Uhr ausgeschaltet wird.

Die Werte für OffsetAus und OffsetEin sind im Kopf des Schaltskripts einzugeben! Sie müssen aber nicht geändert werden und können auch wie dargestellt belassen werden.

```
#!/usr/bin/perl
# Ausschaltzeitpunkt Offset (Gibt die Minuten an die die Heizung grundsätzlich früher ausschalten soll)
integer OffsetAus=25;
# Einschaltzeitpunkt Offset (Gibt die Minuten an die die Heizung grundsätzlich früher einschalten soll)
integer OffsetEin=30;
```

Temperatur Offset

In den Variablen sind die Offsettemperaturen anzugeben bei welcher Außentemperatur um wieviel Minuten die Heizung früher eingeschaltet werden soll. Diese Einstellung ist im Skript Schalten vorzunehmen.

Ein Beispiel:

Terminiert wird ein Raum von 15:00 Uhr -17:00 Uhr und die Variablen für OffsetEin und OffsetAus sind jeweils auf 25 Minuten gesetzt. Die Außentemperatur beträgt 12 °C. Die Variable 4 für den Bereich von 6 - 12 °C Grad ist auf einem Wert von 15 Minuten gesetzt.

Das bedeutet, dass die Heizung um 14:20 Uhr (15:00 Uhr abzüglich 25 Minuten Offset und 15 Minuten Temperaturoffset) ein- und um 16:35 Uhr ausgeschaltet wird.

Die Werte sind im Kopf des Schaltskripts einzugeben! Sie müssen aber nicht geändert werden und können auch wie dargestellt belassen werden.

```
!//#####--Eingabe der Offsetwerte---###--Alle Eingaben in Minuten---#####
!// Aussentemperatur kleiner -8 °C
integer Offset1=30;
!// Aussentemperatur zwischen -8 bis -2 °C
integer Offset2=25;
!// Aussentemperatur zwischen -1 bis +5 °C
integer Offset3=20;
!// Aussentemperatur zwischen 6 und 12 °C
integer Offset4=15;
!// Aussentemperatur zwischen 13 und 18 °C
integer Offset5=10;
!// Ausschaltzeitpunkt Offset (Gibt die Minuten an die die Heizung grundsätzlich früher ausschalten soll)
integer OffsetAus=25;
!// Einschaltzeitpunkt Offset (Gibt die Minuten an die die Heizung grundsätzlich früher einschalten soll)
integer OffsetEin=30;
!// Temperaturgrenze zum Heizen die am Vortag um 20Uhr unterschritten sein muss
integer TG=17.5;
!//#####--Ende Variabler Bereich---#####
```

Heizen bei entsprechender Außentemperatur

Da ein Kalendereintrag in einer Serie vorgelegt werden kann ist es möglich das im Sommer oder Warmen Außentemperaturen geheizt wird, obwohl es nicht nötig ist.

Daher ist es möglich in der Variablen „TG“ im Schaltskript eine Grenztemperatur einzugeben.

Je nach dem ob am Vortag um 20 Uhr diese Temperatur überschritten oder unterschritten war wird geheizt.

Legen sie die Grenztemperatur im Skript fest:

```
!// Temperaturgrenze zum Heizen die am Vortag um 20Uhr unterschritten sein muss
integer TG=17.5;
!//#####--Ende Variabler Bereich---#####
```

Temperatur Flag zum Heizen

ZZ_TemperaturFlagHeizen

Art: Zeichenkette

Legen sie eine Variable mit einem sprechenden Namen in der CCU an.
setzen sie den Inhalt der Variablen auf den Wert: **1;1**
Diese Werteliste wird für das Schalskript benötigt. Hier wird
entsprechende festgehalten ob an einem Tag geheizt wird oder nicht.

Raumliste

ZZ_OrteVarListe

Art: Zeichenkette

In der Variable werden die Bezeichnungen der Räume eingegeben.
Jeder Raum ist dabei mit einem Semikolon zu trennen.

Beispiel:

EG-01;EG-02;EG-03;OG-01;OG-02;OG-03;EG

Außentemperatur

Z_Aussentemp

Art: Integer

In der Variable wird die Außentemperatur gespeichert.
Diese Daten können entweder von einem Wetterdienst per API
abgerufen werden oder von einem Außenfühler entnommen werden.
Es ist ein Programm anzulegen, welches die aktuelle
Außentemperatur in die Systemvariable notiert.

Liste der Raum-Aktoren, Liste der Raumvariablen

ZZ_AktorenVarListeHeizen
ZZ_AktorenVarListeSchalten

Art: Zeichenkette

Es sind zwei Variable anzulegen. In dieser ist eine Liste der Variablenname der Raumvariablen zuschreiben.

Beispiel:

ZZ_EG01Heizen;ZZ_EG02Heizen;ZZ_EG03Heizen;ZZ_OG01Heizen
ZZ_EG01Schalten;ZZ_EG02Schalten;ZZ_EG03Schalten;ZZ_OG01Schalten

Wird eine Raumvariable namens „ZZ-EG01Heizen“ angelegt, die die Liste der Aktoren enthält, so muss deren Name hier abgelegt werden. Alle variablen Namen sind mit einem Semikolon zu trennen. Die Variable der Raumvariablen und der Raumliste müssen gleich sortiert sein!

Beispiel:

In der Raumliste ist an dritter Stelle der Raum „EG-03“ gespeichert, so muss an dritter Stelle in der Variable „Raumvariablen“ der Variablenname stehen, dessen Inhalt die Aktoren des Raum EG-03 entsprechen.

Auch diese Variablen ist für das Schalten analog zum Heizen zu führen.

Schaltliste

ZZ_Schaltliste

Art: Zeichenkette

In der Variable sind keine Eingaben zu machen. Diese werden von Kalenderskript durchgeführt.

Das Skript legt hier die ausgelesenen Kalendertermine aufbereitet ab. Diese Variable ist die Datenübergabe an das Schaltskript.

EG-02;1688411700;1688417100;EG-03;1688411700;1688417100;OG-02;1688411700;1688417100;OG-03;1688414400;1688418000;
--

Zusammenspiel der Variablen

Zum besseren Verständnis ein Beispiel über das Zusammenspiele der Variablen.

Wird ein Schaltauftrag der Schaltliste entnommen so wird in der Ortliste nach dem Bezeichner gesucht. Wird dieser gefunden wird entweder aus den Aktoren für das Schalten oder für das Heizen den Namen der Aktorenliste entnommen. Daraufhin wird die Aktorenliste durchlaufen und jeder Aktor geschaltet.

Man muss sie die AktorenVarListe als eine Liste mit Gruppenbezeichner vorstellen. Wobei die Gruppenmitglieder in der eigentlicher Aktorenliste (z.b ZZ_EG02Heizen) stehen. Auch ein einzelner Aktor muss so aufgelistet werden.

Das folgende Beispiel zeigt eine Konfiguration von Räumen zum Heizen und eine Außenbeleuchtung.

Gibt es für einen Ort nur „eine Sache“ also entweder Heizen oder Schalten so muss in der entsprechend anderen Liste ein Platzhalter geschrieben werden. Denn die Liste für Heizen und Schalten müssen zwingend synchron der OrteVarListe sein, sonst funktioniert das komplette Skript nicht.

ZZ_AktorenVarListeHeizen
ZZ_AktorenVarListeSchalten

ZZ_EG01Heizen;NULL;ZZ_EG02Heizen;ZZ_EG03Heizen;ZZ_OG01Heizen
ZZ_EG01Schalten;ZZ_Aussenbeleuchtung;ZZ_EG02Schalten;ZZ_EG03Schalten;ZZ_OG01Schalten

Es ist zu sehen das in der ListeSchalten an zweiter Stelle die Außenbeleuchtung steht. Da im Außenbereich nicht geheizt wird befindet sich an der zweiten Stelle in der Liste ...Heizen ein Platzhalter. Es ist unerheblich was hier als Platzhalter benutzt wird.

In der Liste ...Schalten befindet sich die Bezeichner der Variable, in der die Namen der Aktoren zu finden sind.

ZZ_EG01Schalten

0;Licht_Essbereich;Licht_Küche;Licht_Decke
--

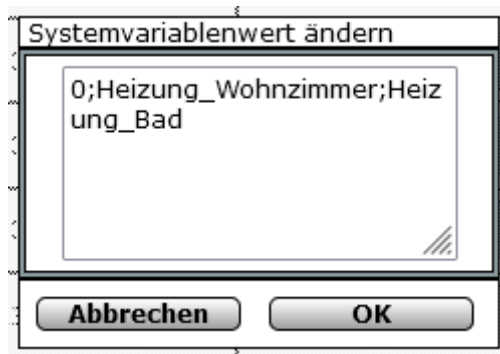
Gibt es einen Termin EG01& der ein Schalten im EG01 darstellen soll so wird das erste Element der Liste ...Schalten benutzt, was wiederum den Namen der Aktoren Liste darstellt. Also in diesem Beispiel die Variablen ZZ_EG01Schalten. In dieser Variablen befinden sich die Aktoren „Licht_Essbereich, Licht_Küche sowie Licht_Decke. Somit wird mit dem Eintrag EG01& diese drei Aktoren geschalten.

2.2.1 Hinweise zum Daten setzen in der CCU3

Leider gibt es in der Firmware der CCU3 von EQ-3 zur Firmware der Raspberry Matic Unterschiede.

In der CCU3 von EQ-3 können Systemvariablen im WebUI angelegt werden. Eine Wertzuweisung kann im WebUI aber nicht durchgeführt werden. Bitte hierzu Kapitel 3.1 beachten!

Benutzer der Raspberry Matic können die Daten im WebUI eingeben. Hierzu im Menü „Status und Bedienung“ auf „Systemvariablen“ klicken. In der Liste die entsprechende Systemvariable im Feld „Status“ anklicken. Es öffnet sich ein Dialog. Hier können die Daten eingegeben werden. Die Eingabe mit „OK“ bestätigen.



2.3 Kalenderskript

Kopieren Sie das Skript aus der Datei Kalender.c heraus (Die Datei lässt sich mit einem normalen Texteditor öffnen)

Legen Sie ein Programm an und fügen sie das Skript ein.

Geben Sie in der Bedingung zur Triggerung des Programms ein, dass es zu jeder Stunde einmal ausgeführt wird.

Im Kopf des Skripts geben Sie die von Ihnen vergebenen Variablennamen ein. Weitere Veränderungen sind zu unterlassen.

2.4 Schaltskript

Kopieren Sie das Skript aus der Datei Schalten.c heraus (Die Datei lässt sich mit einem normalen Texteditor öffnen)
Legen Sie ein Programm an und fügen sie das Skript ein.
Geben Sie in der Bedingung zur Triggerung des Programms ein, dass

es alle 5 Minuten einmal ausgeführt wird.

Zeitmodul einstellen

Zeit

☒ Zeitspanne

☐ Beginn: 18:18
☐ Ende: 18:48
☒ Ganztägig
☐ Von Sonnenaufgang, versetzt um min, bis Sonnenuntergang, oder für min
☐ Von Sonnenuntergang, versetzt um min, bis Sonnenaufgang, oder für min

☐ Zeitpunkt

18:18

Serienmuster

☐ Einmalig

Alle 5 Minuten

☒ Zeitintervall
☐ Täglich
☐ Wöchentlich
☐ Monatlich
☐ Jährlich

Gültigkeitsdauer

Beginn 13.07.2023

☒ Kein Enddatum
☐ Endet nach Terminen
☐ Endet am

Abbrechen

OK

Im Kopf des Skripts geben Sie die von Ihnen vergebenen Variablennamen ein. Weitere Veränderungen sind zu unterlassen. Geben sie zusätzlich in der Variable „aktorenflag“ an, ob es sich bei den verwendeten Aktoren um Aktoren der Homematic Klassik oder Homematic IP handelt. Beachten sie hierzu Punkt 3.2 dieses Dokument.

```

!//-----Beginn Variabler Bereich-----!//
!//Name der Systemvariable für die Ortliste
string NameOrtliste="ZZ_OrtVarListe";
!//Name der Systemvariable für die Schaltliste
string NameSchaltliste="ZZ_Schaltliste";
!//Name der Systemvariable für Liste der Variablenliste Schalten
string NameAktS="ZZ_AktorenVarListeSchalten";
!//Name der Systemvariable für Liste der Variablenliste Heizen
string NameAktH="ZZ_AktorenVarListeHeizen";
!//Name der Systemvariable für die Grundtemperatur
string Grundtemp="ZZ_Grundtemp";
!// Name der Systemvariable für die Aussentemperatur
string Aussentemp="Z_Aussentemp";
!// Name der Systemvariable für die Raumtemperatur
string Raumtemp="ZZ_Raumtemp";
!// Name der Systemvariable für das TemperaturFlag zum Heizen
string THF="ZZ_TemperaturFlagHeizen";
!// Gibt an ob es sich im Homematic Klassik oder Homematic IP Aktoren handelt
!//Geben sie eine 0 an für Klassik Aktoren, geben sie eine 1 an für IP Aktoren
boolean aktorenflag = 0;

```

3.1 Setzen von Daten in Systemvariablen (EQ-3 CCU)

Um Daten in Systemvariablen der CCU3 von EQ-3 zu setzen ist eine Hilfshandlung mittels Hilfs-Skript erforderlich. Nutzer der Raspberry Matic können die Daten direkt im WebUI setzen.

Kopieren sie das Skript aus der Datei Datensetzen.c in das Fenster „Skripte testen“ des WebUI.

In der Liste der Programme unter „Programme und Verknüpfungen“ gibt es am unteren Rand den Button „Skripte testen“. Hierrüber kann das Fenster geöffnet werden.

Geben sie jeweils den Namen der Variable und den zu setzenden Wert in die Befehlszeile ein.

```
#!/Eingabe des Namen der Variable
#!/Schreiben sie den Namen der Variable zwischen die Anführungszeichen!
string Variablename = "" ;

#!/Eingabe des Wert
#!/Schreiben sie den Wert der gesetzt werden soll in die Anführungszeichen!
string Eingabewert = "" ;
```

Beispiel:

Es wird die Variable „ZZ_Grundtemp“ auf 17.0 Grad gesetzt

```
#!/Eingabe des Namen der Variable
#!/Schreiben sie den Namen der Variable zwischen die Anführungszeichen!
string Variablename = "ZZ_Grundtemp" ;

#!/Eingabe des Wert
#!/Schreiben sie den Wert der gesetzt werden soll in die Anführungszeichen!
string Eingabewert = "17.0" ;
```

Beachten sie die Ausgabe im Ausgabefeld:

Ausgabe:

Eingabe wurde gesetzt.

Ihre Eingabe wurde gesetzt. Überprüfen sie das ggf. in der Liste der Systemvariablen

Ausgabe:

Fehler in der Eingabe.
Ueberpruefen sie ihre Eingabe!

Überprüfen sie ihre Eingabe, einer der folgende Fehler liegt vor:

- Es fehlen noch Angaben.
- Es liegt ein Tippfehler im Variablennamen vor.
- Die Variable ist nicht vorhanden.

3.2 Homematic Klassik oder IP-Aktoren

Es können Homematic Klassik oder Homematic IP Aktoren verwendet werden. Unter Umständen ist es nötig mehrere Instanzen des Skripts „Schalten“ und der „Kalender“ in der CCU anzulegen.

Eine Vermischung der Aktoren ist nur bei Schalt Aktoren möglich. Befindet sich in einer Smarthome Installation eine Vermischung von Homematic Klassik und IP-Aktoren bei den Heizungsregler so muss für jede Gruppe eine Instanz der Skripte aufgesetzt werden.

Hier müssen dann einfach die Aktoren jeweils in einer Raumvariable zusammengefasst werden.

Es ist möglich das ein Raum beispielweise je zwei Aktoren Klassik und zwei Aktoren IP hat.

So muss der Raum in beiden Skripten geführt werden mit jeweils nur den Aktoren, die das Skript ansteuern soll.

Ein Skript das gleichzeitig beide Arten ansteuern kann würde die Anzahl von Variablen und zu konfigurierende Zeichenketten Potenzial in die Höhe treiben. Daher muss man sich bei einer solchen Installation mit einer zweiten Skript Instanz behelfen. Auch kann in dieser Doku nicht auf jede mögliche Art von Installation eingegangen werden, da es eine nahezu unbegrenzte Anzahl Möglichkeit gibt, wie man ein Smarthome aufbaut und mit welchen Aktoren.

3.3 Daten eines Temperaturfühler in eine Skriptvariable setzen:

In diesem Beispiel wird erklärt, wie man die Temperatur aus einem Temperatursensor in einem Systemvariable schreibt. An dieser Stelle kann dies auch nur ein Beispiel sein. Hier kann nicht auf alle möglichen Sensoren der Homematic eingegangen werden. Lesen sie hierzu ggf. die Datenpunkt Beschreibung des verwendeten Sensors. Diese steht auf der Homepage von EQ-3 zum Downloadbereich zu Verfügung.

In diesem Beispiel wird der Sensor HmIP-STHO verwendet.

Legen sie ein Programm an, das bei jeder Änderung des Datenpunkt des HmIP-STHO getriggert wird. Fügen sie dem Programm ein Skript hinzu welches die folgende Zeile enthält:


```
dom.GetObject("Z_Aussentemp").State(dom.GetObject("Aussentemperatur").DPByHssDP("TEMPERATURE").State());
```

Ersetzen sie den Namen der Variablen sowie den Kanalnamen durch die verwendeten Namen in ihrer Installation.

Name	Beschreibung	Bedingung (Wenn...)
AussentempTEST		

Bedingung: Wenn...

Geräteauswahl **Aussentemperatur** bei **Temperatur** im Wertebereich / mit Wert **größer oder gleich -40.00°C** bei Änderung auslösen

UND

ODER

Aktivität: Dann... ☒ Vor dem Ausführen alle laufenden Verzögerungen für diese Aktivitäten beenden (z.B. Retriggern).

Skript **dom.GetObject("Z_Aussentemp").State(dom.GetObject("Aussentem...)** sofort

Aktivität: Sonst... ☐ Vor dem Ausführen alle laufenden Verzögerungen für diese Aktivitäten beenden (z.B. Retriggern).

Existiert in der Installation schon eine Systemvariablen, die die aktuelle Außentemperatur enthält so geben sie einfach den Namen der Variablen im Skript Schalten in der Variablen für die Außentemperatur an.

3.4 Wetterdienst:

Um die aktuelle Außentemperatur von einem Wetterdienst abzufragen, legen sie ein Programm an das Stündlichen das Skript aus dem Datei Wetterdienst.c ausführt.

Auf Grund der Beschränkung von 50 Abfragen pro Tag bei der kostenlosen Version sollte das Abfragen nicht häufiger stattfinden.

Wie sie ein API Key vom Wetterdienst Accuweather anlegen und wie sie ihre Location herausfinden ist im Video erklärt:

<https://www.youtube.com/watch?v=fzYoYX9T1wk>

Name	Beschreibung	Bedingung (Wenn...)
AccuWetterTemp		
Bedingung: Wenn... <div> <div>Zeitsteuerung</div> <div>Periodisch Ganztägig beginnend am 26.08.2023</div> <div>zu Zeitpunkten auslösen</div> </div> <div> <div>UND</div> </div> <div> <div>ODER</div> </div>		
Aktivität: Dann... <input checked="" type="checkbox"/> Vor dem Ausführen alle laufenden Verzögerungen für diese Aktivitäten beenden (z.B. Retriggern). <div> <div>Skript</div> <div>!Skript zur ermittlung Temperatur für ein bestimmten Ort vom...</div> <div>sofort</div> </div>		
Aktivität: Sonst... <input type="checkbox"/> Vor dem Ausführen alle laufenden Verzögerungen für diese Aktivitäten beenden (z.B. Retriggern). <div> <div></div> </div>		

Geben sie im Kopf des Skriptes den API Key, die Location sowie den von ihnen vergebenen Namen für die Variabel der Außentemperatur an.

```

//Event orientiertes Heizen/Schalten per Google Kalender Version 1.12 / 27.08.2023 Lukas Helduser (Youtube: https://www.youtube.com/LukasvandeHaag)
//Teil 4 Skript zum auslesen der Temperature vom Wetterdienst Accuweather
//Benutzer Eingaben:
!//#####--Beginn Variabler Bereich--#####
string key="xxxxxxxxxxxxxxxx"; !Eingabe des API key
string ort="xxxxxx"; !Eingabe der Ortskennung
string varTemp = "Z_Aussentemp";
!//#####--Ende Variabler Bereich--#####

```

3.3 Videolinks:

Bleibt aktuell frei