Inhaltsverzeichnis

Verbrauchsauswertung aus Meßdaten der Haustechnik

Erfassen der ersten Daten der Verbräuche

Von einem Messgerät im Netz, in meinem Fall KNX und Loxone Mini Server beziehen wir einen Gesamtverbrauchswert (den Zählerstand), aus dem erzeugen wir einen Tagesverbrauch, um Mitternacht Zählerstand Vortag minus Zähler aktuell, ergibt den Tagesverbrauch.
Ein Speicher history oder (SQL funktioniert bei mir leider nicht) muß installiert und natürlich die Meßdaten verfügbar sein.

1. Ein neues Skript in Blockly erzeugen. Diese Ebene hilft mir als Anfänger sehr das Ganze zu verstehen.
2. Benennen des Skriptes und speichern

Erzeugen des/der Objekte.

3. Dies finden wir nachher im IoBroker unter dem Reiter Objekte, das enthält dann den erzeugten Wert aus unseren Berechnungen. Den Block finden wir im System auf der linken Seite des Skriptes. Dem geben wir den Namen den nachher auch unser Wert hat.

4. 4 Objekte erzeuge ich, Verbrauch und Kosten gestern, Verbrauch und Kosten heute aktuell. Speichern und das Skript starten, dann finden wir die erzeugten Objekte unter dem Reiter Objekte unter Javaskript. Der Wert ist null : ist ja noch nichts drin.

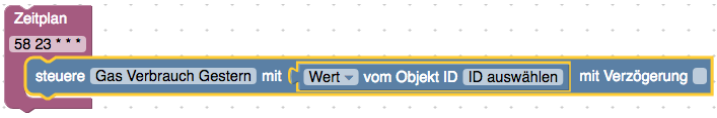
Zeitplan aus Trigger

Da die Erstellung des Wertes ja nur um Mitternacht erzeugt werden soll, setzen wir erst einmal einen Zeitplan aus der Rubrik „Trigger“. Den Block „Zustand erzeugen“ kann man deaktivieren, oder löschen, der hat seinen „Job“ getan. Der Zeitplan wird jetzt eingerichtet, klicken auf die Fenster mit den Sternen, dann kann die Zeit eingegeben werden, bei mir 23.58. Zwei Minuten vor Mittelnacht wird also diese Script ausgeführt.

Block „steuere“ aus der Rubrik System

Nun kommt ein Block „steuere“ aus der Rubrik System. Der wird in den Zeitplan eingesetzt. Wenn man auf Objekte ID klickt, springt die Auswahl auf das Menü Objekte und man kann den zuvor erzeugten Datenpunkt auswählen den man berechnen will. Zum Beispiel „Gas Verbrauch gestern“.

5. In das weiße Puzzle Feld kommt jetzt der „Wert“ Block . Hier soll um Mitternacht der aktuelle Stand gespeichert werden. Also in der Wert ID den Zählerwert der Meßstelle auswählen (wieder unter Objekte). Die erste Steueranweisung ist fertig, um Mitternacht speichern wir den Wert in den Datenpunkt „Gas Verbrauch Gestern“

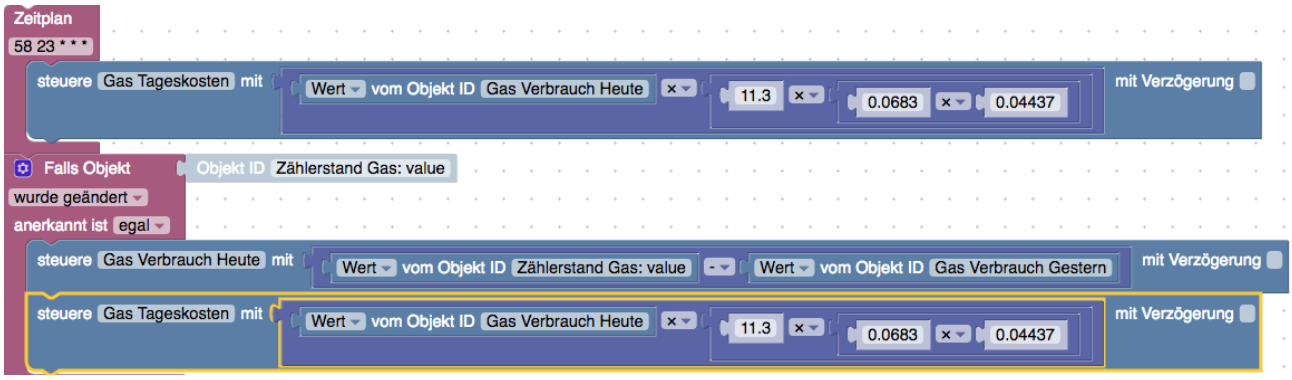


Falls Objekt geändert

6. Um jetzt den heutigen Verbrauch berechnen zu können, muß die Bedingung „Falls Objekt geändert“ darunter setzen (NICHT darein). Es muß natürlich festgelegt werden welcher Wert zugrundegelegt wird für die Prüfung der Änderung, der in diesem Fall der aktuelle Messwert (in > Objekt ID).Wenn also am darauf folgenden Tag sich der „Gas Verbrauch Heute“ zum gestrigen geändert hat, soll er von dem gestrigen abgezogen werden.

ein zweiter Steuerbefehl

7. Jetzt kommt ein zweiter Steuerbefehl, es soll der aktuelle Stand (um 23:58) vom dem gespeicherten Wert gestern (7.) abgezogen werden. Also ein zweiter Block „steuere“ unter den



ersten. Der Wert muß berechnet werden, daher benötigt man einen Rechenblock, aus dem Menüblock Mathematik.

8. Es soll abgezogen werden, den Block mit dem „+“ kann man umstellen auf minus. Er wird wieder in das Puzzle eingesetzt. In dessen linkes Puzzle kommt der aktuelle Zählerstand,

davon wird im rechten Teil der Steuerung der gestrige Wert abgezogen.

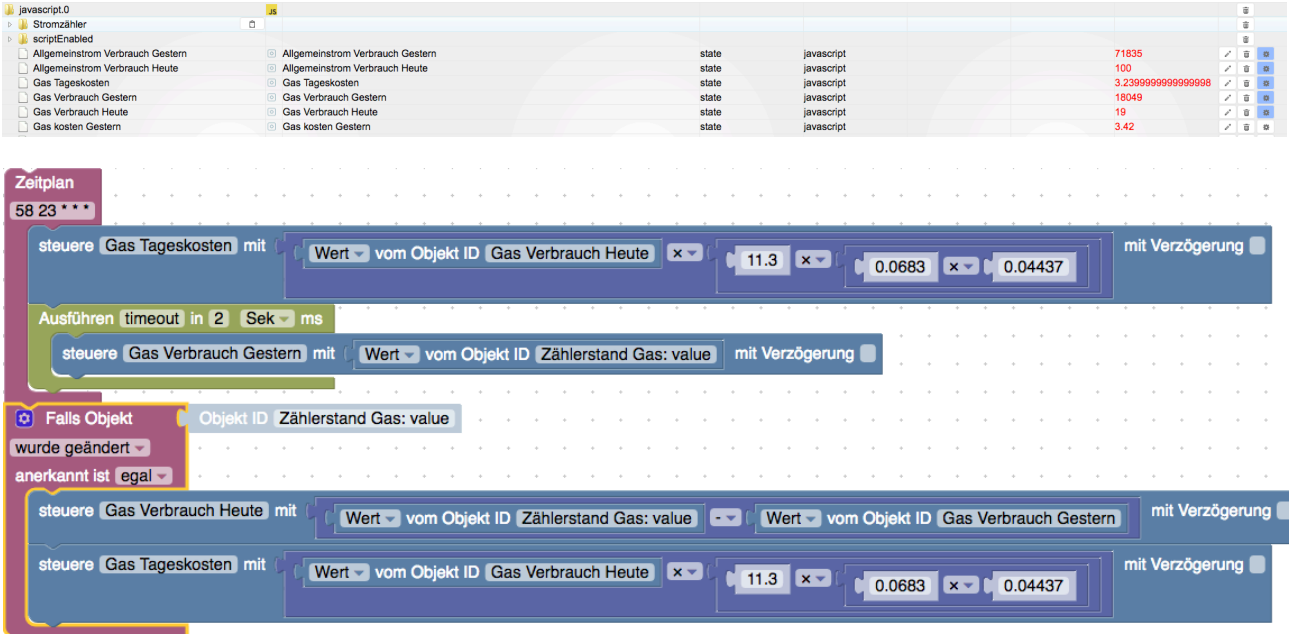
9. Wenn das soweit richtig gesetzt ist haben wir Morgen den Stand des Gaszählers um 23:58 gesichert und Übermorgen den Verbrauch des Vortages.

wieviel ich heute schon für Gas ausgegeben

10. Jetzt möchte ich die Kosten, wieviel ich heute schon für Gas ausgegeben habe ermitteln. Dazu kommt wieder ein „steuere“ Block in den geändert Bereich, diesmal als Multiplikation des heutigen Verbrauches . Bei Gas muß m3 in kW umgerechnet werden, da der Preis pro kW angegeben ist, also m3 x Brennwert x Zustandszahl x € pro kWh.

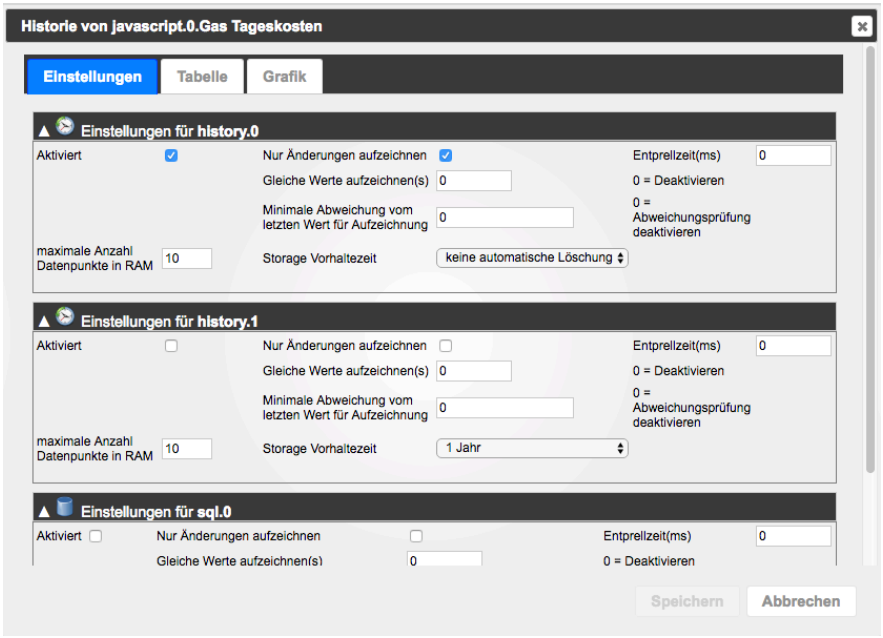
Kosten fürs Gas gestern

11. Nun fehlen noch die Kosten fürs Gas gestern. Diese sind fix, daher müssen für über den „wurde geändert“ Block einfügen. Da die Skripte sehr schnell sind soll eine Verzögerung für die Speicherung des Tageswertes hinein, damit die Berechnung vor der Ermittlung des Zählerstandes erfolgt. Also wieder ein „steuere“ mit der Berechnung in der Verzögerung einsetzen.



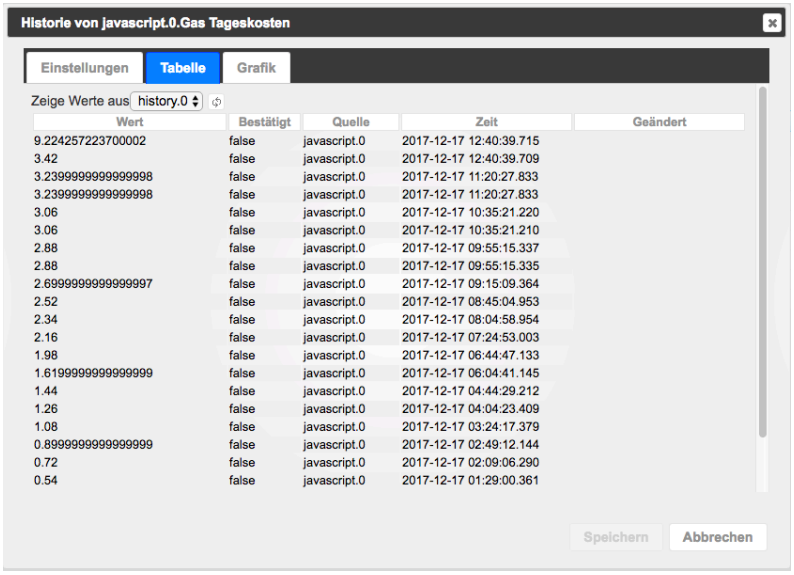
history oder SQL

12. Damit werden alle vier Werte entsprechend gefüllt. Am überwachsten Tag sollten die entsprechenden Zahlen zu finden sein. Ganz rechts am Zahnrad muß noch eingestellt werden, wo die Zahlen zu speichern sind, also history oder SQL.



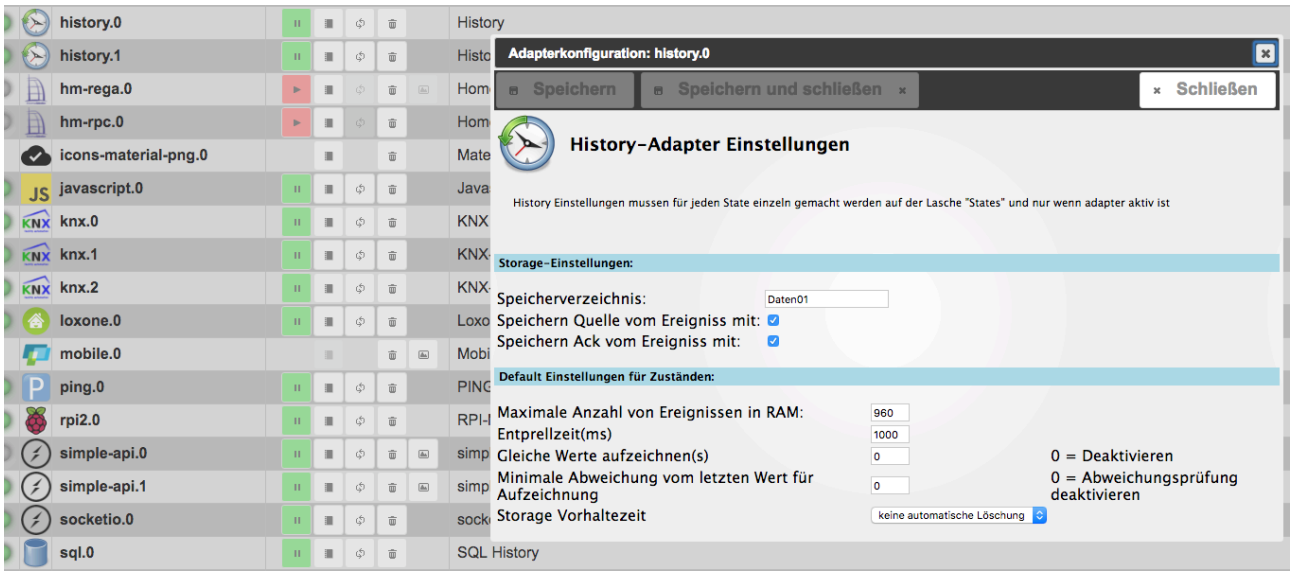
Hier das Einstellungsfenster:

In Tabelle sind dann die gespeicherten Werte zu sehen:

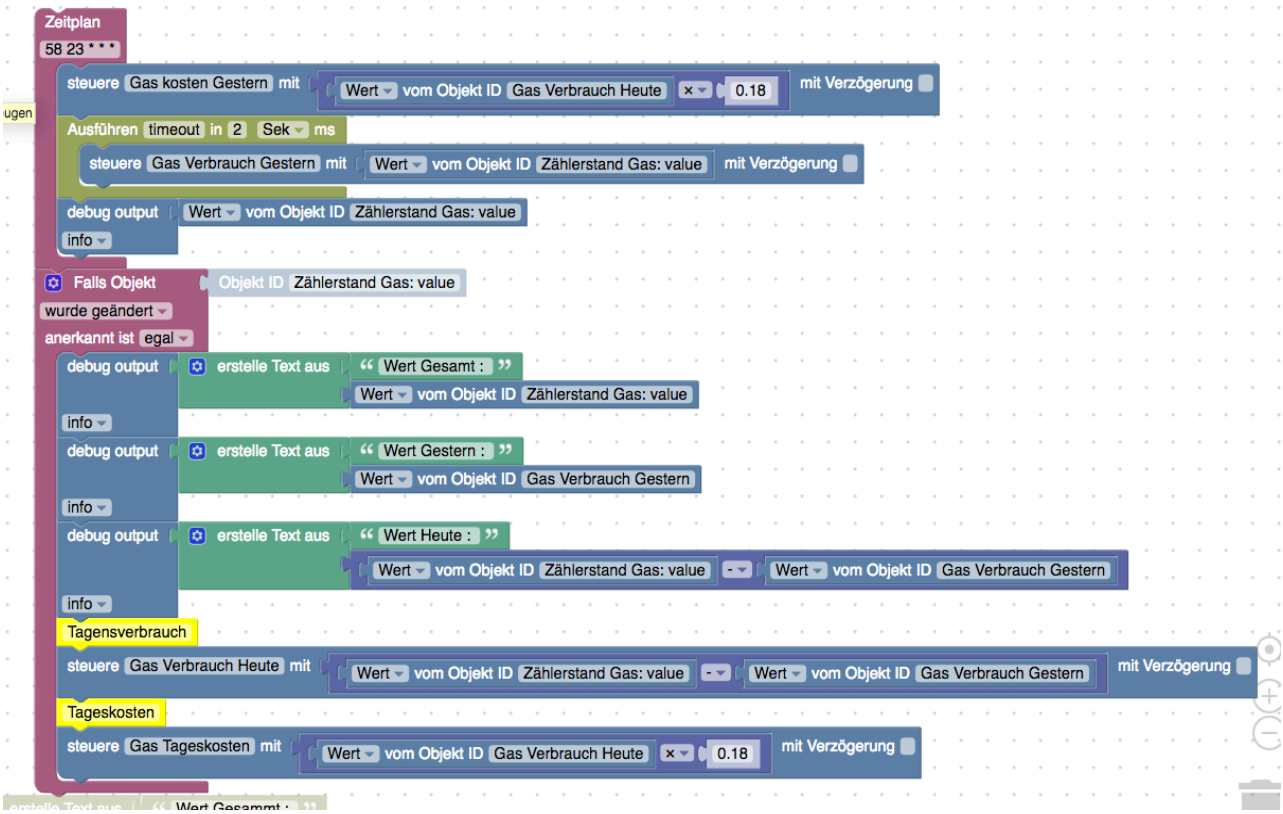


History noch einstellen

In den Instanzen muß History noch Eingestellt werden, also der Speicherort:



Dank der großartigen, selbstlosen Hilfe von Dutchman habe ich jetzt nach nochmaligem durcharbeiten langsam verstanden, es ist auch für „Dummies“ verstehbar. Wir hatten noch einige „Debug“ Felder eingesetzt, dann erscheinen die die Werte in der Logzeile unten zur Überprüfung, das ist sehr hilfreich. Dann sah das gesamte Script zum Schluß so aus:



Eine Pushover Warnmeldung anlegen

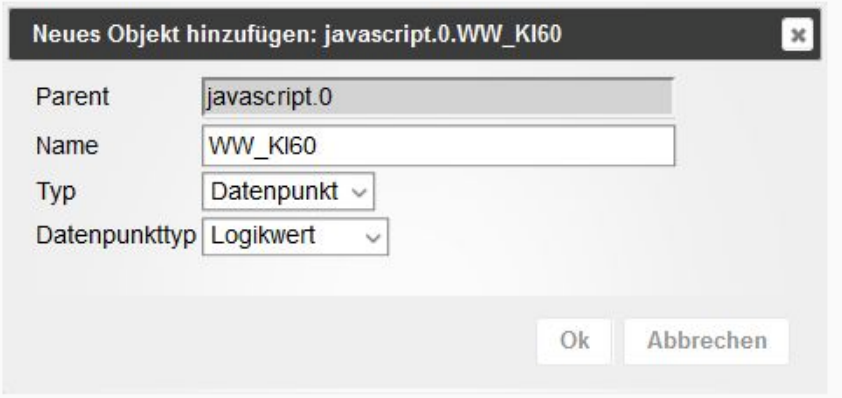
Eine hervorragende YouTube Anleitung gibt es zu diesem Thema

https://www.youtube.com/watch?v=ZeTdEH1SWBg

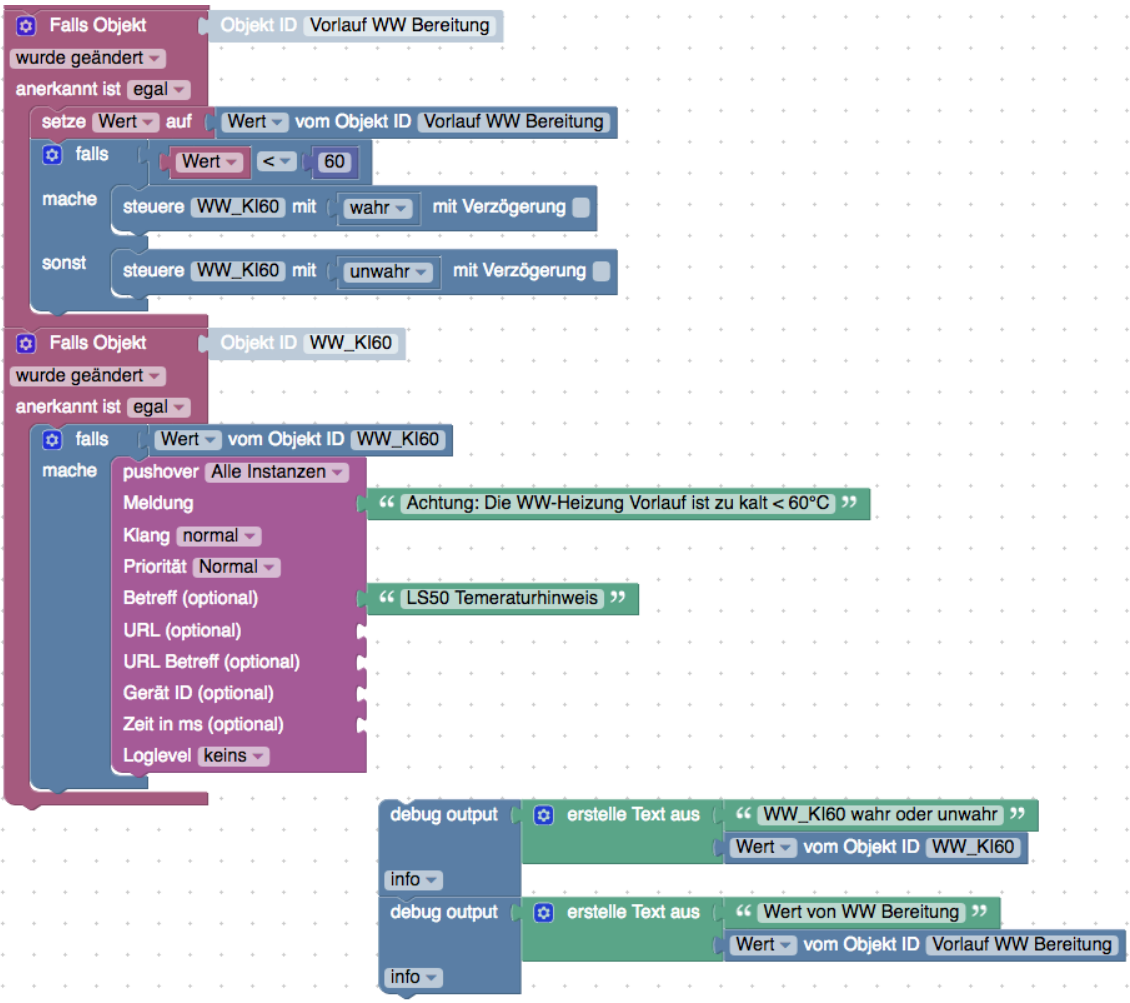
Die vorgestellte Möglichkeit nutze ich um Temperaturunterschreitung im Heizsystem per Push an mein IPhone zu senden, damit werden Störungen rechtzeitig gemeldet, der Service kann erfolgen bevor das ganze Haus kalt ist. Oder, wir haben außen eine elektrische Rampenheizung, um Glatteis zu vermeiden, die war sehr schlecht eingestellt, durch eine Meldung über viel zu hohen Stromverbrauch wurde nachjustiert, andernfalls hätten wir eine Nachzahlung für Strom über viele tausend Euro gehabt. Mit der Hilfe von Dutchman habe ich jetzt noch eine Einschränkung, die Meldung erfogt nur einmal, wenn der Wert unter dem Eingestellten ist, vorher kam alle 2 Sekunden eine Pushnachricht, bis der Wert wieder erreicht war.

Die Blockly Erzeugung des Skriptes sieht jetzt so aus:

Erzeugen eines Objektes WW_Kl60 unter Objekte bei javascript.0 anlegen, unter dem Reiter Objekte, Plus Symbol und nach Bild einrichten.



Dann in Blockly, wie bekannt anlagen, jetzt aber mit der eben angelegten Variablen die auf wahr oder falsch gesetzt wird.


Falls Objekt
Objekt ID Vorlauf WW Bereitung
wurde geändert
anerkannt ist egal
setze Wert auf Wert vom Objekt ID Vorlauf WW Bereitung
falls
Wert < 60
mache
steuere WW_KI60 mit wahr mit Verzögerung
sonst
steuere WW_KI60 mit unwahr mit Verzögerung
Falls Objekt
Objekt ID WW_KI60
wurde geändert
anerkannt ist egal
falls
Wert vom Objekt ID WW_KI60
mache
pushover Alle Instanzen
Meldung
Achtung: Die WW-Heizung Vorlauf ist zu kalt < 60°C
Klang normal
Priorität Normal
Betreff (optional)
LS50 Temeraturhinweis
URL (optional)
URL Betreff (optional)
Gerät ID (optional)
Zeit in ms (optional)
Loglevel keins
debug output
erstelle Text aus
WW_KI60 wahr oder unwahr
Wert vom Objekt ID WW_KI60
info
debug output
erstelle Text aus
Wert von WW Bereitung
Wert vom Objekt ID Vorlauf WW Bereitung
info