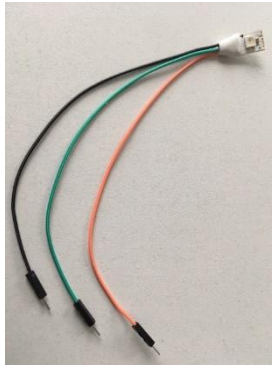


Sensoren/Aktoren?



Sensoren/Aktoren?

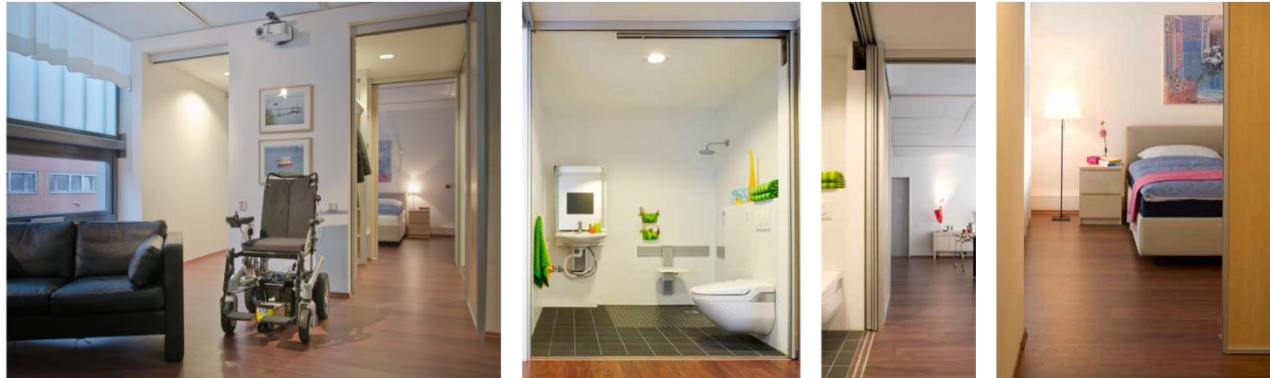
Aktoren:

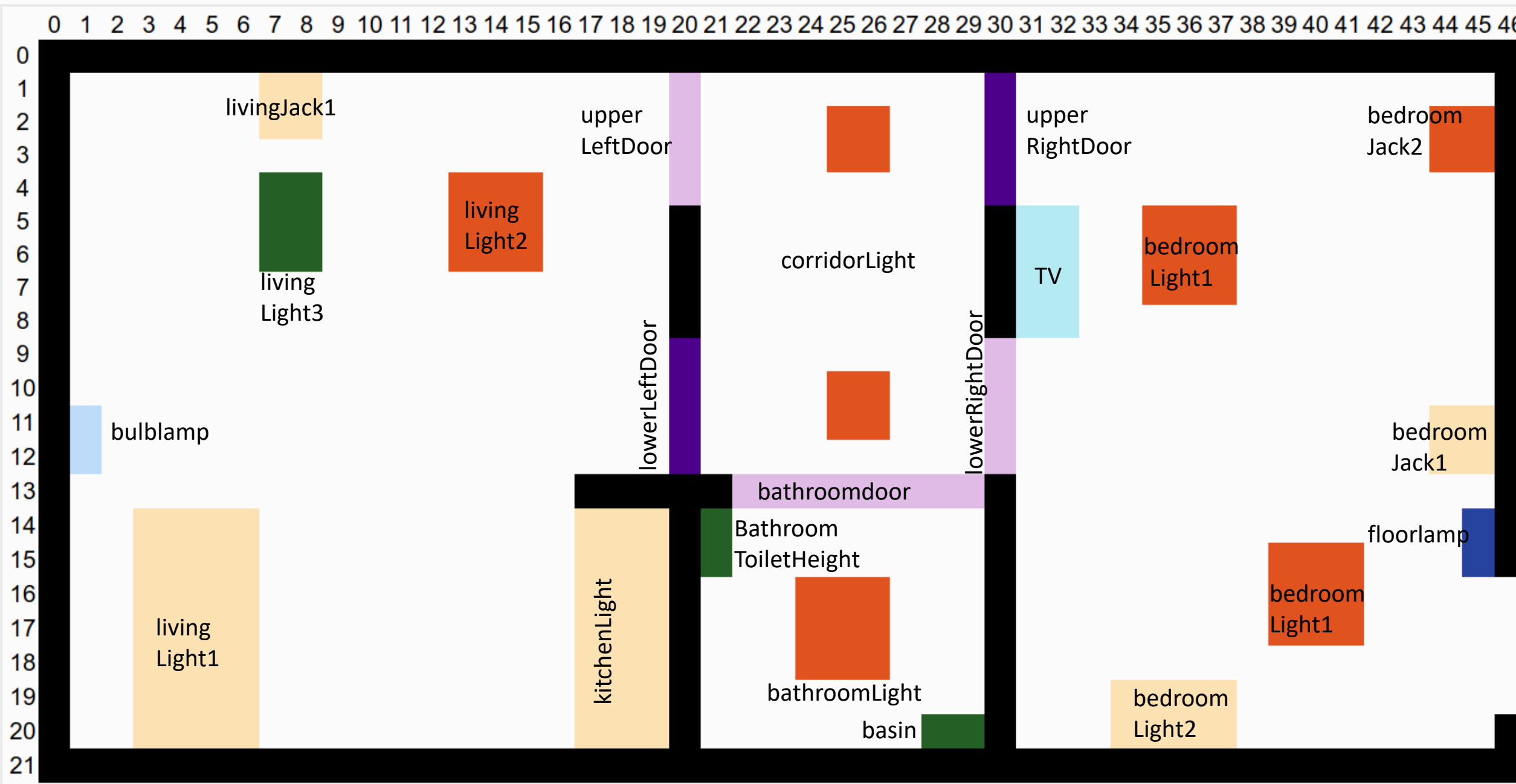


Sensoren: ?



Das BAALL

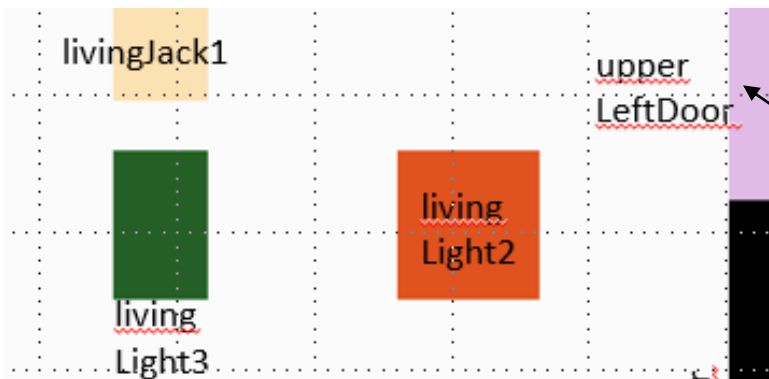




Das BAALL programmieren

Auf dem Handout findet ihr eine Skizze vom BAALL und eine Liste aller Objekte (physische Objekte und weitere Elemente) im BAALL. In der Liste ist jeweils der Status der Objekte angegeben, z.B. ob eine Lampe an- oder ausgeschaltet ist. Den Status können wir mittels Programmierung auslesen und ändern.

Auszug aus der Skizze:



Objekt, z.B.
 basin
 oder
 upperLeftDoor

Auszug aus der Liste:

Status, z.B. 108
 basin: 108
 bathroomLight: off
 bathroomToiletHeight: 111
 bathroomdoor: on
 beaconActive: off
 bed1: 108
 bed4: 96
 bedroomJack1: off
 bedroomJack2: off
 bedroomLight1: off
 bedroomLight2: off
 bedroomTVActive: off
 bulblamp: 0:0:0
 corridorLight: off

Das BAALL programmieren

Die Objekte teilen wir in verschiedene Kategorien ein, die wir unter „Switch“, „Dimmer“, „RGB Farbe“, „Sensoren“ und „TV Programm“ zusammengefasst haben. Diese unterscheiden sich in den Werten, die der Status annehmen kann. „Switch“ bedeutet zum Beispiel, dass das Objekt nur zwei Zustände haben kann: An/Aus, Auf/Zu, off/on oder 1/0.

Den entsprechenden Block findet ihr in BEESM unter BAALL und dann unter „Verbindung“.



.....

Aufgaben

Aufgabe: Schaut euch die BAALL Skizze und die Liste an – achtet dabei besonders auf die Werte, die der Status der Objekte annehmen kann. Welche Objekte aus der Liste würdet Ihr den Kategorien „Switch“, „Dimmer“, „RGB Farbe“, „Sensoren“ und „TV Programm“ zuordnen? Warum? Notiert ein paar Beispiele.

„Switch“: z.B. bathroomdoor, _____

„Dimmer“: z.B. basin, _____

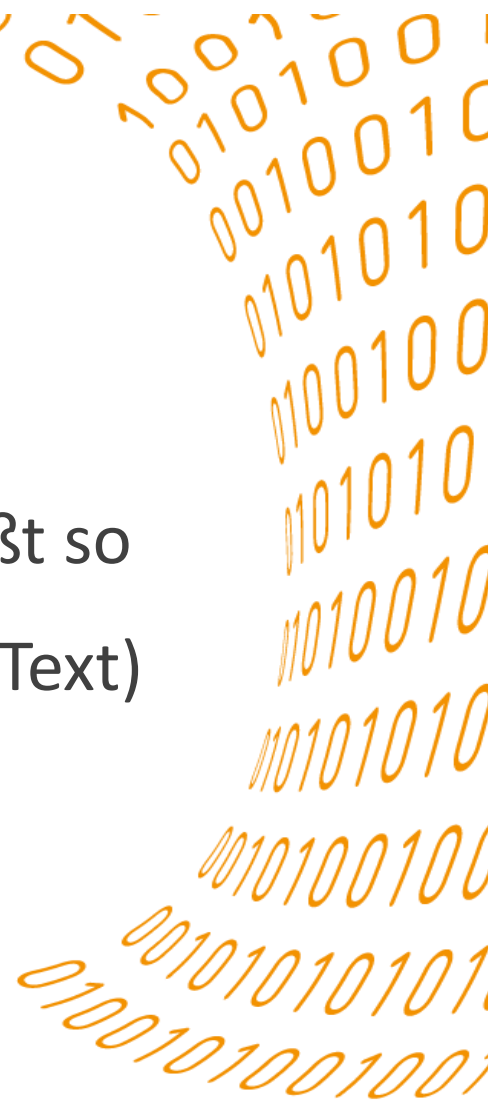
„RGB Farbe“: z.B. bulblamp, _____

„Sensoren“: z.B. LuxOutside, _____

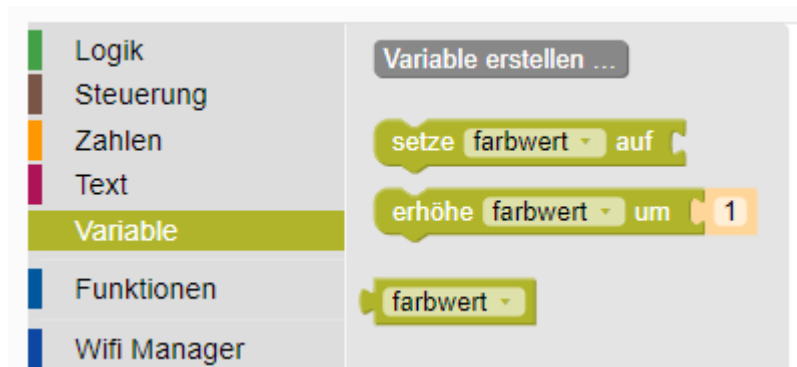
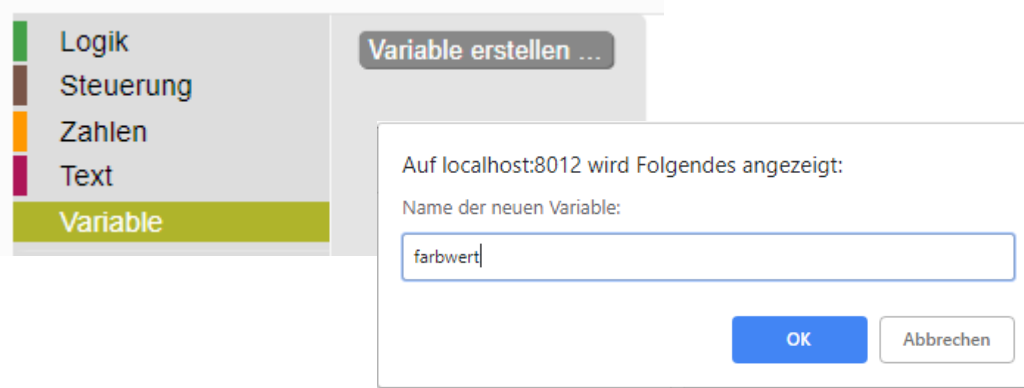
„TV Programm“: _____

Einführung: Variablen

- Eine Variable ist ein Platz, um Werte zu speichern
- Ähnlich wie ein Karton, in den ich etwas hinein tun kann
- Dieser Karton hat einen eindeutigen Namen, also nur er heißt so
- Bei BEESM können Variablen **NUR** Zahlen enthalten (keinen Text)

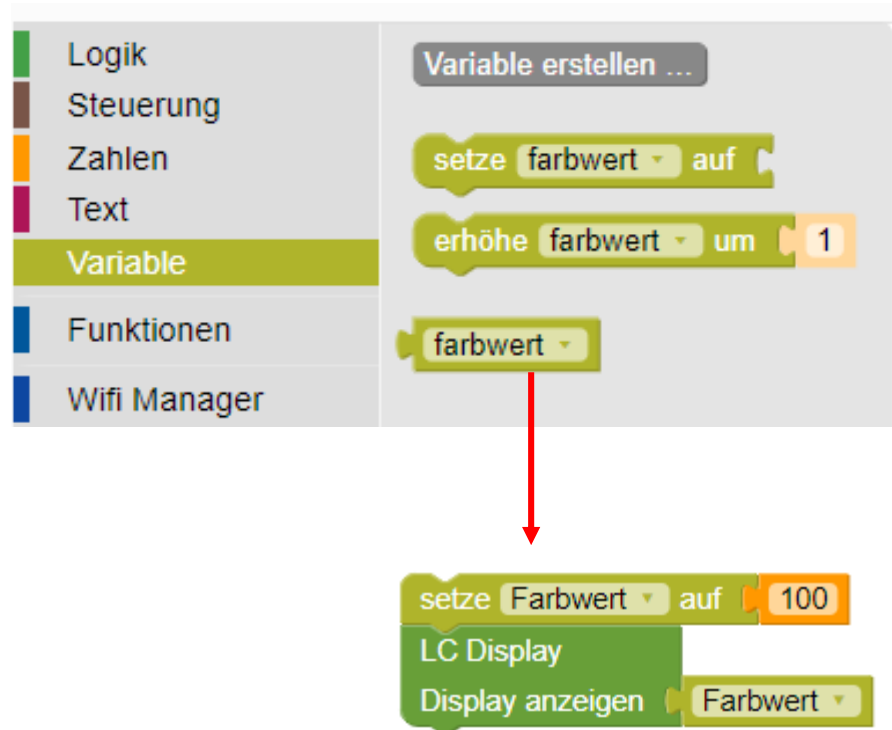


Variablen anlegen

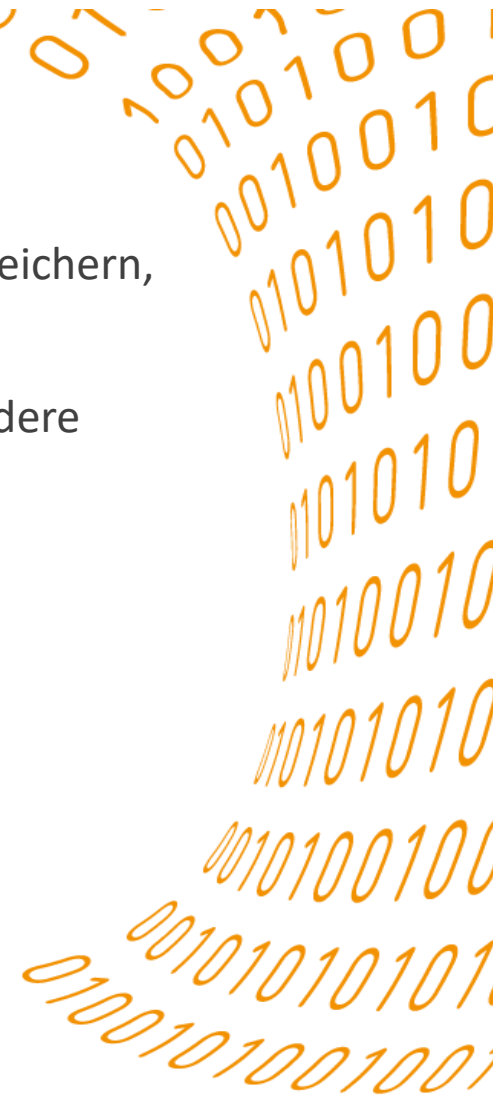


- Erstelle eine Variable im Punkt „Variable“
- Dabei muss der Variable ein Name gegeben werden.
- Danach erscheinen neue Befehle unter „Variable“ für diese Variable

Variablen anlegen

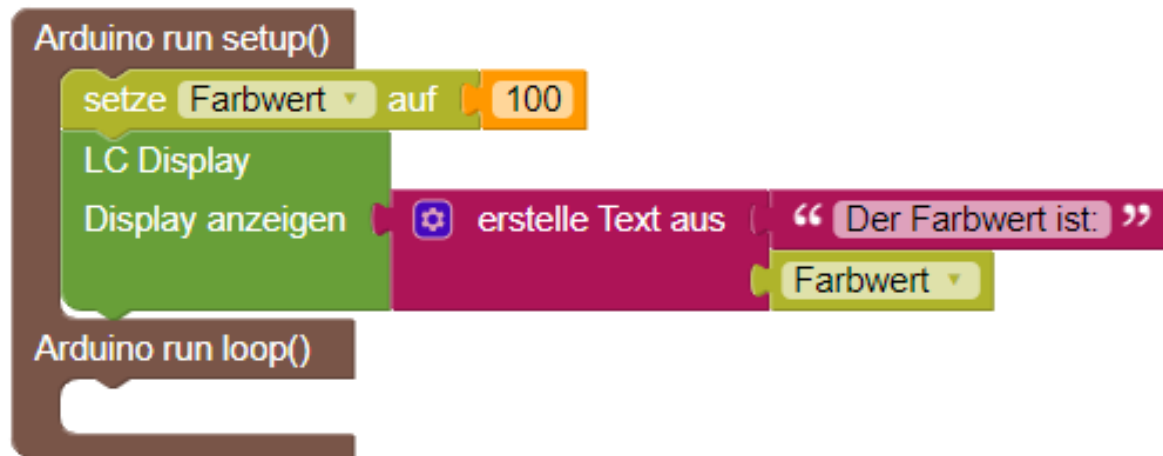


- Um einen Wert in einer Variablen zu speichern, nehmen wir den Block „setze auf x“.
- Zum Lesen zieht die Variable in eine andere Operation rein.



Übungen: Variablen

Erstellt eine Variable, speichert einen Wert (Zahl) in der Variablen und lässt den Wert der Variablen auf dem Display ausgeben!



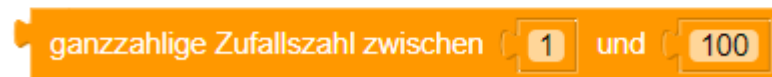
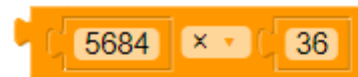
Was gibt der Code auf dem Display aus?

Übung 1: Variablen

Nun lassen wir den Computer rechnen.

Anstelle einer einzigen Zahl wie im vorherigen setzen wir die Variable auf komplexere Terme („mathematische Gebilde“). Entsprechende Blöcke findet ihr unter Zahlen.

Zum Beispiel folgende:



Übung 2: Variablen

Ändert euer Programm:

- Warte nach der Ausgabe der Variable für eine Sekunde
- Erhöhe dann den Wert der Variable um eins
- Gebe die geänderte Variable aus
- Wiederhole dieses drei mal



Übung 3: Variablen

Erweitert euer Programm und programmiert einen Zähler:

- Wartet nach der Ausgabe der Variable für eine Sekunde (wie gehabt)
- Erhöht dann den Wert der Variable um eins
- Gebt die geänderte Variable aus
- Diese Schritte sollen unendlich oft wiederholt werden



Benutzt nebenstehende Blöcke
jeweils nur einmal. Wie geht ihr vor?

Übung 3: Variablen

Erweitert euer Programm und programmiert einen Zähler:

- Wartet nach der Ausgabe der Variable für eine Sekunde (wie gehabt)
- Erhöht dann den Wert der Variable um eins
- Gebt die geänderte Variable aus
- Diese Schritte sollen unendlich oft wiederholt werden



Benutzt nebenstehende Blöcke

jeweils nur einmal. Wie geht ihr vor?



Tipp!
Programmiert hierzu in der **loop()**

Status holen

Im Setup:

- Wir verbinden uns mit dem BAALL Server 

In der loop:

- Wir legen eine Variable an. In dieser speichern wir den Statuswert, den wir holen. Zum Beispiel setzen wir die Variable auf: 
- Wir geben den Wert der Variable auf dem Display/über die serielle Schnittstelle aus.
- **!!! Setzt in der loop ein Verzögerung von mehreren Sekunden, z.B.**  **!**
(Wir müssen schließlich den Status nicht in jeder Millisekunde abrufen.)

Aufgaben

- Hole den aktuellen Status und gebe diesen auf dem Display aus:
 - vom Waschbecken (basin)! _____
 - von der dimmbaren Lampe (livingLight3)! _____
 - vom Schlafzimmerlicht (bedroomLight1 oder 2)! _____
 - Von der Temperatur draußen (tempOutside)! _____
 - Von der Emotion, die der Badezimmerspiegel gerade „erkennt“! _____
(→ Benutze hierzu den Block „Erkenne Gesichtszüge“)
 - Und was euch noch interessiert...

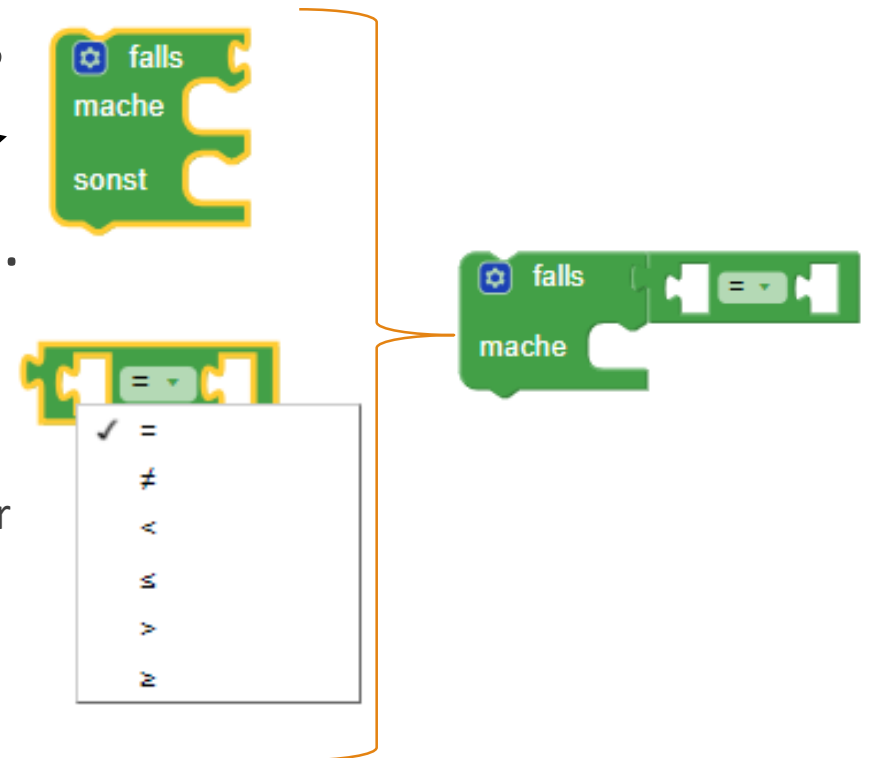
Bedingungen

Bedingungen dienen zum Steuern des Programmflusses. Sie sind so etwas wie „Falls dies zutrifft, mache dies, sonst das“.

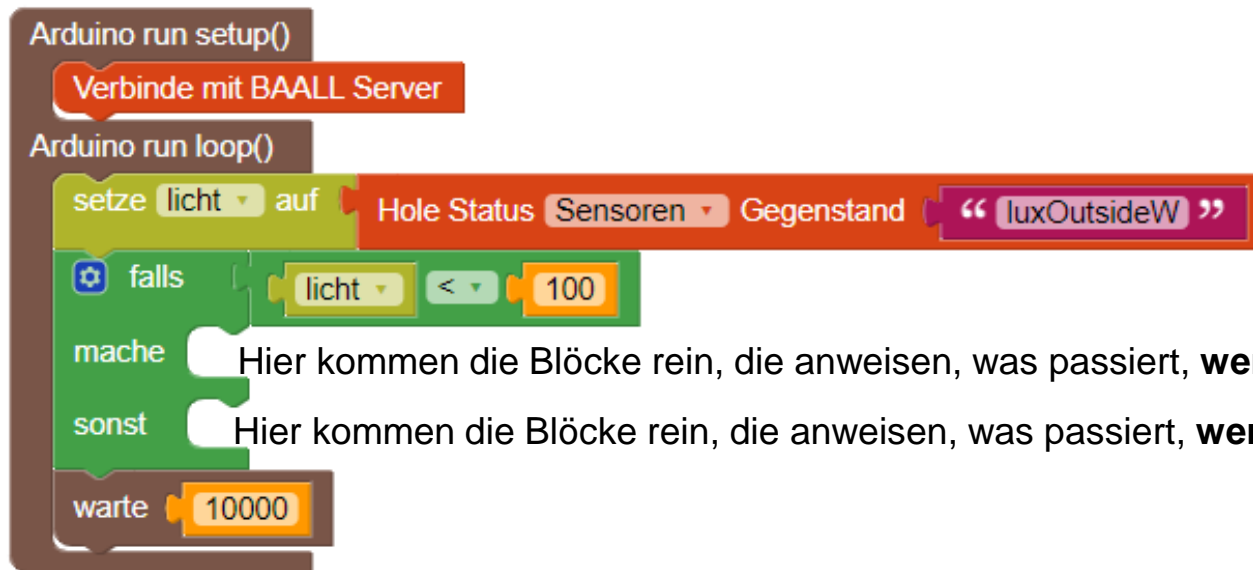
Dazu benötigen wir einen **Bedingungsblock** ...

...und einen **Vergleichsblock**.

- Es gibt einfache Vergleiche, wie „kleiner“, „größer“ und „gleich“
- Damit überprüfen wir, ob der Vergleich zweier Werte wahr ist oder falsch. Ist er wahr, werden die Anweisungen hinter „mache“ ausgeführt.
- Beide Blöcke findet ihr unter „Logik“



Bedingungen



Ein kleines Projekt

Aufgabe: Intelligente Lichtsteuerung! Wenn es draußen dunkel wird, dann soll das Licht automatisch angehen

- Wenn die Helligkeit draußen (luxOutside) weniger als 100 Lux beträgt, soll euer LED Lämpchen angehen. Ansonsten soll das Lämpchen ausgeschaltet bleiben.
- Es reicht, wenn wir den Status alle 10 Sekunden überprüfen. **Also den Delay in der loop nicht vergessen!!**



Challenge

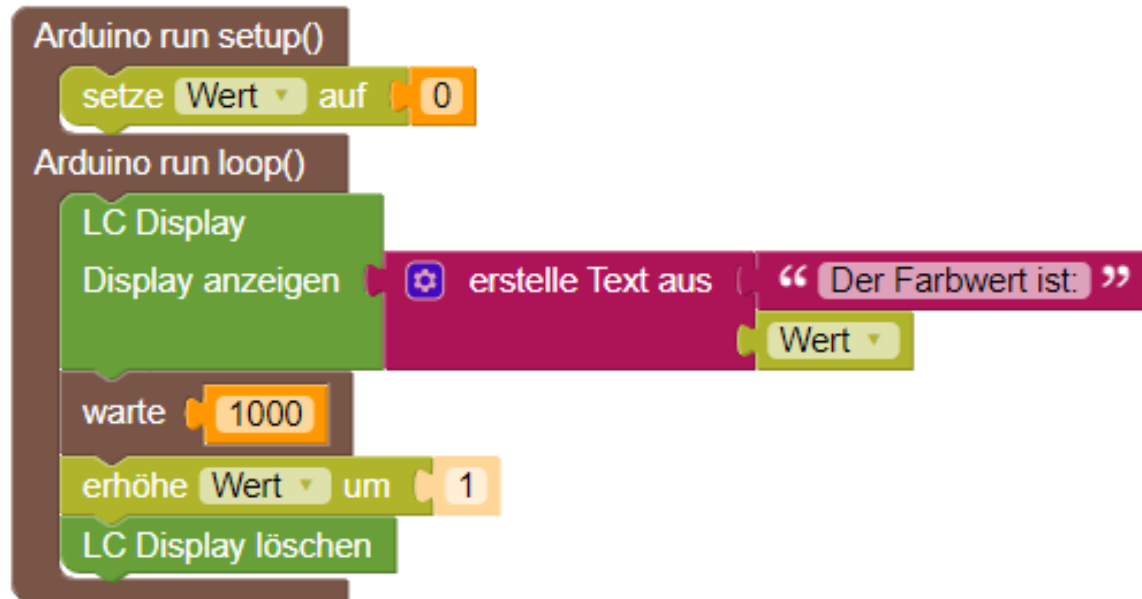
Lösungsvorschläge

Übung 2: Variablen



Lösungsvorschlag

Übung 3: Variablen (Zähler)



Lösungsvorschlag

Kleines Projekt

