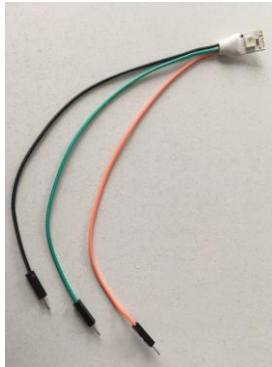


# Sensoren/Aktoren?



# Sensoren/Aktoren?

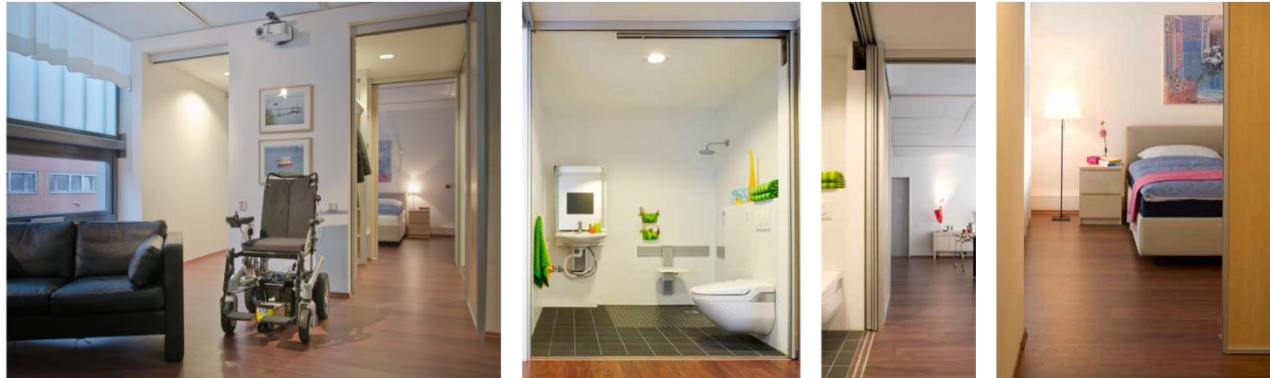
Aktoren:



Sensoren: ?

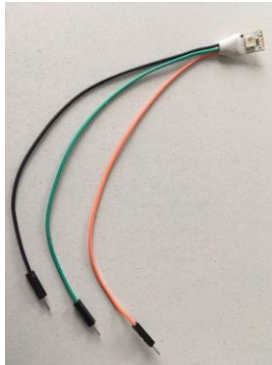


# Das BAALL



# Sensoren/Aktoren?

Aktoren:

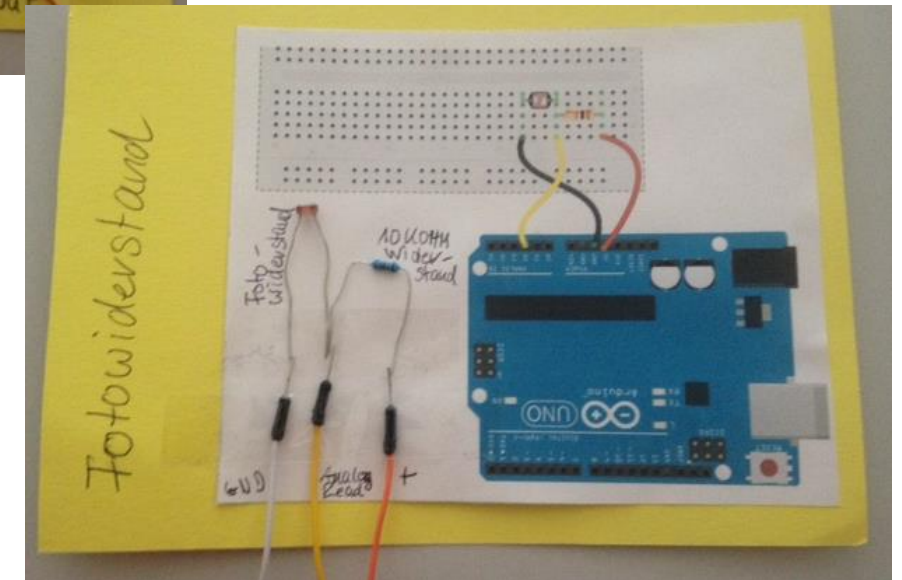
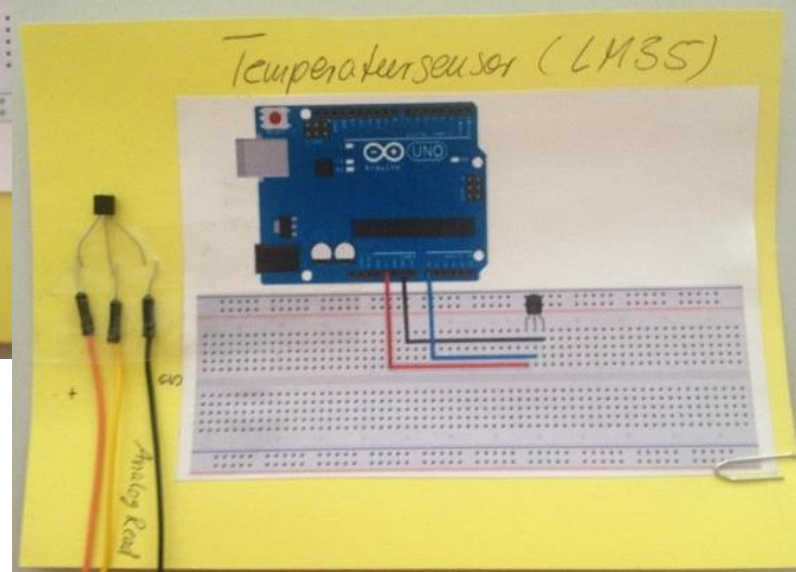
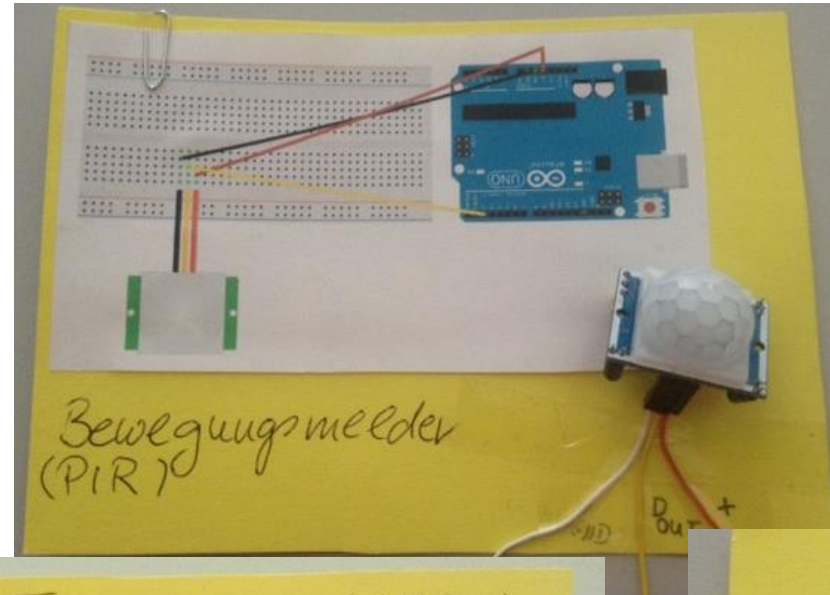
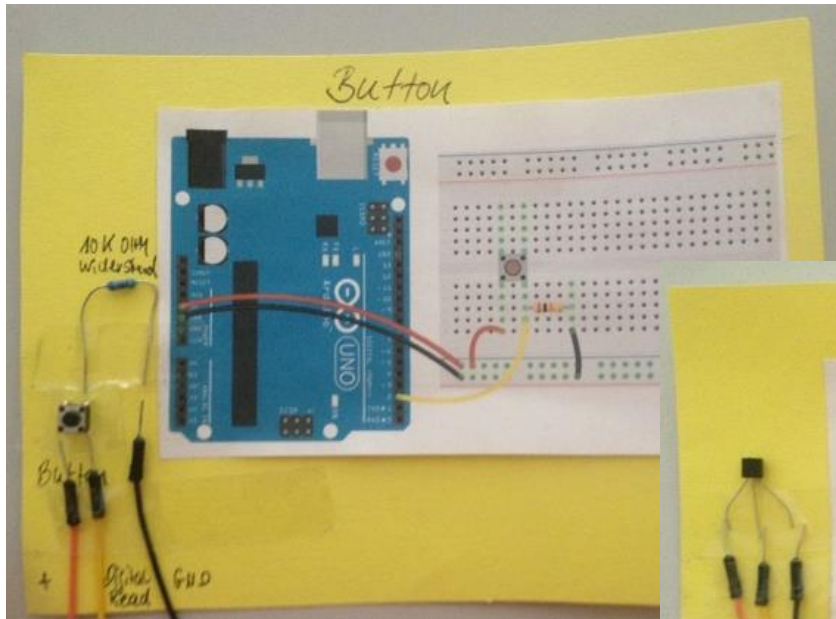


Sensoren:

- PIR Bewegungssensor
- Temperatursensor
- Fotowiderstand (Lichtsensor)
- Button



# Sensoren



# Sensor-Werte auslesen

Anweisungen im Code legen fest, ob ein Pin Ein- oder Ausgang ist.

An einem Eingangspin wird die angelegte Spannung (Volt) eingelesen.

Damit können Messdaten von Sensoren erfasst und verarbeitet werden.

lese digital pin# 1 ▾

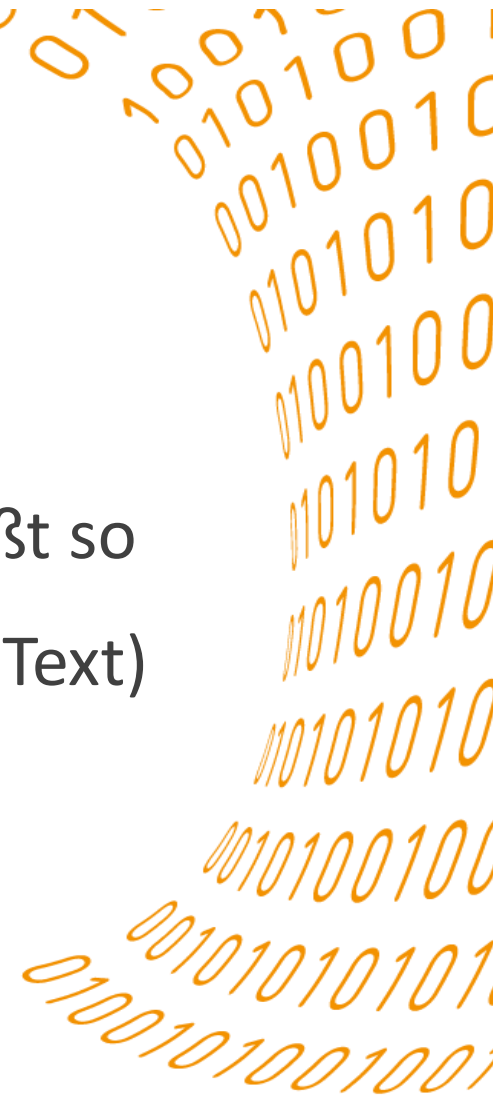
Liest einen Wert von einem Digitalpin ein. Es gibt nur zwei Zustände: Null, (0/LOW) oder Eins (1/HIGH); dabei entspricht der Zustand 0 einer Spannung von 0 Volt und der Zustand 1 einer Spannung von 5 Volt (oder 3,3 Volt).

lese analog pin# A0 ▾

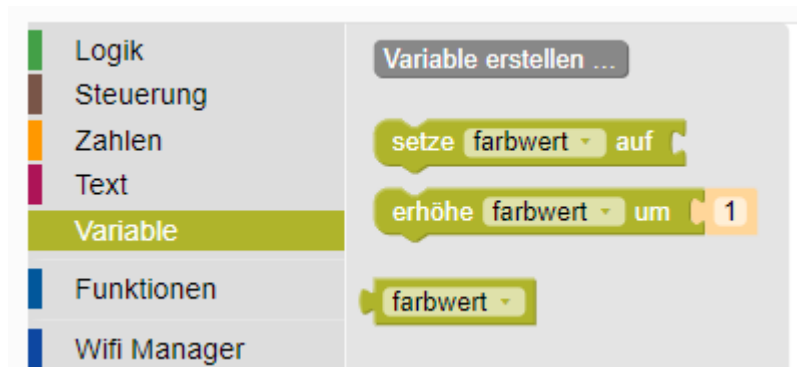
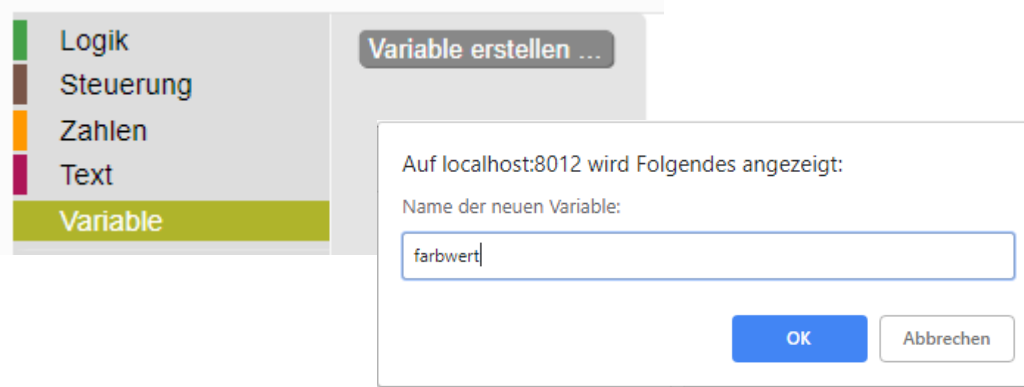
Sensoren liefern die Messwerte häufig als analoge Größen. An den Analogpins befindet sich ein Analog-Digital-Wandler (ADC), der den analogen Messwert linear in eine digitale Größe überführt. Bei dem Arduino Uno sind das Werte von 0 bis 1023 (10 Bit). 5 V am Eingang wandelt der ADC in 1023 um, 0 V entsprechend in 0.

# 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)



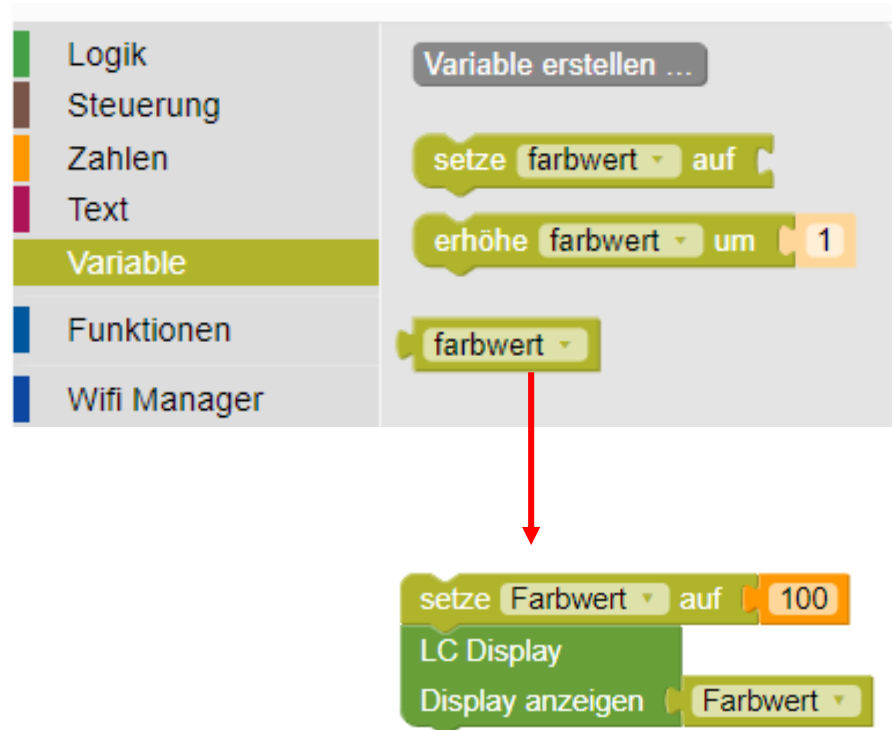
# Variablen anlegen



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



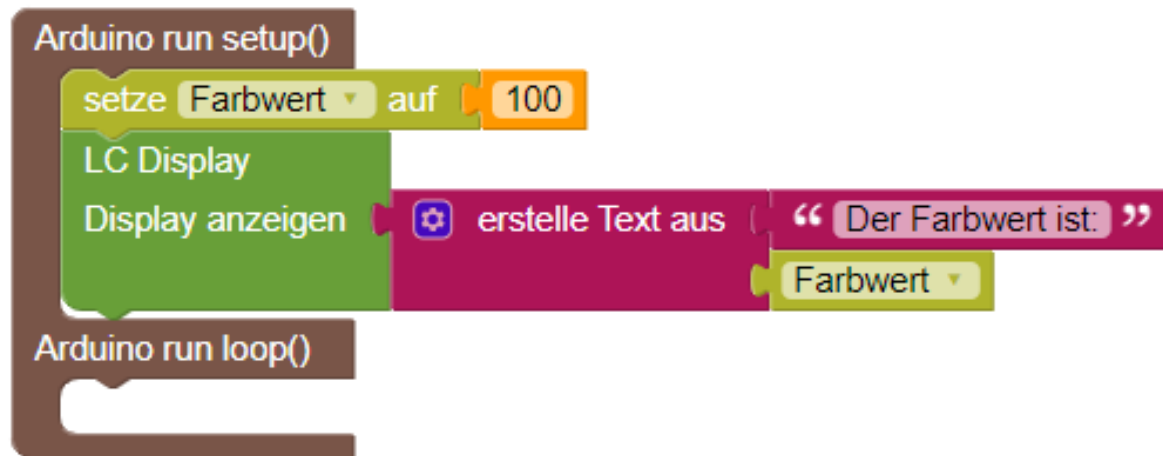
# Variablen anlegen



- 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!



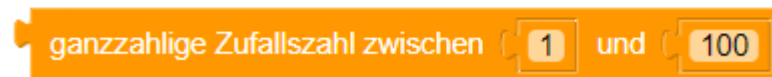
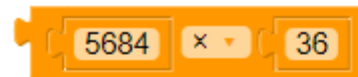
Was gibt der Code auf dem Display aus?

# 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



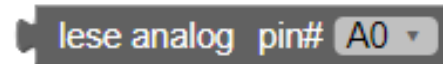
Benutzt nebenstehende Blöcke


jeweils nur einmal. Wie geht ihr vor?

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

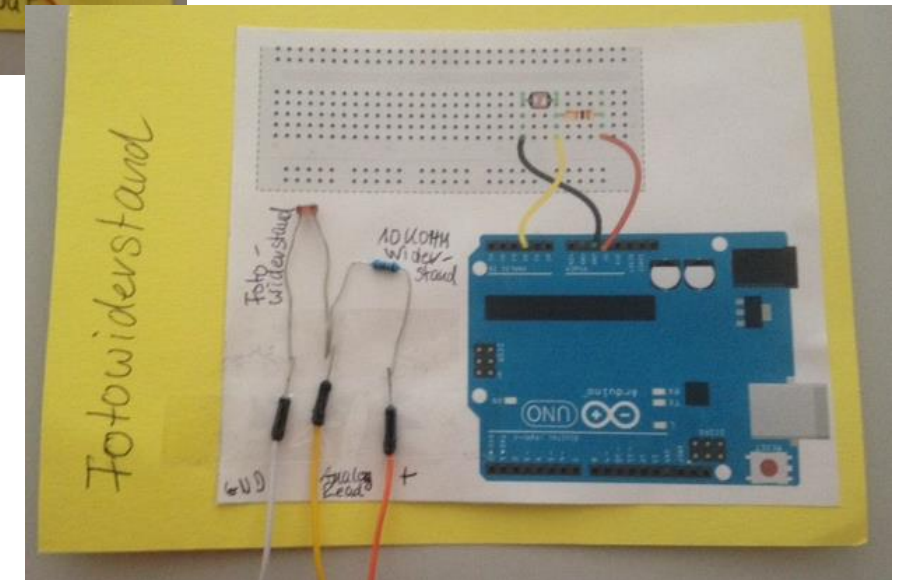
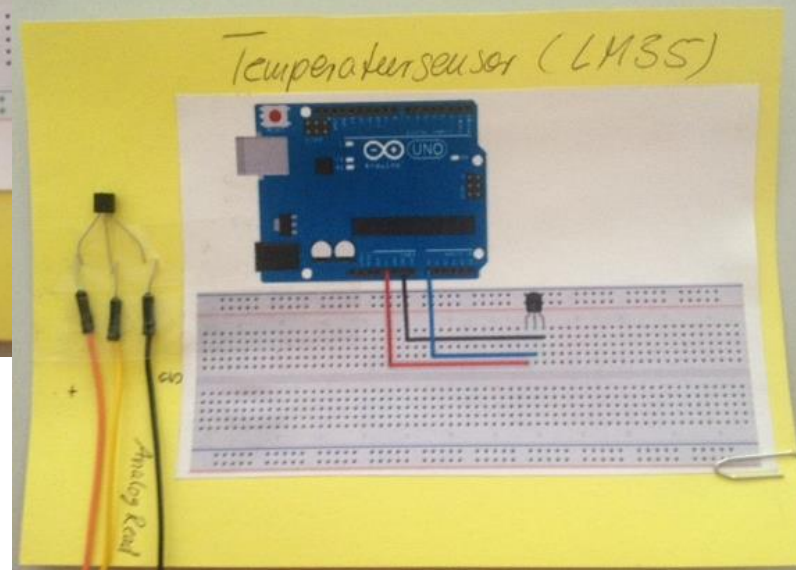
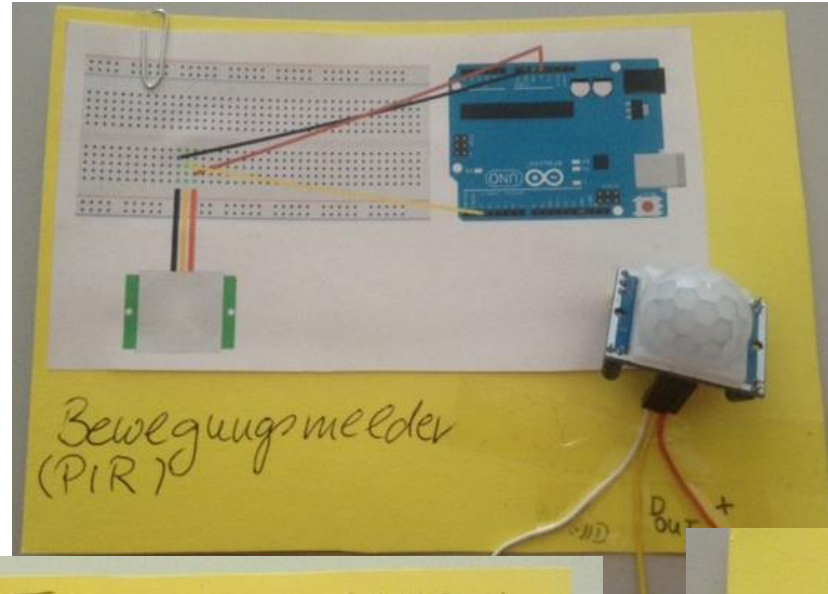
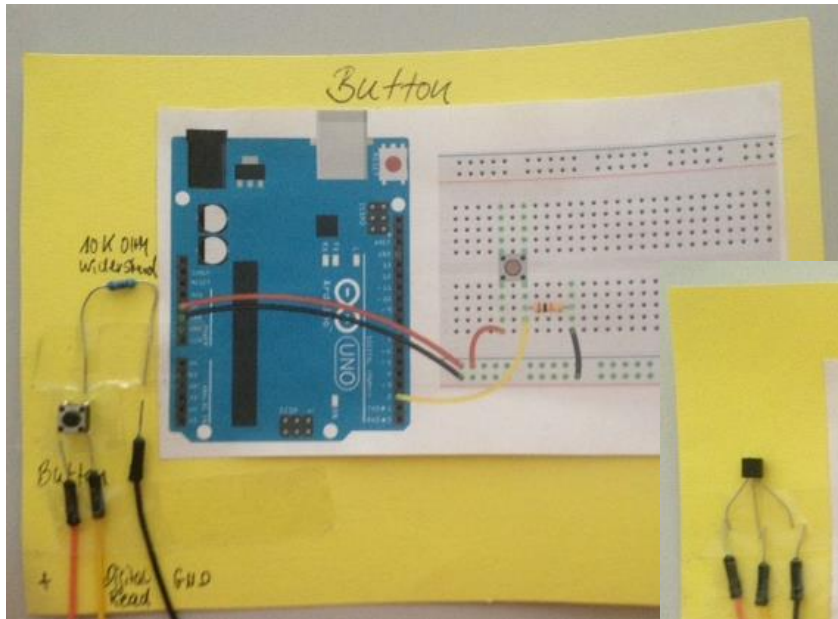
# Status holen

- Wir legen eine Variable an. In dieser speichern wir den gemessenen Wert, der an dem Pin A0 anliegt.



- Wir geben den Wert der Variable auf dem Display/über die serielle Schnittstelle aus.
- **!!! Setzt in der loop ein delay von mindestens 1 Sekunde**  **!**
- Was wird ausgegeben?

# Sensoren



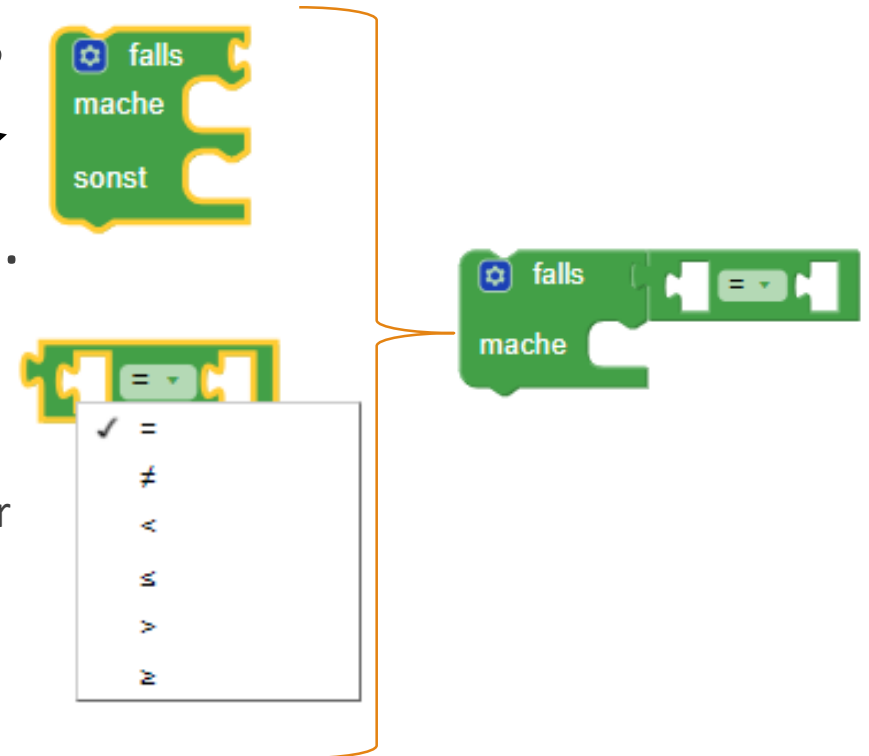
# Bedingungen

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

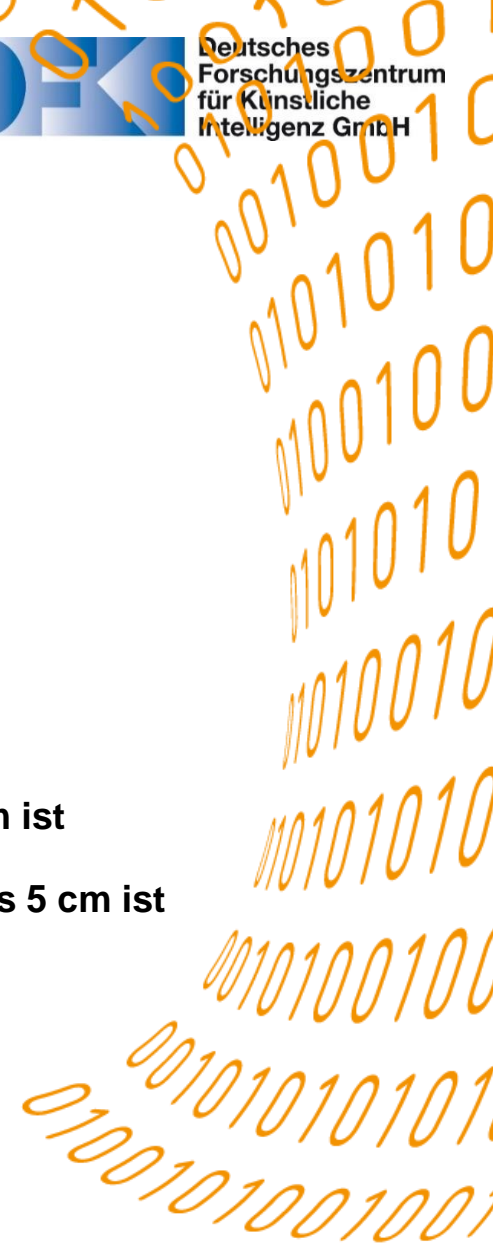
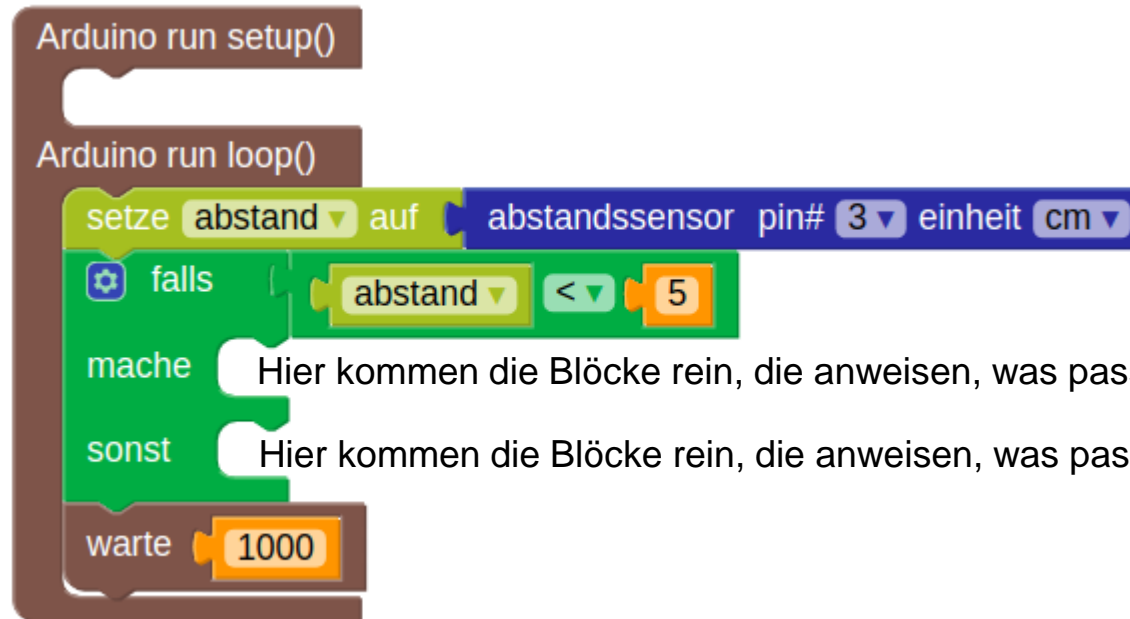
Dazu benötigen wir einen **Bedingungsblock** ...

...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





# Lösungsvorschlag

## Übung 2: Variablen



# Lösungsvorschlag

## Übung 3: Variablen (Zähler)



# Lösungsvorschlag

# Status holen

