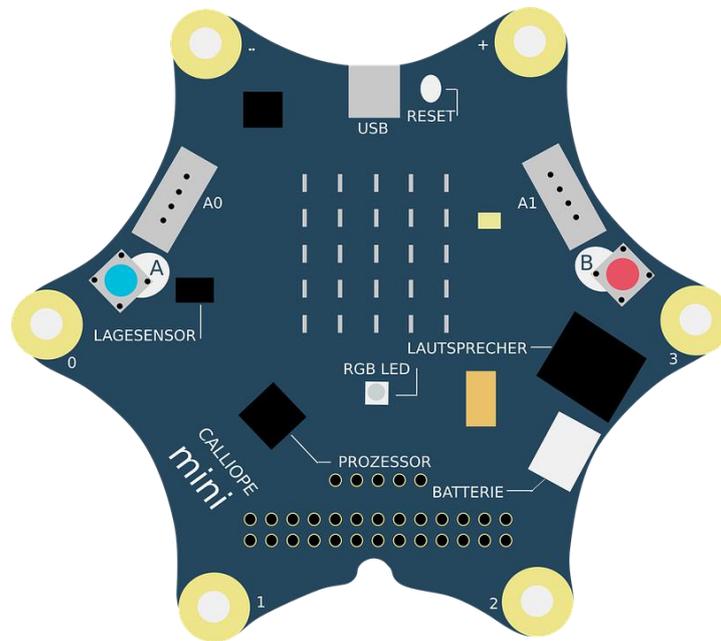


Coden

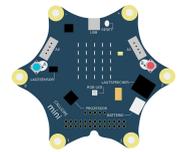
mit dem

Calliope



Inhaltsverzeichnis:

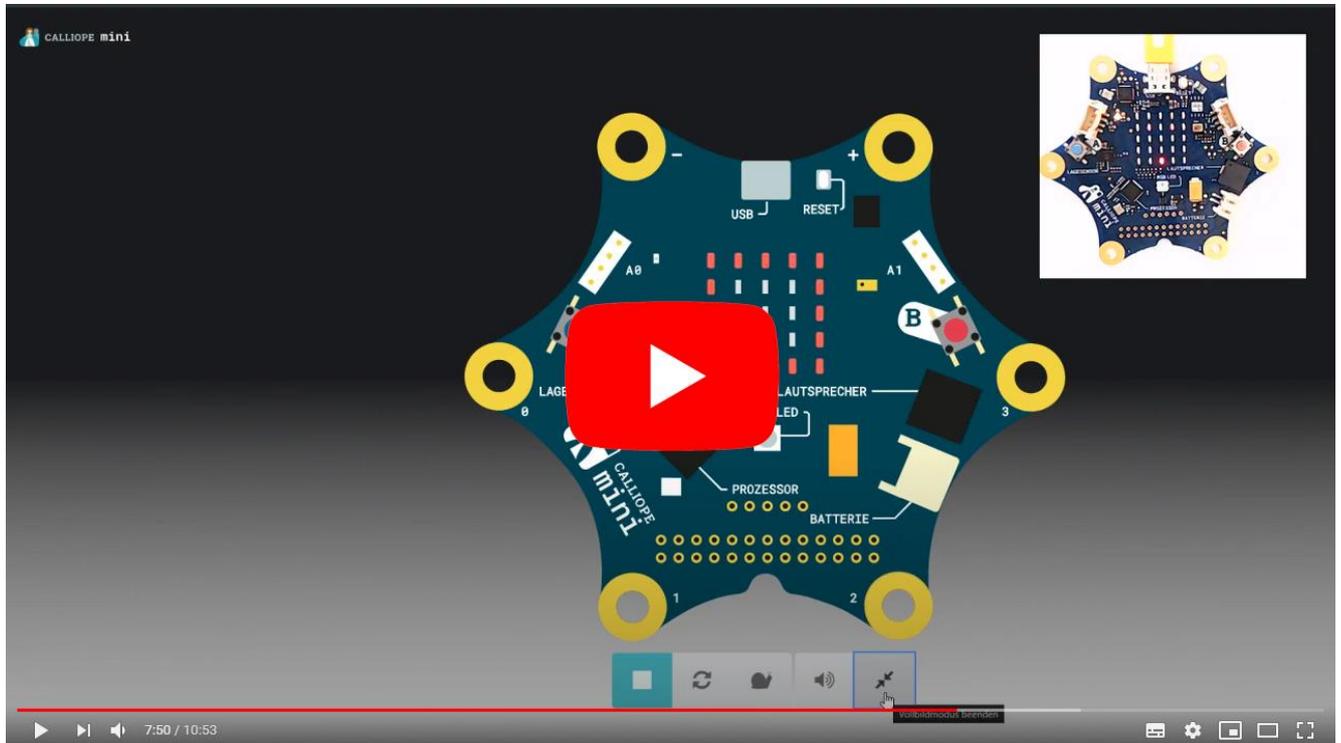
Der Einstieg.....	2
Was ist „coden“?	3
Mein erster Code: Die Ampel	4
Den Calliope anschließen	5
Leuchtdioden (LED) auf dem Calliope	6
Schleifen	7
Sensoren.....	8
bedingte Anweisung und Verzweigung.....	9
Variablen.....	10
Du hast alle Anleitungen und Aufgaben erledigt! Was nun?	11



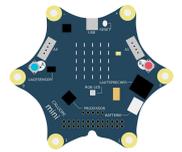
Der Einstieg

Um dir den Einstieg zu erleichtern, habe ich ein kurzes Video gedreht. Ich erkläre dir, was eigentlich ein Calliope ist, wie er funktioniert und zeige dir auch, wie die ersten „Codes“ programmiert werden.

Klicke auf das Video und schau es dir bei Youtube an!



(<https://youtu.be/Jly1mip5pJI>)

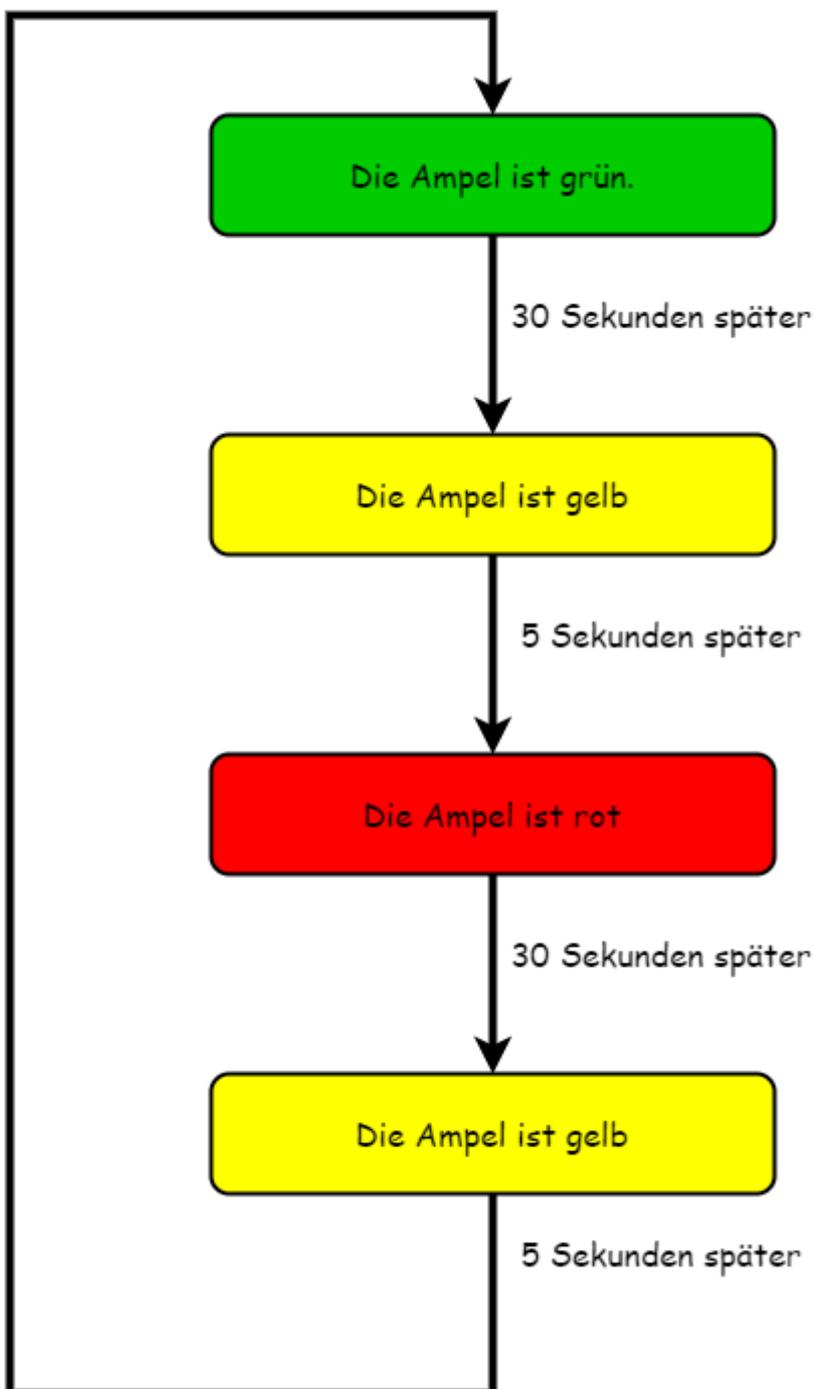


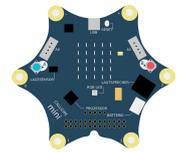
Was ist „coden“?

„Coden“ ist ein eingedeutschtes Wort. Auf Englisch heißt programmieren: „to code“. „Coden“ heißt also nichts anderes als „programmieren“. Doch was ist eigentlich programmieren? Beim Programmieren gibt man einem Computer Befehle vor, die in einer Reihe nacheinander abgearbeitet werden. Ohne die richtige Reihenfolge, kann die Aufgabe nicht erledigt werden. Du kannst dir nicht erst die Schuhe anziehen und dann die Socken. Eine richtige Reihenfolge nennt man einen Algorithmus.

Beispiel Algorithmus einer einfachen Ampel:

(eine richtige Ampel hat auch Phasen mit in denen 2 Lampen an sind.
So kompliziert machen wir uns das jetzt aber erstmal nicht)

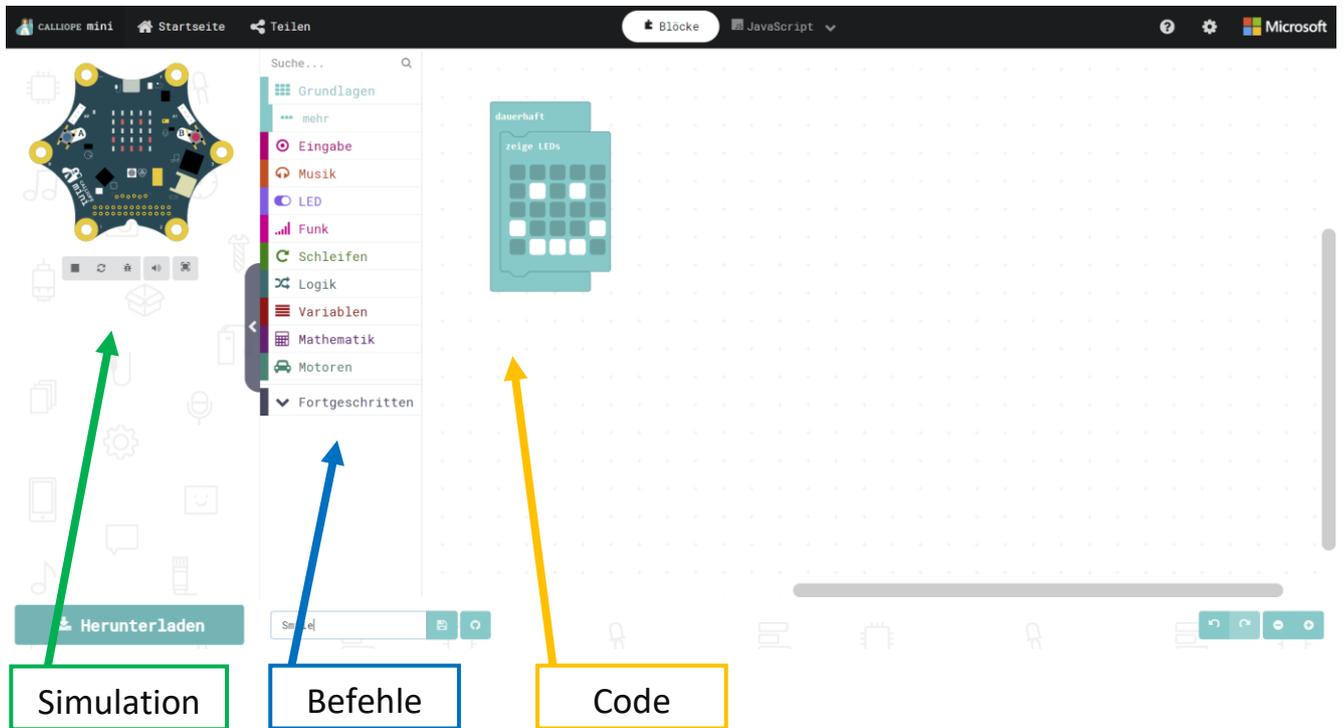




Mein erster Code: Die Ampel

Der erste Code soll eine Ampel sein. Dazu wird die RGB-LED des Calliope genutzt. Die RGB-LED kann jede Farbe darstellen. Zuerst müssen wir einen Editor öffnen, mit dem wir den Calliope programmieren können.

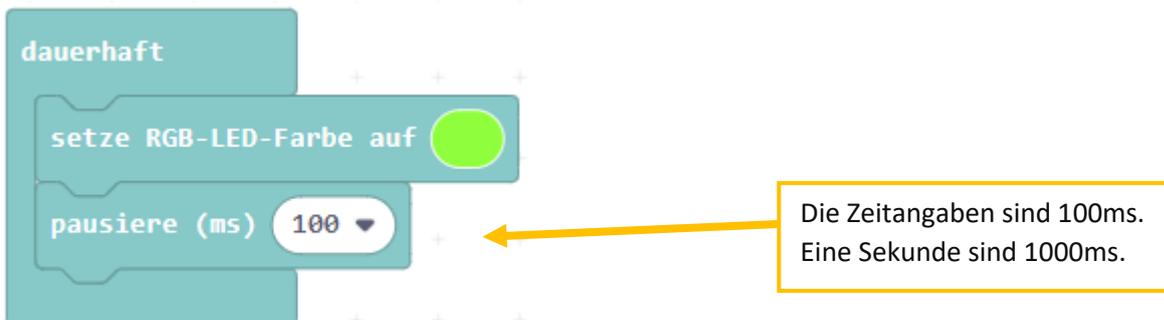
Gehe dazu im Chrome-Browser auf: makecode.calliope.cc

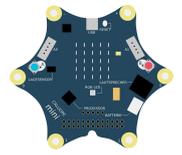


Hinweis: Du kannst Befehle löschen, indem du sie zurück in die Befehle schiebst. Neue Befehle fügst du ein, indem du sie in den Code ziehst.

Aufgabe:

Programmiere eine Ampel so, wie der Algorithmus auf der letzten Seite funktioniert. Den Anfang des Codes siehst du in der Abbildung unten. Du musst nun weitere Befehle hinzufügen, bis die Simulation die richtigen Ampelfarben anzeigt.





Den Calliope anschließen

WICHTIG!

Wenn du keinen Calliope besitzt, mach weiter auf der nächsten Seite!
Alle Codes kannst du auch simulieren.

Schritt 1.: Beachte folgendes Warnsymbol:



Das Warnsymbol bedeutet:
Achtung ESD-empfindliche Bauteile
Lese den Text „Erklärung ESD“!

Schritt 2.: **3 Sekunden die Heizung berühren!**
(um elektrostatische Ladung „abzuleiten“)

Erklärung ESD:

ESD = electrostatic discharge

(deutsch: elektrostatische Entladung)

Elektrostatische Aufladung kennen wir aus dem Alltag. Wir bekommen Einen „gewischt“. Dazu kommt es, wenn durch Reibung zum Beispiel von Socken auf einem Teppich wir uns statisch aufladen. Uns Menschen schaden diese kleinen Stromschläge nicht, einem elektronischen Bauteil schon. Elektronische Bauteile sind ESD-empfindlich.

Grade im Winter, sind wir Menschen öfter aufgeladen. Trockene Haut und trockene Heizungsluft leiten Strom sehr schlecht und verhindern, dass die Ladung über die Haut und Luft entweichen kann. Fassen wir dann ein elektronisches Bauteil an, entladen wir uns an diesem. Der Strom kann für einen sehr kurzen Augenblick sehr

Schritt 3.: Verbinde den Calliope mit dem USB-Kabel mit dem Computer



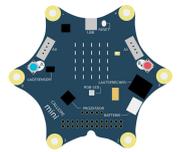
Schritt 4.: Klicke auf  und wähle links „MINI“ aus. Klicke dann auf „speichern“.

Schritt 5.: Warte kurz ab, dein Code wird gleich ausgeführt.

Nun kann der Code auf ohne Computer ausgeführt werden. Dazu muss der Calliope vom USB-Kabel getrennt und an die Batterie angeschlossen werden.



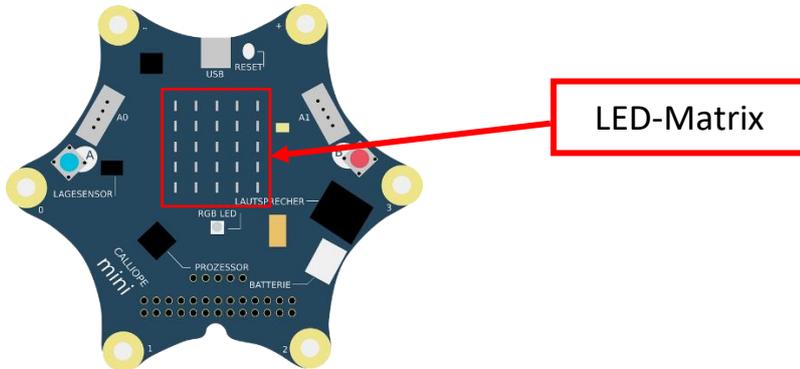
Der Calliope darf nie an Batterie und Computer gleichzeitig angeschlossen sein!



Leuchtdioden (LED) auf dem Calliope

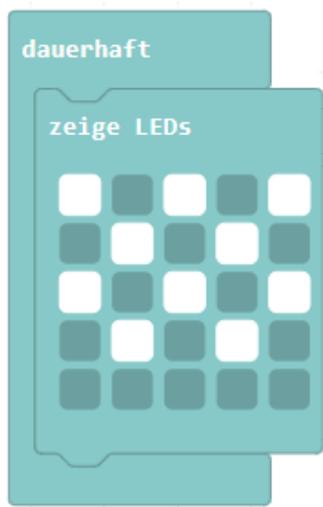
Die LED-Matrix

Neben der RGB-LED, die du schon für deine Ampel benutzt hast, hat der Calliope eine **LED-Matrix**. Diese hat 25 LED: 5 in Reihe und 5 Reihen übereinander:



Du hast 3 Möglichkeiten die LED-Matrix zu benutzen:

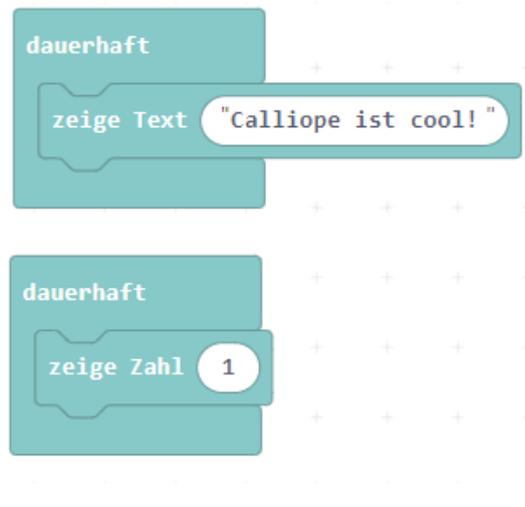
eigenes Muster oder Symbol anzeigen:



ein Symbol aus vielen auswählen:

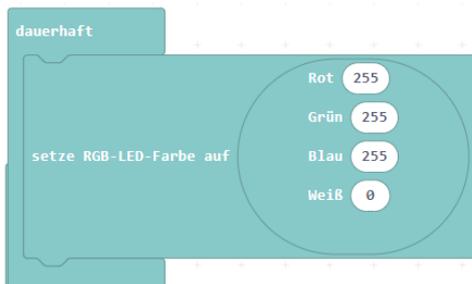


einen Text oder eine Zahl anzeigen lassen:

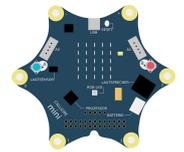


Die RGB-LED

Die Farbe der RGB-LED zu ändern hast du schon gelernt. Die RGB-LED kann aber noch viel mehr Farben anzeigen. Du musst sie nur selbst mischen. Eine RGB-LED hat in einem Bauteil eine **Rote**, eine **Grüne** und eine **Blaue** LED. Für jede Farbe kannst du ihre Leuchtstärke einstellen. 255 heißt volle Power, 0 heißt aus. Mit folgendem Code kannst du dir selbst Farben mischen:

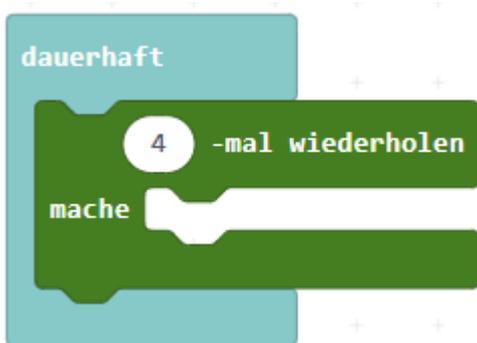


Aufgabe: Lasse die LED-Matrix und die RGB-LED verrückt blinken wie in einer Disco.

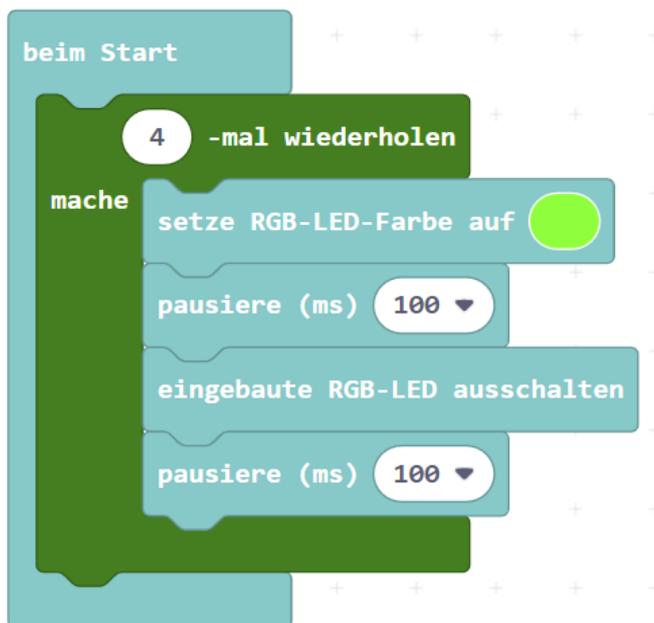


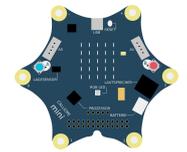
Schleifen

Bisher haben wir die „dauerhaft“-Schleife genutzt. Sie sorgt dafür, dass der Code wieder von oben beginnt, sobald er unten angekommen ist und nie endet. Wollen wir die LED 4x grün und dann 2x rot blinken lassen, wird das schwierig. Dafür benötigen wir eine andere Schleife. Du findest sie unter „Schleifen“ ganz oben:



Aufgabe: Programmiere den Calliope so, dass die RGB-LED 4x grün blinkt, dann 2x rot und anschließend wieder von vorne beginnt. Den Anfang siehst du unten. Jetzt musst du nur das rote Blinken hinzufügen.





Sensoren

Auf dem Calliope sind verschiedene Sensoren verbaut. Du kannst mit den folgenden Anleitungen die Temperatur, die Lichtstärke, die Lautstärke, die Kompassausrichtung und die Beschleunigung messen.

Aufgabe: Programmiere die Sensoren und trage deine Messwerte in die Tabelle unten ein. Versuche bei jedem Sensor einen möglichst hohen und einen möglichst niedrigen Wert zu messen.

```

dauerhaft
  zeige Zahl Temperatur (°C)
  pausiere (ms) 100
  
```

Temperatur

Die Temperatur wird mit dem Mikrocontroller gemessen. Die Temperatur ist also die des Prozessors. Mit folgendem Code kannst du dir die Temperatur auf der Matrix anzeigen lassen.

```

dauerhaft
  zeige Zahl Lichtstärke
  pausiere (ms) 100
  
```

Lichtstärke

LED können nicht nur Licht erzeugen, sondern auch Licht messen. Die LED-Matrix ist gleichzeitig der Lichtsensor. Versuche einen möglichst hohen und einen möglichst niedrigen Wert zu erreichen.

```

dauerhaft
  zeige Text Lautstärke
  pausiere (ms) 100
  
```

Lautstärke

Das Mikrofon misst die Lautstärke. Du findest den Baustein „analoge Werte von Pin ...“ unter „Fortgeschritten“ und dann „Pins“.

```

dauerhaft
  zeige Zahl Kompassausrichtung (°)
  pausiere (ms) 100
  
```

Kompassausrichtung

Die Kompassausrichtung wird vom Lagesensor gemessen. Es kann sein, dass du deinen Kompass zuerst kalibrieren musst. Dazu muss der Calliope in alle Richtungen bewegt werden, bis alle LED leuchte.

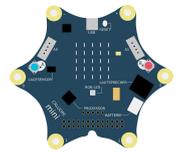
```

dauerhaft
  zeige Zahl Beschleunigung (mg) x
  pausiere (ms) 100
  
```

Beschleunigung

Die Beschleunigung wird ebenfalls vom Lagesensor gemessen. Du kannst die Beschleunigung in eine Richtung (x, y oder z) messen oder die gesamte Beschleunigung (Stärke).

Meine Messungen		
Temperatur	min:	max:
Lichtstärke	min:	max:
Lautstärke	min:	max:
Kompassausrichtung	min:	max:
beschleunigung	min:	max:



bedingte Anweisung und Verzweigung

Du kannst jetzt mit dem Calliope LED steuern und Sensordaten erfassen. Gut wäre es jetzt, wenn wir den Calliope so programmieren, dass er aufgrund von unseren Eingaben oder Sensorwerten etwas ausführt. Dazu benutzen wir ein bisschen Logik. Unter „Logik“ findest du die bedingte Anweisung und die Verzweigung. „Bedingte Anweisung“ bedeutet, der Calliope führt die nächsten Befehle nur aus, wenn ein bestimmtes Ereignis eintritt.

Aufgabe 1: Programmiere die bedingte Anweisung (Bild links) und teste den Code.

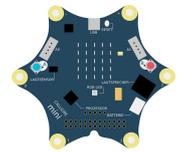
Aufgabe 2: Programmiere die Verzweigung (Bild rechts) und teste den Code.

Bedingte Anweisung: wenn	Verzweigung: wenn ... dann
<p>The image shows a Scratch-style code block for a 'Bedingte Anweisung: wenn' (conditional instruction). It starts with a 'dauerhaft' (forever) loop block. Inside the loop, there is a 'wenn Knopf A ist gedrückt dann' (if button A is pressed then) block. Below this, there is a 'setze RGB-LED-Farbe auf' (set RGB-LED color to) block with a red circle selected.</p>	<p>The image shows a Scratch-style code block for a 'Verzweigung: wenn ... dann' (branch: if ... then). It starts with a 'dauerhaft' (forever) loop block. Inside the loop, there is a 'wenn Knopf A ist gedrückt dann' (if button A is pressed then) block. Below this, there is a 'setze RGB-LED-Farbe auf' (set RGB-LED color to) block with a red circle selected. Below that, there is an 'ansonsten' (otherwise) block with a minus sign icon. Below the 'ansonsten' block, there is an 'eingebaute RGB-LED ausschalten' (turn off built-in RGB-LED) block.</p>

Überlege immer zuerst, ob du die **bedingte Anweisung** oder eine **Verzweigung** brauchst!

Ideen, wenn du schon fertig bist:

- 1.) Programmiere eine Lärmampel mit Mikrofon und der RGB-LED
- 2.) Programmiere einen Bewegungsmelder mit dem Lichtsensor und dem Lautsprecher (Musik).
- 3.) Sei kreativ, denk dir ein eigenes Programm aus.



Variablen

Variablen musst du dir vorstellen, wie kleine Kisten, in die etwas reingepackt werden kann. Verschiedene Kisten werden für verschiedene Sachen genutzt. Eine vielleicht für Spielsachen und eine andere für Bastelmaterialien. Beim Coden braucht man Kisten, um Zahlen oder Wörter darin zu lagern. Legt man sie nicht in eine Kiste, vergisst der Computer sie wieder. In einer Variablen werden sie also gespeichert. Die Bezeichnung einer Variablen sollte also beschreiben, was in dieser Kiste drin ist. Beim Coden brauchen wir Variablen, wenn wir **Sensordaten** speichern oder etwas **zählen** wollen.

Beispiel für eine Sensor-Variable:

```

dauerhaft
  setze Helligkeit auf Lichtstärke
  zeige Zahl Helligkeit
  pausiere (ms) 100
  
```

Jetzt kann „Helligkeit“ an vielen anderen Stellen im Code verwendet werden. Zum Beispiel, um bei Dunkelheit automatisch eine Lampe anzuschalten:

```

dauerhaft
  setze Helligkeit auf Lichtstärke
  wenn Helligkeit < 100 dann
    setze RGB-LED-Farbe auf [blau]
  ansonsten
    eingebaute RGB-LED ausschalten
  pausiere (ms) 100
  
```

Beispiel für eine Zähler-Variable:

Mit diesem Code wird gezählt, wie oft der Taster A gedrückt wurde. Mit Taster B wird das Zählen auf 0 zurückgesetzt.

```

dauerhaft
  zeige Zahl Taster Zähler

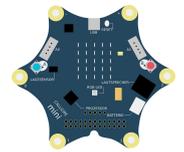
wenn Knopf A gedrückt
  ändere Taster Zähler um 1

wenn Knopf B gedrückt
  setze Taster Zähler auf 0
  
```

Aufgabe:

Probiere aus, wie du die Variablen benutzen kannst.

- 1) Programmiere eine LED, die auf Knopfdruck die Farbe wechselt (noch schwieriger: mit dem anderen Knopf wird die letzte Farbe wiederhergestellt)
- 2) Programmiere eine Alarmanlage: Wenn jemand den Calliope in seine Tasche steckt, soll ein Alarm ertönen (Nutze Befehle aus „Musik“)



Du hast alle Anleitungen und Aufgaben erledigt!

Was nun?

Jetzt geht es erst richtig los! Du kannst kreativ sein und dir **eigene Projekte** ausdenken.

Hier ein paar Ideen:

Lärmampel → Wenn es zu laut ist, leuchtet die LED rot, ansonsten grün.

Schrankwächter → Ein lautes Signal ertönt, wenn jemand deinen Schrank öffnet.

Würfel → Der Calliope zeigt eine Zahl oder ein Symbol, wenn er geschüttelt wird.

Schere, Stein, Papier → Spiele gegen den Calliope. Nach 3x Schütteln zeigt er ein Symbol an.

Liegestützzähler → Zähle deine Liegestütze. Immer wenn der Lichtsensor verdeckt ist, zählt der Calliope einen Liegestütz.

Sprungzähler → Lasse den Calliope zählen, wieviel Sprünge du mit dem Sprungseil schaffst.

Weitere Projektideen findest du unter:

<https://calliope.cc/projekte>