

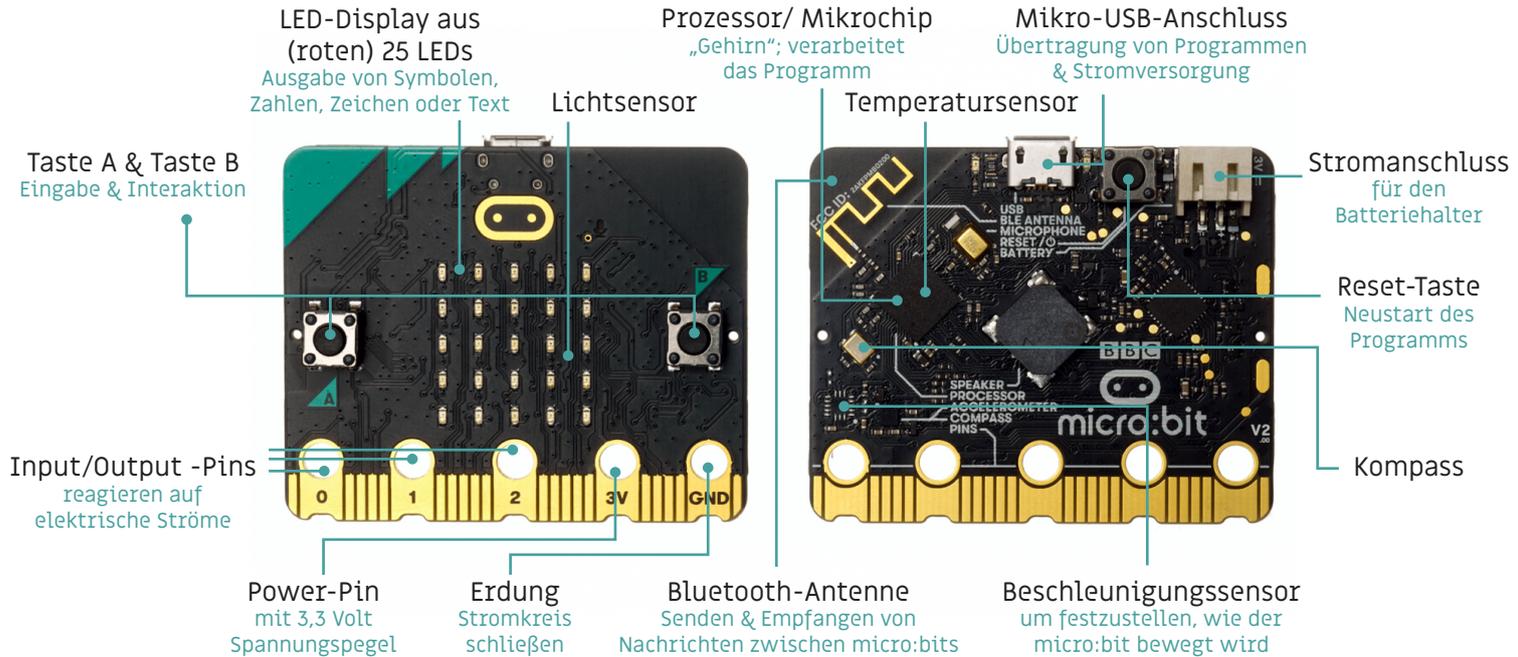


**Digital  
Literacy  
Lab**

## **MICRO:BIT LERNKARTEN**

Mit diesem Lernkartenset bekommst du alle Voraussetzungen beigebracht, um mit dem micro:bit selbstständig kleine Projekte umzusetzen.







Mit Hilfe der Lernkarten lernst du, wie der micro:bit programmiert wird. Dazu nutzt du den Browser des Computers.

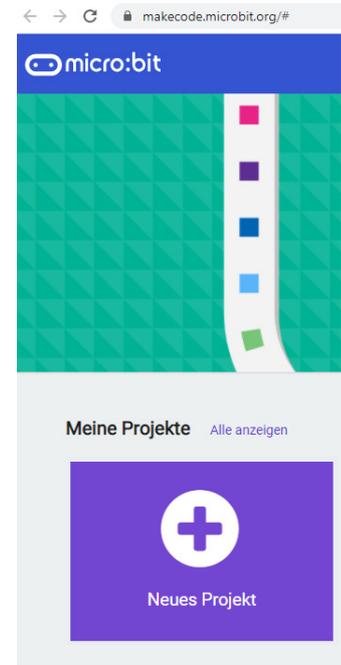
1. Gehe zu »makecode.microbit.org«.
2. Klicke »Neues Projekt« an.
3. Gib deinem Projekt einen Namen:

## Erstelle ein Projekt ✕

Gib deinem Projekt einen Namen.

[Code-Optionen](#)

**Erstelle** ✓



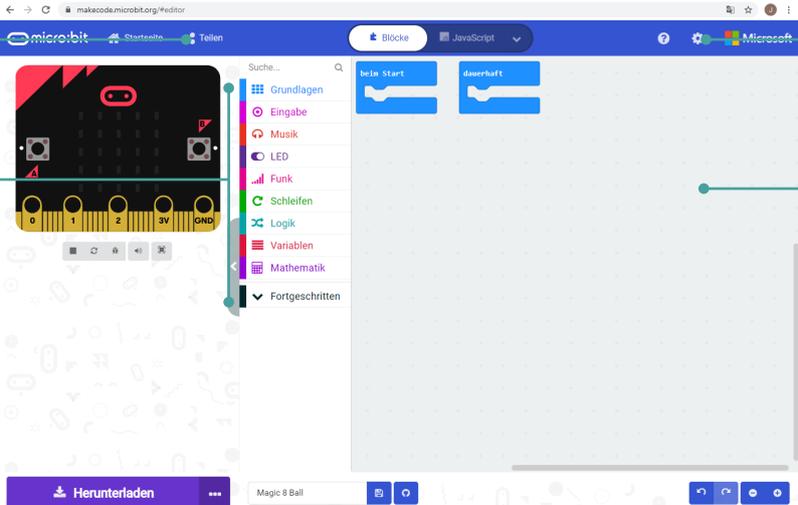


# SO SIEHT DIE OBERFLÄCHE VON MAKECODE AUS

EINFÜHRUNG | 3/4

Die Programmierumgebung findest du hier: <https://makecode.microbit.org/>

Programmiere per Drag-and-Drop (ziehe die Blöcke aus der Blockbibliothek in den Codebereich und füge sie aneinander).



The screenshot shows the MakeCode Microbit editor interface. On the left, there is a 'Blockbibliothek' (Block Library) with various categories like 'Grundlagen', 'Eingabe', 'Musik', 'Funk', 'Schleifen', 'Logik', 'Variablen', and 'Mathematik'. A red arrow points to the 'Grundlagen' section, which contains a Microbit icon. A text label 'Projekt veröffentlichen' points to the top left corner. Another text label 'Einstellungen um z.B. die Sprache zu ändern' points to the gear icon in the top right. A third text label 'Fläche zum Erstellen der Codes' points to the main workspace area. A fourth text label 'Verschiedene Arten von Codeblöcken und Erweiterungen' points to the 'Grundlagen' section of the block library.

Projekt veröffentlichen

Einstellungen um z.B. die Sprache zu ändern

Fläche zum Erstellen der Codes

Verschiedene Arten von Codeblöcken und Erweiterungen



# HIER SIEHST DU DIE UNTERSCHIEDLICHEN BEREICHE IM ÜBERBLICK

EINFÜHRUNG | 3/4

The image shows a screenshot of the online micro:bit editor interface. The browser address bar shows 'makecode.microbit.org/#editor'. The interface is divided into several sections:

- Menü:** A blue header bar at the top containing the 'micro:bit' logo, 'Startseite', 'Teilen', 'Blöcke', and 'JavaScript' dropdown menu.
- Simulation:** A window on the left showing a 3D model of a micro:bit board with a red LED and a blue button.
- Block-Bibliothek:** A central sidebar with a search bar and a list of block categories: Grundlagen, Eingabe, Musik, LED, Funk, Schleifen, Logik, Variablen, Mathematik, and Fortgeschritten.
- Codebereich:** A large workspace on the right where code blocks are assembled. It shows two blue blocks: 'beim Start' and 'dauerhaft'.

At the bottom, there is a purple 'Herunterladen' button and a status bar showing 'Magic 8 Ball' and navigation icons.



Das Programm vom Computer auf den micro:bit laden:

1. Verbinde den micro:bit mit dem USB-Kabel am Computer.
2. Klick auf den lila Balken >>Herunterladen<< links unten auf dem Bildschirm.
3. Im neuen Fenster werden die einzelnen Schritte erklärt. Folge einfach den Anweisungen. Falls du nicht weiter kommst, klicke das Fragezeichen an.
4. Verschiebe die .hex -Datei von deinem Computer auf das Laufwerk MICROBIT. Wenn die orangefarbene LED auf dem micro:bit nicht mehr blinkt, startet das Programm automatisch.

Download abgeschlossen...



**1** Verbinde den micro:bit per USB-Kabel mit deinem Computer  
Benutze den microUSB-Anschluss oben auf dem micro:bit



**2** Verschiebe die .hex-Datei auf den micro:bit  
Suche die heruntergeladene .hex-Datei und ziehe sie auf das MICROBIT-Laufwerk



Gerät koppeln



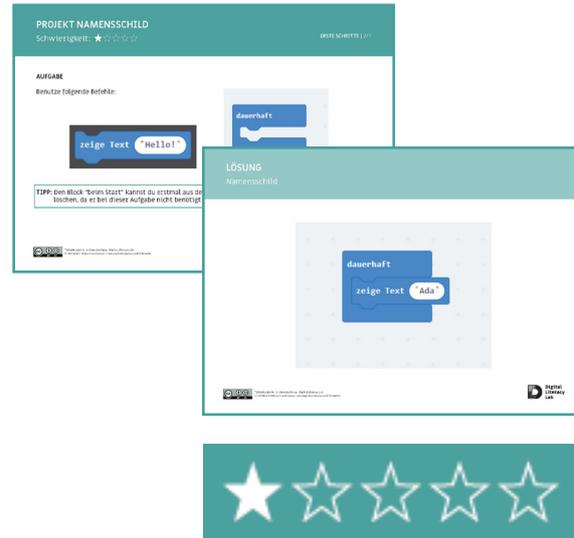
Erneut herunterladen





Jetzt geht's los!

1. Die Lernkarten haben eine Vorder- und eine Rückseite. Auf der Vorderseite befindet sich die Aufgabe, auf der Rückseite ein Lösungsvorschlag.
2. Die Sterne geben den Schwierigkeitsgrad an; je mehr Sterne abgebildet sind, desto anspruchsvoller ist die Aufgabe.
3. Die Lernkarten können als Set an die Schülerinnen und Schüler verteilt werden und dienen dem eigenständigen Lernen.





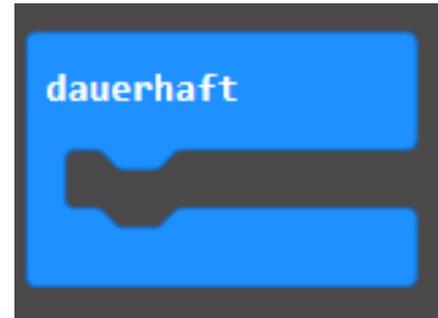
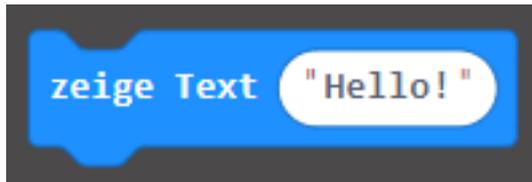
# PROJEKT NAMENSSCHILD

Schwierigkeit: ★☆☆☆☆

ERSTE SCHRITTE | 2/7

## AUFGABE

Benutze folgende Befehle:



**TIPP:** Den Block "beim Start" kannst du erstmal aus dem Codebereich löschen, da er bei dieser Aufgabe nicht benötigt wird.



# LÖSUNG

## Namensschild



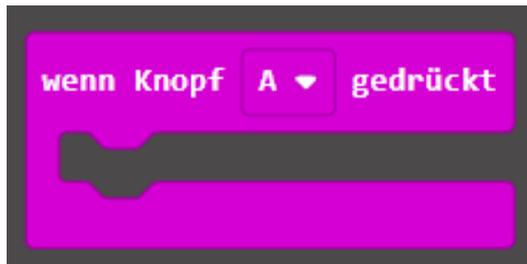
# PROJEKT SMILEY PROGRAMMIEREN

Schwierigkeit: ★★☆☆☆

ERSTE SCHRITTE | 3/7

**AUFGABE:** Programmiere die Tasten "A" und "B" auf dem micro:bit, um zu zeigen dass er glücklich oder traurig ist.

Benutze folgende Befehle:



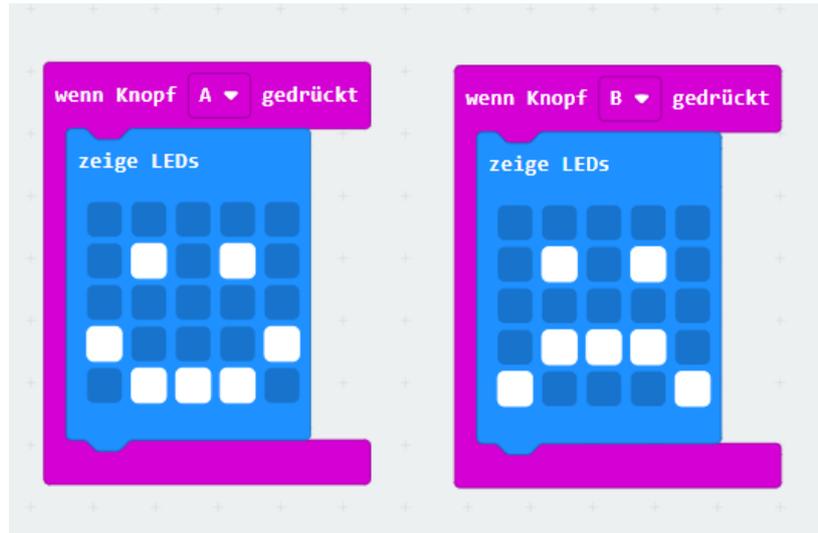
**TIPP:** Die Blöcke "beim Start" und "dauerhaft" kannst du erstmal aus dem Codebereich löschen, da sie bei dieser Aufgabe nicht benötigt werden.



TüftelAkademie | 1. Überarbeitung: Digital Literacy Lab  
CC-BY-SA 4.0: <https://creativecommons.org/licenses/by-sa/4.0/deed.de>

# LÖSUNG

## Smiley programmieren



# PROJEKT WÜRFEL PROGRAMMIEREN

Schwierigkeit: ★★☆☆☆

ERSTE SCHRITTE | 4/7

**AUFGABE:** Programmiere einen Würfel auf deinen micro:bit.

Benutze folgende Befehle:



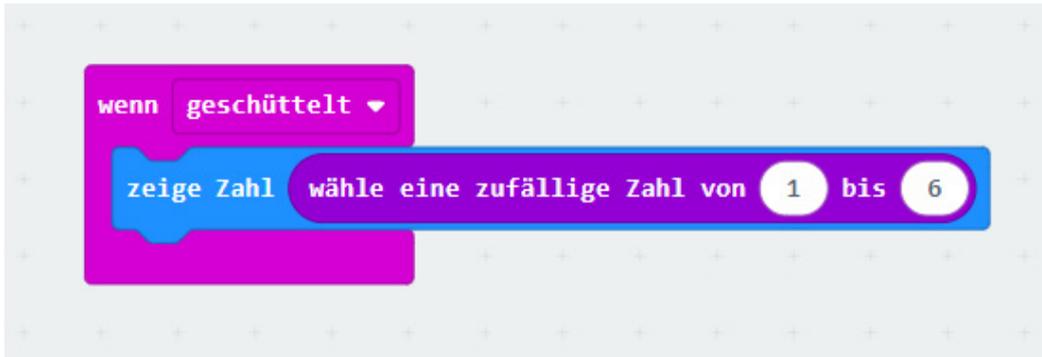
**TIPP:** Ein typischer Würfel hat die Werte 1-6.



TüfelAkademie | 1. Überarbeitung: Digital Literacy Lab  
CC-BY-SA 4.0: <https://creativecommons.org/licenses/by-sa/4.0/deed.de>

# LÖSUNG

## Würfel programmieren



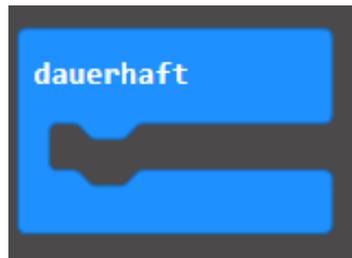
# PROJEKT NACHTLICHT PROGRAMMIEREN

Schwierigkeit: ★★☆☆☆

ERSTE SCHRITTE | 5/7

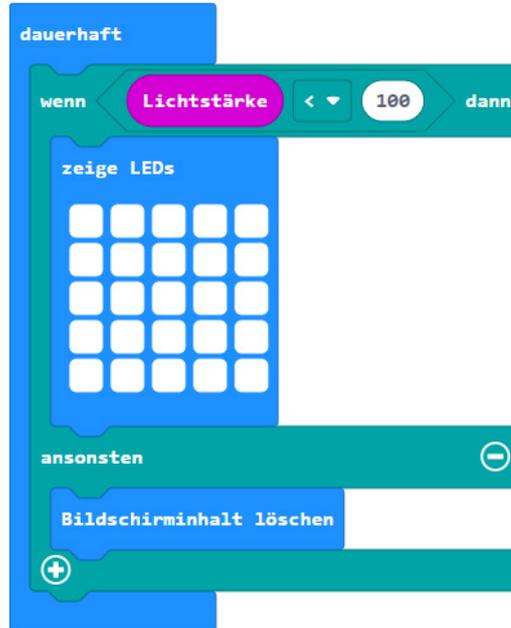
**AUFGABE:** Programmiere ein Nachtlcht: Wenn es dunkel ist, soll das LED Display vom micro:bit leuchten.

Benutze folgende Befehle:



# LÖSUNG

## Nachtlicht programmieren



# VARIABLEN

Schwierigkeit: ★★☆☆☆

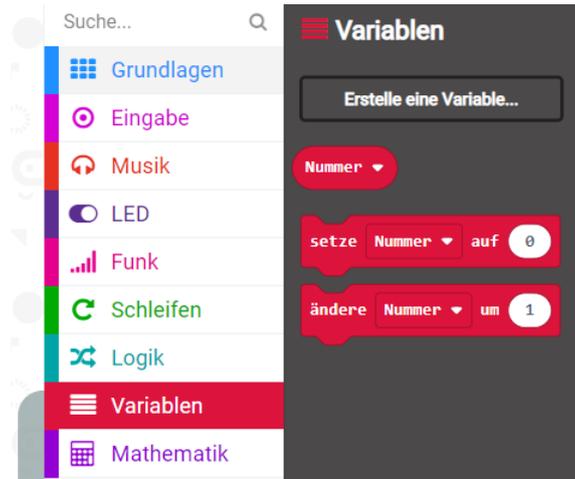
ERSTE SCHRITTE | 6/7

Eine Variable ist in der Programmierung eine Schublade zum speichern für verschiedene Daten, zum Beispiel für Zahlen (7, 2.5, 159.89532147), Zeichen (h, e, y, !) oder Zeichenfolgen (hey, Anna, HALLO!).

Diese Informationen, können während der Ausführung des Programms variieren oder sich verändern.

Du kannst dir eine Variable wie eine Box mit bestimmten Eigenschaften vorstellen. Bei der Erstellung einer Variable wird im Speicher eine bestimmte Box reserviert.

Jede Box hat einen **eindeutigen Namen**, der ihren Inhalt gut beschreibt. Das bedeutet, dass es keine zweite Variable mit demselben Namen geben darf.(1)



<https://microbit.education.at/wiki/Variablen>

**TIPP:** Unter dem Punkt Variablen in der Blockbibliothek erstellst du eine neue Variable.





# PROJEKT MAGIC 8 BALL

Schwierigkeit: ★★☆☆☆

ERSTE SCHRITTE | 7/7

**AUFGABE:** Programmiere den micro:bit so, dass wenn er geschüttelt wird, der Mikrocontroller per Zufallsauswahl eine Antwort auf eine Frage gibt mit den Antwortmöglichkeiten "ja" oder "nein".

Benutze folgende Befehle:

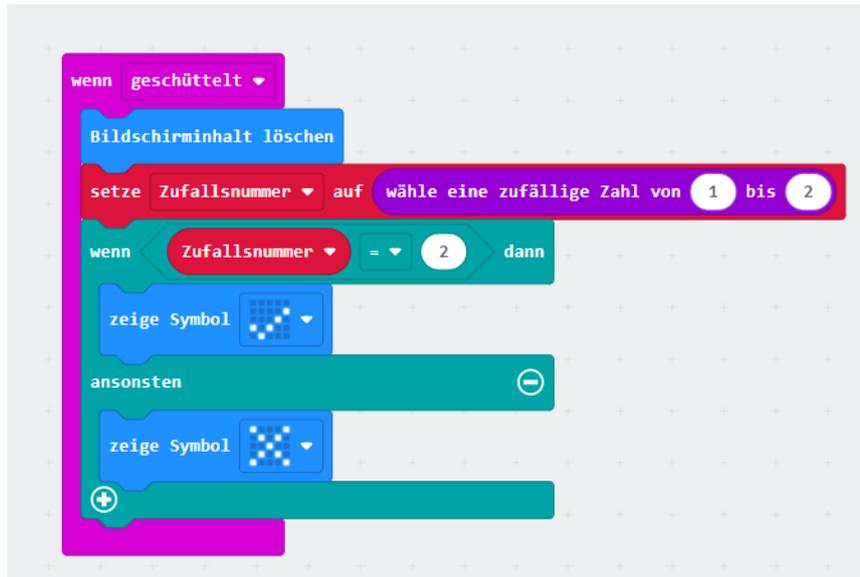


**TIPP:** Für dieses Projekt musst du eine Variable erstellen.



# LÖSUNG

## Magic 8 Ball



# PROJEKT BIENZÄHLUNG

Schwierigkeit: ★★★★★☆

BIENZÄHLUNG | 1/5

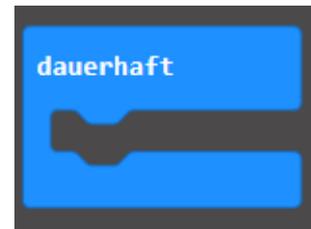
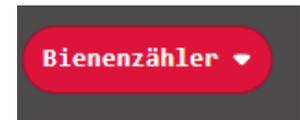
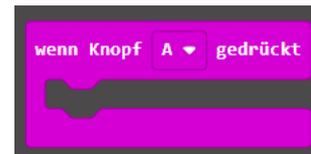
**AUFGABE:** Zähle mit dem micro:bit die Bienen und lasse dir die Zahl auf dem Display anzeigen. Angefangen bei 0 erhöht sich die Zahl auf dem Display immer um eins, wenn die Taste A gedrückt wird.

**SCHRITT 1:** Erstelle eine Variable für den Zähler.  
Welchen Wert hat diese Variable am Anfang?

**SCHRITT 2:** Erstelle einen Block, der den Zähler um eins erhöht, wenn Taste A gedrückt wird.

**SCHRITT 3:** Lasse dir den Zähler die ganze Zeit anzeigen.

Benutze folgende Befehle:



# LÖSUNG

## manuelle Bienenzählung



# PROJEKT BIENZÄHLUNG

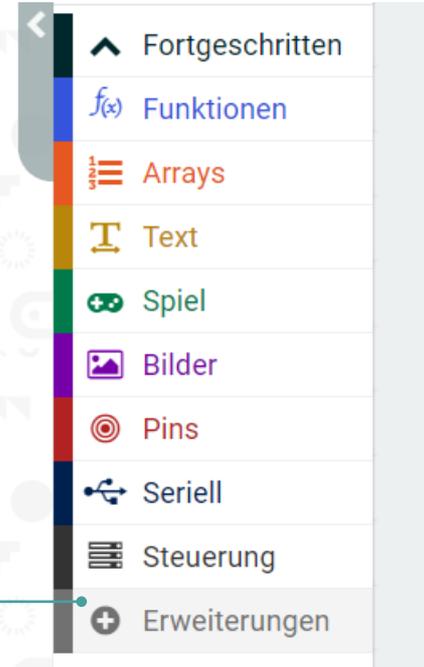
Schwierigkeit: ★★★★★☆

BIENZÄHLUNG | 2/5

**AUFGABE:** Programmiere mit dem micro:bit eine automatische Bienenzählung. Immer wenn eine Biene an dem externen Ultraschallsensor vorbeifliegt, erhöht sich die Zahl auf dem Display um 1.

**SCHRITT 1:** Füge die Programmierblöcke für die Ansteuerung der Grove-Sensoren hinzu.

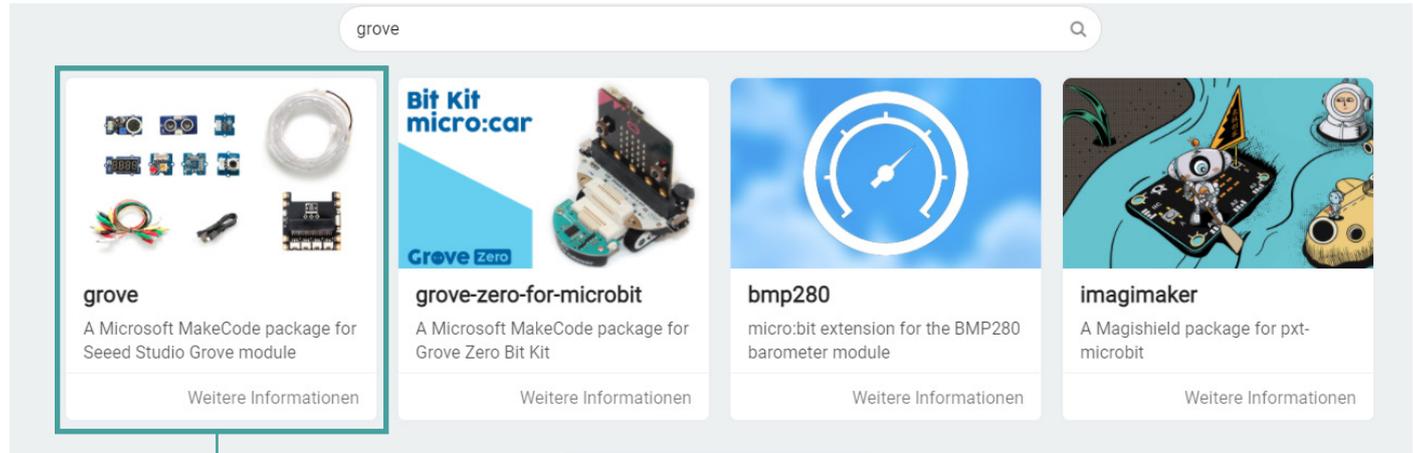
**TIPP:** Die Grove-Sensoren können über den Punkt "Erweiterungen" zur Blockbibliothek hinzugefügt werden.



# LÖSUNG

## Grove Sensoren hinzufügen

grove



**grove**  
A Microsoft MakeCode package for Seed Studio Grove module  
Weitere Informationen

**Bit Kit micro:car**  
Grove Zero  
**grove-zero-for-microbit**  
A Microsoft MakeCode package for Grove Zero Bit Kit  
Weitere Informationen

**bmp280**  
micro:bit extension for the BMP280 barometer module  
Weitere Informationen

**imagimaker**  
A Magishield package for pxt-microbit  
Weitere Informationen

auswählen



# PROJEKT BIENZÄHLUNG

Schwierigkeit: ★★★★★☆

BIENZÄHLUNG | 3/5

**SCHRITT 2:** Erstelle eine Variable "Bienenzähler".

**SCHRITT 3:** Füge in den Startblock den Befehl Bienenzähler auf 0 setzen ein.

**SCHRITT 4:** Der Bienenzähler soll dauerhaft angezeigt werden.

**SCHRITT 5:** Programmiere, dass der Bienenzähler sich um eins erhöht, wenn der Ultraschallsensor (unter Grove Blöcken) ein Signal erhält welches kleiner als 15 cm (entspricht Länge Eingangsbereich Bienenhaus) ist.

Wichtig: Wähle einen Pin aus, bei diesem Projekt P0.

Benutze folgende Befehle:

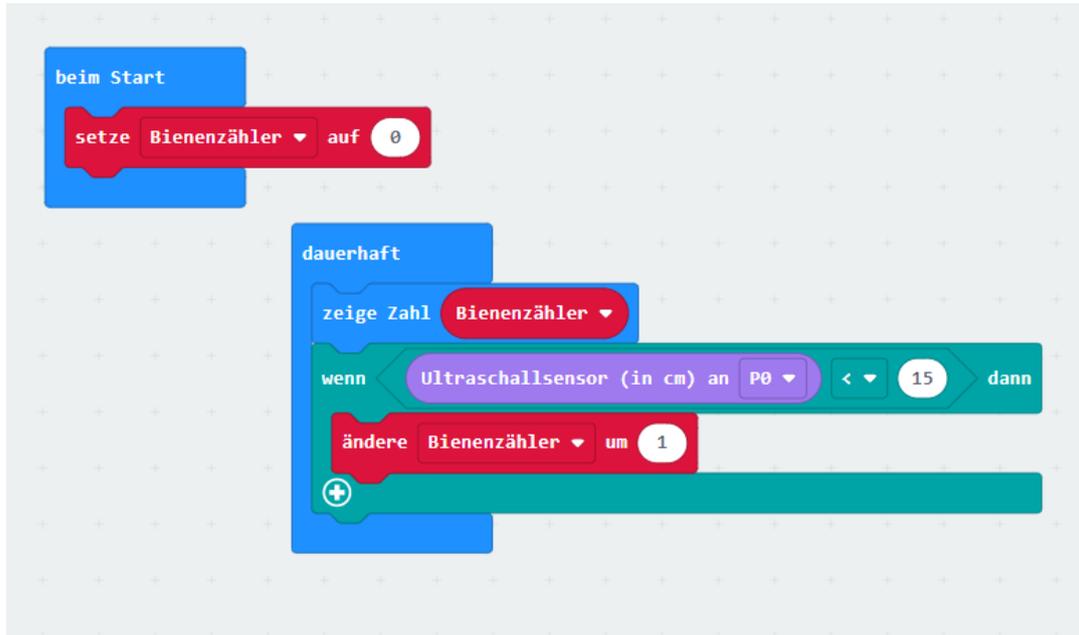


**TIPP:** Benutze die Steckverbindung P0/P14 wenn du den micro:bit mit Grove Shield und Sensor verbindest.



# LÖSUNG

## automatische Bienenzählung



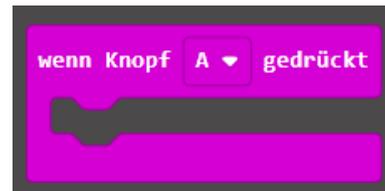
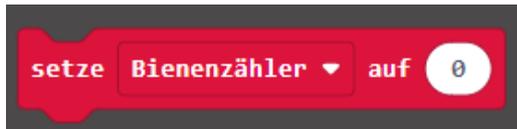
# PROJEKT BIENZÄHLUNG

Schwierigkeit: ★★☆☆☆

BIENZÄHLUNG | 4/5

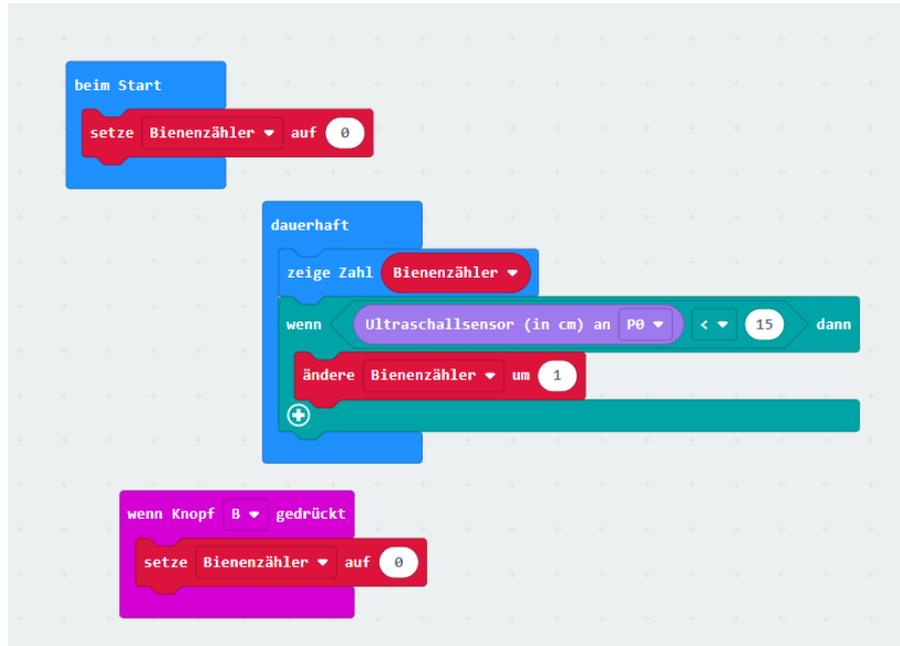
**ERWEITERUNG 1:** Der Zähler soll gelöscht werden und wieder bei 0 anfangen wenn die Taste B gedrückt wird.

Benutze folgende Befehle:



# LÖSUNG

## Bienenzähler programmieren I Erweiterung 1



# PROJEKT BIENZÄHLUNG

Schwierigkeit: ★★☆☆☆

BIENZÄHLUNG | 5/5

**ERWEITERUNG 2:** Programmiere eine Auswertung mit dem micro:bit z.b. wenn Taste A gedrückt wird zeigt der micro:bit einen lachenden Smiley bei einer hohen Bienenaktivität.

Benutze folgende Befehle:



# LÖSUNG

## Bienenzähler programmieren I Erweiterung 2

The image displays a Scratch script for a bee counter. It is organized into three main sections:

- beim Start (When Green Flag Clicked):** A red block 'setze Bienenzähler auf 0' initializes the counter to zero.
- dauerhaft (Forever Loop):** A blue loop block containing:
  - A 'zeige Zahl Bienenzähler' block to display the current count.
  - A 'wenn Ultraschallsensor (in cm) an P0 < 15 dann' block that triggers when the sensor is within 15 cm.
  - An 'ändere Bienenzähler um 1' block that increments the counter by 1.
- Event Listener:** A purple block 'wenn Knopf A gedrückt' that triggers when button A is pressed. It contains:
  - A 'wenn Bienenzähler > 100 dann' block that checks if the counter has reached 100.
  - Inside this 'wenn' block:
    - A 'zeige Symbol' block with a bee icon.
    - An 'ansonsten' block (indicated by a minus sign) containing:
      - A 'zeige Symbol' block with a bee icon.
      - A 'pausiere (ms) 100' block that pauses the program for 100 milliseconds.

