

Mein erstes Programm 2

Voraussetzung: In dieser Anleitung lernst du wie du dein Taschenlampenprogramm aus [03 Mein erstes Programm](#) erweitern kannst. Daher solltest du dieses natürlich bereits schon haben!

Ziel: Weitere Prinzipien des Programmierens kennen lernen.

Anleitung

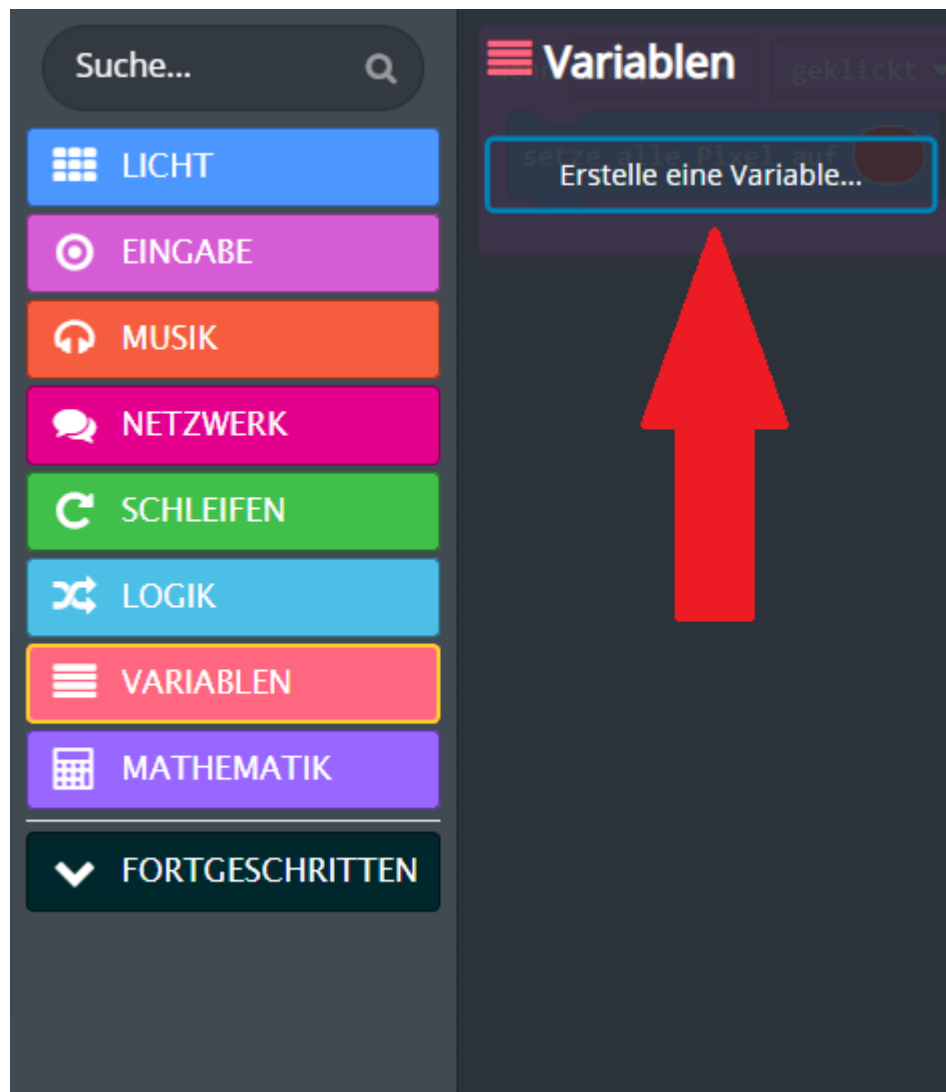
Vielleicht dachtest du schon beim Programmieren der Taschenlampe, dass normale Taschenlampen nur einen Knopf haben anstatt zwei. Bei einer herkömmlichen Taschenlampe aktiviert und deaktiviert der selbe Schalter die Lampe(n). Genau das setzen wir nun in deinem Taschenlampenprogramm um:

Überlegung: Unser Programm muss sich nun verändern. Zunächst darf nur ein Taster verwendet werden. Das Programm muss sich auch merken, ob die LEDs an oder aus sind und sich entsprechend verhalten.

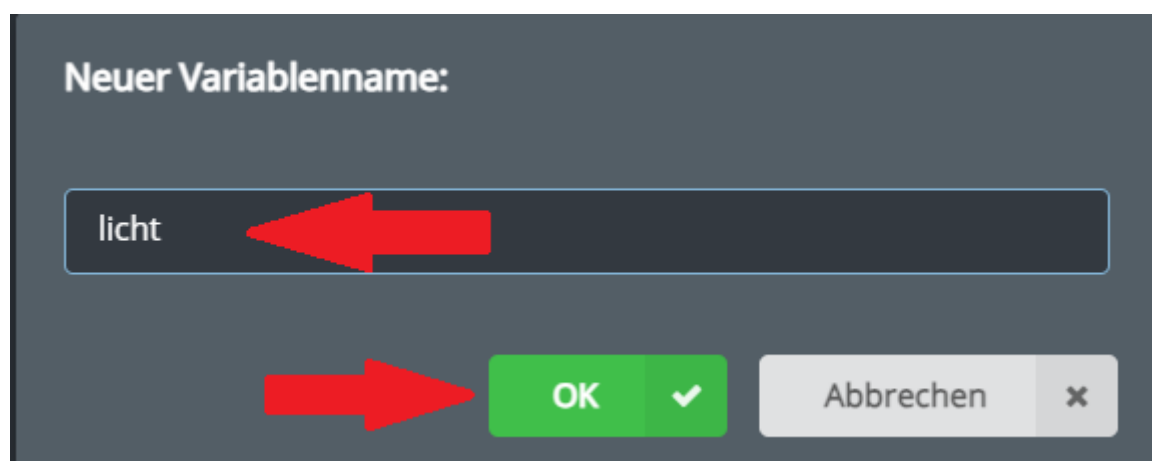
1. Variablen

Um sich zu merken, ob die LEDs an sind benötigen wir eine sogenannte "Variable". In Variablen können wir uns Daten speichern, wie zum Beispiel den Status unserer LEDs.

1. Erstellen einer Variable: Klicke in deiner Entwicklungsumgebung auf das "Variablen" Menu und dann auf "Erstelle eine Variable".



2. Danach öffnet sich ein Menu, um unserer Variablen einen Namen zu geben. Da wir damit den Status unserer LEDs speichern wollen, nennen wir sie "licht" und drücken auf "OK".



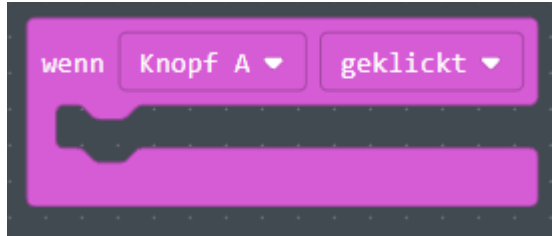
Perfekt! Du hast deine erste Variable erstellt!

2. Logik

Damit jetzt unsere Variable zum Einsatz kommen kann, brauchen wir Logik! Wir bauen Logik in unser Programm ein, damit es sich "Verzweigen" kann. Mit "Verzweigen" ist so viel gemeint, dass unser Programm unterschiedliche Dinge tut unter unterschiedlichen Umständen. In unserem Fall ist es der

Status der Variable aus Schritt 1 "licht". Ist das Licht an, soll das Programm beim nächsten Knopfdruck diese ausschalten und anders herum!

1. Zu erst müssen wir alte Funktionen entfernen: Lösche also den "setze alle Pixel auf rot" Block aus dem "wenn Knopf A geklickt" Block. Das tust du, indem du mit rechter Maustaste auf ihn drückst und danach mit der linken Maustaste die Option "Block löschen" auswählst.



2. Fügen wir anstelle nun ein Logik-Block hinzu! Dazu wählst du unter dem Logik-Menü den zweiten Block aus. Dieser ist mit "wenn wahr dann ... ansonsten" beschriftet. Danach ziehst du ihn in den "wenn Knopf A geklickt" Block.

Suche...

LICHT

EINGABE

MUSIK

NETZWERK

SCHLEIFEN

LOGIK

VARIABLEN

MATHEMATIK

FORTGESCHRITTEN

Logik

Bedingungen

wenn wahr dann

wenn wahr dann

ansonsten

Vergleich

=

<

=

Boolescher Wert

und

oder

nicht

wenn Knopf B

alle Lichter aus

Wenn ein Wert wahr ist, dann starte den ersten Block von Befehlen. Andernfalls starte den zweiten Block von Befehlen.



Dieser Block macht so ziemlich genau das was auf ihm steht. Wenn eine Variable "wahr" ist führt er dass aus, was im oberen Teil des Blockes hinzugefügt wurde, ansonsten führt er den unteren Teil aus (die Variable hat den Wert "falsch"). In unserem Fall wollen wir also wenn das Licht an ist -> es deaktivieren und es aktivieren wenn es deaktivieren.

Wichtig: Gerade steht noch "wahr" als Variable in unserem Logikblock. Das führt dazu, dass nur der erste Blockteil ausgeführt werden würde.

3. Kombinieren

1. Anstatt "wahr" fügen wir jetzt unsere "licht" Variable ein um unseren Logik-Block abhängig von dieser zu machen. Das machst du, indem du unter "Variablen" den ovalen "licht" Block an die Stelle von "wahr" ziehst.



2. Fügen wir nun unsere Funktionen hinzu: Wenn das Licht an ist wollen wir es deaktivieren, wenn es deaktiviert ist, wollen wir es aktivieren:



Ausprobieren: Starte den Simulator und steuere deine Taschenlampen nur über den A-Knopf.

Beobachtung: Die Taschenlampe geht zwar an, aber jedoch nicht aus. Das liegt daran, dass der Wert deiner "licht" Variable immer auf "Falsch" bleibt und immer der zweite Block in deinem Logik-Block ausgeführt wird.

4. Variablenwerte ändern

1. Damit unsere Taschenlampe korrekt funktioniert musst du den Wert der "licht" Variable ändern. Das tust du indem du den Block "setze Licht auf 0" zwei Mal in deinen Logik-Block einfügst. Du findest diesen Block im "Variablen" Menu.



2. Jedoch soll der Wert nicht auf "0" geändert werden, sondern auf "wahr" oder "falsch". Dafür benötigen wir die diamantförmigen Wahr- und Falsch-Blöcke aus dem Logik-Menü.



Ausprobieren: Starte den Simulator und steuere deine Taschenlampen nur über den A-Knopf.

Fertig: Deine Taschenlampe sollte nun allein über den A-Knopf steuerbar sein. (Falls nicht, vergleiche dein Code mit dem letzten Bild.)

Weiter geht es nur noch mit "Ideen und Aufgaben" Teil, indem du für dein eigenes Projekt Inspiration und Ideen sammeln kannst: [Nächster Schritt](#)

