

Dash-Crash-Kurs

ein Robotik-Workshop mit dem Roboter Dash





Übung 1: „Disco Dash“

App: Blockly

Dein Roboter hat Lust zu feiern! Lass ihn tanzen und sein Augenmuster ändern. Verändere auch die Lichtfarbe.

> *Tipp: Es gibt mehrere Möglichkeiten die Lichtfarben deines Roboters zu verändern!*

Erweiterung 1: Programmiere Dash so, dass er endlos: tanzt, sein Augenmuster und seine Lichtfarbe ändert.

Erweiterung 2: Programmiere deinen Roboter so, dass er seinen Tanz so lange wiederholt, bis du den oberen Knopf des Roboters drückst.



Übung 2: „Reihenfolge“

App: Blockly

Der Roboter soll ein Rechteck fahren. Stimmt die vorgegebene Reihenfolge? Nutze den farbigen Zettel als Start- bzw. Endpunkt. Wenn der Roboter nicht ungefähr dorthin zurück kommt, stimmt hier etwas nicht, aber was?



Übung 3: „Ampel-Roboter“

App: Blockly

Programmieren die Lichter des Roboters: Alle Lichter sollen zuerst Rot dann Gelb und dann Grün leuchten.

Erweiterung 1: Jetzt steuerst du die Ampel. Programmieren die Knöpfe so, dass er die **Farben** wechselt, **wenn** du den richtigen **Knopf** drückst.

Knopf 1: rot

Knopf 2: gelb

Knopf 3: grün

Blöcke die du brauchst:

Tanzen Dash Zuversichtlich

Augenmuster Alle an

Alle Lichter



Blöcke die du brauchst:

Wenn Dash Oberer Knopf



Alle Lichter



Übung 4: „Labyrinth“

App: Blockly

Das Startfeld ist das rote X!

Hilf Dash den Weg durchs Labyrinth zu finden. Wenn er an das Ende gelangt, soll er etwas sagen.

Erweiterung 1:

Wenn am Ende des Weges ein Monster steht, soll Dash das Hindernis erkennen, „Oh Oh“ sagen und dann den Weg rückwärts laufen.



Übung 5: „Fangen spielen“

App: Blockly

Spiele Fangen mit deinem Roboter, du bist der Fänger bzw. die Fängerin. Programmiere deinen Roboter so, dass er losfährt, wenn du dich hinter dem Roboter befindest.

Erweiterung 1: Programmiere deinen Roboter so, dass er ein Geräusch macht und die Räder stoppt, wenn du ihn aufhebst.

Erweiterung 2: Was müsstest du verändern, damit der Roboter dich fängt?



Übung 6: „Fill the Gap“

App: Blockly

Füge den Code so zusammen, dass Dash genau dort aufhört wo er begonnen hat!

Baue folgende Programmierbefehle nach und ergänze die fehlenden.

Denk' daran, manchmal fehlt nur ein Schritt, manchmal vielleicht auch 2! Welche geometrische Form fährt Dash?

Erweiterung 1: Wie ist es möglich das gleiche Ergebnis mit nur 6 Befehlsblöcken zu bauen, wobei jeder Block nicht mehr als einmal verwendet werden soll?

Blöcke die du brauchst:



Blöcke die du brauchst:





Übung 7: „Yes or No!“

App:Blockly

Dein Roboter trifft von nun an die Entscheidungen und beantwortet dir all deine Fragen. Er hat 4 Tasten auf seinem Kopf, durch die er unterschiedliche Antworten geben kann. Programmiere die 4 Knöpfe so, dass beim Drücken eines Knopfs der Roboter eine Antwort wie „Ja“, „Nein“, „Vielleicht“ usw. gibt. Du kannst eigene Antworten erfinden und aufnehmen. Ein Kind stellt die Fragen, das andere Kind darf durch das Drücken einer Taste antworten.



Übung 8: „Weg auf Knopfdruck“

App: Blockly

Programmiere die Knöpfe so, dass du den Roboter durch Drücken der Knöpfe steuern kannst:
Wenn...

Oberer (weißer) Knopf: 40cm vorwärts fahren

Knopf 1: um 90° Grad nach rechts drehen

Knopf 2 um 90° Grad nach links drehen

Knopf 3: 40cm rückwärts fahren

Fahre mit deinem Roboter einmal durch den Raum und weiche allen Hindernissen aus.

Erweiterung 1: (Zusätzlich zu den vorhandenen Blöcken): lass den Roboter anzeigen in welche Richtung er fährt. Tipp: am Ende jedes Blocks müssen die Lichter wieder weiß gedreht werden.

Oberer Knopf: vorne Lichter an; Knopf 1: Linkes Ohr rot;

Knopf 2: Rechtes Ohr rot; Knopf 3: Lichter hinten an



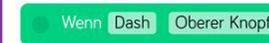
Übung 9: „Achtung Auto!“

App: Blockly

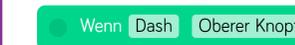
Dein Roboter ist zu Fuß unterwegs und möchte die Straße sicher überqueren.

Achtung: Manchmal kann es sein, dass sich ein Auto auf der Straße befindet, dann darf der Roboter natürlich nicht losgehen. Wenn ein Auto da ist, dann soll er seine Lichter auf Rot drehen, „Oh Oh“ sagen und warten bis das Auto weg ist. Sonst darf er über die Straße gehen.

Blöcke die du brauchst:



Blöcke die du brauchst:



Blöcke die du brauchst:





Übung 10: „What is wrong?“

App: Blockly

Dash soll ein eigenes Augenmuster haben, außerdem sollen die Lichter abwechselnd grün und rot leuchten, bis oberer Knopf gedrückt wird und Dash albern zu tanzen beginnt. Was ist an dem folgenden Bild falsch und wie gehört es richtig programmiert?



Übung 11: „Farb-Labyrinth“

App: Blockly

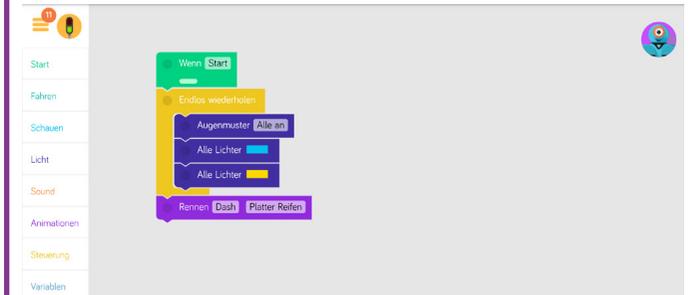
Programmiere den Roboter so, dass er den Weg des Labyrinths fahren kann. Dabei muss er sich auch nach links und rechts drehen. Seine Lichter sollen immer in der passenden Farbe zum Weg-Abschnitt leuchten.



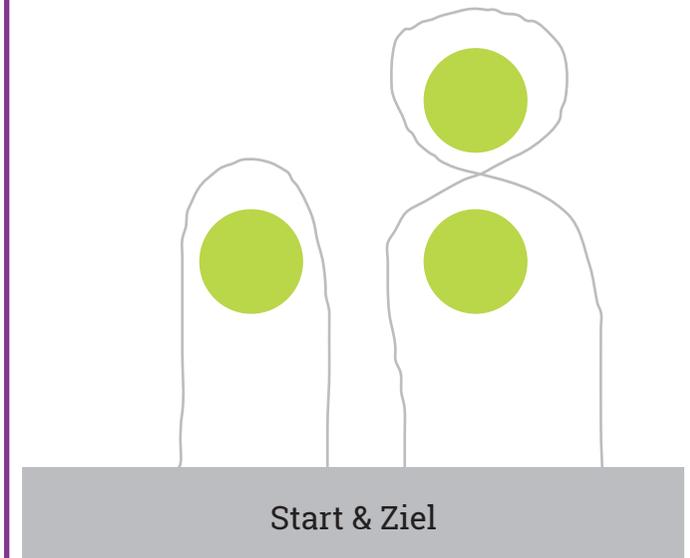
Übung 12: „Wettrennen“

App: Go

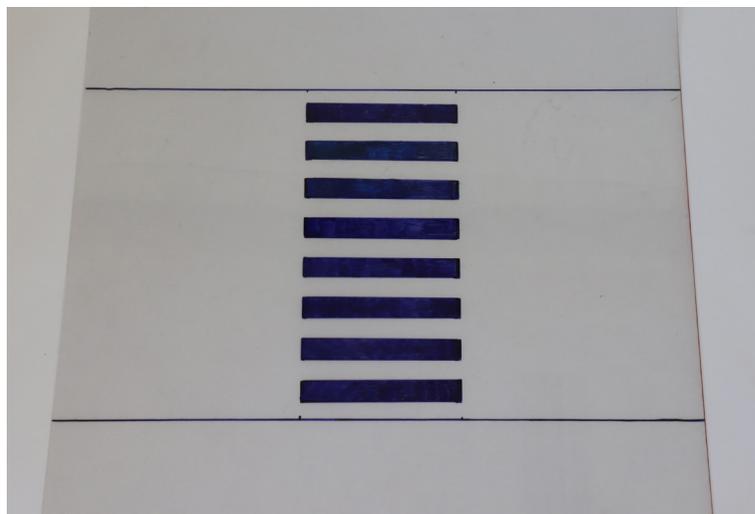
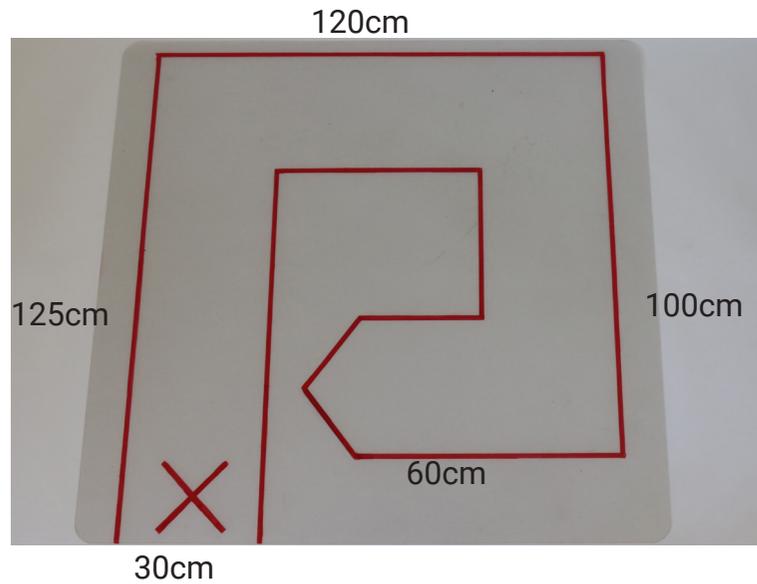
Finde ein zweites Team! Lenkt die Roboter mit der Go-App von einem festgelegten Startpunkt aus einen kurzen Parcours entlang und wieder zum Startpunkt zurück. Damit je 2 Kinder gleichzeitig gegeneinander antreten können, müssen 2 Parcours aufgebaut werden.



Blöcke die du brauchst:



Fotos zu den erstellten Matten der Übung „Labyrinth“ und „Achtung Auto“:
Das Farblabyrinth ist eine Variante des Labyrinths, wo die Außenlinien in unterschiedlichen Farben den Weg begrenzen.





Übung 1: „Disco Dash“

Du brauchst: Roboter, Tablet

App: Blockly

Vorbereitung: Suche dir einen Platz!

Dein Roboter hat Lust zu feiern! Lass ihn **tanzen** und sein **Augenmuster ändern**.
Verändere auch seine **Lichtfarbe**.

> Tipp: Es gibt mehrere Möglichkeiten die Lichtfarben deines Roboters zu verändern.

Erweiterung 1: Programmiere deinen Roboter so, dass er **endlos** tanzt, sein Augenmuster und seine Lichtfarbe ändert.

Erweiterung 2: Programmiere deinen Roboter so, dass er seinen Tanz so lange **wiederholt, bis du den oberen Knopf** des Roboters drückst.

Blöcke die du auf jeden Fall brauchst:



Blöcke die du für Erw. 1 brauchst:



Blöcke die du für Erw. 2 brauchst:





Übung 2: „Reihenfolge“

Du brauchst: Roboter, Tablet, farbiges Blatt

App: Blockly

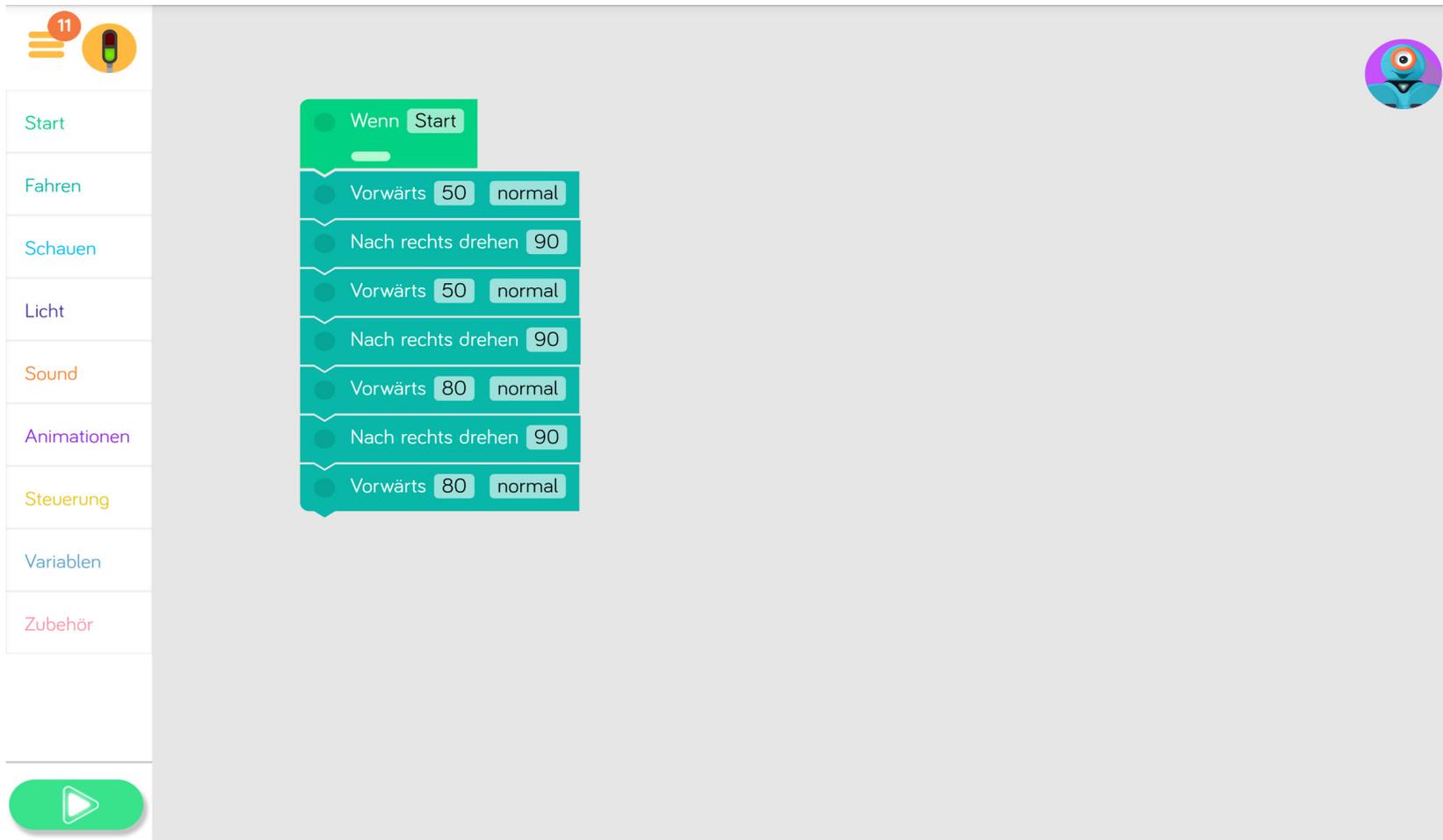
Vorbereitung: Suche dir einen Platz!

Dash soll ein Rechteck fahren.

An der vorgegebenen Reihenfolge ist etwas falsch, aber was?



Nutze den farbigen Zettel als Start- und Endpunkt. Wenn der Roboter nicht ungefähr dorthin zurück kommt, wo er startet, stimmt hier etwas nicht, aber was?



The screenshot shows the Blockly programming environment. On the left is a sidebar with categories: Start, Fahren, Schauen, Licht, Sound, Animationen, Steuerung, Variablen, and Zubehör. The main workspace contains a sequence of blocks: a green 'Wenn Start' block, followed by a teal 'Vorwärts 50 normal' block, a teal 'Nach rechts drehen 90' block, another teal 'Vorwärts 50 normal' block, another teal 'Nach rechts drehen 90' block, a teal 'Vorwärts 80 normal' block, another teal 'Nach rechts drehen 90' block, and finally a teal 'Vorwärts 80 normal' block. A play button is visible at the bottom left of the workspace.



Übung 3: „Ampel-Roboter“

Du brauchst: Roboter, Tablet

App: Blockly

Vorbereitung: Suche dir einen Platz!

Programmiere die Lichter des Roboters: **Alle Lichter** sollen zuerst **Rot** dann **Gelb** und dann **Grün leuchten**.

Erweiterung 1: Jetzt steuerst du die Ampel. Programmiere die Knöpfe so, dass er die **Farben** wechselt, **wenn** du den richtigen **Knopf** drückst.

Knopf 1: rot

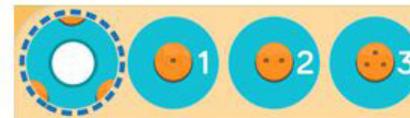
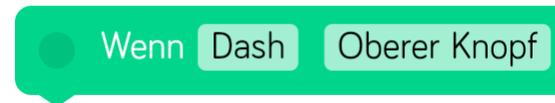
Knopf 2: gelb

Knopf 3: grün

Blöcke die du auf jeden Fall brauchst:



Erweiterung 1:





Übung 4: „Labyrinth“

Du brauchst: Roboter, Tablet, Labyrinth Matte

App: Blockly

Vorbereitung: Gehe zur Labyrinth Matte!

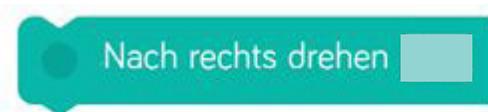
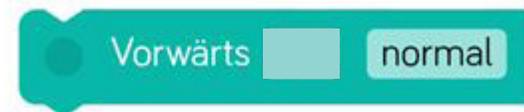
Das Startfeld ist das rote X!

Hilf dem Roboter den Weg durchs Labyrinth zu finden. Wenn er an das Ende gelangt, soll er „**Hi**“ sagen.

Erweiterung 1:

Wenn am Ende des Weges ein Monster steht, soll Dash das **Hindernis** erkennen, „**Oh Oh**“ sagen und dann den Weg rückwärts laufen.

Blöcke die du auf jeden Fall brauchst:





Übung 5: „Fangen spielen“

Du brauchst: Roboter, Tablet

App: Blockly

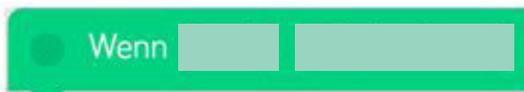
Vorbereitung: Suche dir einen Platz!

Spiele Fangen mit deinem Roboter, du bist der Fänger bzw. die Fängerin.
Programmiere deinen Roboter so, dass er losfährt, **wenn du dich hinter dem Roboter** befindest.

Erweiterung 1: Programmiere deinen Roboter so, dass er ein **Geräusch macht** und die **Räder stoppt, wenn du ihn aufhebst.**

Erweiterung 2: Was müsstest du verändern, damit der Roboter dich fängt?

Blöcke die du auf jeden Fall brauchst:





Übung 6: „Fill the Gap“

Du brauchst: Roboter, Tablet

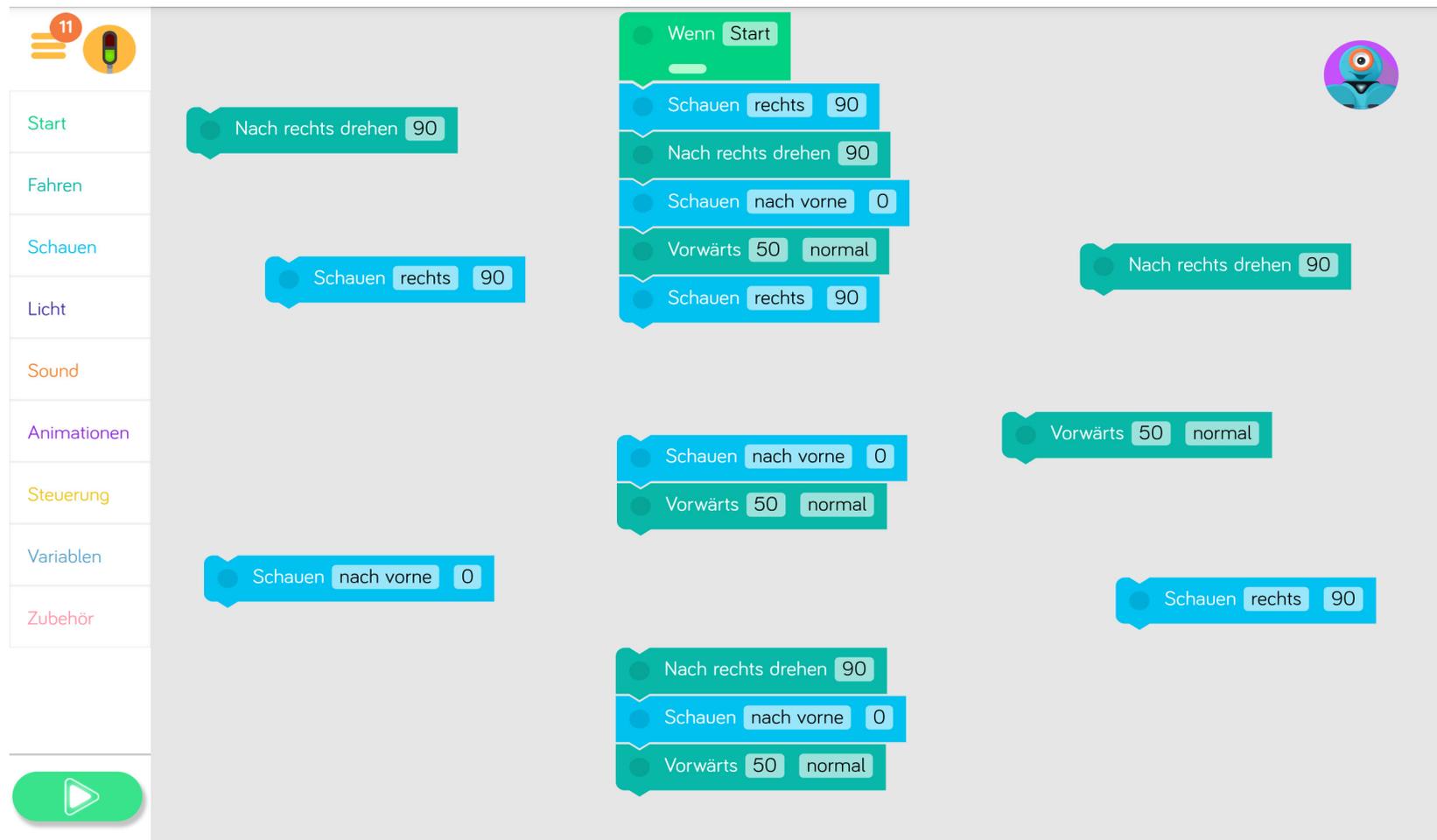
App: Blockly

Vorbereitung: Suche dir einen Platz!

Lückentext: Füge die fehlenden Blöcke so in die Lücken ein, dass der Roboter genau dort stehen bleibt, wo er startet!

Manchmal fehlt nur ein Block, manchmal vielleicht auch 2! Welche geometrische Form fährt der Roboter?

Erweiterung 1: Wie ist es möglich das gleiche Ergebnis mit **nur 6 Befehlsblöcken** zu bauen, wobei jeder Block nicht mehr als einmal verwendet werden soll?



The screenshot shows the Blockly interface with a central workspace containing several code blocks. On the left, a sidebar lists categories: Start, Fahren, Schauen, Licht, Sound, Animationen, Steuerung, Variablen, and Zubehör. A play button is at the bottom left. The workspace contains the following blocks:

- Start** (green)
- Nach rechts drehen 90** (teal)
- Schauen rechts 90** (blue)
- Schauen nach vorne 0** (blue)
- Nach rechts drehen 90** (teal)
- Vorwärts 50 normal** (teal)
- Schauen rechts 90** (blue)
- Schauen nach vorne 0** (blue)
- Vorwärts 50 normal** (teal)
- Nach rechts drehen 90** (teal)
- Vorwärts 50 normal** (teal)
- Schauen rechts 90** (blue)
- Nach rechts drehen 90** (teal)
- Schauen nach vorne 0** (blue)
- Vorwärts 50 normal** (teal)



Übung 7: „Yes or No!“

Du brauchst: Roboter, Tablet

App: Blockly

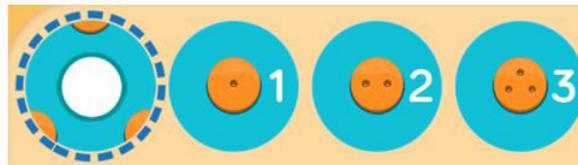
Vorbereitung: Suche dir einen Platz!

Programmiere die 4 Knöpfe am Kopf deines Roboters: **Wenn ein Knopf** gedrückt wird, soll ein eigener **Sound** erklingen.

Du kannst eigene Antworten erfinden und aufnehmen oder einfach Antworten wie „Ja“, „Nein“, „Vielleicht“ programmieren.

Nachfolgendes Spiel: Dein Roboter trifft von nun an die Entscheidungen und beantwortet dir all deine Fragen. Ein Kind stellt die Fragen, das andere Kind darf durch das Drücken einer Taste antworten.

Blöcke die du auf jeden Fall brauchst:





Übung 8: „Weg auf Knopfdruck“

Du brauchst: Roboter, Tablet

App: Blockly

Vorbereitung: Suche dir einen Platz!

Programmiere die Knöpfe so, dass du den Roboter durch Drücken der Knöpfe steuern kannst: **Wenn...**

Oberer (weißer) Knopf: 40cm vorwärts fahren

Knopf 1: um 90° Grad nach **rechts drehen**

Knopf 2 um 90° Grad nach **links drehen**

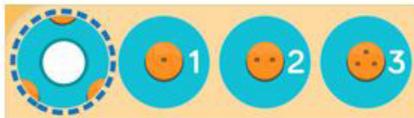
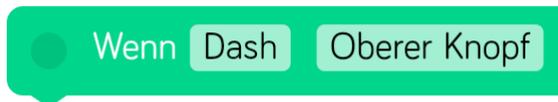
Knopf 3: 40cm rückwärts fahren

Fahre mit deinem Roboter einmal durch den Raum und weiche allen Hindernissen aus.

Erweiterung 1: (Zusätzlich zu den vorhandenen Blöcken): lass den Roboter anzeigen in welche Richtung er fährt. Tipp: am Ende jedes Blocks müssen die Lichter wieder weiß gedreht werden. **Oberer Knopf: vorne Lichter an; Knopf 1: Linkes Ohr rot;**

Knopf 2: Rechtes Ohr rot; Knopf 3: Lichter hinten an

Blöcke die du auf jeden Fall brauchst:



Erweiterung 1:





Übung 9: „Achtung Auto“

Du brauchst: Roboter, Tablet, Verkehrsmatte

App: Blockly

Vorbereitung: Gehe zu dem Platz mit der Verkehrsmatte!

Der Startpunkt ist auf einer Seite des Zebrastrreifens. Dein Roboter ist zu Fuß unterwegs und möchte die Straße sicher überqueren. Er muss daher **links, rechts** und wieder **links** schauen.

Achtung: **Wenn** sich ein **Auto vor dem Roboter** auf der Straße befindet, dann darf er natürlich nicht losgehen, dann soll er seine **Lichter auf Rot** drehen, „**Oh Oh**“ sagen und warten bis das Auto weg ist. **Sonst** darf er über die Straße gehen und „Hi“ sagen. (Das Auto darf nicht mehr als 10cm vom Roboter entfernt stehen!)

Blöcke die du auf jeden Fall brauchst:





Übung 10: „What is wrong“

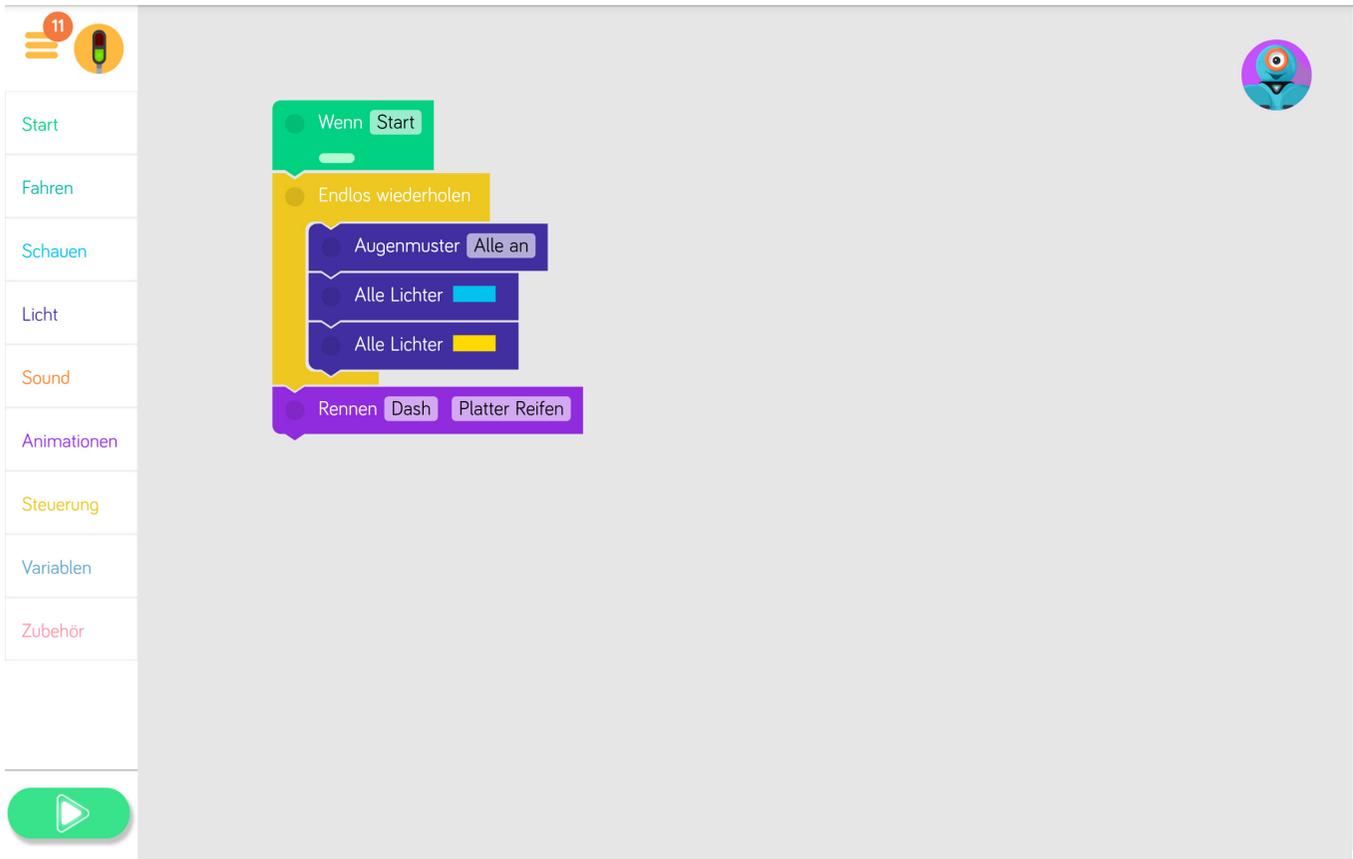
Du brauchst: Roboter, Tablet

App: Blockly

Vorbereitung: Suche dir einen Platz!

Der Roboter soll ein **eigenes Augenmuster** haben, außerdem sollen die **Lichter abwechselnd grün und rot** leuchten, bis der **obere Knopf** gedrückt wird und er **albern zu tanzen** beginnt.

Was ist an dem folgenden Bild falsch und wie gehört es richtig programmiert?



The screenshot shows the Blockly programming environment. On the left is a sidebar with categories: Start, Fahren, Schauen, Licht, Sound, Animationen, Steuerung, Variablen, and Zubehör. The main workspace contains a script starting with a green 'Wenn Start' block. This is followed by a yellow 'Endlos wiederholen' loop block. Inside the loop, there are three blue blocks: 'Augenmuster Alle an', 'Alle Lichter' (with a green light icon), and 'Alle Lichter' (with a red light icon). Below the loop is a purple 'Rennen Dash Platter Reifen' block. A play button is visible at the bottom left of the workspace.



Übung 11: „Farb-Labyrinth“

Du brauchst: Roboter, Tablet, Farb-Labyrinth-Matte

App: Blockly

Vorbereitung: Gehe zu dem Platz mit dem Farb-Labyrinth.

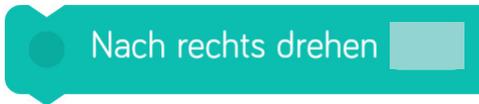
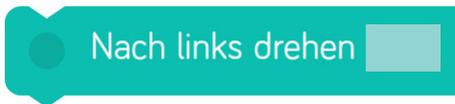
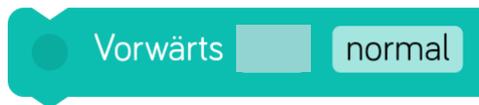
Das Startfeld ist das große X!

Programmiere den Roboter so, dass er den Weg des Labyrinths **fahren** kann. Dabei muss er sich auch nach **links** und **rechts drehen**.

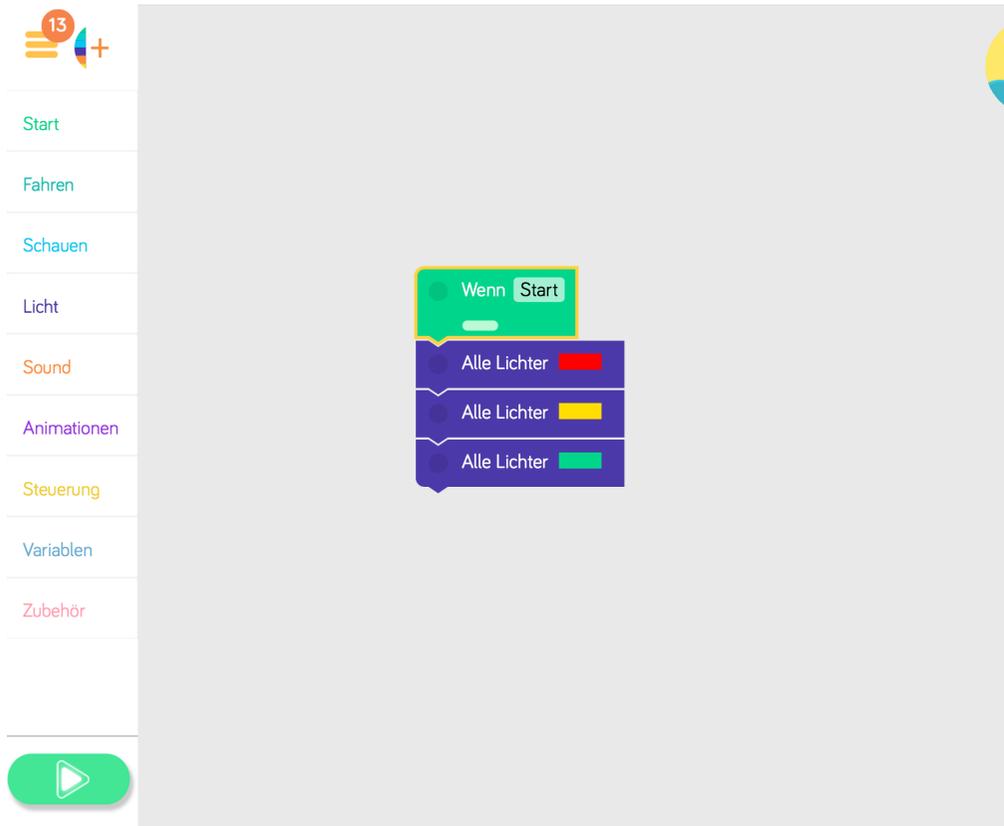
Seine Lichter sollen immer in der passenden Farbe zum Weg-Abschnitt leuchten.

Tip: Denk daran, eine Viertel-Drehung hat 90°, die Hälfte davon sind 45°!

Blöcke die du auf jeden Fall brauchst:



„Ampel-Roboter“



13 +

- Start
- Fahren
- Schauen
- Licht
- Sound
- Animationen
- Steuerung
- Variablen
- Zubehör

Wenn Start

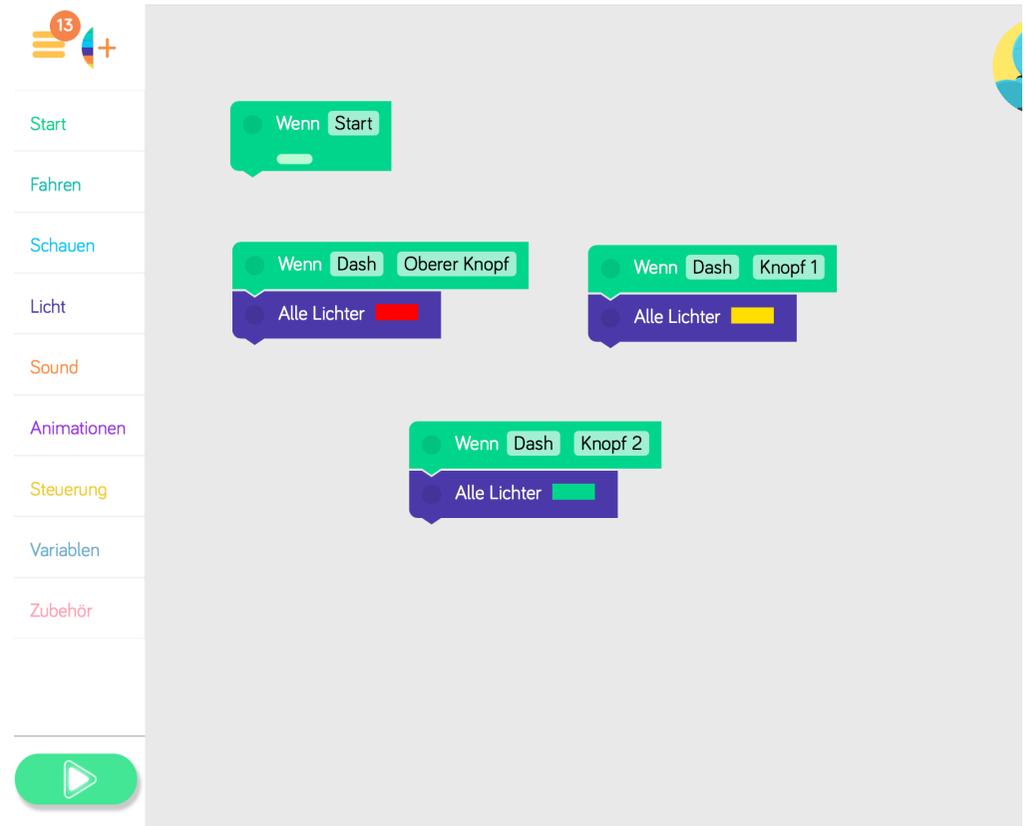
Alle Lichter ■

Alle Lichter ■

Alle Lichter ■

▶

Erweiterung 1:



13 +

- Start
- Fahren
- Schauen
- Licht
- Sound
- Animationen
- Steuerung
- Variablen
- Zubehör

Wenn Start

Wenn Dash Oberer Knopf

Alle Lichter ■

Wenn Dash Knopf 1

Alle Lichter ■

Wenn Dash Knopf 2

Alle Lichter ■

▶

„Labyrinth“

11 

- Start
- Fahren
- Schauen
- Licht
- Sound
- Animationen
- Steuerung
- Variablen
- Zubehör



```
Wenn Start
  Vorwärts 90 normal
  Nach rechts drehen 90
  Vorwärts 80 normal
  Nach rechts drehen 90
  Vorwärts 70 normal
  Nach rechts drehen 90
  Vorwärts 40 normal
  Sag Dash Hi
```

Erweiterung 1:

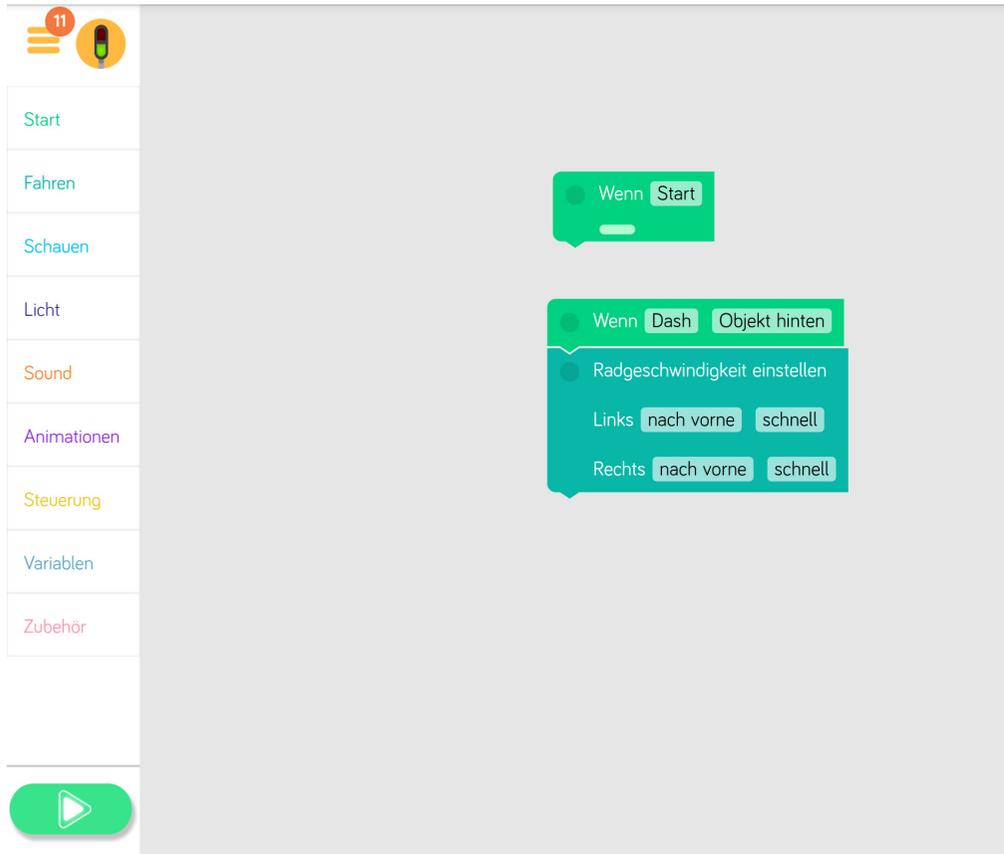
11 

- Start
- Fahren
- Schauen
- Licht
- Sound
- Animationen
- Steuerung
- Variablen
- Zubehör



```
Wenn Start
  Vorwärts 80 normal
  Nach rechts drehen 90
  Vorwärts 80 normal
  Nach rechts drehen 90
  Vorwärts 60 normal
  Nach rechts drehen 90
  Vorwärts 40 normal
  Wenn Dash Hindernis vorne
    Sag Dash Oh, oh!
    Rückwärts 40 normal
    Nach links drehen 90
    Rückwärts 60 normal
    Nach links drehen 90
    Rückwärts 80 normal
    Nach links drehen 90
    Rückwärts 80 normal
  Sonst
    Sag Dash Hi
```

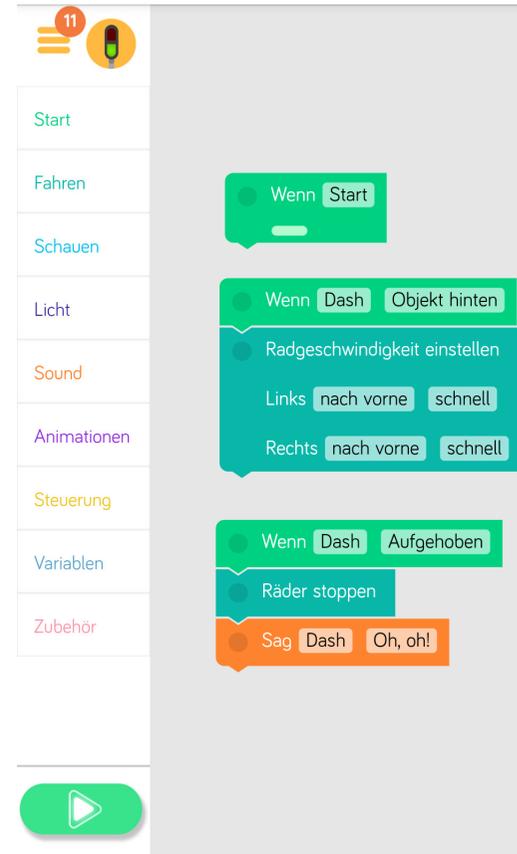
„Fangen spielen“



Scratch script for the 'Fangen spielen' project. The script is located in the 'Steuerung' (Control) category and consists of the following blocks:

- Start
- Wenn Start
- Wenn Dash Objekt hinten
- Radgeschwindigkeit einstellen
- Links nach vorne schnell
- Rechts nach vorne schnell

Erweiterung 1:



Scratch script for the 'Erweiterung 1' project. The script is located in the 'Steuerung' (Control) category and consists of the following blocks:

- Start
- Wenn Start
- Wenn Dash Objekt hinten
- Radgeschwindigkeit einstellen
- Links nach vorne schnell
- Rechts nach vorne schnell
- Wenn Dash Aufgehoben
- Räder stoppen
- Sag Dash Oh, oh!

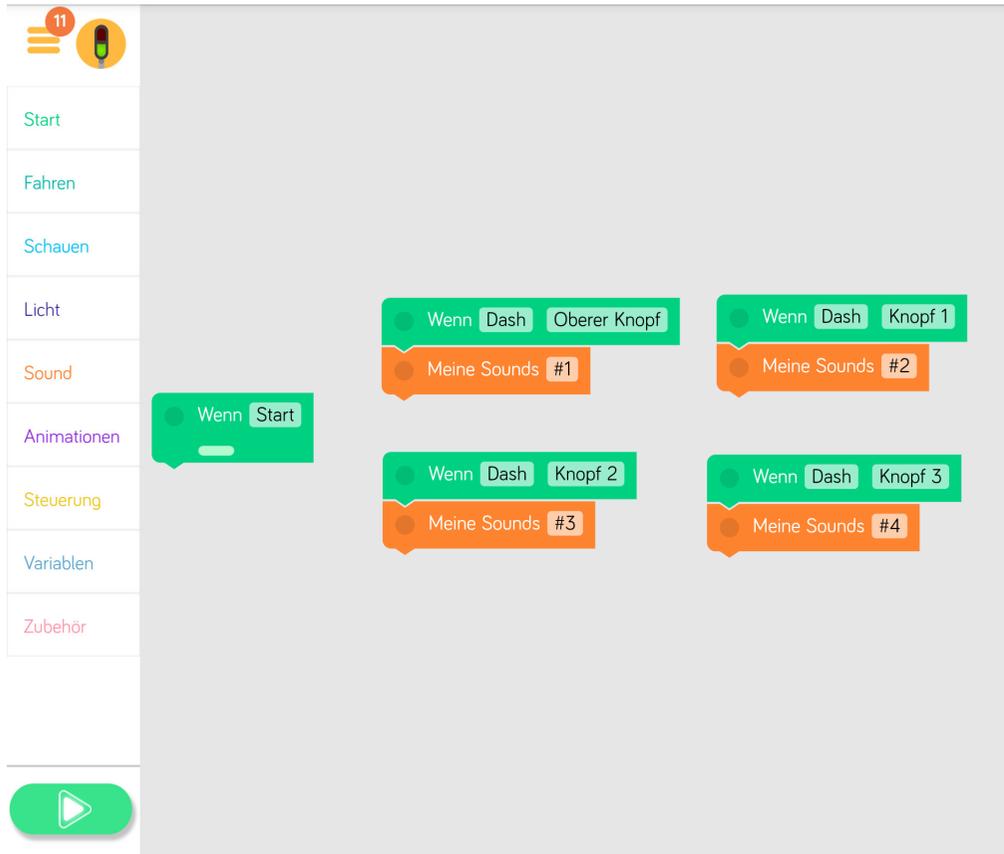
Erweiterung 2:



Scratch script for the 'Erweiterung 2' project. The script is located in the 'Steuerung' (Control) category and consists of the following blocks:

- Wenn Start
- Wenn Dash Hindernis vorne
- Vorwärts 70 schnell
- Wenn Dash Aufgehoben
- Räder stoppen
- Sag Dash Oh, oh!

„Yes or No!“



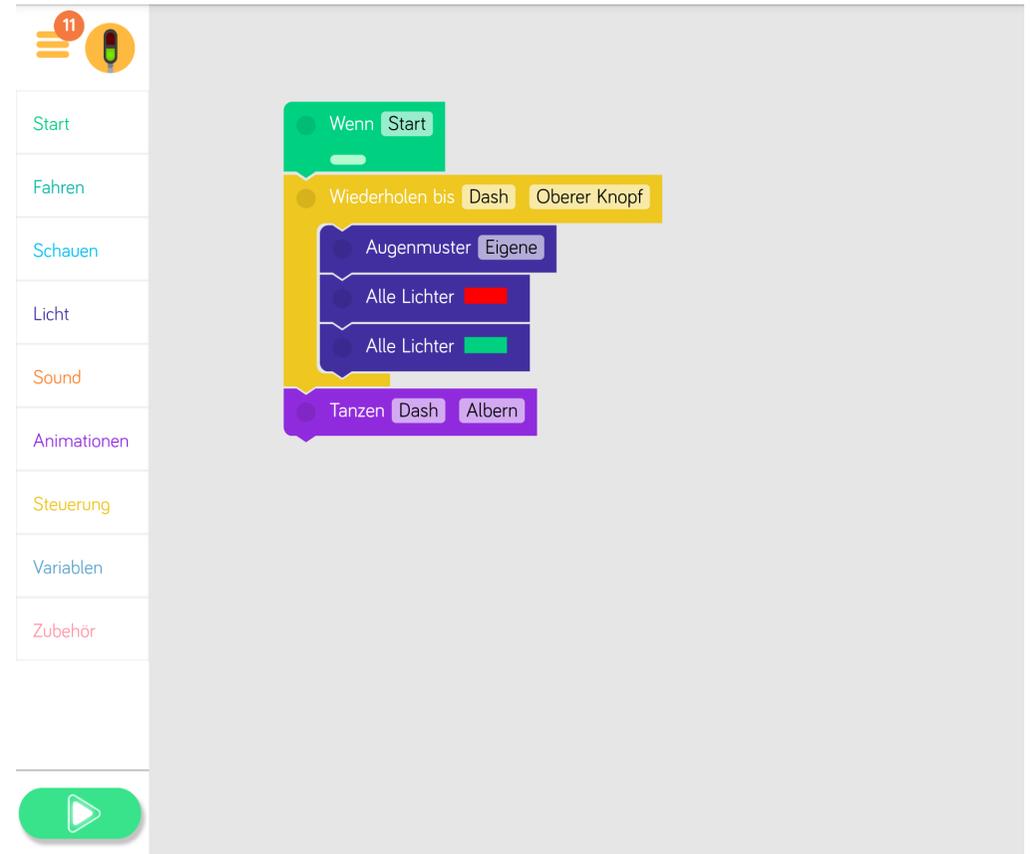
Scratch script for the challenge „Yes or No!“:

- Start
- Fahren
- Schauen
- Licht
- Sound
- Animationen
- Steuerung
- Variablen
- Zubehör

Script blocks:

- Wenn Start (green flag clicked) → empty block
- Wenn Dash Oberer Knopf → Meine Sounds #1
- Wenn Dash Knopf 1 → Meine Sounds #2
- Wenn Dash Knopf 2 → Meine Sounds #3
- Wenn Dash Knopf 3 → Meine Sounds #4

„What is wrong?“



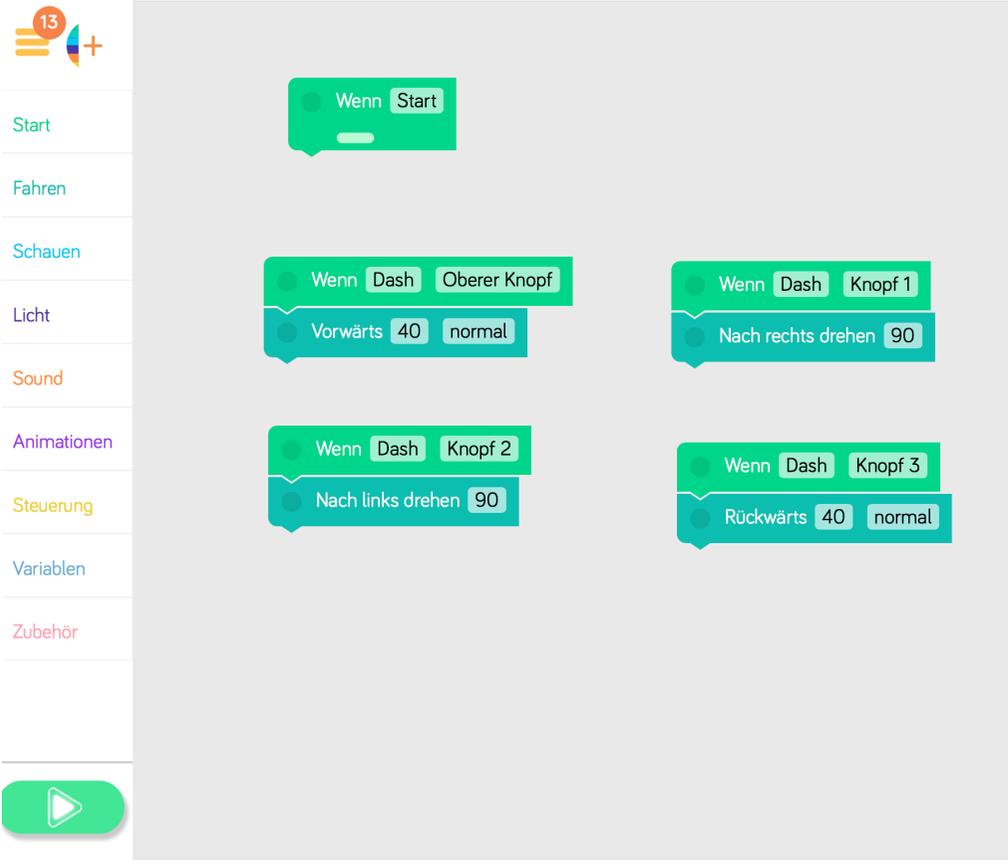
Scratch script for the challenge „What is wrong?“:

- Start
- Fahren
- Schauen
- Licht
- Sound
- Animationen
- Steuerung
- Variablen
- Zubehör

Script blocks:

- Wenn Start (green flag clicked) → Wiederholen bis Dash Oberer Knopf
 - Augenmuster Eigene
 - Alle Lichter (red)
 - Alle Lichter (green)
- Tanzen Dash Albern

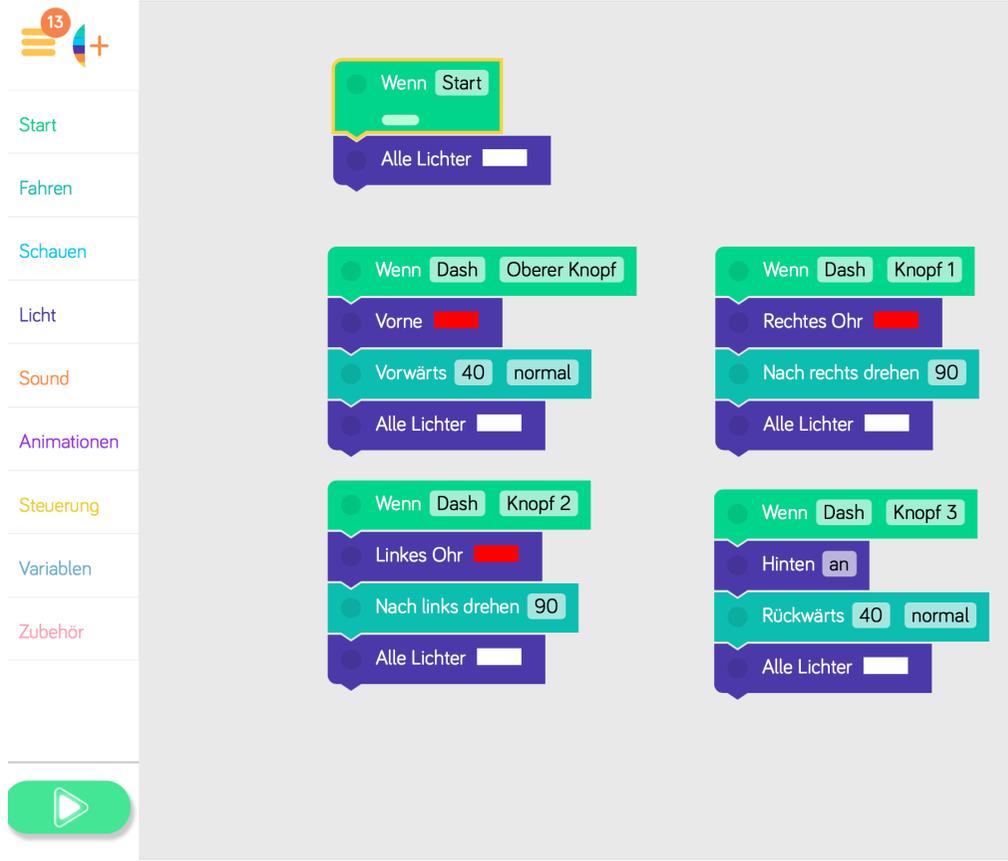
„Weg auf Knopfdruck“



Scratch script for "Weg auf Knopfdruck". The script starts with a "Start" event block. It then branches into four "Wenn Dash" (When Green Flag Clicked) blocks, each corresponding to a button: "Oberer Knopf", "Knopf 1", "Knopf 2", and "Knopf 3".

- Wenn Start** (Event)
- Wenn Dash Oberer Knopf** (Action): Vorwärts 40 normal
- Wenn Dash Knopf 1** (Action): Nach rechts drehen 90
- Wenn Dash Knopf 2** (Action): Nach links drehen 90
- Wenn Dash Knopf 3** (Action): Rückwärts 40 normal

Erweiterung 1:



Scratch script for "Erweiterung 1". The script starts with a "Start" event block. It then branches into four "Wenn Dash" (When Green Flag Clicked) blocks, each corresponding to a button: "Oberer Knopf", "Knopf 1", "Knopf 2", and "Knopf 3".

- Wenn Start** (Event): Alle Lichter []
- Wenn Dash Oberer Knopf** (Action): Vorne [red], Vorwärts 40 normal, Alle Lichter []
- Wenn Dash Knopf 1** (Action): Rechtes Ohr [red], Nach rechts drehen 90, Alle Lichter []
- Wenn Dash Knopf 2** (Action): Linkes Ohr [red], Nach links drehen 90, Alle Lichter []
- Wenn Dash Knopf 3** (Action): Hinten [an], Rückwärts 40 normal, Alle Lichter []

„Achtung Auto“

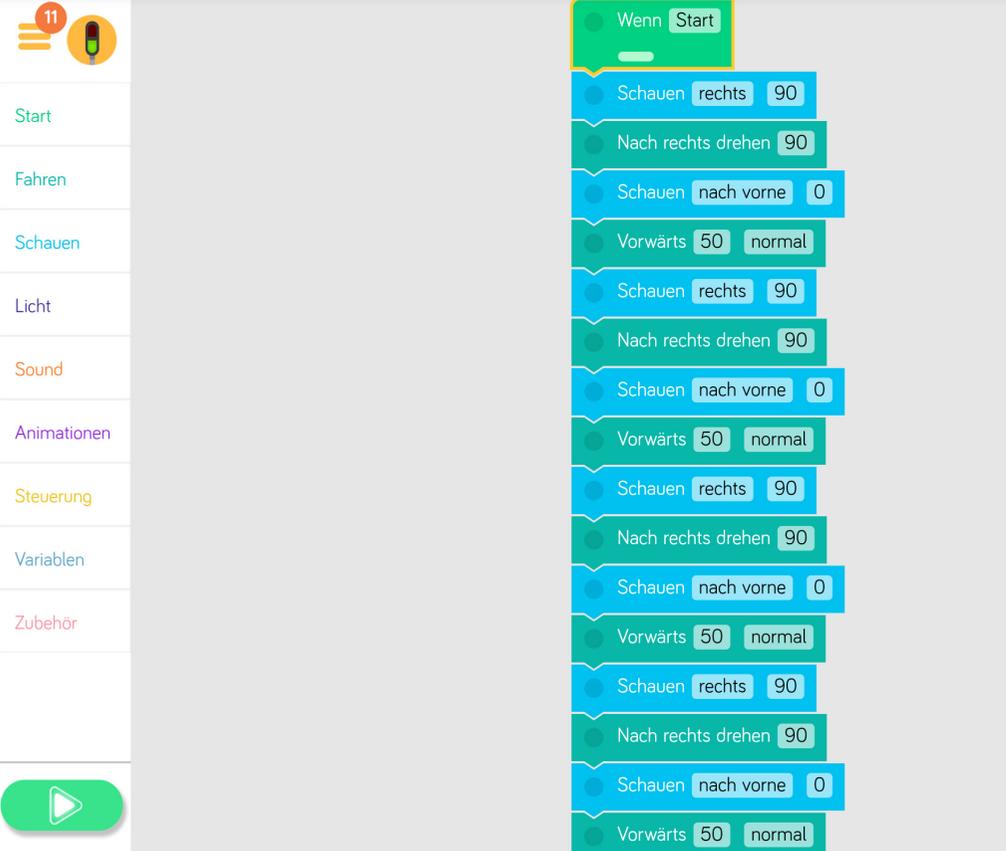
11 

- Start
- Fahren
- Schauen
- Licht
- Sound
- Animationen
- Steuerung
- Variablen
- Zubehör



```
Wenn Start
  Schauen links 90
  Schauen rechts 90
  Schauen links 90
  Schauen nach vorne 0
  Wenn Dash Hindernis vorne
    Alle Lichter
    Sag Dash Oh, oh!
  Sonst
    Vorwärts 80 schnell
```

„Fill the Gap“



11

Start

Fahren

Schauen

Licht

Sound

Animationen

Steuerung

Variablen

Zubehör

Wenn Start

Schauen rechts 90

Nach rechts drehen 90

Schauen nach vorne 0

Vorwärts 50 normal

Schauen rechts 90

Nach rechts drehen 90

Schauen nach vorne 0

Vorwärts 50 normal

Schauen rechts 90

Nach rechts drehen 90

Schauen nach vorne 0

Vorwärts 50 normal

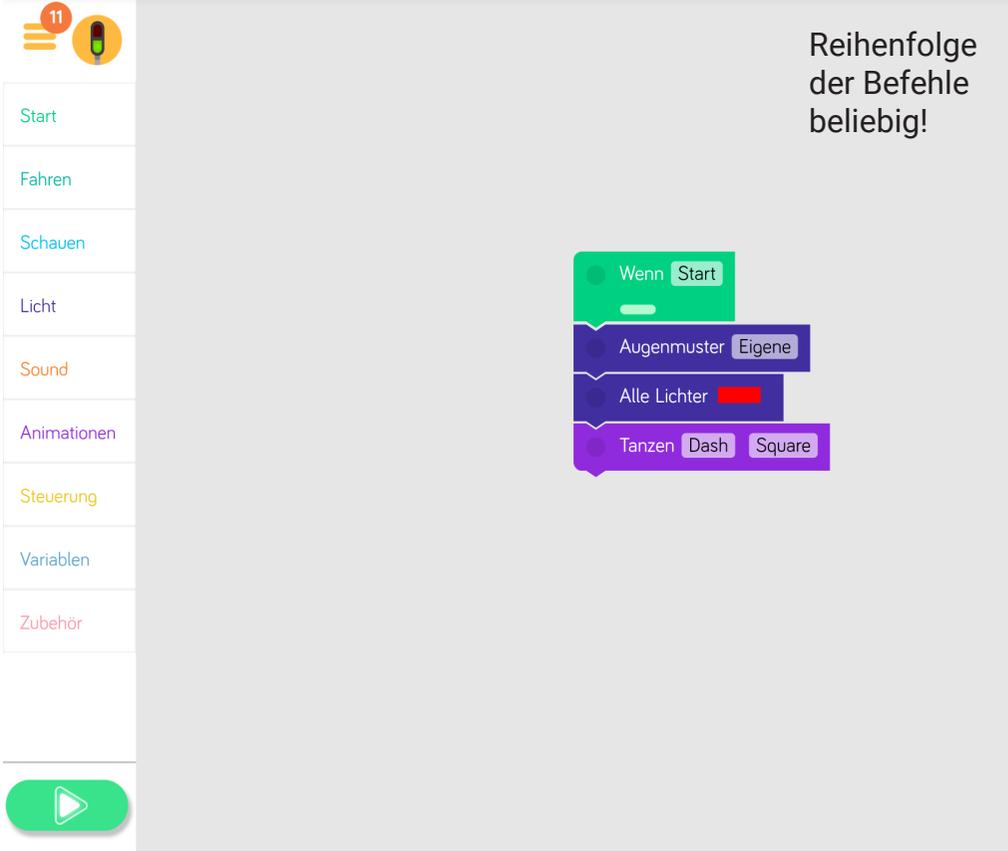
Schauen rechts 90

Nach rechts drehen 90

Schauen nach vorne 0

Vorwärts 50 normal

„Disco Dash“



11

Start

Fahren

Schauen

Licht

Sound

Animationen

Steuerung

Variablen

Zubehör

Reihenfolge der Befehle beliebig!

Wenn Start

Augenmuster Eigene

Alle Lichter

Tanzen Dash Square

Erweiterung 1:



Wenn Start

Wiederhole 4 Mal

Schauen rechts 90

Nach rechts drehen 90

Schauen nach vorne 0

Vorwärts 50 normal

Erweiterung 1:



Wenn Start

Endlos wiederholen

Augenmuster Eigene

Alle Lichter

Tanzen Dash Square

Erweiterung 2:



Wenn Start

Wiederholen bis Dash Oberer Knopf

Alle Lichter

Augenmuster Eigene

Tanzen Dash Square



Wenn Start

Endlos wiederholen

Tanzen Dash Zuversichtlich

Augenmuster Eigene

Alle Lichter

Wenn Dash Oberer Knopf

Räder stoppen