

## Task 1 - Let's get to know our Robbies!

In this task, we will be building a Robby, connecting it to the app, then controlling it remotely. We will also look at the installed sensors in more detail.

1. Build a model of your choice - Bob, Lilly, and Gerry are all looking forward to some exciting adventures with you!  
Use the building instructions to do so.



### 2. Connecting to the First Coding App

You can use the First Coding App to create a variety of programs for your robot to execute. To do so, the app needs to tell the robot what it should do, and when.

We could just connect the robot to the tablet and the app using a cable. However, that wouldn't be very practical, since we want to drive all around the classroom without having to run after the Robby all the time.

Work together to come up with a way to replace the cable:

---

---

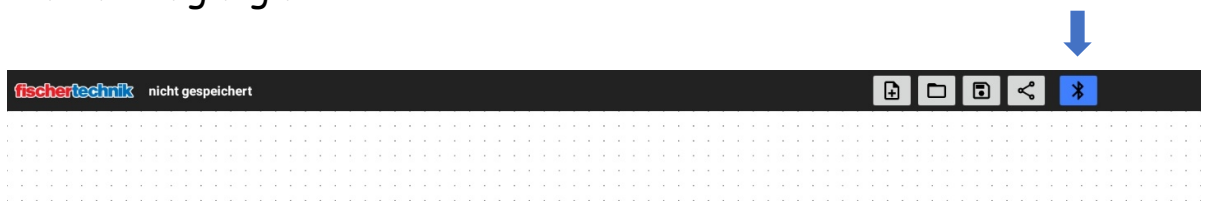
---

We will replace the cable with Bluetooth. But what is Bluetooth, exactly?

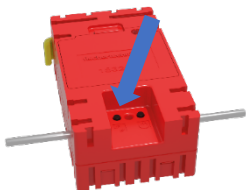
Bluetooth can be used to transmit data over short distances. This is done using radio signals. In our case, the program you create will be transmitted to the robot via Bluetooth - without a cable. The robot needs to be connected to the app to do so.

Now, let's look at how we can create this connection.

- Open the app on your smartphone or table
- After you have opened the app, click the Bluetooth symbol which is highlighted blue:



- Press the left button on the Robby until the LED starts flashing green:



- Then, select your robot

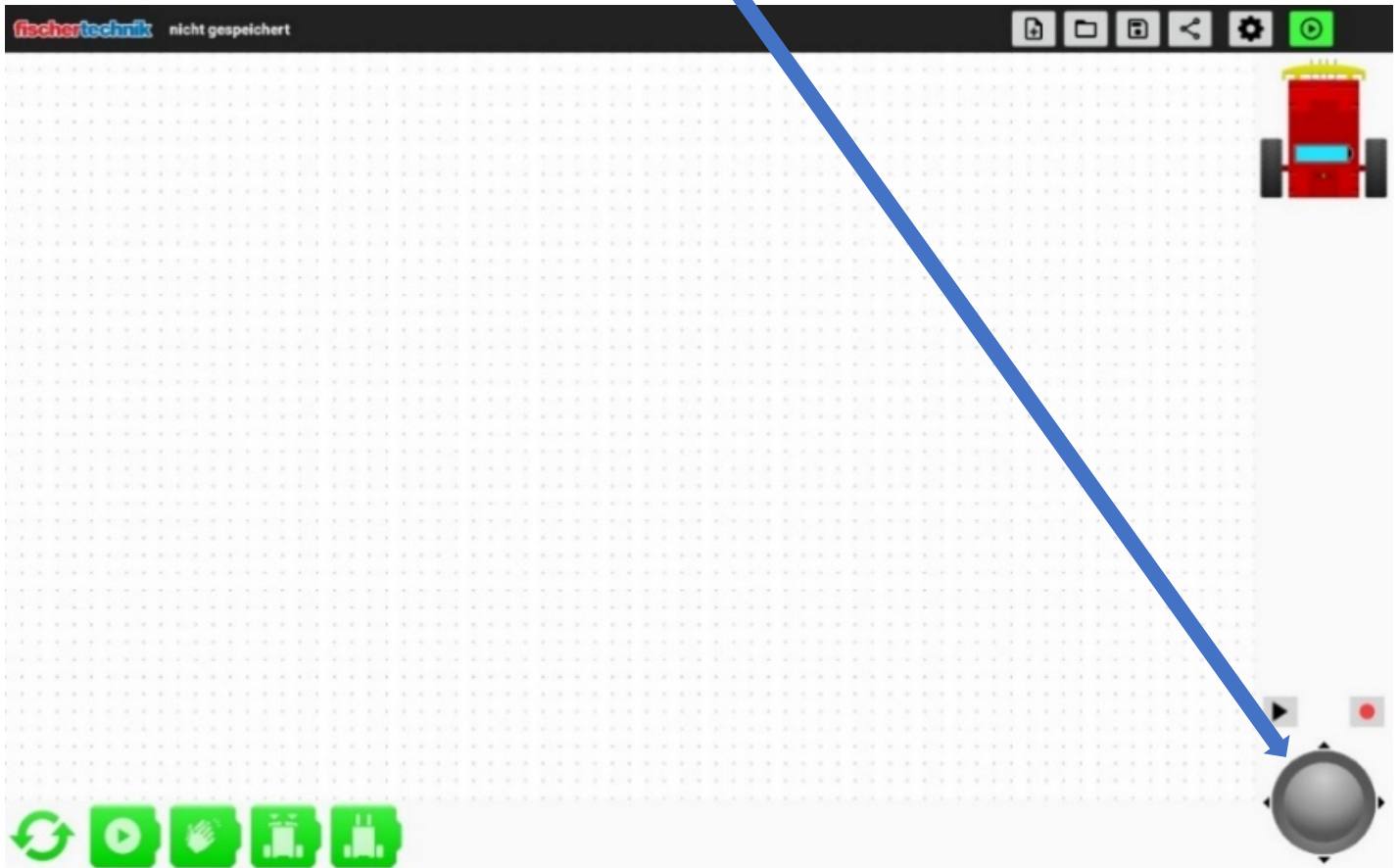
#### Mit Robby verbinden

Schalte Robby ein, indem du den rechten Knopf kurz drückst. Ein grünes Licht leuchtet auf. Drücke den Knopf links vom grünen Licht und halte ihn für 5 Sekunden gedrückt. Das grüne Licht beginnt zu blinken und Robby taucht in der Liste auf. Tippe auf den Eintrag in der Liste.

Robby [D8:74:F1:46:95:0D]

ABBRUCH

- After the connection is complete, the app interface should look like the image below.  
Use the track ball at the bottom right side of the display to steer your Robby



Now, let's go on an adventure!

Use your Robby to discover the classroom, run a race, and learn how to control it via the trackball.

### 3. Trying out sensors

Sensors are components the Robby uses to perceive its environment. It can use them to see, hear, or touch something. That means they are the eyes and ears of the robot.

A variety of sensors are installed in our Robby. The first one is the bumper.

- When you press it, there are two sensors inside the Robby that detect whether the bumper is pressed on the left, right or in the center.
- The image at the top right of the screen in the app shows you the side on which the bumper was pressed. Try it out. What do you observe?

---

---

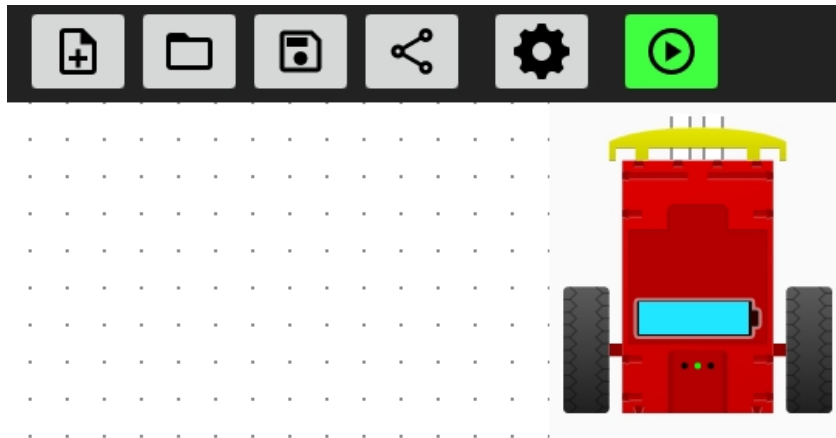
- There are also additional sensors on the bottom of the Robby. It can use these to differentiate between light and dark, so they function as its eyes. This allows it to follow a black track, for instance.

Do you have a piece of paper with a black track on it?  
Hold your Robby over it, and see what you observe in the app:

---

---

**Note:** The Battery symbol indicates how much power your Robby has left.:



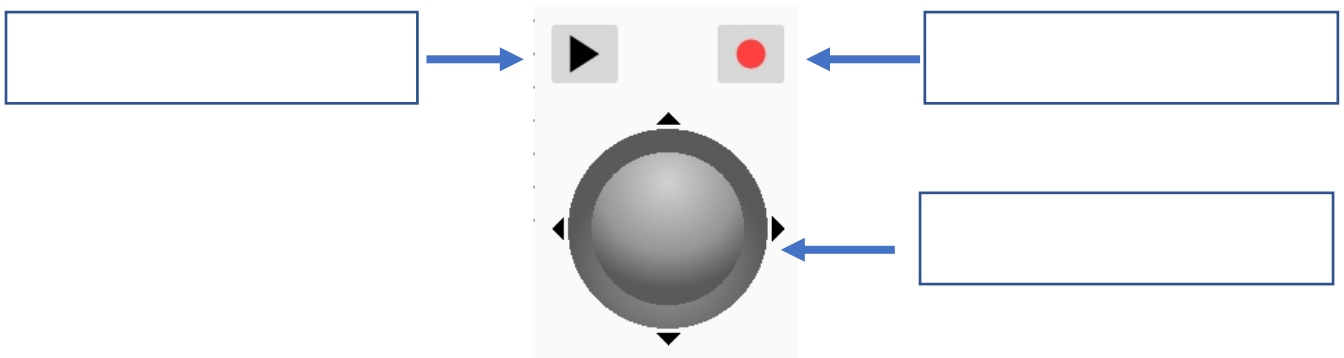
## Task 2: Recording and saving motions

We are going to teach the Robby a certain path, which it should remember and repeat.


We already know how to control the robot remotely using the trackball. Now, we will use the trackball to teach the Robby a path, which it will remember.


To do so, we will need the app functions shown below.

1. Work together to think about which function you will use, and write it in the box.



2. Now, let's start recording by clicking this symbol: 

The robot will remember everything you do until you click Stop .

Then you can press Play  to start traveling along the stored path.

Now, try out the function a few times.

3. Try to teach the Robby a certain path as precisely as you can. To do so, make markings on the floor, for instance using tape, or travel along the black line of the course. Then let your Robby travel the path alone.

This programming has been fun and easy too, right? In technical jargon, what you just did is called "**teaching-in**" a program. We taught the robot something exciting!

### Notes:

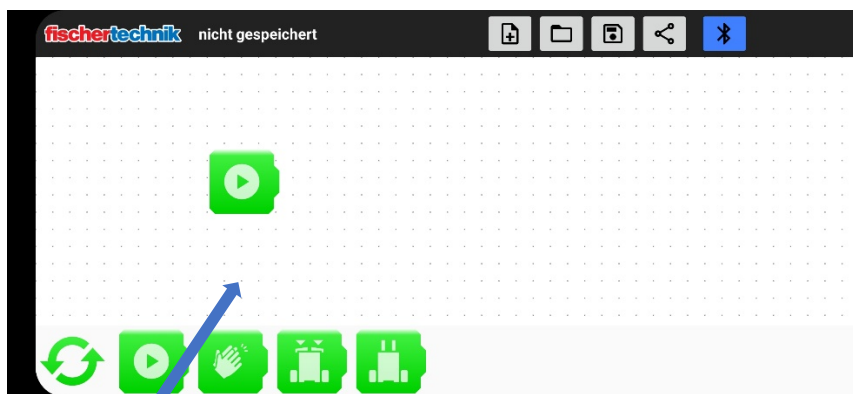
The program will be retained until the Robby is switched off, or the First Coding App is ended.

## Task 3: Handling the app, learning about the blocks and functions

Now, let's start by learning how to handle the app. Then we will learn about the programming blocks and functions.

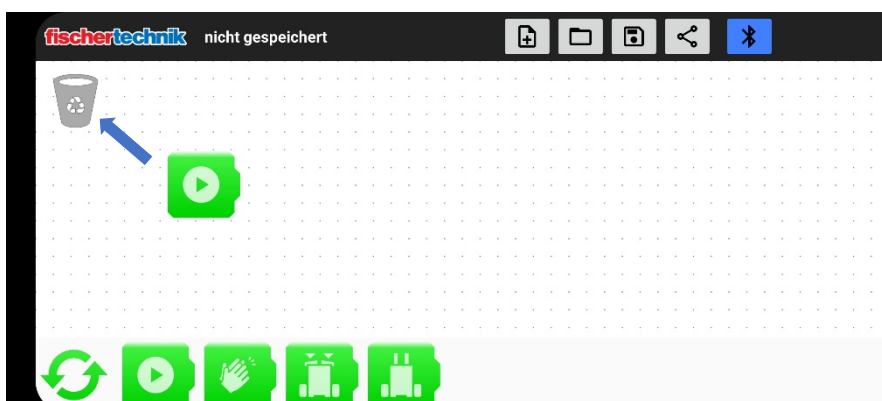
### 1. Using blocks

To use a programming block, use your fingers to drag it to the dotted area. This is the so-called programming interface.



### 2. Deleting blocks

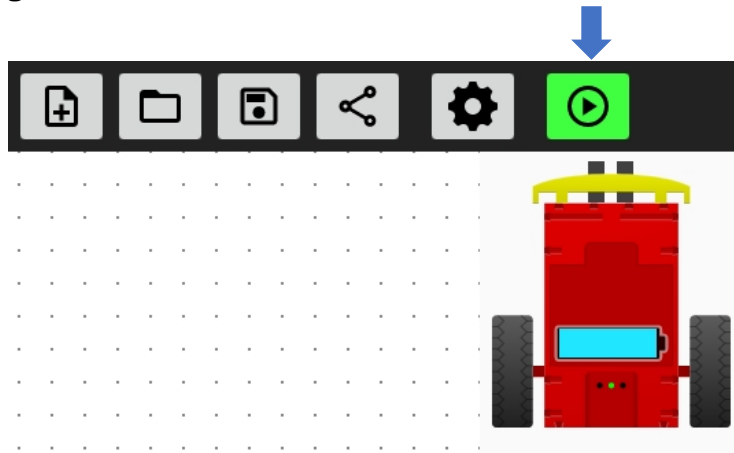
To delete a block, you must place your finger on the block and move it slightly. Keep your finger on the block. When you do so, a garbage can will appear at the top left. Now drag the block to the garbage can and drop it.





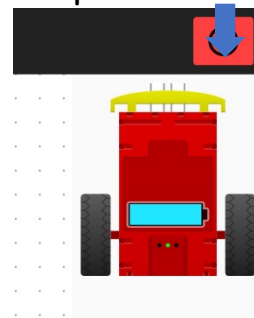
### 3. Starting a program

To start a program, click the "Play" button highlighted in green



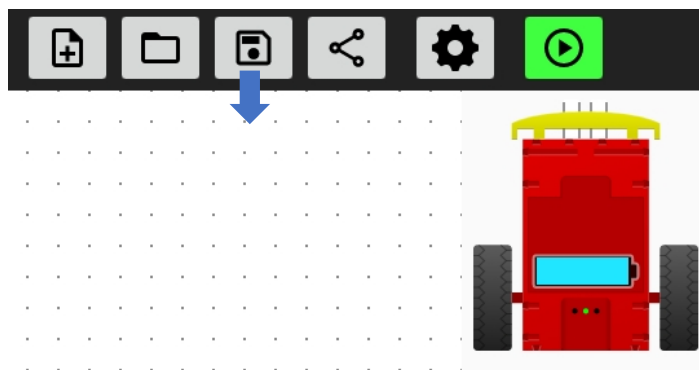
### 4. Stopping a program

To stop a program, click the "Stop" button highlighted in red




### 5. Saving programs

You can use the Save symbol at the top right to save the program on your smartphone or tablet and give it a name.



## 6. Programming block categories

Programming blocks come in five different colors. You can use the arrows  at the left of the blocks to change categories.

Each color stands for a different category. Work together to think about what functions the different blocks stand for, and enter your ideas beside the blocks.

### Green

The green blocks are event blocks. When the event occurs, the blocks attached to the event block are executed.



### Blue

The blue blocks are command blocks.





*Grey*



*Yellow*



*Red*



## Task 4: Programming using programming blocks and traveling along a path

Next, we will use programming blocks to make the Robby travel along a certain path

Experiment with the app, and try to solve the following tasks together.

If you need help, look at the command overview from task 3.

1. **Make the Robby travel 20 cm forward.**  
Drag the blocks to the programming interface, and see whether the commands you have selected fit.  
  
**Tip:** To set the distance, you will need to touch the right block.
2. Of course, we don't want to just drive straight ahead. Now make the Robby turn around **and** drive back 20 cm.
3. We want the robot to **wait briefly** before starting off again after turning. That means we need a **block for a delay**. Now add the delay to your program, and give the Robby a little breather between the commands.
4. Save the program.

## Task 5: Let's follow a track

You have a course with a black track in your materials.  
Now, we want our Robby to follow the track.

In task 1, we learned about the track sensor, as one of the sensors that can detect if something is light or dark. This sensor helps the Robby to follow the black track.

1. Use the app to create a program that will make the Robby follow a track.  
**Tip:** You will need two blocks to do so.
2. Save the program.

## Task 6: A sound is played when the bumper is touched

The goal is for your Robby to play a sound when the bumper is touched. First, start the app.

1. Where do you see bumpers in your everyday life - do you know what their purpose is?

---

---

2. Once the bumper is touched, the robot should play a sound. Create this program.  
Experiment with the app, and try to find a solution together.

What is your favorite sound? 😊

## Task 7: Detecting an obstacle on the track and turning around

The Robby should follow the track and turn around when it encounters an obstacle.

1. The Robby should follow a track.
2. Once it hits an obstacle with its bumper, our Robby should turn around and follow the track back.

**Tip:** We will need two different programs.

## Task 8: Following a track - detecting an obstacle - playing a sound

Now, we will add another function to the last task. Once the bumper hits an obstacle, the robot should play a sound.

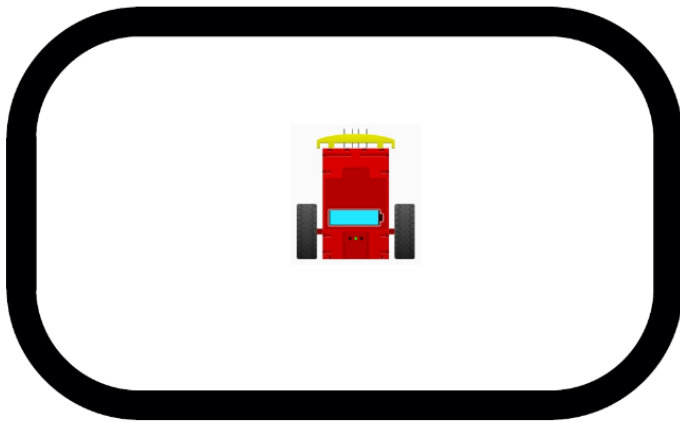
1. The Robby should follow a track, and play a sound when it detects an obstacle.  
Then the Robby should turn around and follow the track back

**Tip:** Add a little breather for your Robby after it turns around.



## Task 9: The Robby gets caged in

If you have a pet, like a rabbit, and you want to let it play outdoors, you will need to put up a fence all around it so it doesn't run away. We're going to do something similar with our Robby. It should be allowed to drive around on the course, but should always stay inside the black line



It's going to get a little tricky. Are you ready?

1. Experiment with the app, and try to create a program together. Use the overview from task 3 if you need help.

**Tip 1:** You will need a block that is assigned the right function when you touch it.

**Tip 2:** We will need the yellow block for the first time

2. Save the program for the next task.

## Task 10: Sound activation

For this task, we will need the program from task 9. We are going to add sound activation to it.

1. The Robby is now restricted by the black line: when it reaches the line, it should **first drive backwards a short distance**, then **turn around**. Work together to add this function to the program from task 9.
2. In addition, our Robby should **go to sleep** when we **clap our hands**. Going to sleep means that the program should end. Use the app to experiment, and add this function to your program.

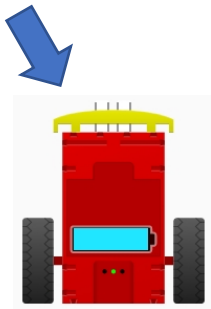
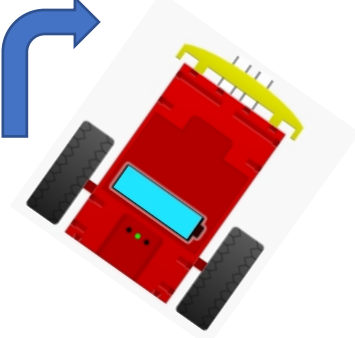
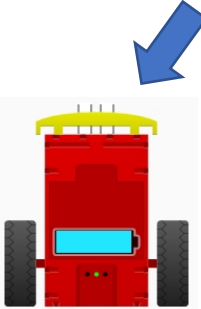
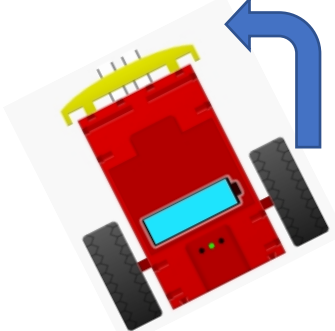
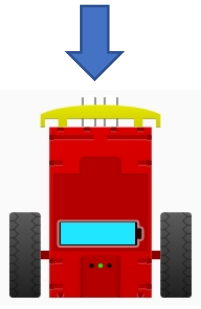
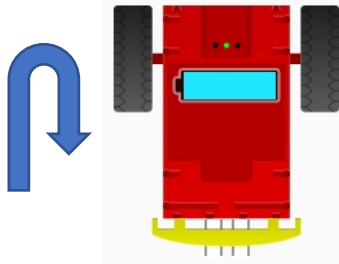
## Task 10: Sound activation

Now, it's time for your Robby to explore the classroom! It should start driving. When it comes into contact with an obstacle, it should stop briefly, drive backwards a short distance, turn and then continue driving in another direction.

Of course, we don't want to make things quite so easy on ourselves. It should turn in different directions, depending on which side of the bumper the obstacle is located.

Might sound tough at first, but you can do it!

- The following image shows what the Robby should do when the bumper is touched on different sides.

What side of the robot hits the obstacle?		How should it react?	
Obstacle on the left		Turn to the right	
Obstacle on the right		Turn to the left	
Obstacle in front		Turn around	

- Now, program the Robby so that it evades the obstacle to the side shown in the image.  
Use the app to experiment together, and add this function to your program.

**Tip:** If you set the turn to the left slightly smaller than the turn to the right, then the Robby will be able to get out of any corner.

This document provides preparatory information, tips, and possible solutions for tasks one to eleven. This document is designed as a guide to support instructors.

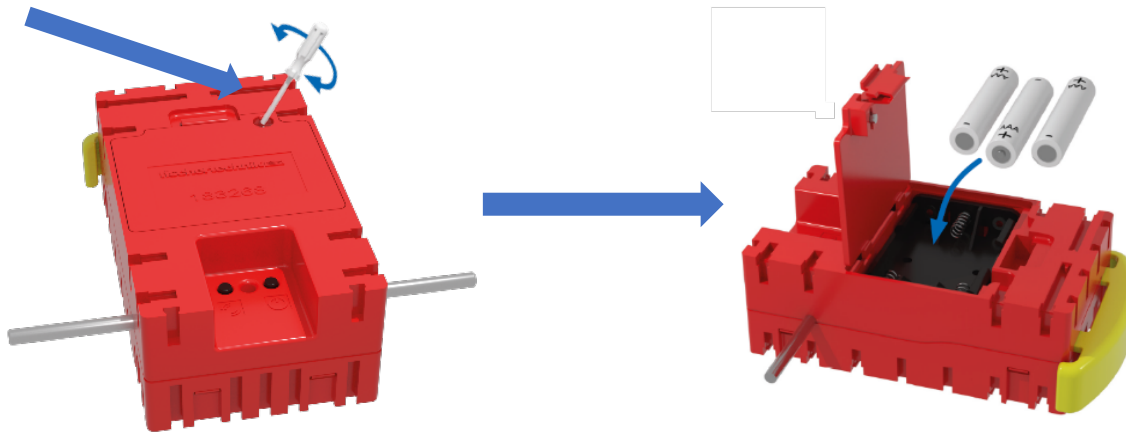
All documents are provided in editable format. This allows you to adapt them to meet your needs.

## Contents

Vorbereitung.....	2
Aufgabe 1.....	3
Aufgabe 2.....	4
Aufgabe 3.....	4
Aufgabe 4.....	6
Aufgabe 5.....	7
Aufgabe 6.....	7
Aufgabe 7.....	7
Aufgabe 8.....	8
Aufgabe 9.....	8
Aufgabe 10.....	9
Aufgabe 11.....	9

## Preparation

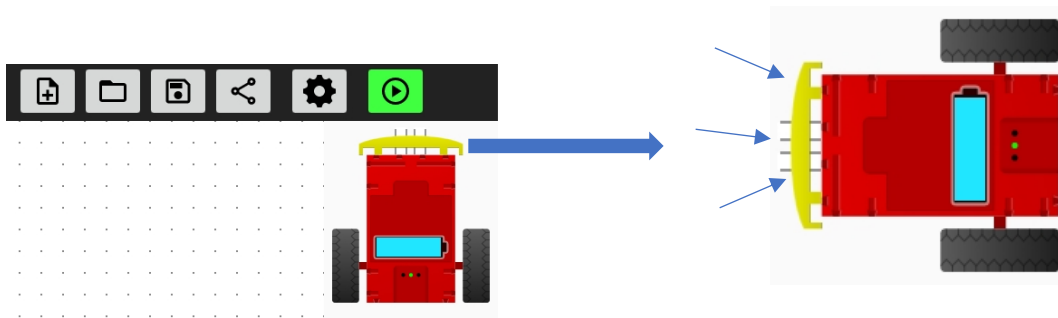
- ➔ Place the batteries in the robot chassis:



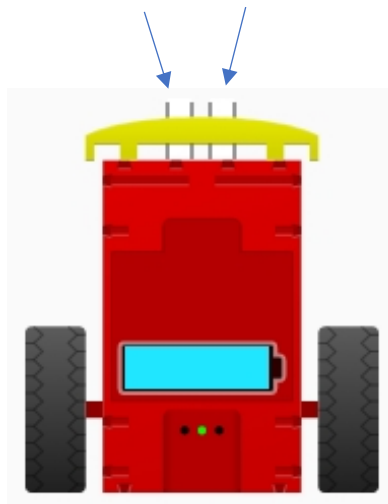
- ➔ Install the First Coding App on the mobile devices. Links to the Stores are provided on our homepage.
- ➔ When you open the app, you will see an option offering a brief introduction. You can use it in the classroom, or allow students to gain knowledge as they go through the tasks.

## Task 1

- Building the robot, using the building instructions
- Brainstorming in groups on transmission options
  - Transmission option used in the building set: Bluetooth
- Getting to know the app and connecting the robot following the description provided in the task
- Remote control via the trackball.
- Getting to know the sensors  
Display which side the bumper is touched on in the app:

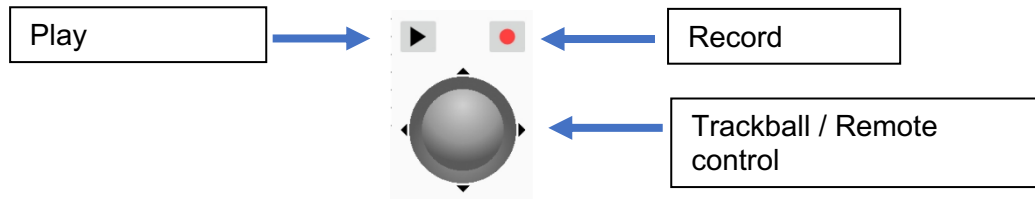


- Track sensor underneath the robot. If the robot is held over the black line of the course, it will detect the line and display it in the app.



## Task 2

- Labeling the functions of the trackball:



- Teach in a free or previously defined path. The path can be defined using tape, stopping points, or the course, for instance.
- The task provides an explanation of the teach-in function.
- Teach-in program will be retained until the app is ended or the robot is switched off.

## Task 3

The introductory video when starting the app can be used in this task if needed.

- **Block labels and functions**

### Green



Program start: The program begins here with the program start



Clap: When you clap ...



Bumper: When the bumper is pressed ...



Track: When the robot finds a track ...

### Blue



Sound output: A sound is played



Speed: Set the speed





Forwards/backwards:

Travel forwards / backwards until a condition is fulfilled:

- always,
- until bumper is touched,
- until line is found



Distance: travel the set distance forwards or backwards



Turn: turn to the right or left



Follow a track: follow a black track

## Grey



Wait block: Wait until the set time has expired

## Yellow



Loop: The blocks in the loop are repeated.  
The number of repetitions can be set.

## Red

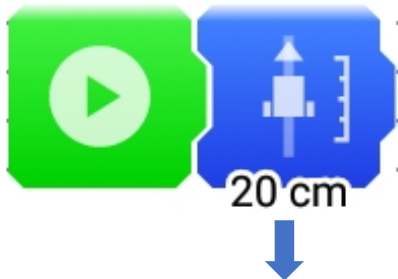


Stop: ends the program

## Task 4

- First chance to use the programming blocks students have learned about
- Sample solutions for the task steps:

### 1. Travel forward 20 cm



### 2. Build in the turn and the delays



## Task 5

- Programming to follow the black line:



## Task 6

- Everyday examples of bumpers: for instance on cars and trucks, front and rear, functions of a bumper include:
  - Protective barrier for passengers in case of a crash
  - Protective barrier for pedestrians and cyclists in case of a crash
  - Place to mount sensors (such as proximity sensors)
  - Improved aerodynamics
  - Design aspect
- Program to play the sound when the bumper is touched:



## Task 7

- Program to follow the track, detect an obstacle, turn around



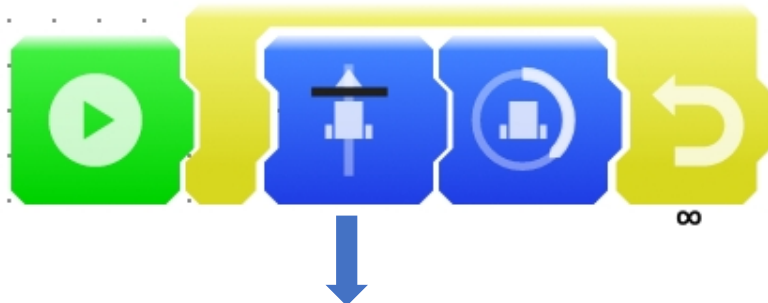
## Task 8

- Program to follow the track, detect an obstacle, play a sound, turn around




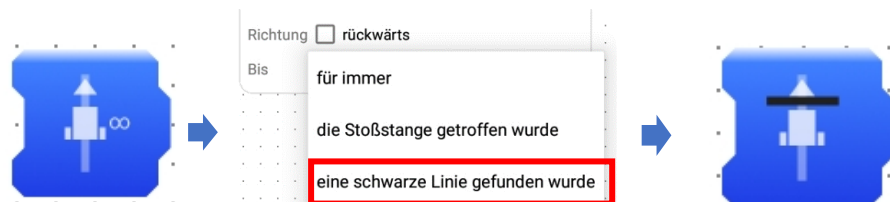
## Task 9

- Program to fence the robot inside the black line



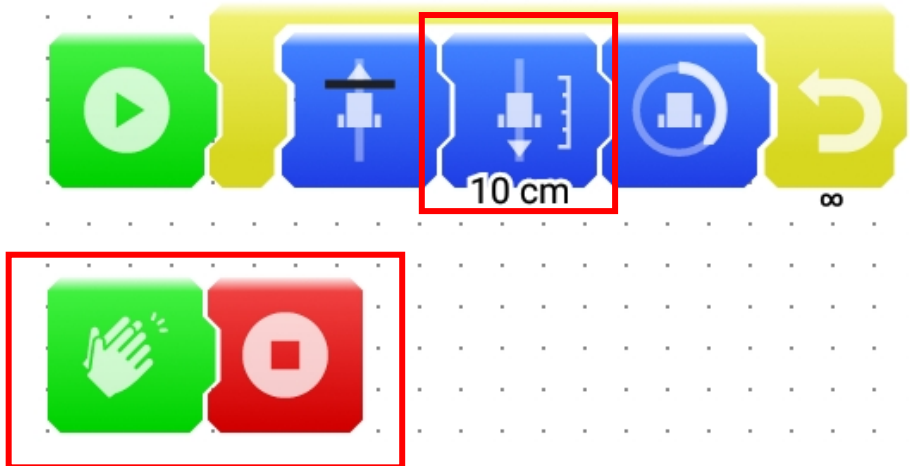
This block is created as follows:

1.  select
2. Click block
3. Select the option "a black line was found"
- 4.



## Task 10

- Programming sound activation  
The program from task 9 is expanded



## Task 11

- Programming wanderer

