Hi!

A STARTER GUIDE

Mind+ and Calliope mini

Special features and fast transmission of the code

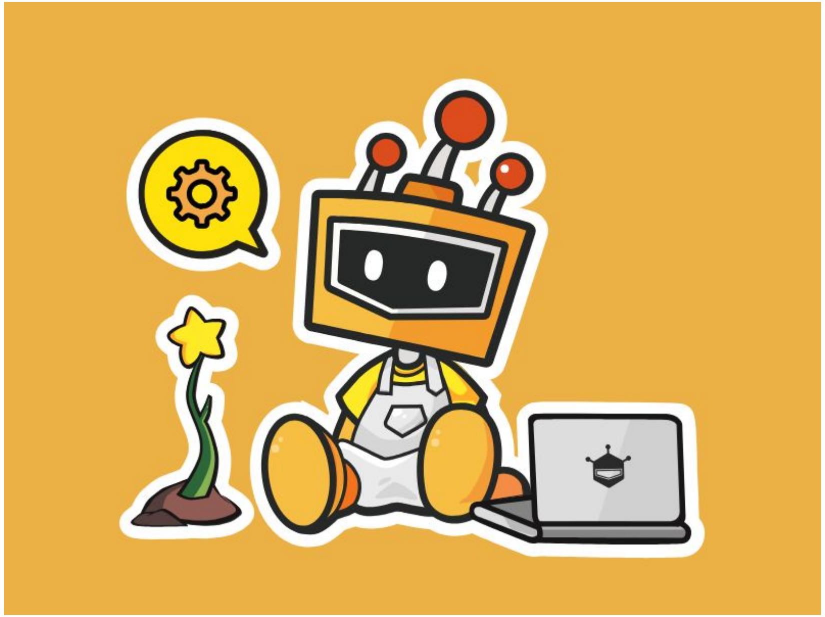
Similar to known Scratch environment

Offline mode (burn codes into board and run without a PC)
Online mode (run codes with board always connected to PC)

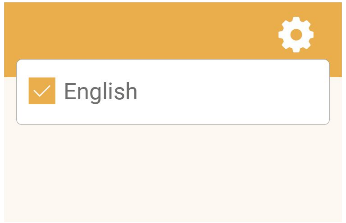
Interaction application on the screen and the Calliope mini

Such as (1) control games on the screen with the Calliope mini, (2) use Calliope mini as a sensor for the application on the screen, (3) Calliope mini as a machine on the screen control takes place.

Intuitive programming of the Calliope mini



Download and Install



☑ English

1 **Download**

Select your operating system and download the Mind+ Editor [BETA].

MACOS   WINDOWS

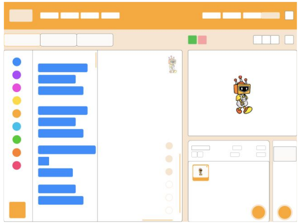
2 **Install**

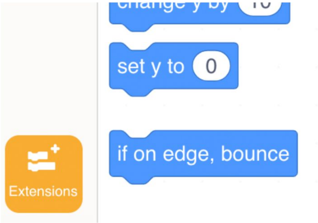
Now follow the instructions and install the Mind+ software.

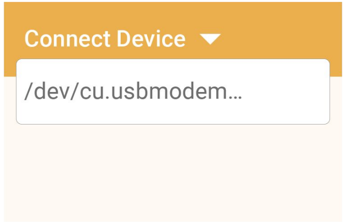
3 **Language**

Open the Mind+ Editor and select a language.

Download and Install







1 You need:

Calliope mini
USB cable
Computer
Mind+ software

2 Calliope mini Extensions

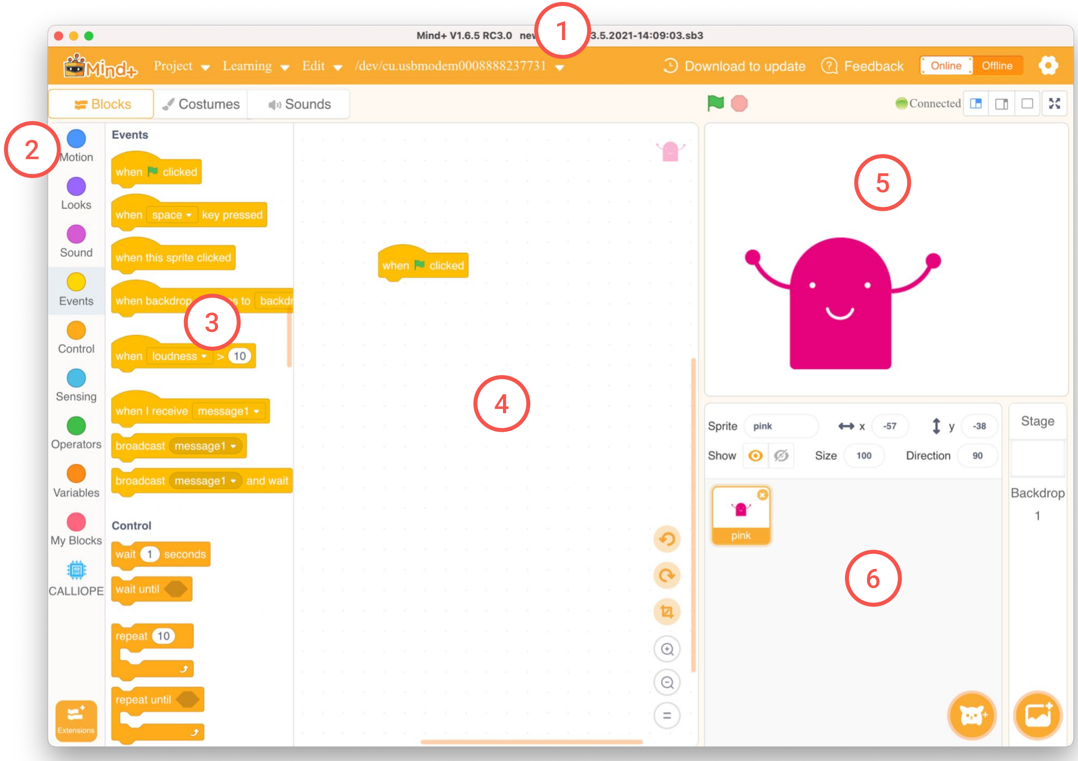
- Click on "Extensions" in the lower left corner of the preview
- Click on “Board”
- Then select “Calliope mini”

3 Connect

Now the Calliope mini can be connected to Mind+.
To do this, select "Connect Device".

The user interface

1. **Menu bar**
2. **Categories**
3. **Programming blocks**
4. **Programming area**
5. **Stage**
6. **Characters library**



Categories

A program code consists of commands that are executed in sequence. You put these commands together in the form of program blocks like a puzzle.

The program blocks are grouped into different categories/groups in the block library. They have an individual color depending on the category.

Blocks that control the Calliope mini can be found at the bottom after you selected the Calliope mini board in Extensions ▸ Board.

- ● Position, motion...
- ● Display, scaling, color...
- ● Sounds and music
- ● Events: Start, key...
- ● Control: waits, branches, loops...
- ● System functions: Time, cursor position...
- ● Logic, comparison, mathematical functions, chance...
- ● Variables
- ● Own functions
- ● Blocks for the Calliope mini

60 MIN

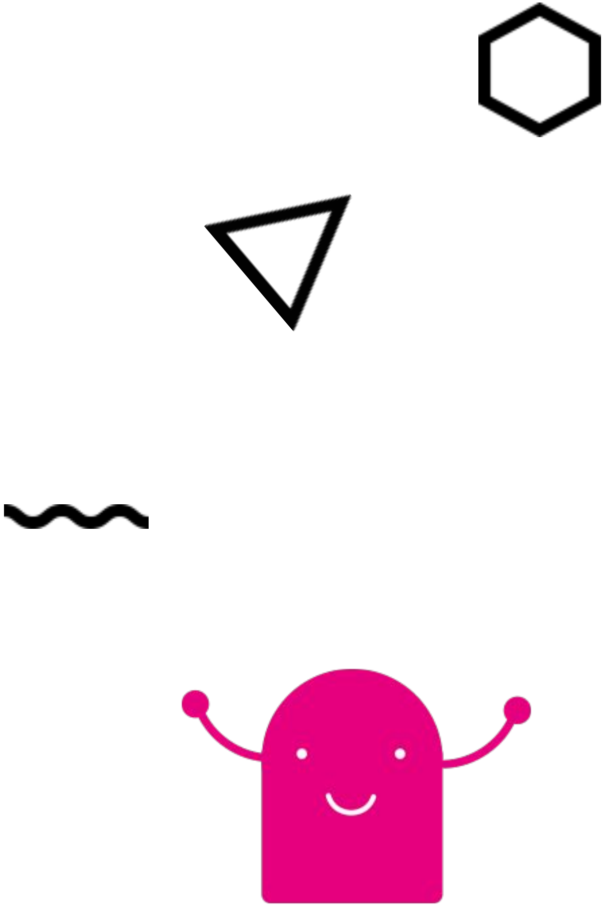
LEVEL
CHALLENGE

Catch

Now you code your first game!

The mini character catches 5 of the objects falling from the sky as fast as possible. You can move it to the left and right using the A and B buttons on your Calliope mini.

Have fun!



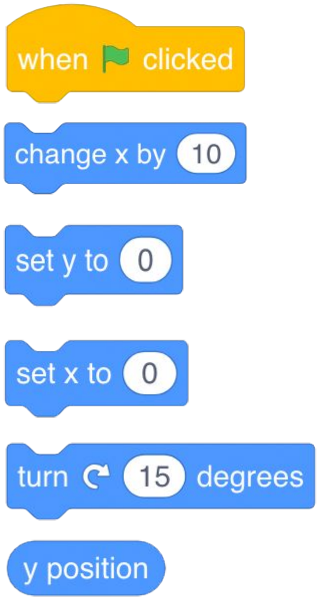
CALLIOPE.CC

Code blocks

You can find the blocks you need in the following categories:

Events

Motion

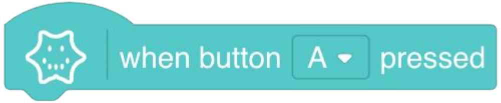


when 🏳 clicked	To start the script, the flag has to be clicked.
change x by (10)	Changes the sprite's X position by the amount.
set y to (0)	Sets the sprite's X position to the amount.
set x to (0)	Sets the sprite's Y position to the amount.
turn ↻ (15) degrees	Turns the sprite (clockwise) by the specified amount.
(y position)	Displays the Y position of the sprite.

Code blocks

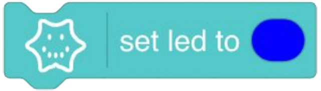
You can find the blocks you need in the following categories:

Calliope



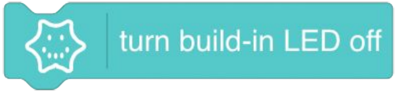
when button A ▾ pressed

Event - when Calliope mini Button A is pressed.



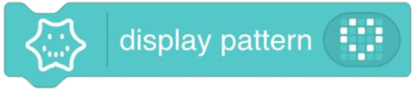
set led to ●

Turns on the Calliope mini RGB LED in a specific color.



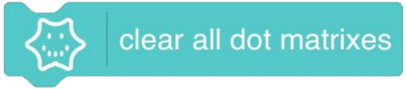
turn build-in LED off

Turns off the Calliope mini RGB LED.



display pattern

A symbol appears on the Calliope mini LED matrix.



clear all dot matrixes

Clears the Calliope mini LED matrix.

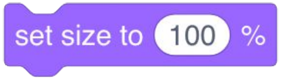
Code blocks

You can find the blocks you need in the following categories:

Looks

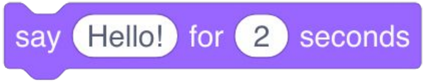
Variables

Sensing



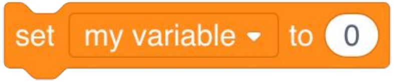
set size to 100 %

Sets the sprite's size to the specified amount.



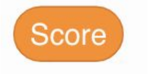
say Hello! for 2 seconds

A speech bubble appears over the sprite and stays for the specified amount of time.



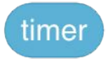
set my variable ▾ to 0

Defines the amount of the specified variable.



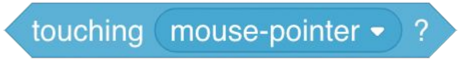
Score

The variable's value.



timer

How much time has passed since the program was opened or the timer was reset.



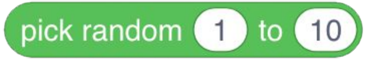
touching mouse-pointer ▾ ?

The condition to check if the sprite is touching the mouse-pointer or another sprite.

Code blocks

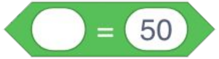
You can find the blocks you need in the following categories:

Operators



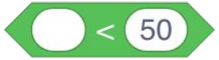
pick random 1 to 10

Selects a random number between the two specified limits.



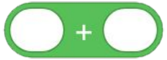
= 50

The condition to check if two values are equal.



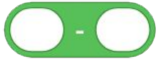
< 50

The condition to check if a value is smaller than the other.



+

The value of an addition.



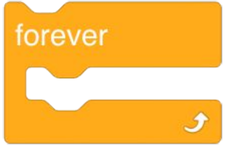
-

The value of a subtraction.

Code blocks

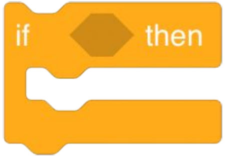
You can find the blocks you need in the following categories:

Control



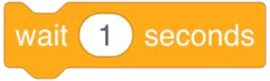
forever

With “forever” you create a loop, that will never end unless the stop sign is pressed.



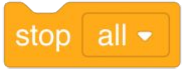
if then

This is a function to check a condition. That means, “if” the condition is true, the blocks inside it will be activated.



wait 1 seconds

Pauses the script for a specified period of time.



stop all

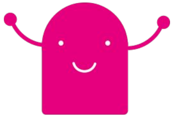
Stops the scripts chosen through the drop-down menu.

Catch

If you want to develop a game, it is advisable to divide the game mechanics into individual packages. You work through these individual packages in small steps. It is important that you always check your code and test the program.

A Control your character

The A and B keys move the character on the X axis



B Control obstacles

Continuous movement on the Y-axis (from top to bottom)

If they fall below a value, they start again from the top

If they touch the character, the score is increased by 1

C Gameplay

Score is displayed and queried

Playing time is queried continuously

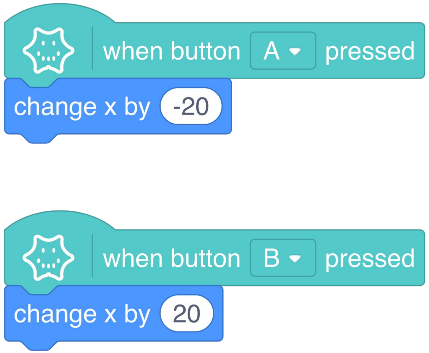
Catch

A Control mini character

Select the mini character in the character library and control the mini character with the **buttons A and B** of your Calliope mini.

Assign a direction change on the X-axis to both key events.

The larger the change value, the larger the steps.



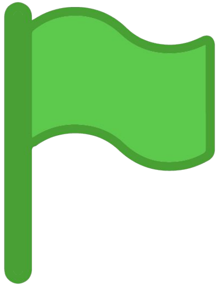
```
when button A pressed
change x by -20

when button B pressed
change x by 20
```

Debugging

Test your program.

Does your mini character move from right to left when you press the buttons on your Calliope mini?



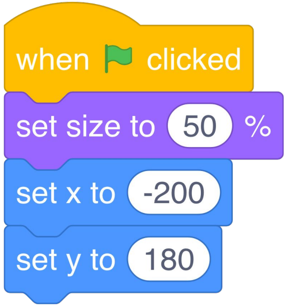
Catch

B Position obstacle on the stage

Select an obstacle in the character library and determine the following properties:

- Scaling / Size
- Position X-axis
- Position Y-axis

Start with a “when 🚩 clicked” event and use the “set size to”, “set x to” and “set y to” blocks for the properties settings.



```
when 🚩 clicked
set size to (50) %
set x to (-200)
set y to (180)
```

Catch

B Move obstacle from top to bottom

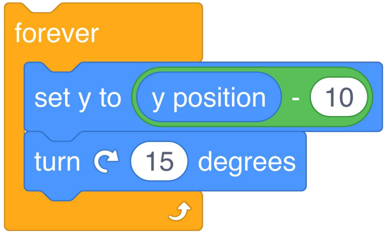
For a continuous movement you need an infinity loop.

1. Add a "Forever" block.

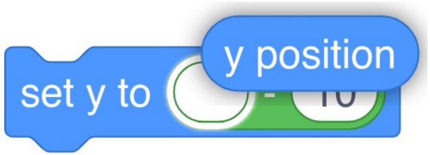
   Decrease the position on the Y-axis in steps of 10 with the block "set y to" in which you query the current "Y-position".

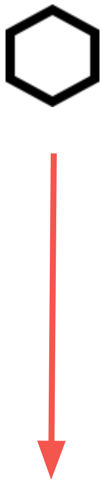
   Rotate the obstacle in 15 degree steps by adding the block "turn".

   Test your program.



```
forever
    set y to (y position) - (10)
    turn ↻ (15) degrees
```





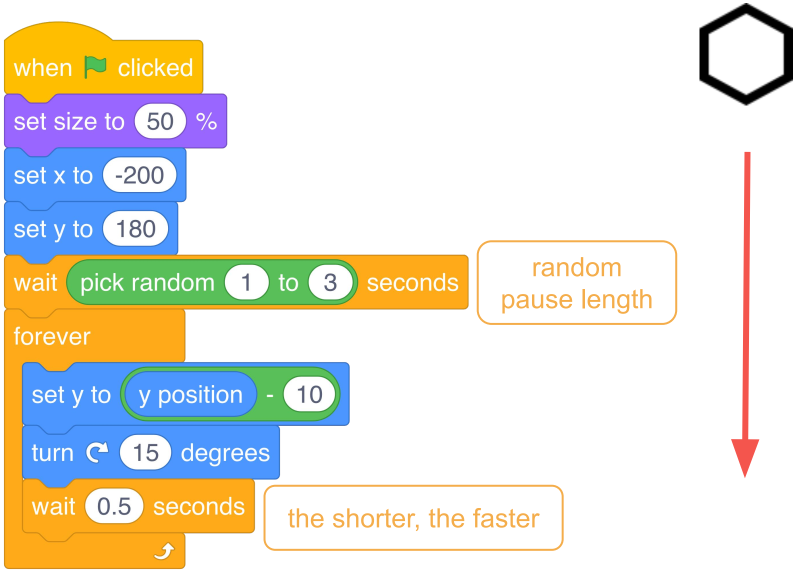
Catch

B Define speed

Add a pause at the end of the infinity loop to control the speed of the obstacles.

Add a "wait" block to do this.

HINT! To make the obstacles more unpredictable you can add a short random pause in the beginning.



```
when ⚑ clicked
set size to (50) %
set x to (-200)
set y to (180)
wait (pick random (1) to (3)) seconds
forever
    set y to ((y position) - (10))
    turn ↻ (15) degrees
    wait (0.5) seconds
```

random pause length

the shorter, the faster

Catch

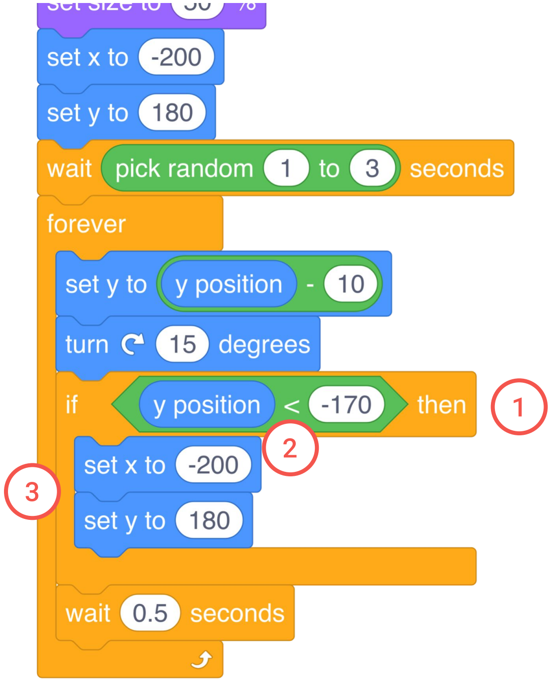
B **Has the obstacle reached the bottom of the stage?**

The position on the Y-axis shows you if the obstacle has reached the bottom of the stage.

1. Add an "if...then" branch to the persistent loop

2. Define using the condition by comparing the current Y position of the obstacle with the threshold. Use the block "less than (<)" for this purpose.

3. If this condition is met, set the obstacle back to the start position with the blocks "set y to" and "set x to".



```
set size to 50 %
set x to -200
set y to 180
wait pick random 1 to 3 seconds
forever
    set y to y position - 10
    turn ↻ 15 degrees
    if y position < -170 then
        set x to -200
        set y to 180
    wait 0.5 seconds
```

Debugging

Test your program.

Is the lower stage border as desired?

Does the obstacle reappear at the top of the stage?

Does the obstacle restart?

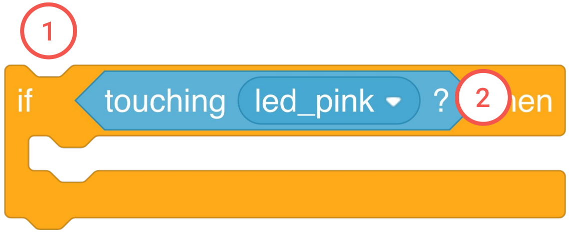
Catch

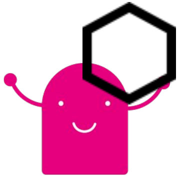
B **Did the obstacle touch the mini character?**

The goal of the game is to catch as many obstacles as possible. In the next step you check if the obstacle and the mini character are touching.

1 To do this, add another "if...then" branch to the infinity loop.

2 Define the condition by using the "touching" block to query whether your obstacle is touching the mini character.



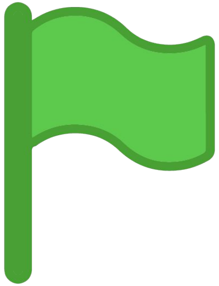


Debugging

Test your program.

Is the touch detected?

You can also have a sound output for control purposes.



Catch

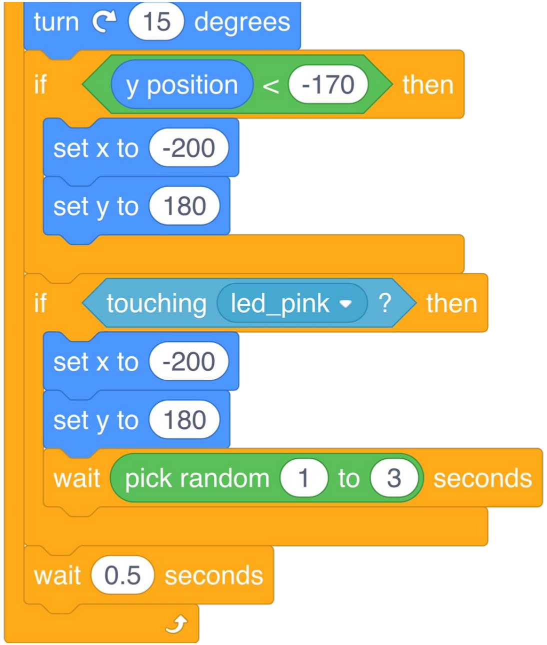
B Back to start?

Set the obstacle back to the start position after a collision and add a small delay.

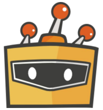
Set the following properties:

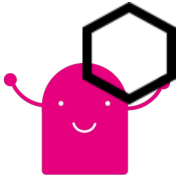
- X-axis position
- Y-axis position
- Pause

If the obstacle and the mini character touch you get a point. You will learn how to do this in the next step.



```
turn ↻ 15 degrees
if y position < -170 then
    set x to -200
    set y to 180
if touching led_pink ? then
    set x to -200
    set y to 180
    wait pick random 1 to 3 seconds
wait 0.5 seconds
```





Catch

C **Define start of game**

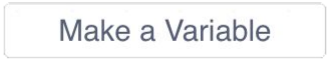
1 In order to store and display the score, a variable is needed.

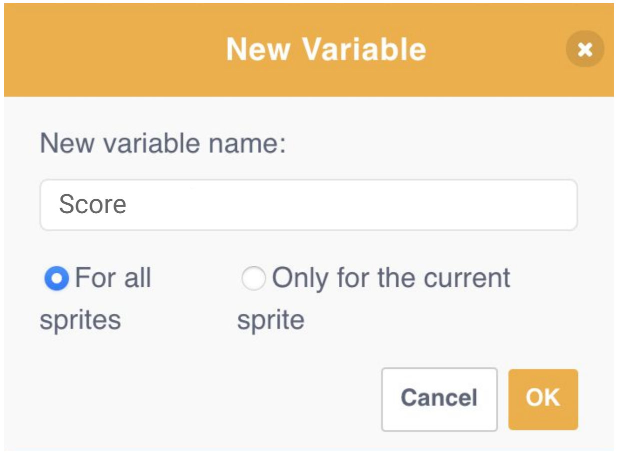
2 Create a new variable by clicking on "make a variable" and assigning the name "Score".

3 **IMPORTANT**! Define the variable for all characters.

Display the variable on the stage.

1





New Variable

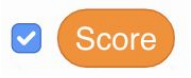
New variable name:

Score

2

● For all sprites ○ Only for the current sprite

3


☑ Score

Catch

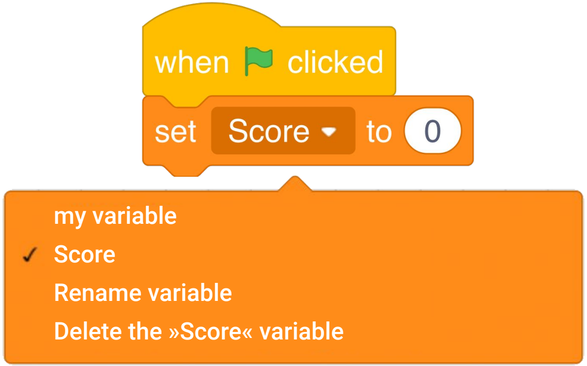
C Set score to 0

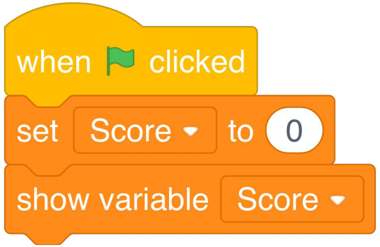
1. Add a “when 🚩 clicked” block

2. Add the block "set my variable to" and select the variable "Score" in the pulldown menu

3. Set the value of the variable "score" to 0 at the beginning of the game

With the block "show variable" you can show the score on the stage





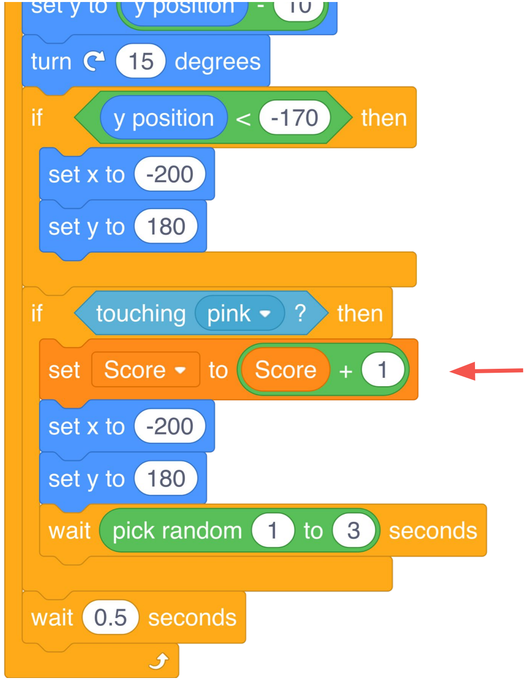
Catch

B **Keep score**

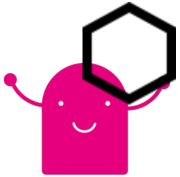
Now that you've created the score variable, go back to your obstacle and add one more command to your mini character´s code.

The score variable will be incremented by 1 as soon as the obstacle and the mini character touch each other.

To do this, insert a block "set variable" and increase the current score by 1.



```
set y to (y position - 10)
turn ↻ 15 degrees
if <(y position) < -170> then
    set x to -200
    set y to 180
if <touching pink ▾ ?> then
    set Score ▾ to (Score + 1)
    set x to -200
    set y to 180
    wait (pick random 1 to 3) seconds
wait 0.5 seconds
```



Catch

C Query score

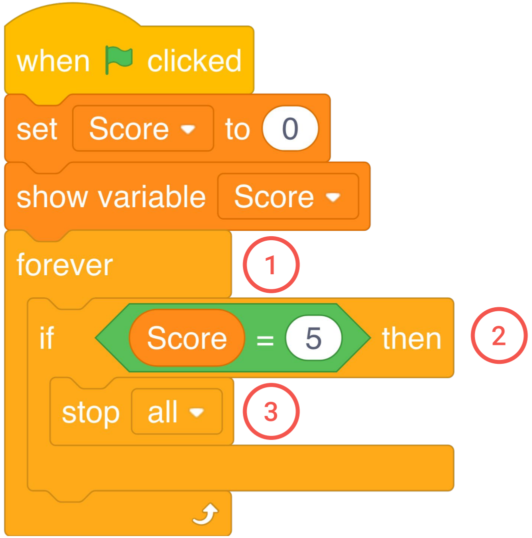
Back to the mini character...

1 Add an infinity loop to query the score

2 In a branch you ask if a score of 5 has been reached

Add a "if... then" branch and define the condition

3 If the condition is met, the game is won and over.



```
when ⚑ clicked
set Score ▾ to 0
show variable Score ▾
forever
    if Score = 5 then
        stop all ▾
```

Debugging

Test your program.

Is the score displayed on the stage?

Is the score 0 at the beginning?

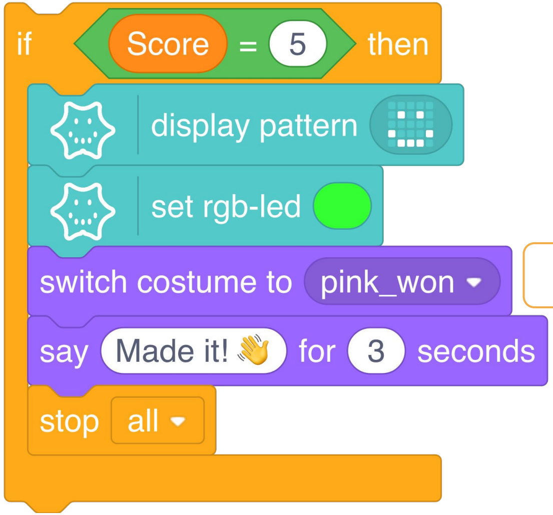
Does the score count up as soon as an obstacle is caught?

Catch

C Design game end

When the game is won the mini character raises its arms in joy and shouts "Made it!". A smiley face appears on your Calliope mini and the RGB LED lights up green.

1. Add the corresponding blocks from the "Looks" category.
2. Add the corresponding blocks from the "Calliope" category



```
if Score = 5 then
    display pattern
    set rgb-led 🟩
    switch costume to pink_won ▾
    say Made it! 👋 for 3 seconds
    stop all ▾
```

Arms up

Speech bubble

Debugging

Test your program and try to win the game

Is the RGB LED glowing green?

Is a smiley displayed on your Calliope mini?

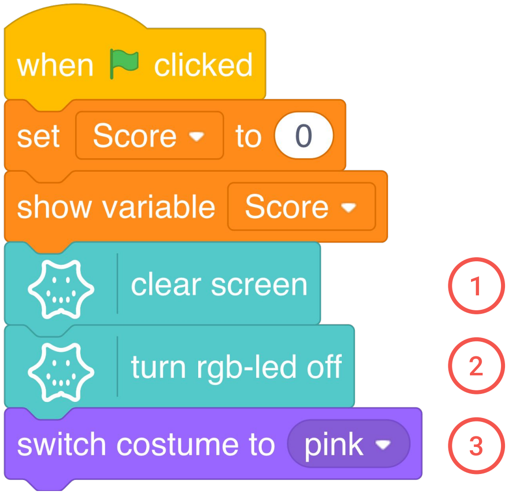
Is the appropriate text displayed in the speech bubble?

Catch

C Reset game at start

When the game is started, everything should be set to RESET.

1. The LED matrix on the Calliope mini is deleted
2. The RGB LED is switched off
3. The mini character is set to the default image



```
when 🏁 clicked
set Score ▾ to 0
show variable Score ▾
clear screen
turn rgb-led off
switch costume to pink ▾
```

In the next step, look at how you can make the game even harder with a limited playtime.

Debugging

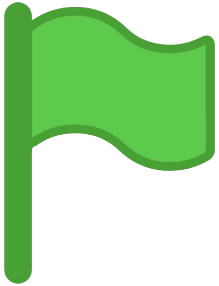
Test your program.

Is everything reset when you press the start flag?

Is the LED matrix cleared?

Is the RGB LED off?

Is the mini character in the start costume?

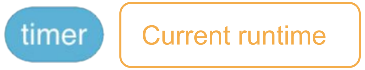


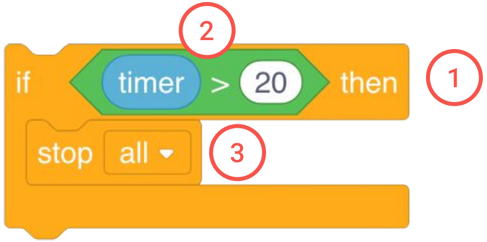
Catch

C Game against time

Increase the difficulty by limiting the game time. Use the code block "timer" to query the game time and define the end of the game when the game time is over and the game is lost.

1. Add a "if... then" branch to this and define the condition
2. Compare if a game time of 20 seconds is exceeded
3. If the condition is met, the game is lost and over





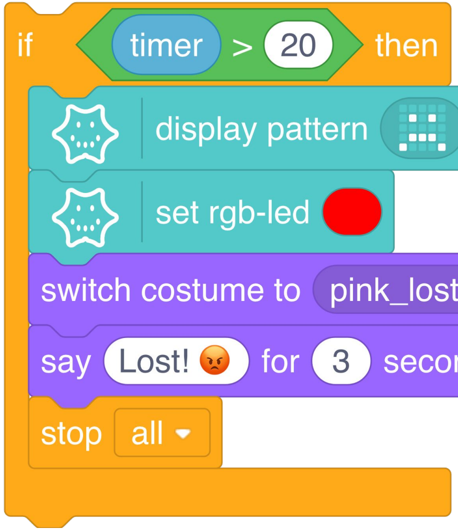
```
if <(timer) > (20)> then
  stop [all ▾]
```

Catch

C Design game end

If the game is lost the mini character hangs its arms in exhaustion and shouts "Lost!”. A sad smiley appears on your Calliope mini and the RGB LED lights up red.

1. Add the appropriate blocks from the “Looks” category
2. Add the appropriate blocks from the “Calliope” category



```
if timer > 20 then
    display pattern
    set rgb-led
    switch costume to pink_lost
    say Lost! 😡 for 3 seconds
    stop all
```

Arms down

Speech bubble

Debugging

Test your program and let the time go by.

Does the RGB LED light up red OR green?

Is the matching face displayed on your Calliope mini?

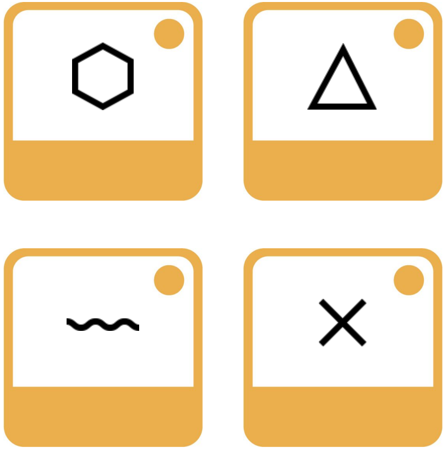
Is the matching text displayed in the speech bubble?

Catch

C More obstacles

You can add as many obstacles as you want. To do this, follow the steps below:

1. Load another obstacle into the character library.
2. Click the first block of the program code of your first obstacle and press ctrl + c / cmd+c (copy)
3. Select your new obstacle and paste the copied blocks with ctrl + v / cmd+v



CALLIOPE

If you have any questions, please do not hesitate to contact us.

You can contact us at +49 (0)30 4849 2030

or by e-mail: info@calliope.cc

CALLIOPE.CC