



愛動智教育系統

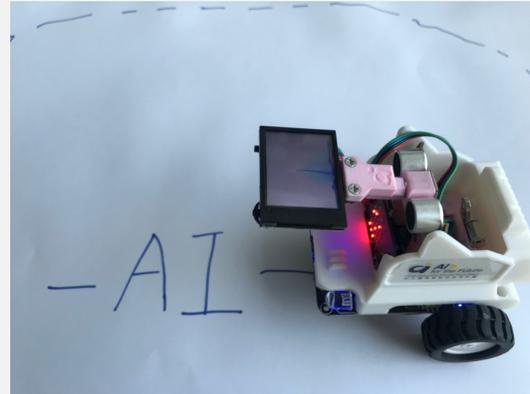
CUHK iCar Experiment Manual
Experiment 3: Moral Dilemma Experiment

Write Your Own Code
(Evade Kids)

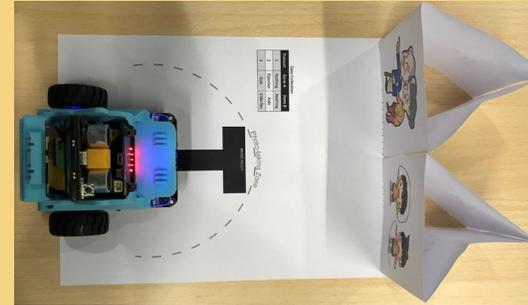
CUHK iCar



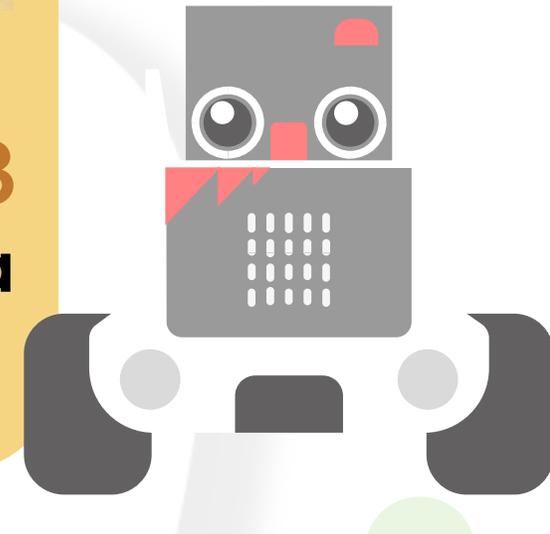
Experiment 1
Face Following



Experiment 2
Line Tracking



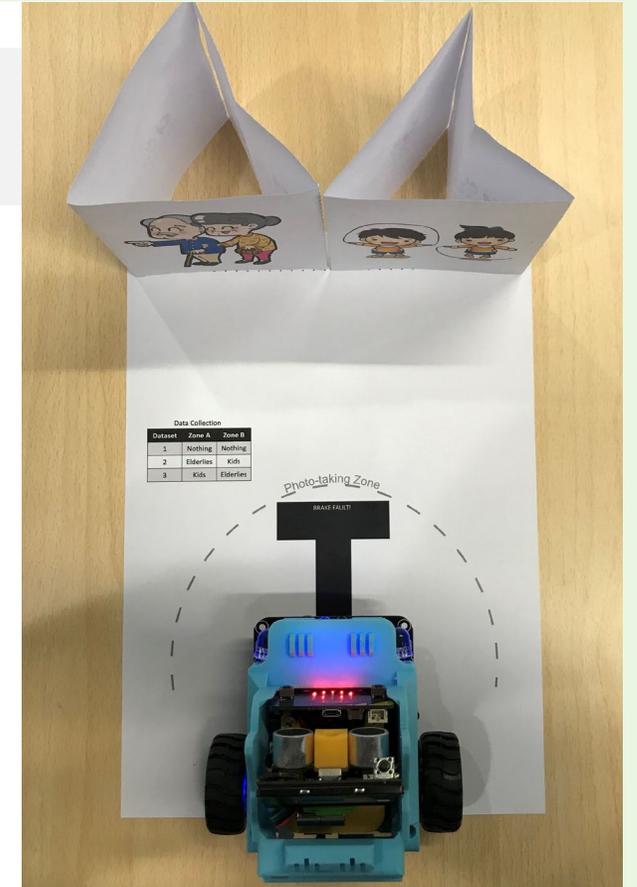
Experiment 3
Moral Dilemma



Moral Dilemma Experiment

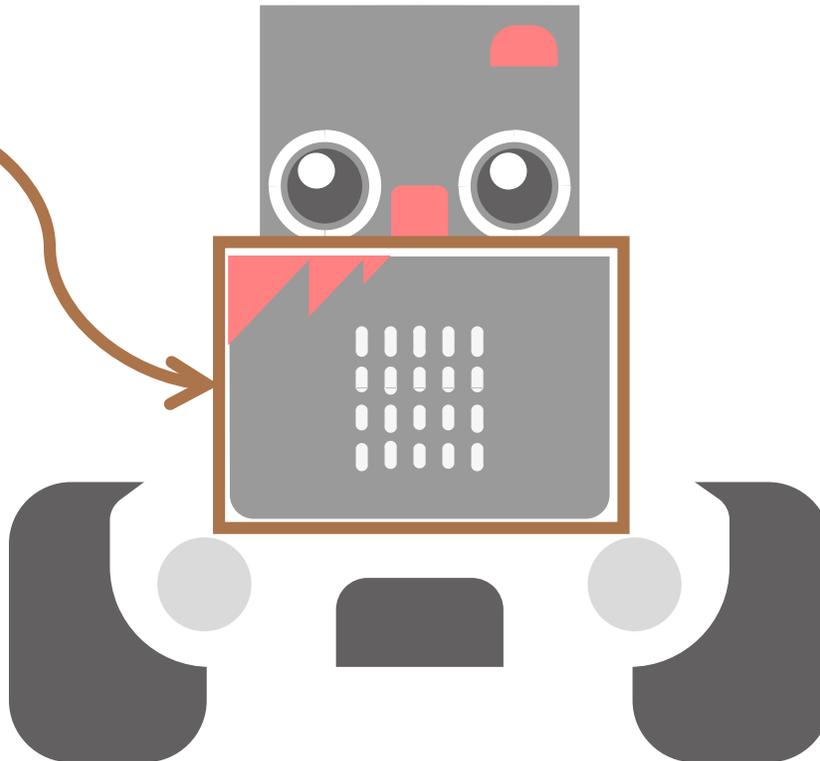
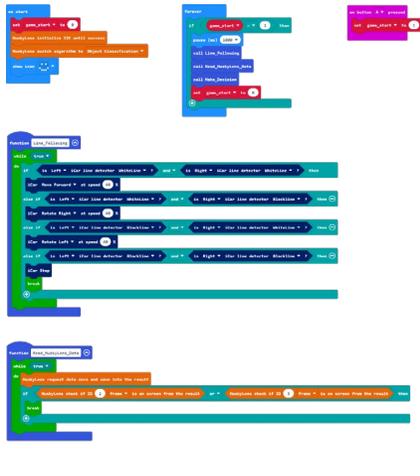
Introduction Of The Experiment

This experiment simulates the predicament of the failure of the self-driving car brake system, leaving it with two options: turn left or right while there are children and elderlies in front of you, which one will you choose to evade?



Download Program To micro:bit

Evade_kids.hex



Method 1

Clone the .hex to micro:bit directly

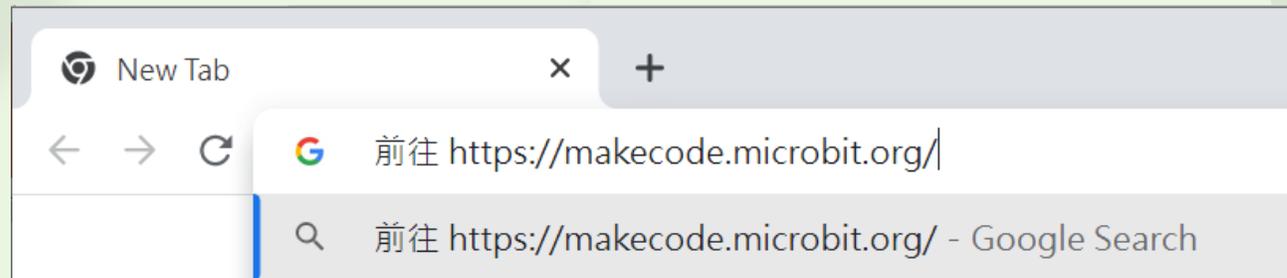
Method 2

Write your own code on MakeCode

Code On MakeCode

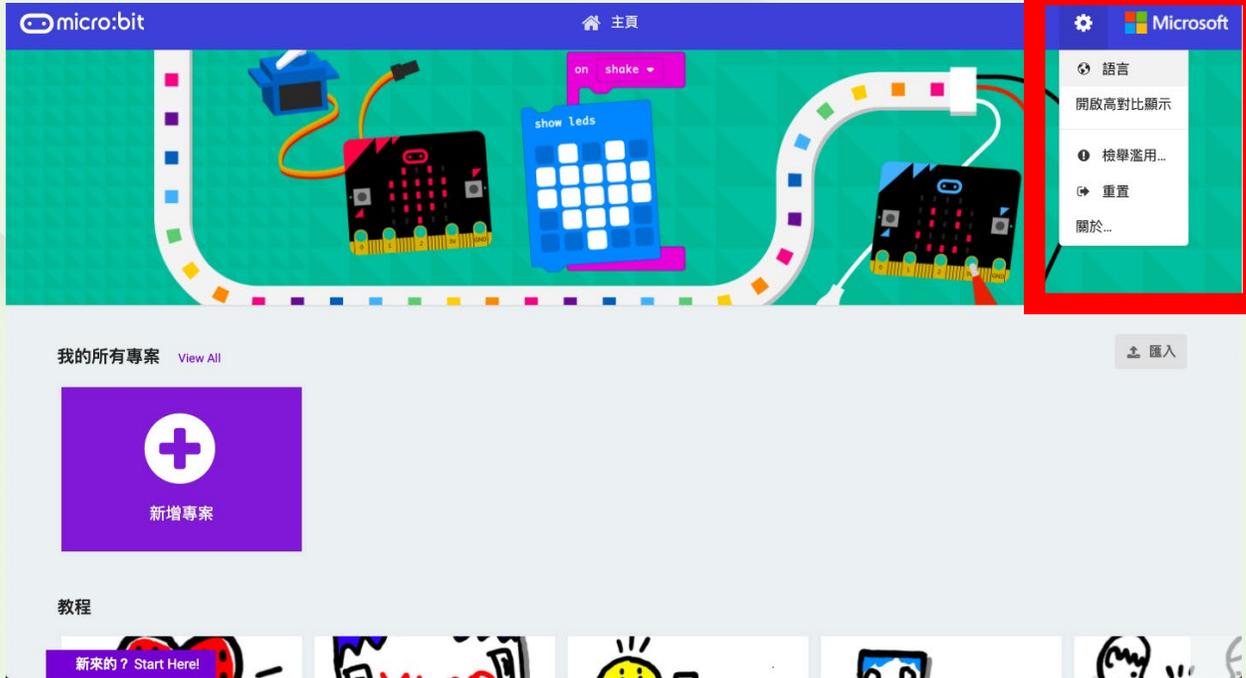


Browse <https://makecode.microbit.org/>





Caution
Please Set **English** As The Language!



1. Click
2. Click

English must be set as the language, otherwise the program may fail.

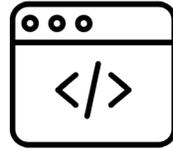
micro:bit 主頁 Microsoft

選擇語言

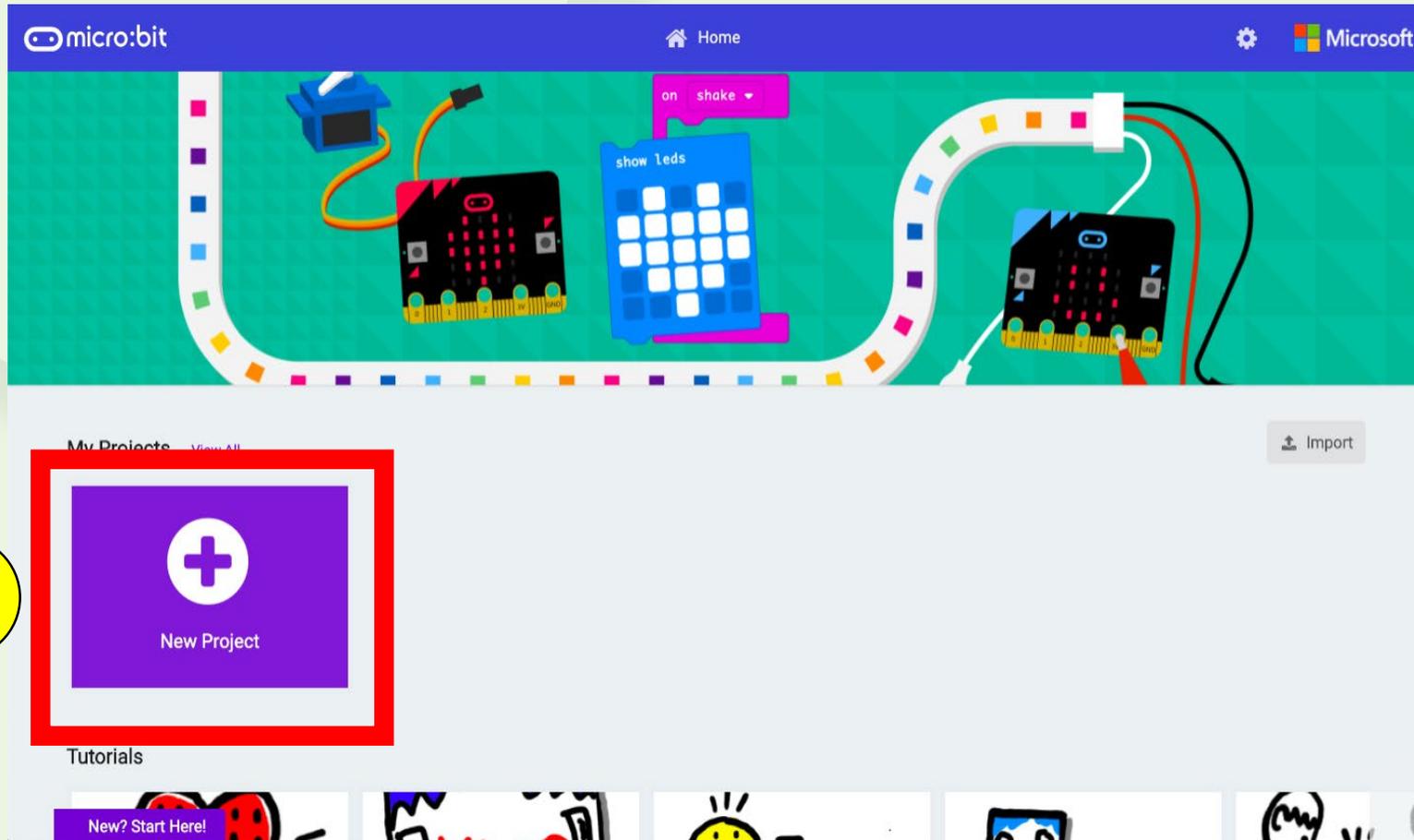
English English	العربية Arabic	български Bulgarian
Čeština Czech	Dansk Danish	Deutsch German
Ελληνικά Greek	Español (España) Spanish (Spain)	Suomi Finnish
Français French	עברית Hebrew	Magyar Hungarian

我的所有專案 匯入

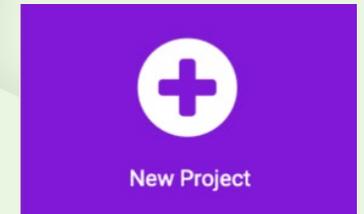
3. Click English



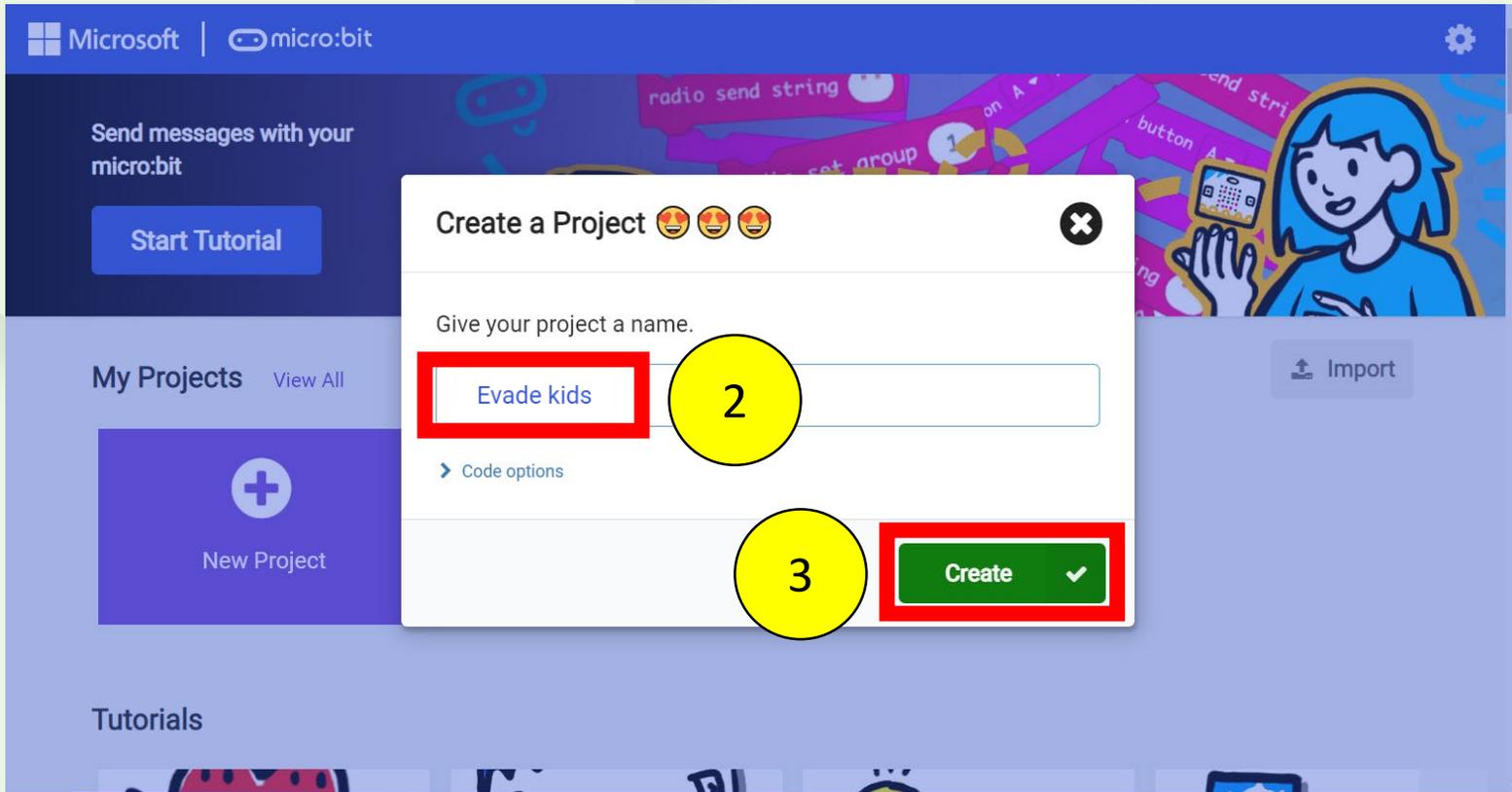
New Project



1. Click New Project

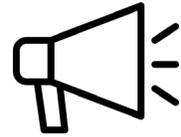


1

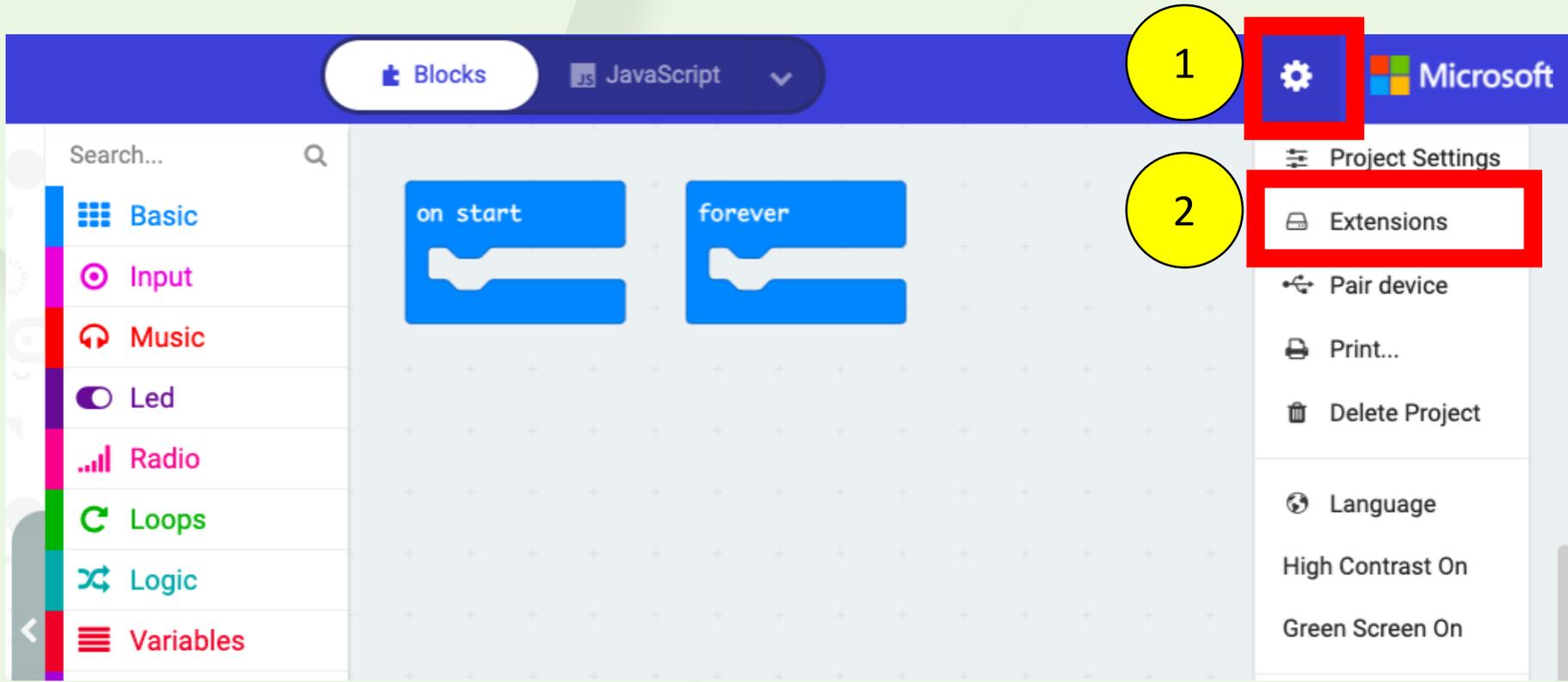


2. Enter “Evade kids”

3. Click 



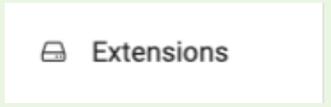
Prepare CUHK-JC-iCar Extension

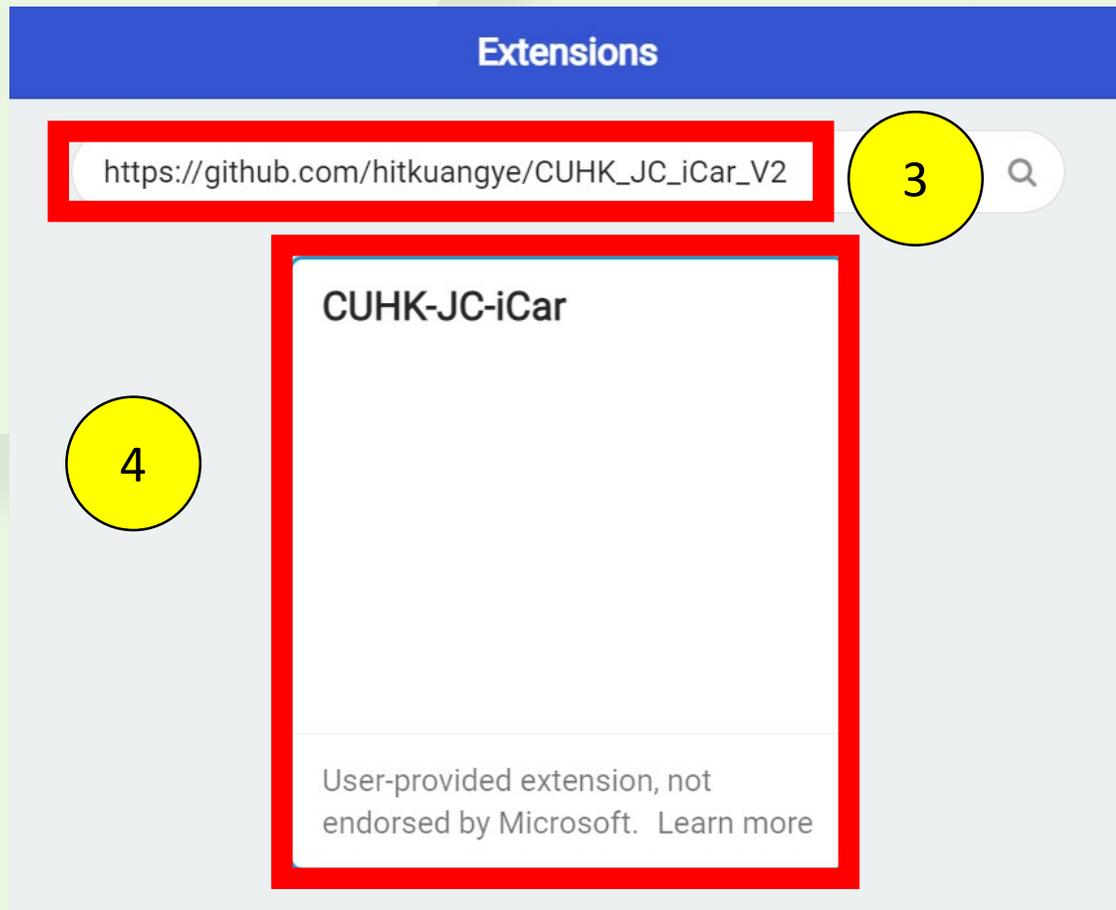


1. Click

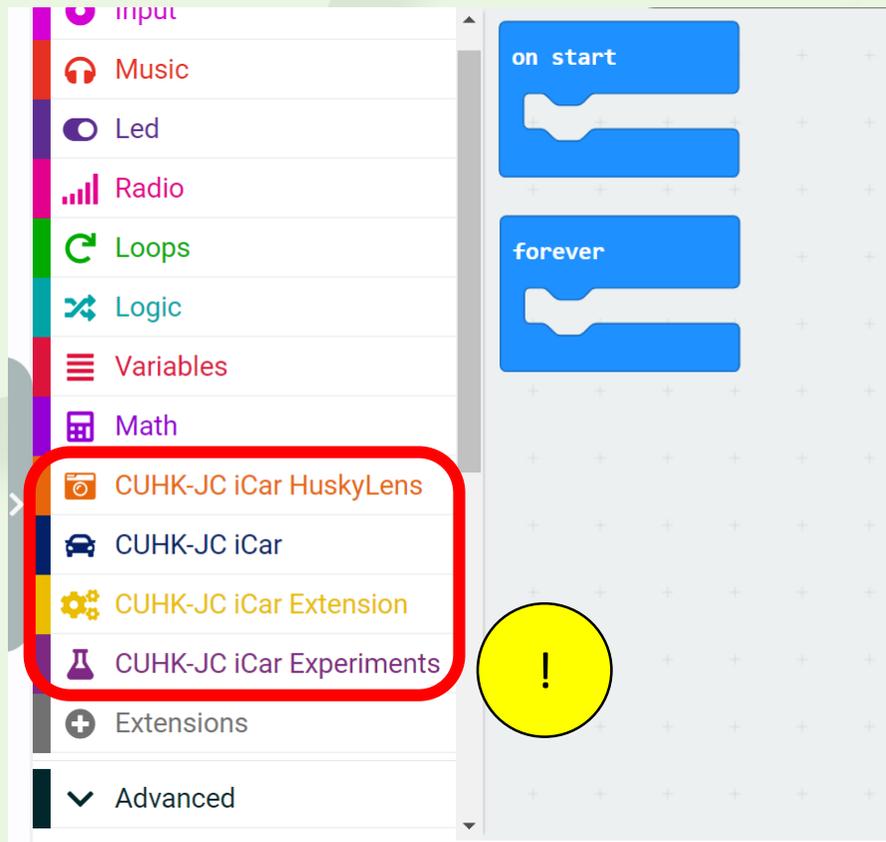


2. Click





3. Paste the following link
https://github.com/hitkuangye/CUHK_JC_iCar_V2
4. Click CUHK-JC-iCar extension



Extensions are included!

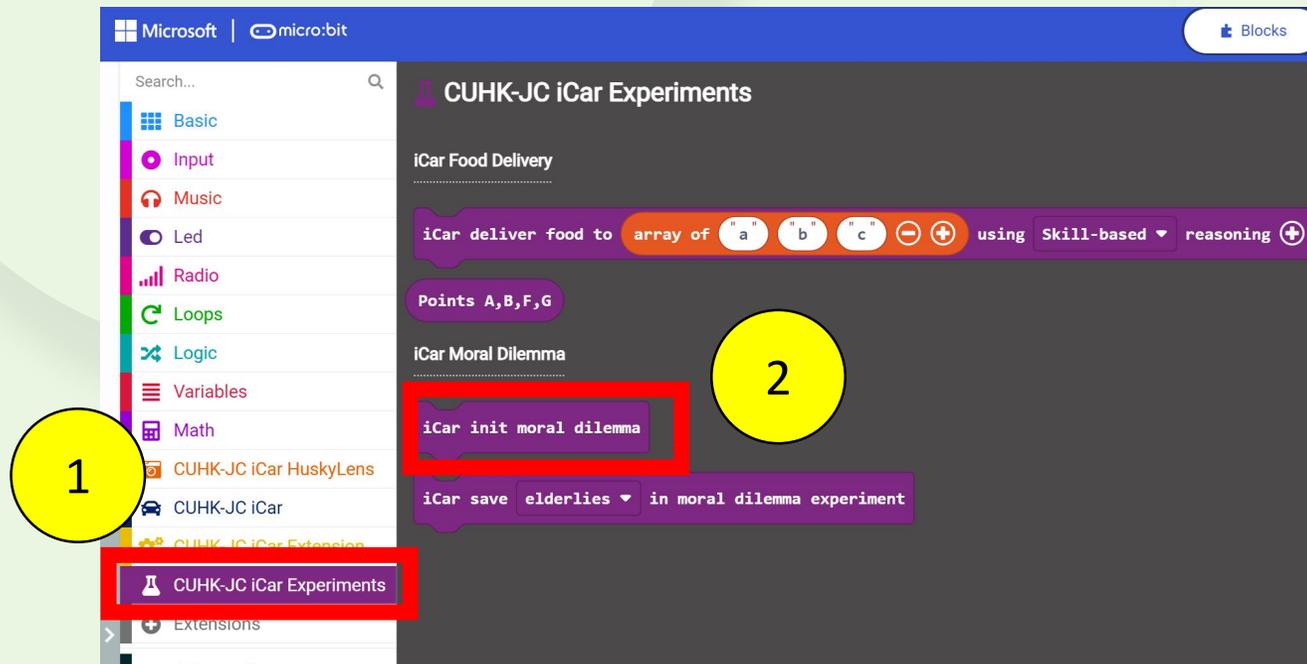
Complete the program in two ways



We prepared two options for you:

1. Use our prebuilt blocks
2. Complete the conditions by yourself

Option 1: Use Prebuilt Blocks



1. Click  from the module list

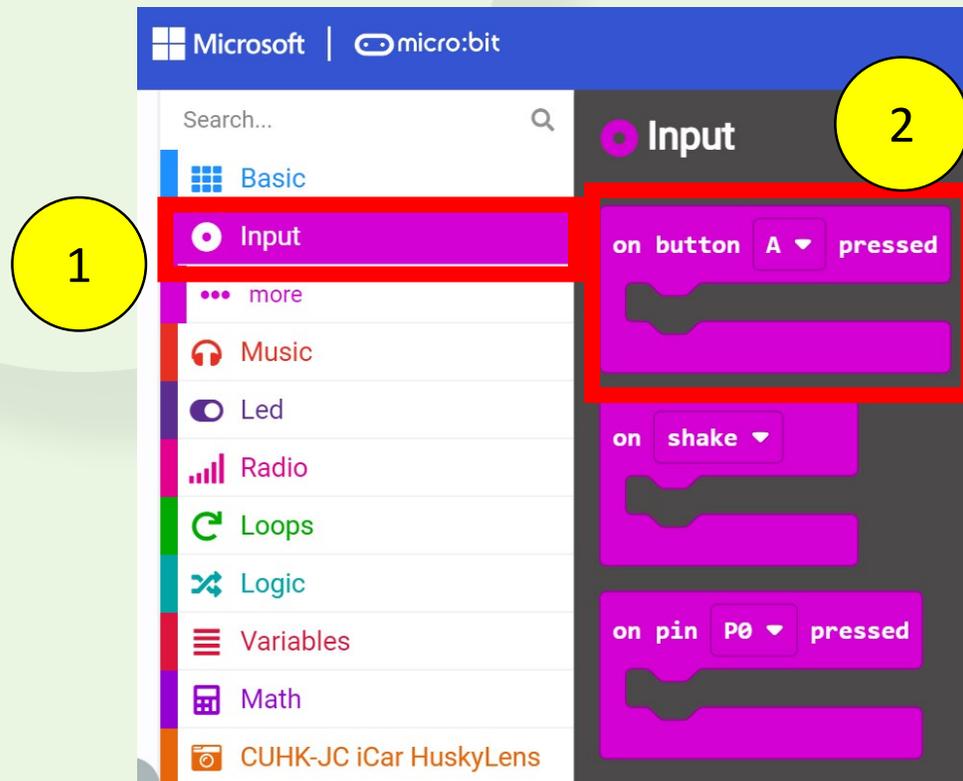
2. Click 

A blue 'on start' block with a purple 'iCar init moral dilemma' block nested inside it. The blocks are positioned on a light green background with abstract white shapes.

```
on start
  iCar init moral dilemma
```

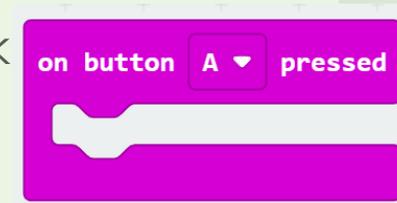
Drag the previous block into the "On Start" Block

Create on button pressed event



1. Click  from the module list

2. Click 



```
on button A ▼ pressed
  iCar save kids ▼ in moral dilemma experiment
    elderlies
    ✓ kids
```

You can find the blocks from the following:

Input

CUHK-JC iCar Experiments



Finished!

EduAIR

on start

iCar init moral dilemma

on button pressed

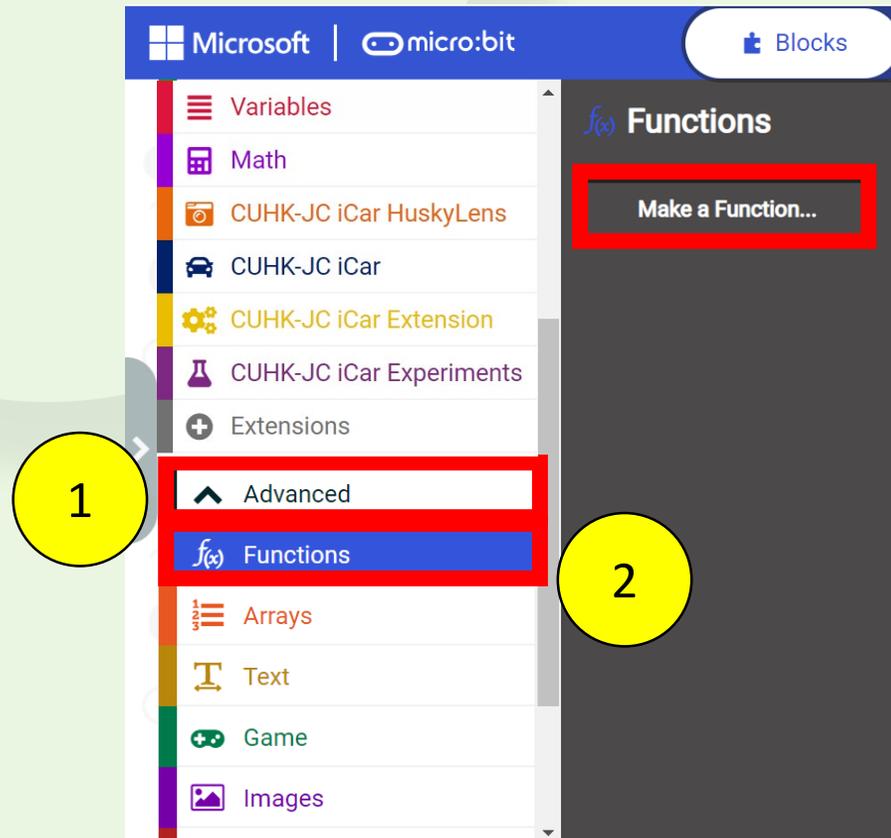
iCar save in moral dilemma experiment

Option 2: Program conditions by yourself



Before we start programming, let's learn more about micro:bit!

Functions Modules



1. Scroll to the bottom of module list and click 

2. Click 

3. Click 



4. Name the function
"Read_HuskyLens_Data"

5. Click 

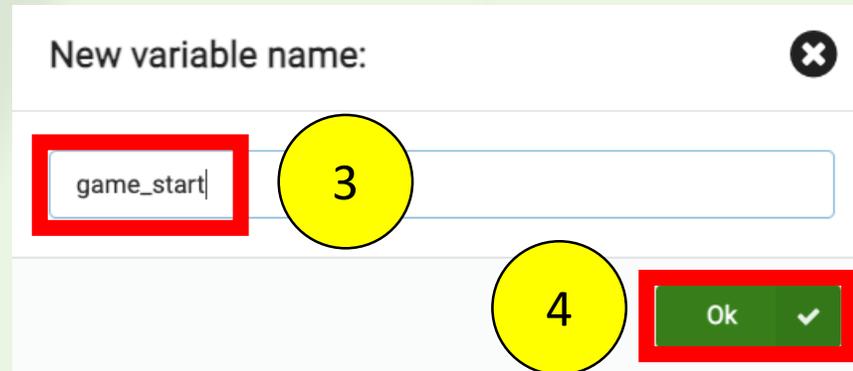
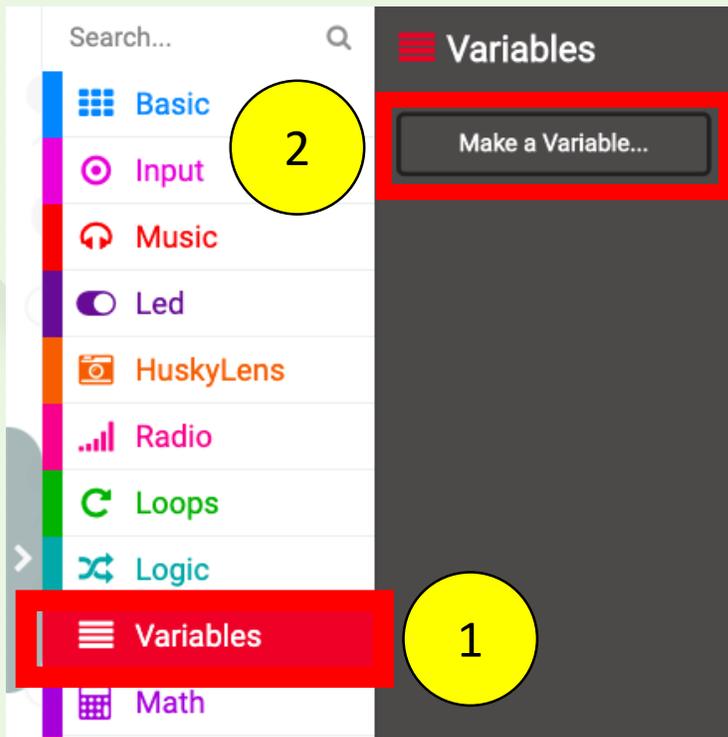
6. "Read_HuskyLens_Data"
function will appear on
the screen

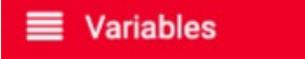
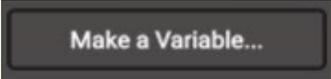
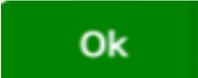


5

6

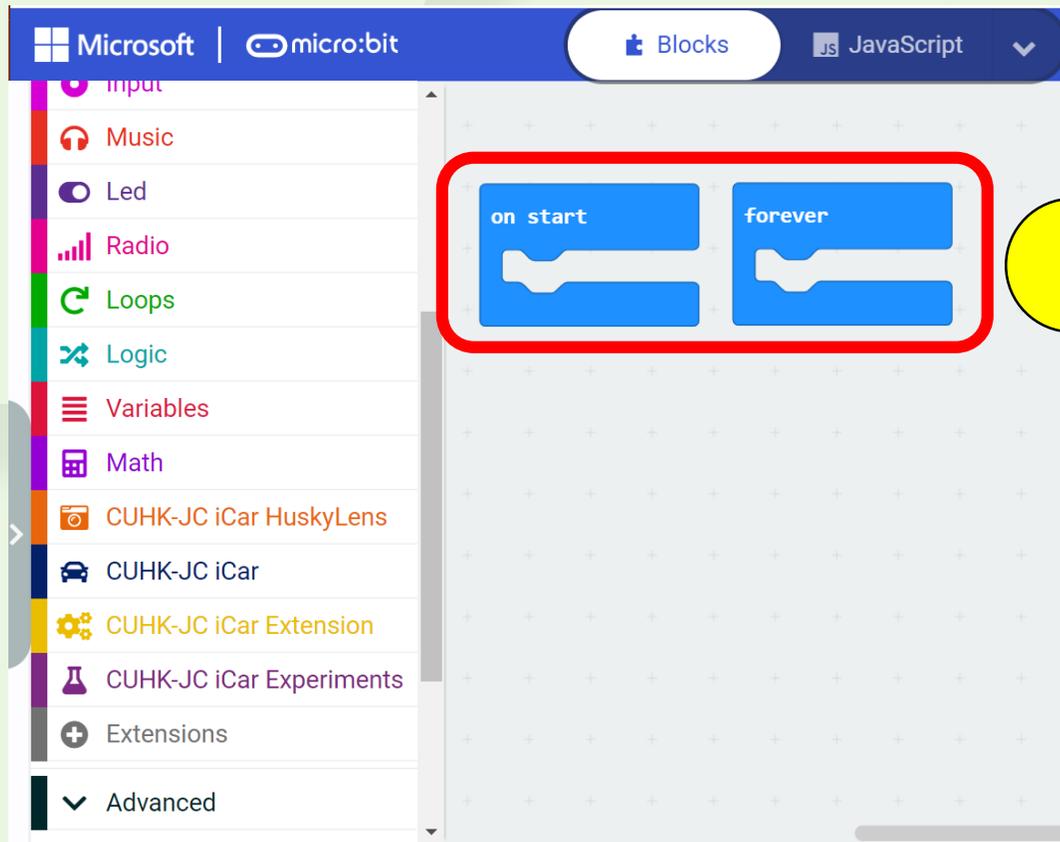
Variables Modules



1. Click 
2. Click 
3. Name the variable "game_start"
4. Click 
5. Programming blocks related to "game_start" will appear on the list

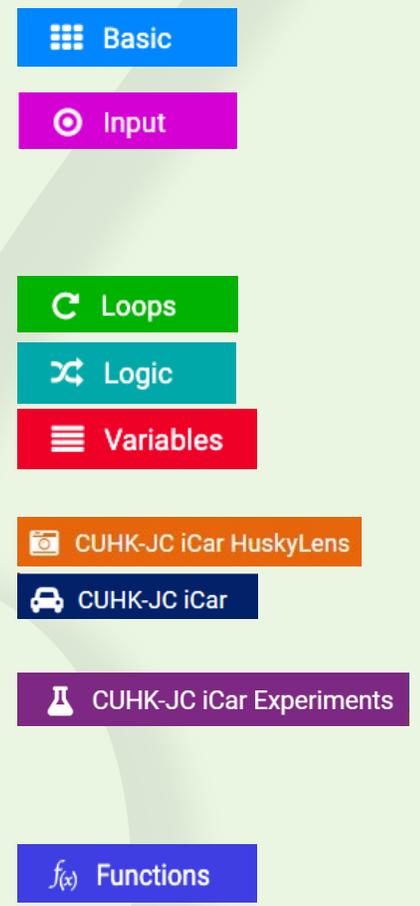
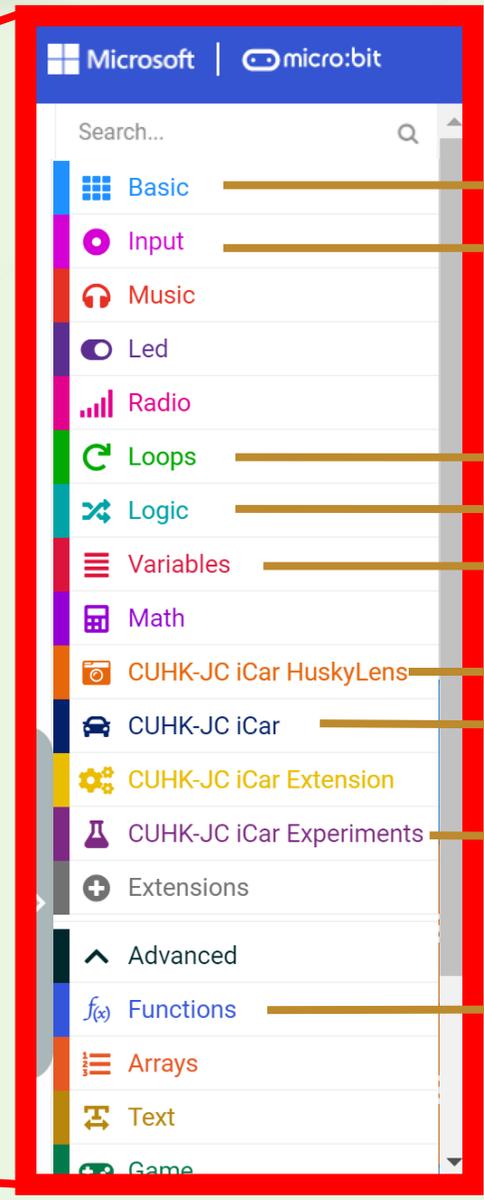
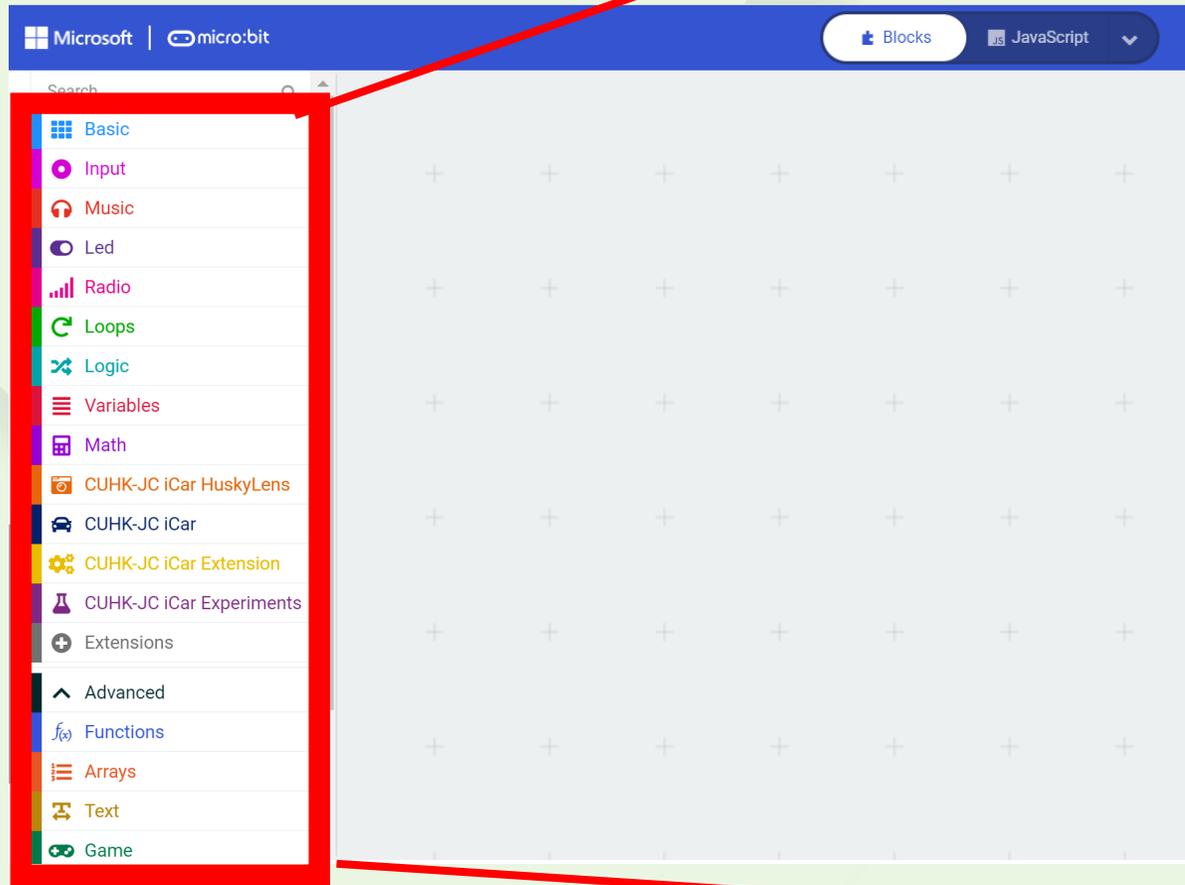


Let's start coding!



The program will pre-set “on start” and “forever” at the beginning.

We can ignore them for now.
Please don't delete!



We will use the above modules to do programming.

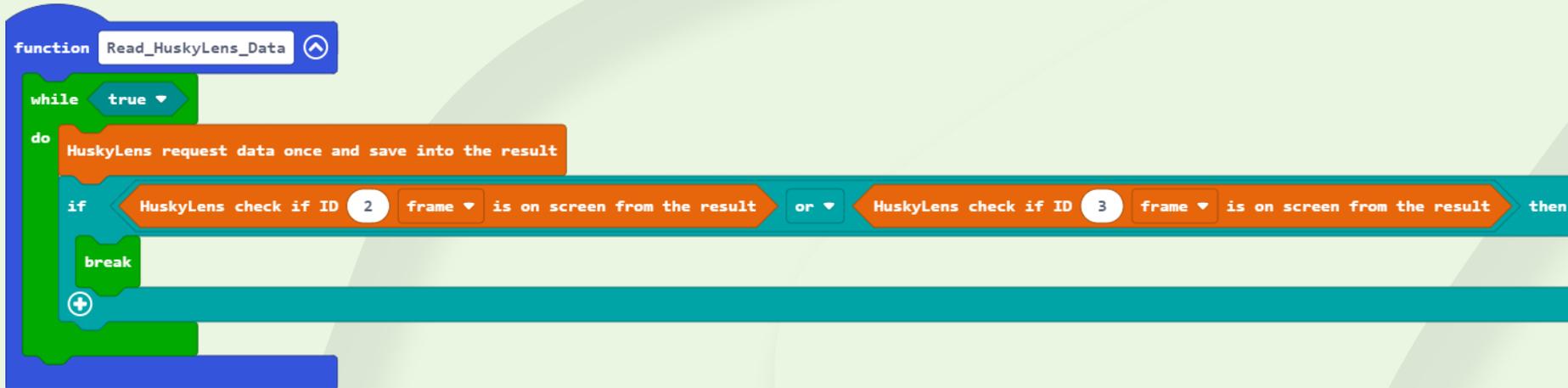

```

function Line_Following
while true
do
if is Left iCar line detector WhiteLine ? and is Right iCar line detector WhiteLine ? then
iCar Move Forward at speed 40 %
else if is Left iCar line detector WhiteLine ? and is Right iCar line detector Blackline ? then
iCar Rotate Right at speed 40 %
else if is Left iCar line detector Blackline ? and is Right iCar line detector WhiteLine ? then
iCar Rotate Left at speed 40 %
else if is Left iCar line detector Blackline ? and is Right iCar line detector Blackline ? then
iCar Stop
break

```

You can find the blocks from the following:

- Functions
- Loops
- Logic
- CUHK-JC iCar



You can find the blocks from the following:

$f(x)$ Functions

Loops

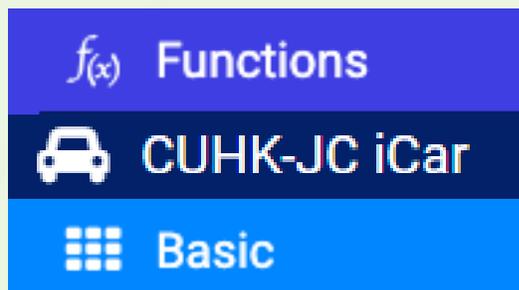
CUHK-JC iCar HuskyLens

Logic

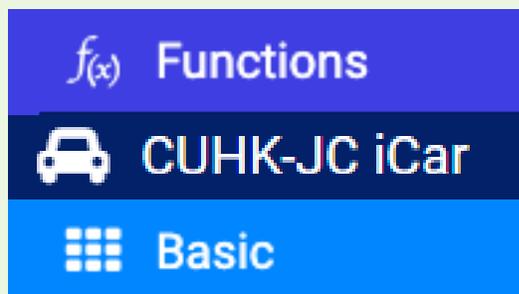
```
function Turn_Left  
iCar Turn Left at speed 70 %  
pause (ms) 200  
iCar Move Forward at speed 60 %  
pause (ms) 1000  
iCar Stop
```

```
function Turn_Right  
iCar Turn Right at speed 70 %  
pause (ms) 200  
iCar Move Forward at speed 60 %  
pause (ms) 1000  
iCar Stop
```

You can find the blocks from the following:



You can find the blocks from the following:



```
function Make_Decision ^
  if HuskyLens check if ID 2 frame is on screen from the result then
    show number 2
    call Turn_Left
  else if HuskyLens check if ID 3 frame is on screen from the result then -
    show number 3
    call Turn_Right
  +
```

You can find the blocks from the following:

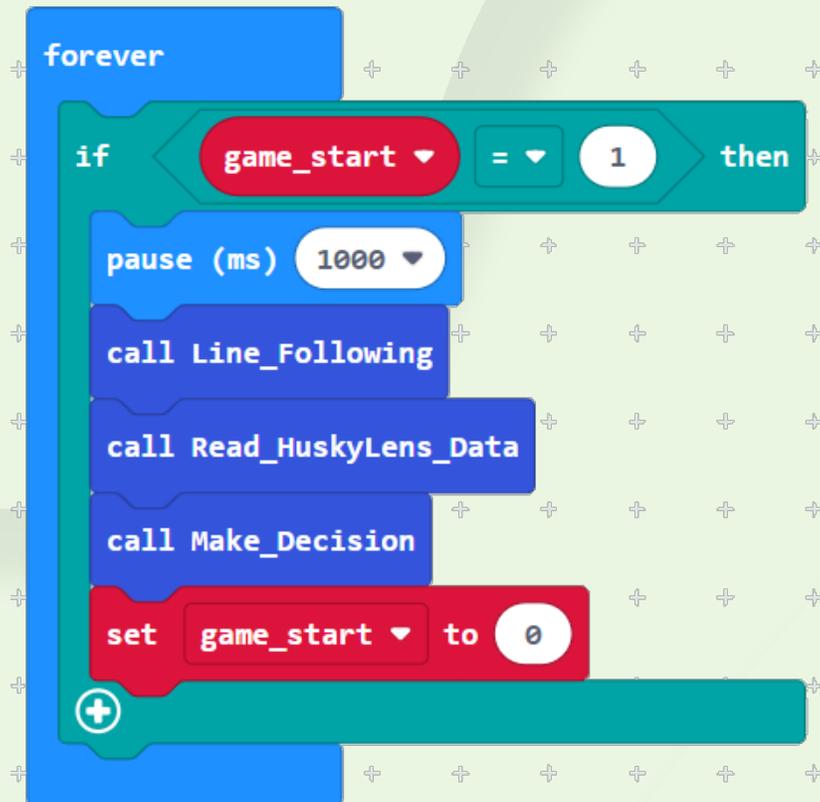
- Functions
- Logic
- CUHK-JC iCar HuskyLens
- Basic



You can find the blocks from the following:

 Input

 Variables



You can find the blocks from the following:

 Logic

 Variables

 Basic

 Functions



Finished!



```
on start
  set game_start to 0
  HuskyLens initialize I2C until success
  HuskyLens switch algorithm to Object Classification
  show icon
```

```
forever
  if game_start == 1 then
    pause (ms) 1000
    call line_following
    call Read_HuskyLens_Data
    call Make_Decision
    set game_start to 0
```

```
on button A pressed
  set game_start to 1
```

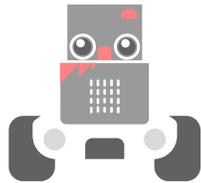
```
function Line_Following
  while true
    do
      if is Left iCar line detector Whiteline ? and is Right iCar line detector Whiteline ? then
        iCar Move Forward at speed 40 %
      else if is Left iCar line detector Whiteline ? and is Right iCar line detector Blackline ? then
        iCar Rotate Right at speed 40 %
      else if is Left iCar line detector Blackline ? and is Right iCar line detector Whiteline ? then
        iCar Rotate Left at speed 40 %
      else if is Left iCar line detector Blackline ? and is Right iCar line detector Blackline ? then
        iCar Stop
      break
```

```
function Read_HuskyLens_Data
  while true
    do
      HuskyLens request data once and save into the result
      if HuskyLens check if ID 2 frame is on screen from the result or HuskyLens check if ID 3 frame is on screen from the result then
        break
```

```
function Make_Decision
  if HuskyLens check if ID 2 frame is on screen from the result then
    show number 2
    call Turn_Left
  else if HuskyLens check if ID 3 frame is on screen from the result then
    show number 3
    call Turn_Right
```

```
function Turn_Left
  iCar Turn Left at speed 70 %
  pause (ms) 200
  iCar Move Forward at speed 50 %
  pause (ms) 1000
  iCar Stop
```

```
function Turn_Right
  iCar Turn Right at speed 70 %
  pause (ms) 200
  iCar Move Forward at speed 50 %
  pause (ms) 1000
  iCar Stop
```



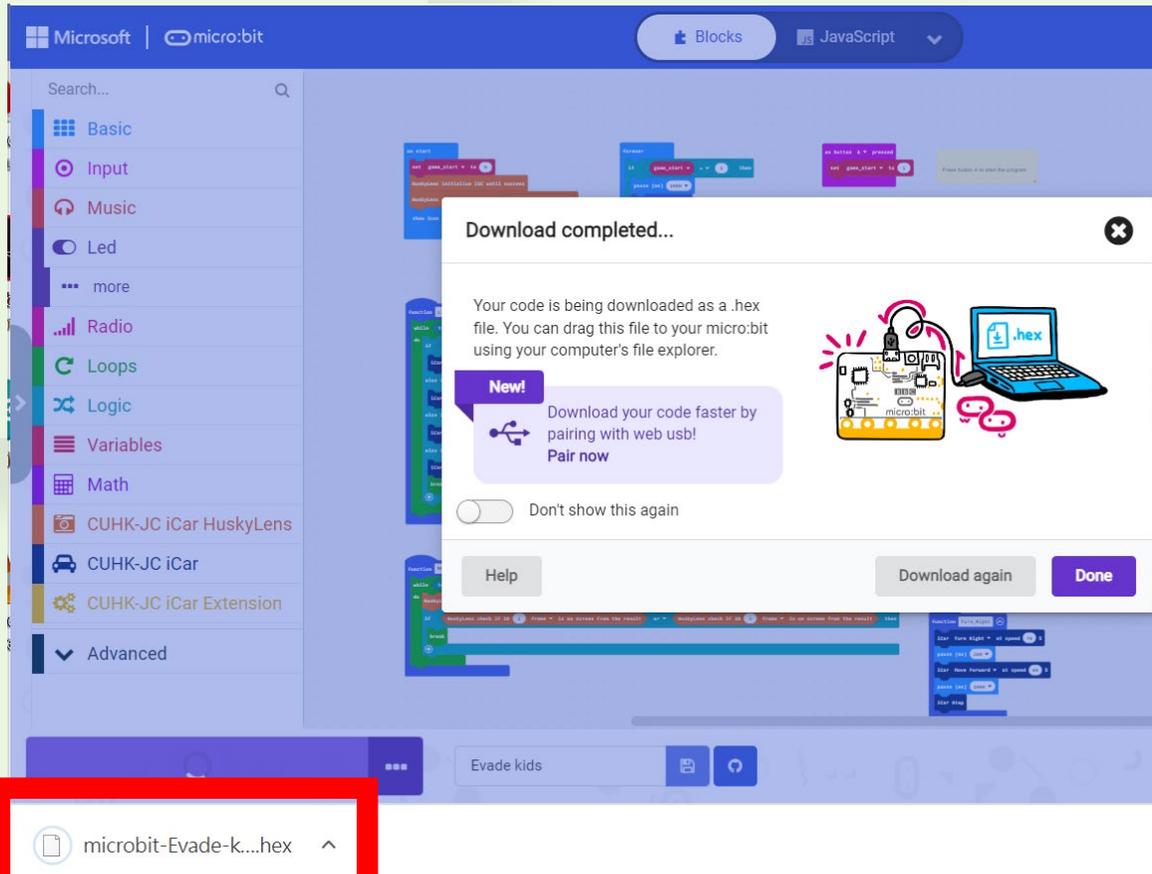
Download The Program To CUHK iCar

Step 1:

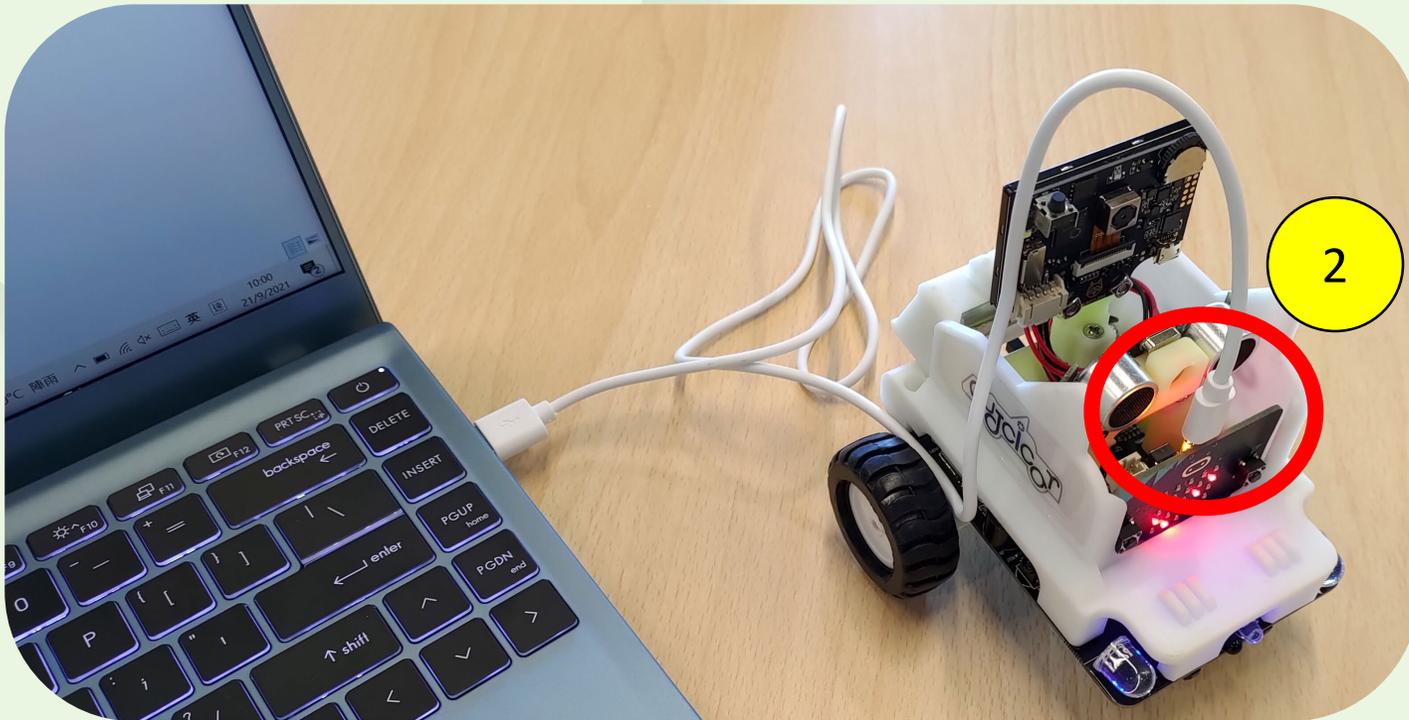
Click

 Download

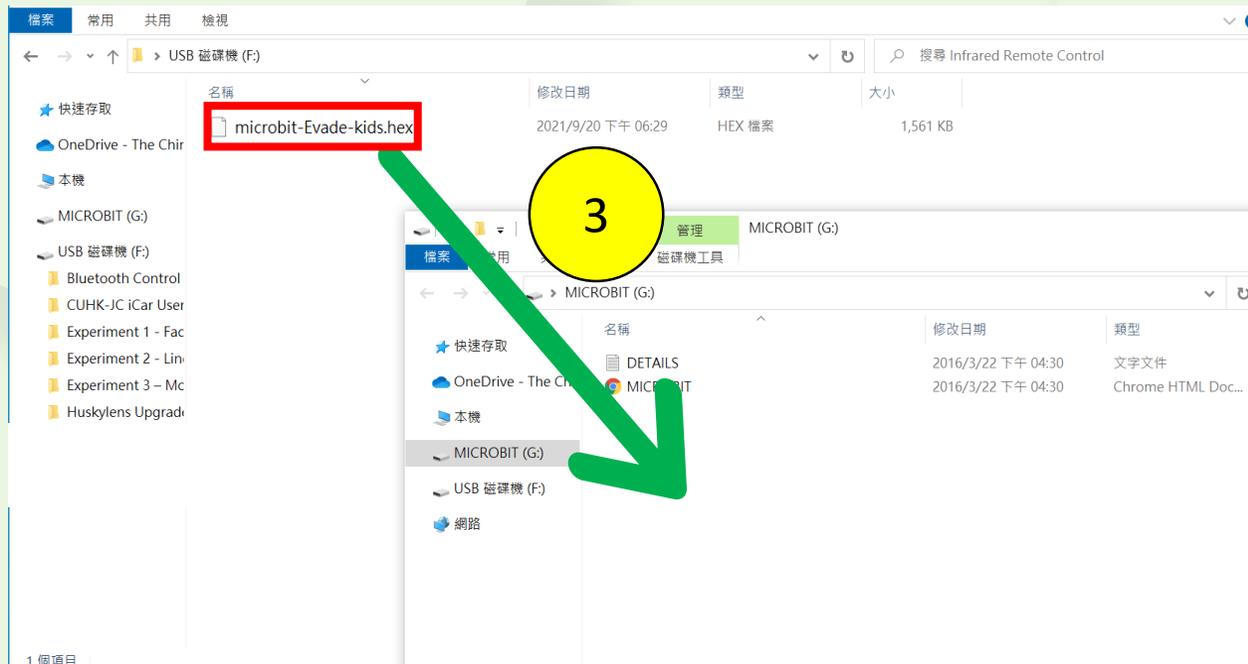
1



The hex file is downloaded!



Step 2:
Connect the micro:bit to
computer by a micro USB cable



Step 3:
Drag the downloaded hex file
into the micro:bit window



Step 4:
Wait for the completion of cloning process

Caution:

- The micro:bit window will potentially disappear after the completion
- After the completion of cloning process, the hex file will not be displayed in the micro:bit window

Step 5:
Disconnect the micro:bit from your computer

If the connection between CUHK iCar and the computer is failed:

- Restart the computer
- Try another USB port
- Change the micro USB cable



You have finished programming!

Let's start collecting data!



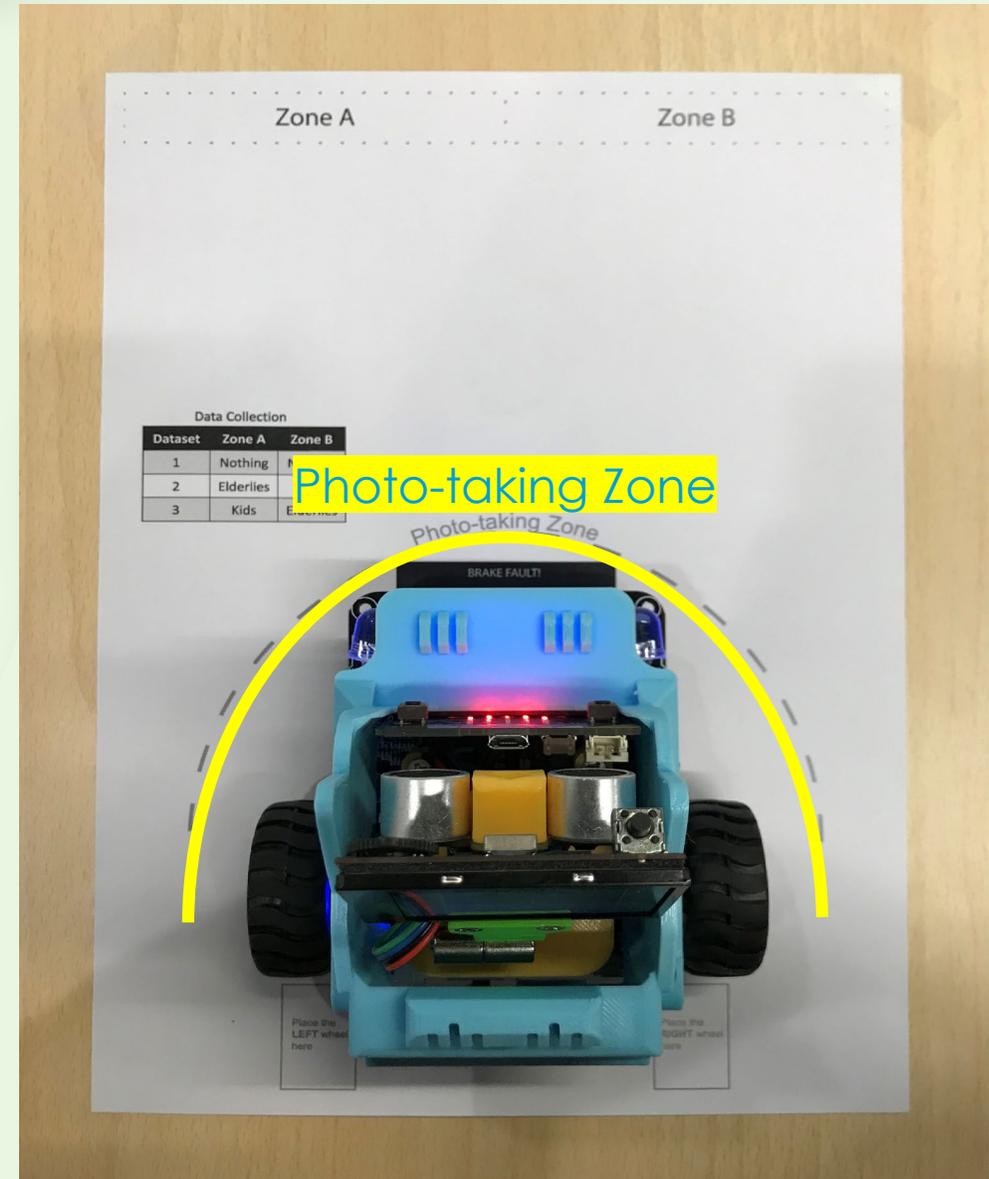


Data Collection



Step 1:

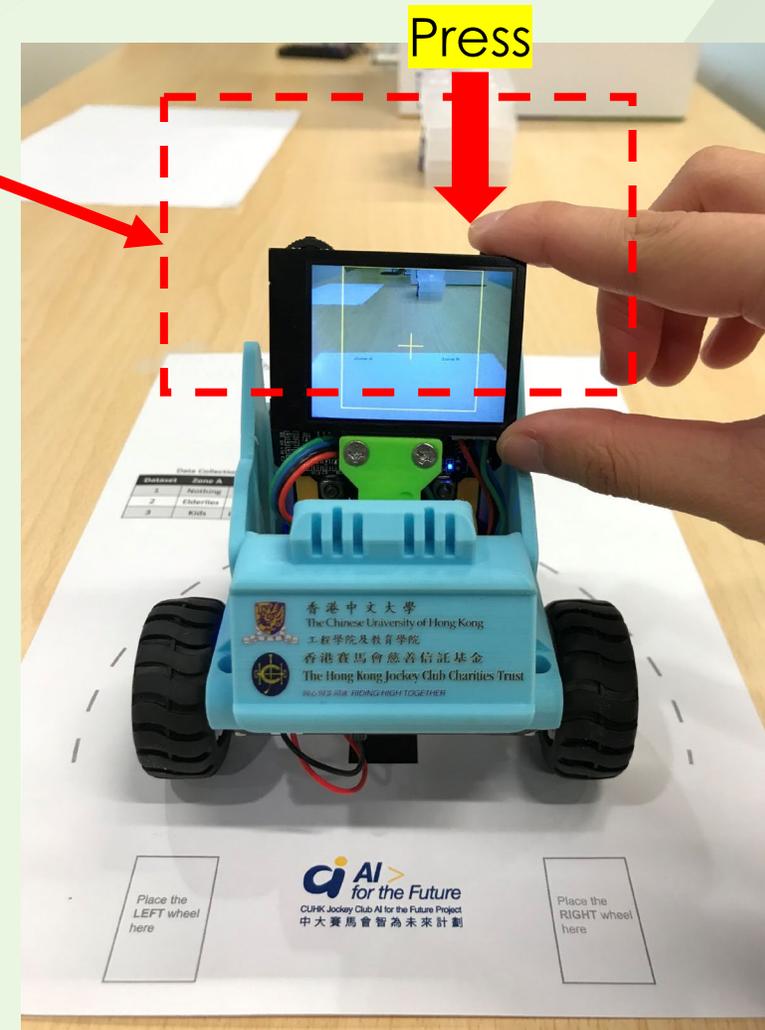
Place the track onto the table, and the CUHK iCar on the Photo-taking Zone



Dataset	Zone A	Zone B
1	Nothing	Nothing
2	Elderlies	Kids
3	Kids	Elderlies

Step 2 - Collect Dataset 1:

- CUHK iCar should face forward
- **Press** the learning button to collect data



Tips:

Rotate the CUHK iCar to obtain a better result

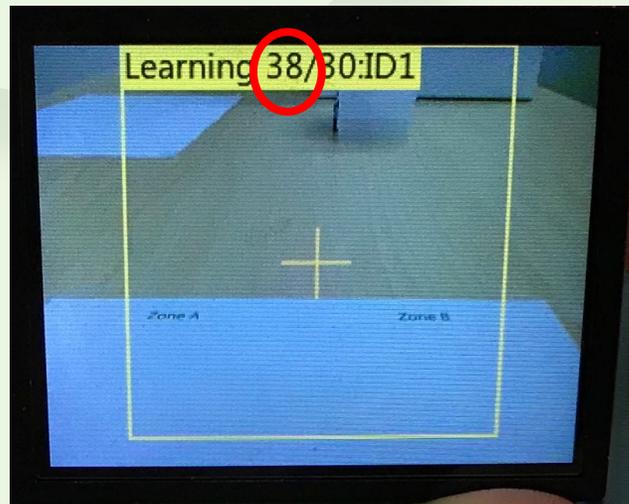
Make sure you keep pressing the button while you're rotating!

Press



Step 3:

Release the learning button after the number reaches 30



Step 4:

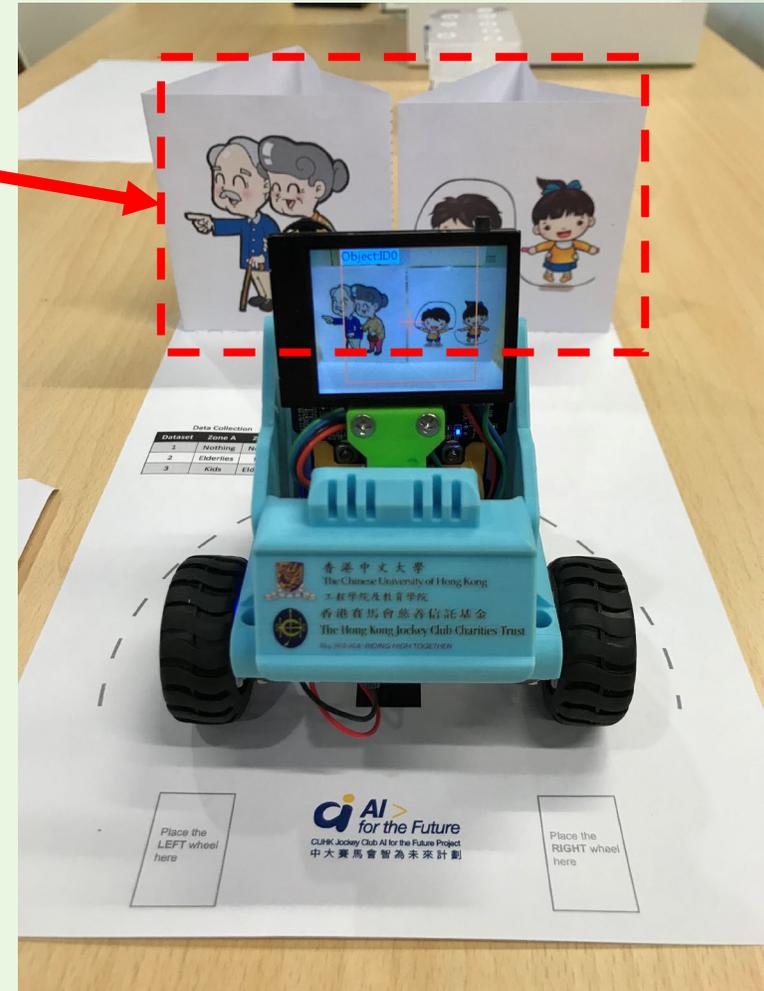
Press the learning button again
before the countdown ends



Dataset	Zone A	Zone B
1	Nothing	Nothing
2	Elderlies	Kids
3	Kids	Elderlies

Step 5 - Collect Dataset 2:

- CUHK iCar should face forward
- Place the paper model accordingly
- **Press** the learning button to collect data



Tips:

Rotate the CUHK iCar to obtain a better result

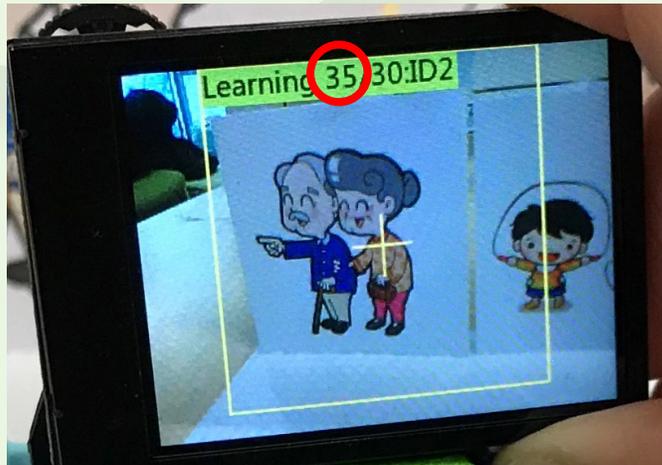
Make sure you keep pressing the button while you're rotating!

Press



Step 6:

Release the learning button after the number reaches 30



Step 7:

Press the learning button again
before the countdown ends



Dataset	Zone A	Zone B
1	Nothing	Nothing
2	Elderlies	Kids
3	Kids	Elderlies



Step 8 - Collect Dataset 3:

- CUHK iCar should face forward
- Place the paper model accordingly
- **Press** the learning button to collect data



Tips:

Rotate the CUHK iCar to obtain a better result

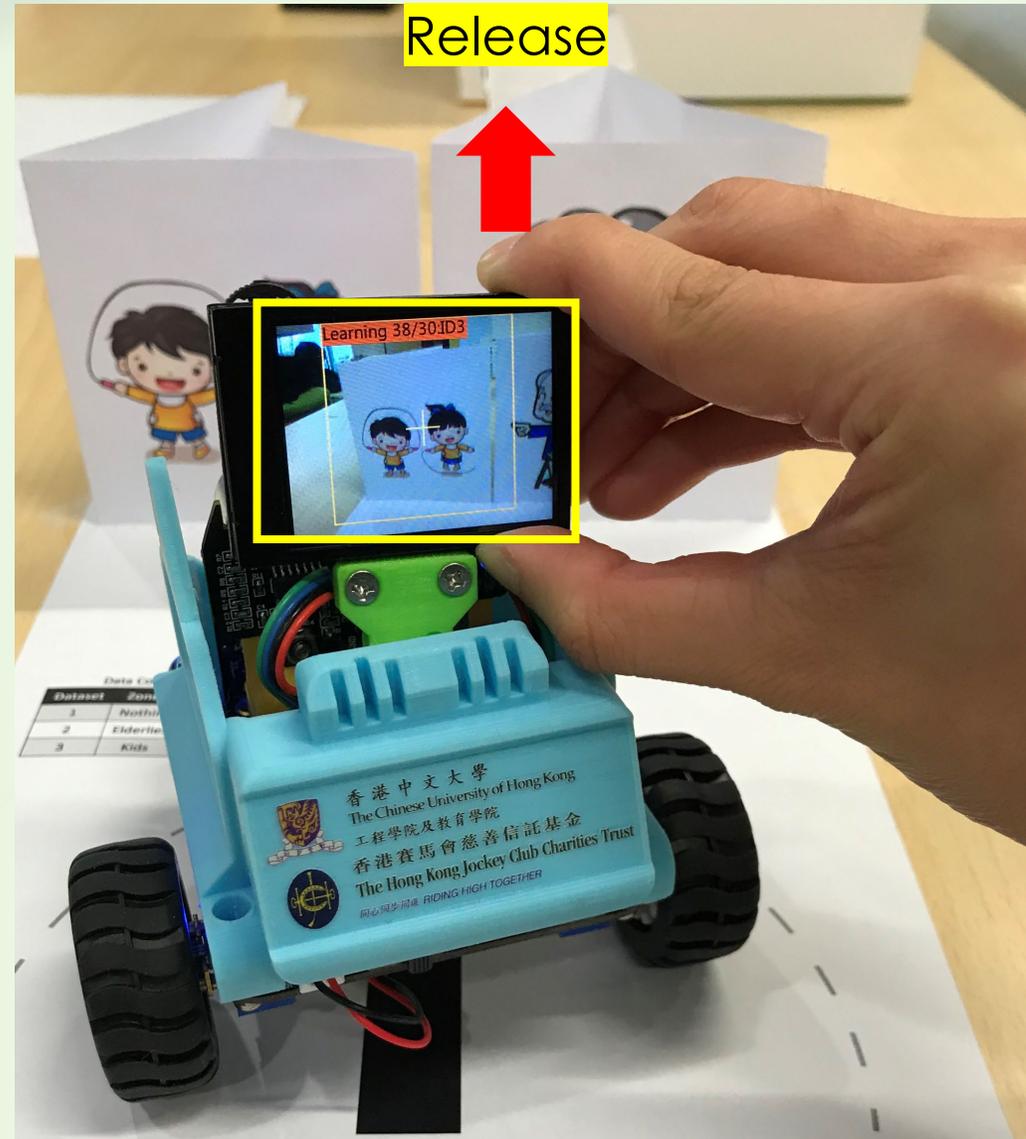
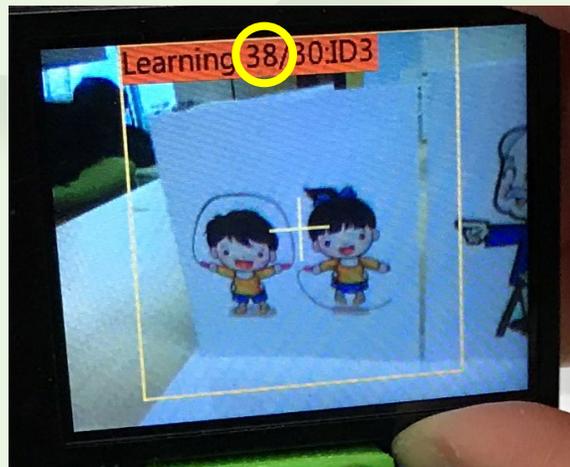
Make sure you keep pressing the button while you're rotating!

Press



Step 9:

Release the learning button after the number reaches 30



Dataset	Zone
1	North
2	Elderly
3	Kids

Step 10:

Wait for the

countdown ends





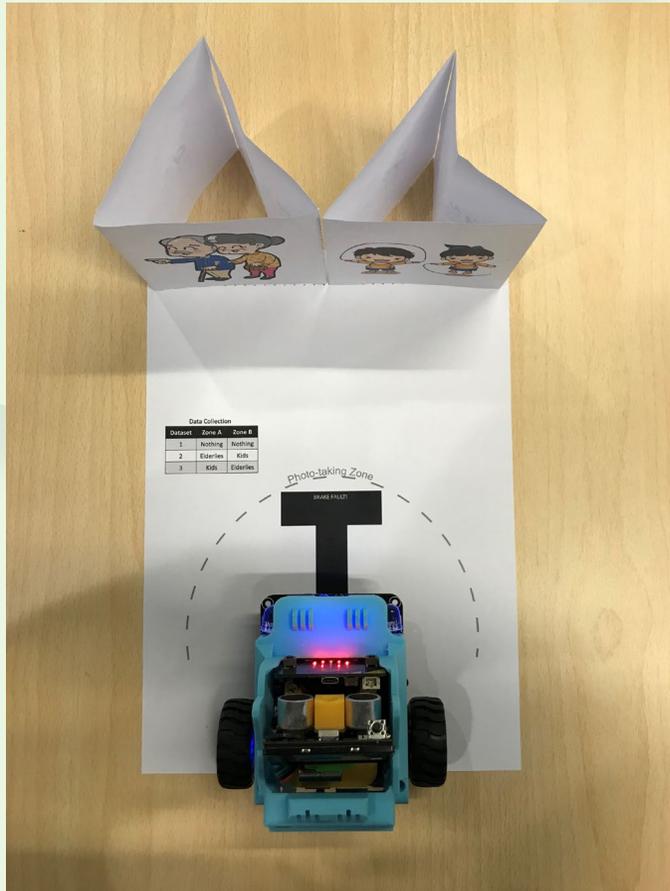
You've collected all the data needed!

Let's conduct the experiment!



Step 1:

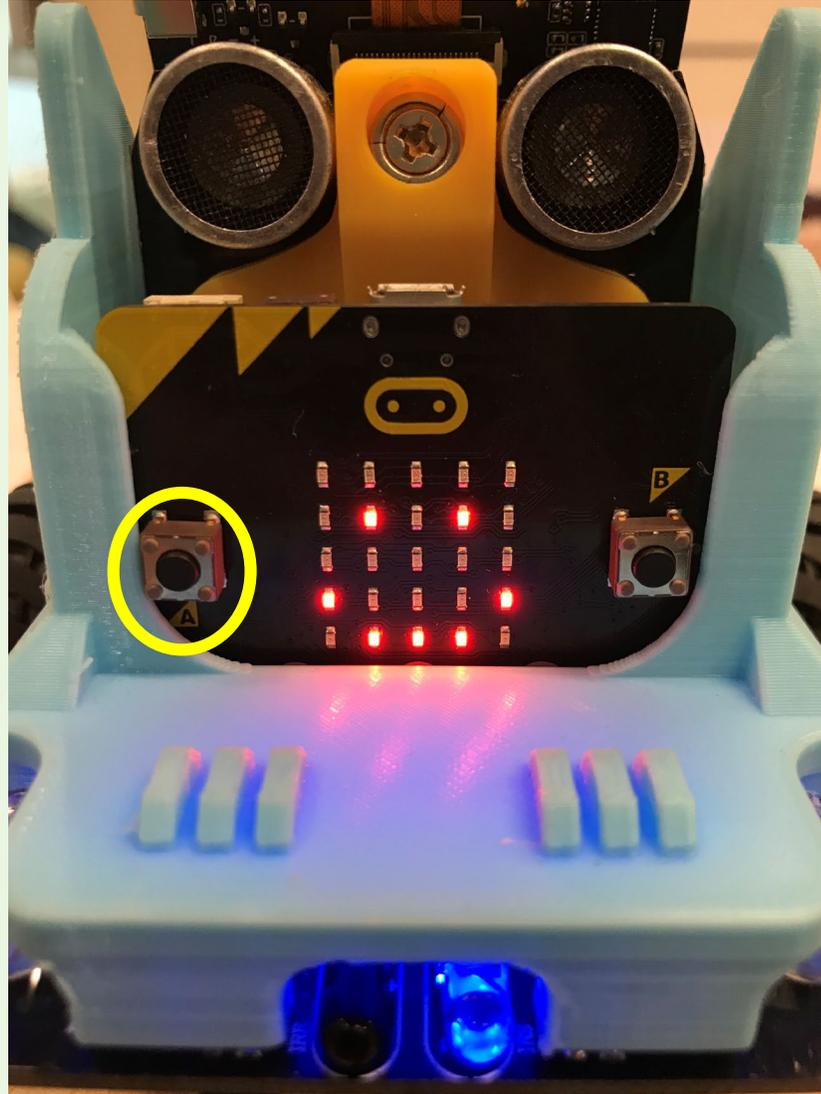
Place the paper models in the designated zones and place CUHK iCar on the starting point



Attention:
Place the two
wheels according
to the instructions
on the paper

Step 2:

Press **button A** on micro:bit



Simulation

Program: Evade Kids

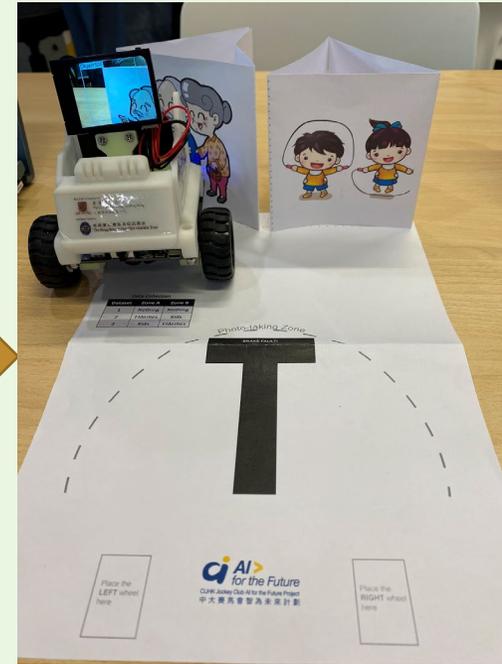
Paper Model: Dataset 2



Place the CUHK iCar at the starting position and press button A.



CUHK iCar has trouble braking!



It will evade the kids.

Simulation

Program: Evade Kids
Paper Model: Dataset 3



Place the CUHK iCar at the starting position and press button A.



CUHK iCar has trouble braking!



It will evade the kids.

If CUHK iCar does not work as expected:

- Try to fully charge the CUHK iCar
- If the CUHK iCar still does not work as expected, then you can try to recollect data by pressing the learning button twice to forget the recorded data, then to recollect data. For details, please refer to the slide 40

