



# 愛動智教育系統

## CUHK iCar 實驗操作手冊 實驗一：人面識別追蹤實驗

---

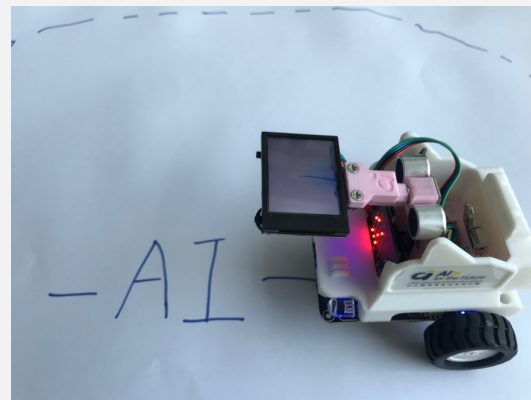
### 自行編寫

# CUHK iCar

EdwAIR



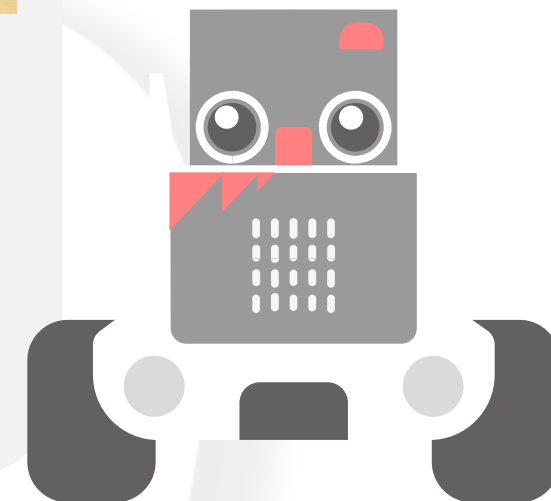
**實驗一**  
人面識別追蹤實驗



**實驗二**  
手畫線循跡實驗



**實驗三**  
道德困境實驗



# 人面識別追蹤實驗

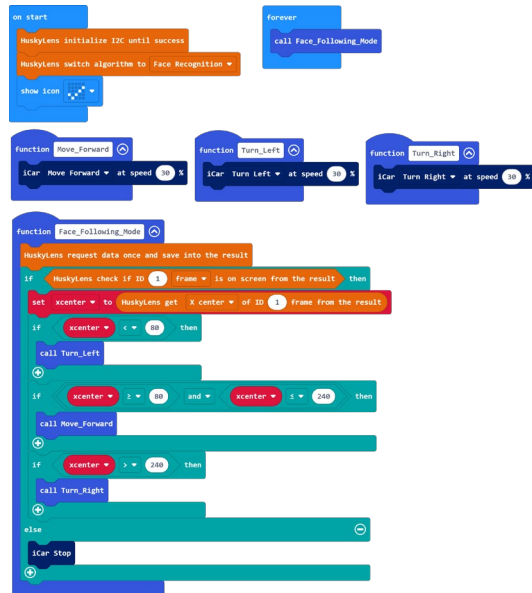
## 實驗簡介

這實驗模擬自動駕駛汽車跟隨特定的人臉行駛



# 下載程式到 micro:bit

人面識別追蹤實驗.hex



## 方法一

直接燒錄 .hex 到  
micro:bit

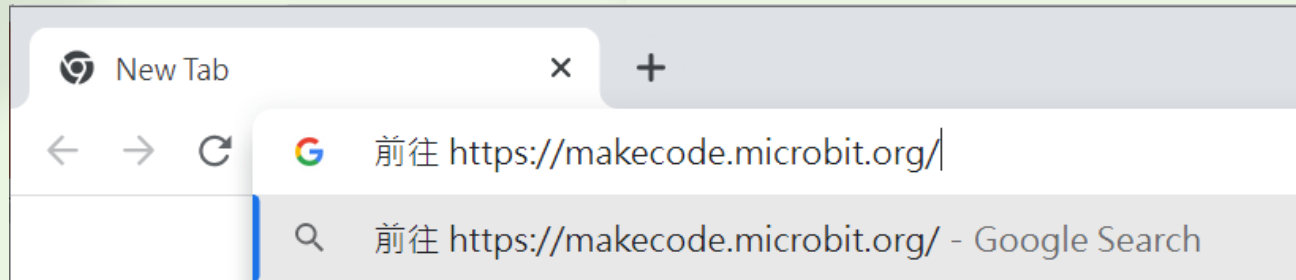
## 方法二

自行在 MakeCode 上  
編寫

# 自行在 MakeCode 上編寫

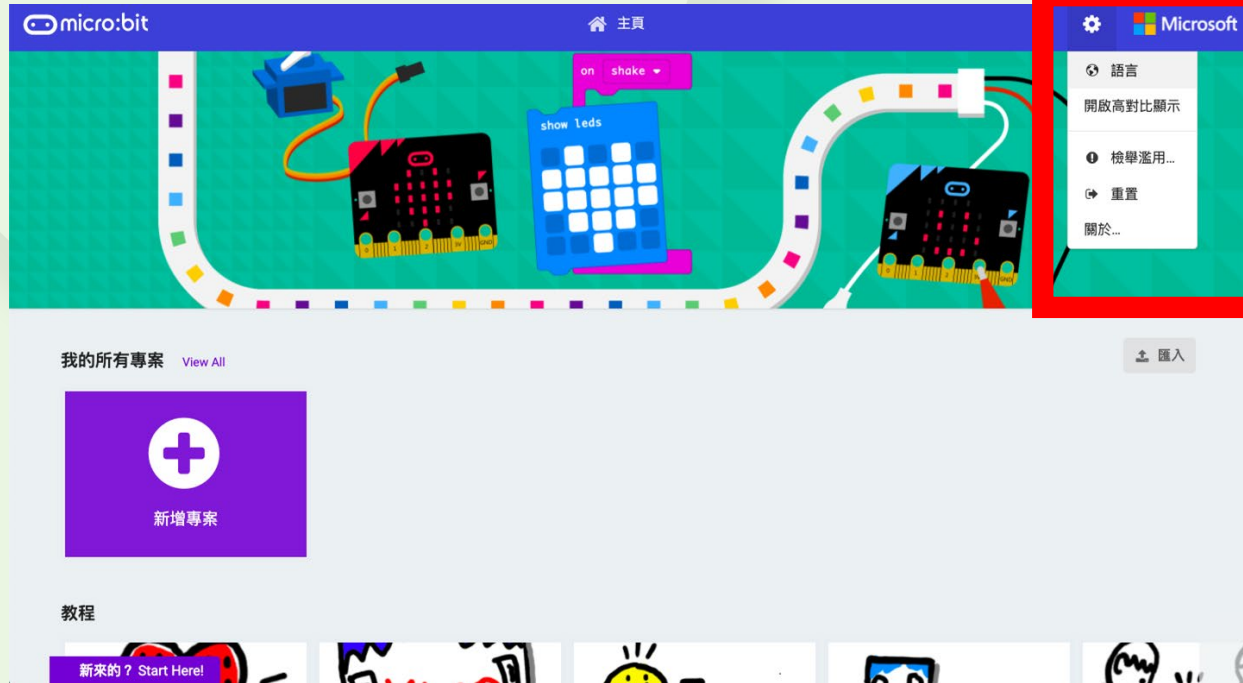


前往 <https://makecode.microbit.org/>





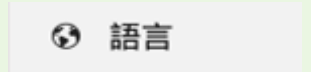
注意  
請設定**英文**為語言！



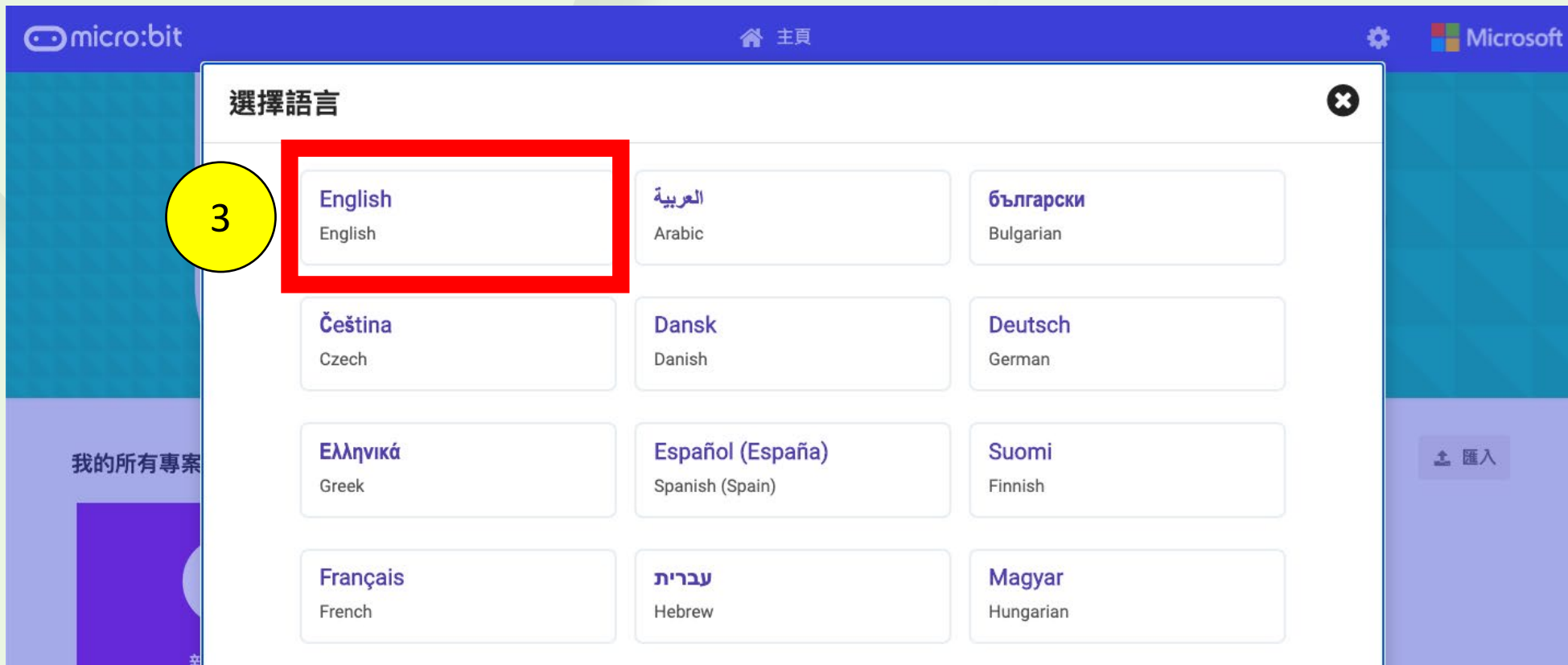
1. 按



2. 按

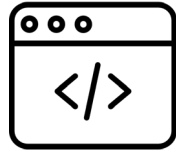


**必須設定英文為語言，  
否則程式可能會出現錯誤**

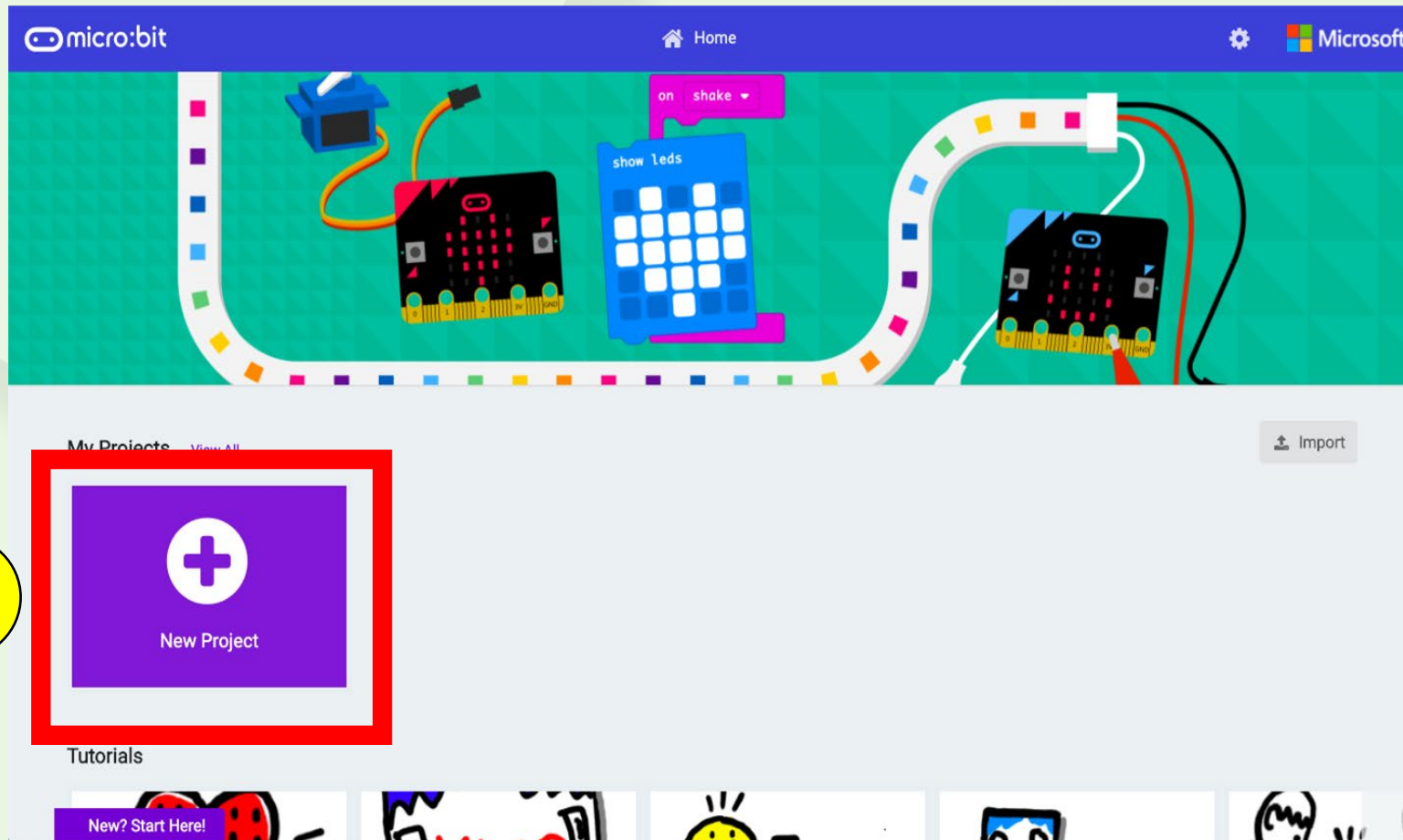


### 3. 選擇 English

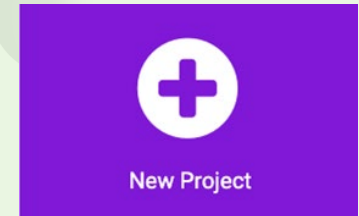




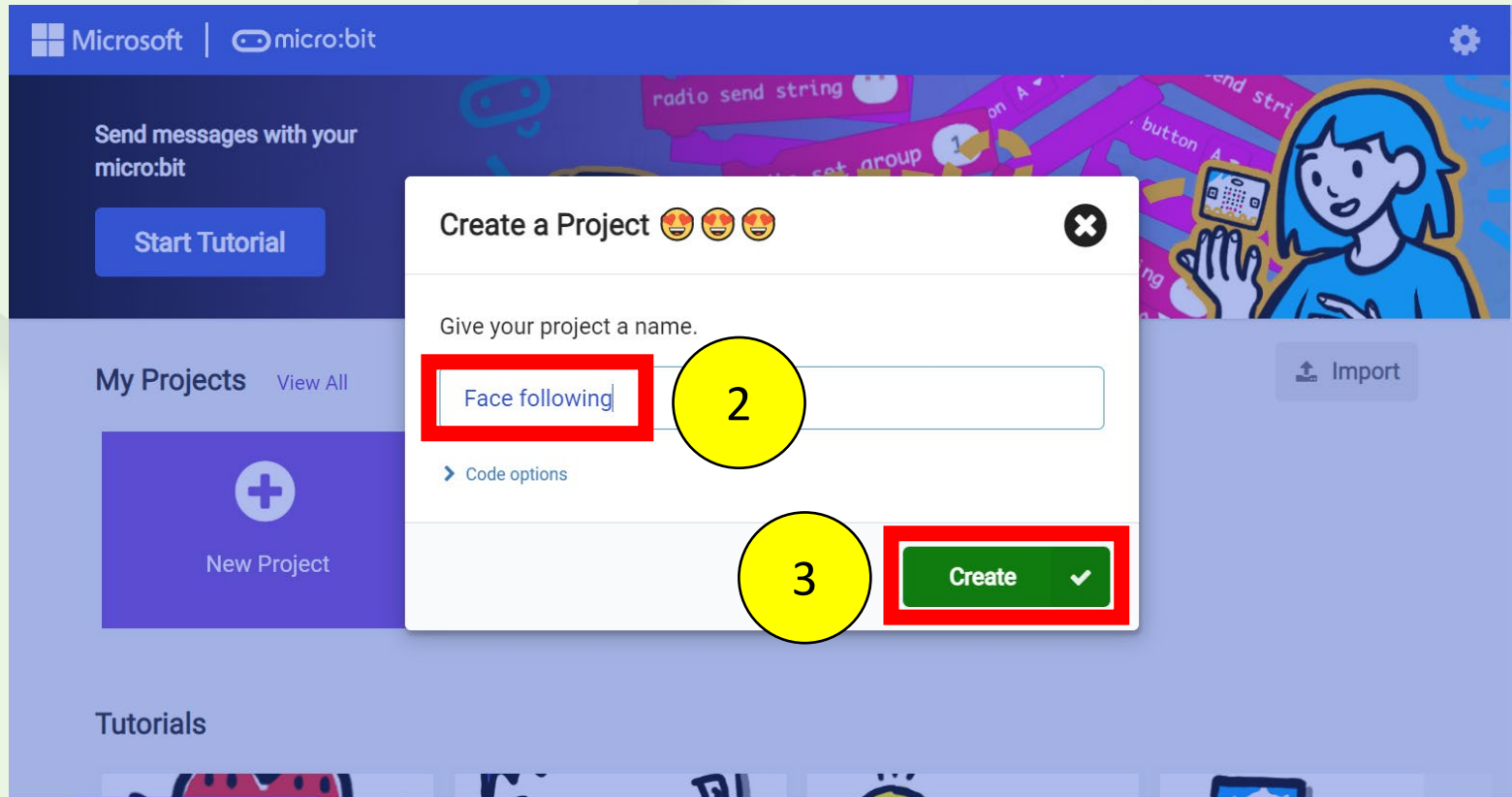
開啟新項目



1. 按 New Project

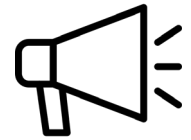




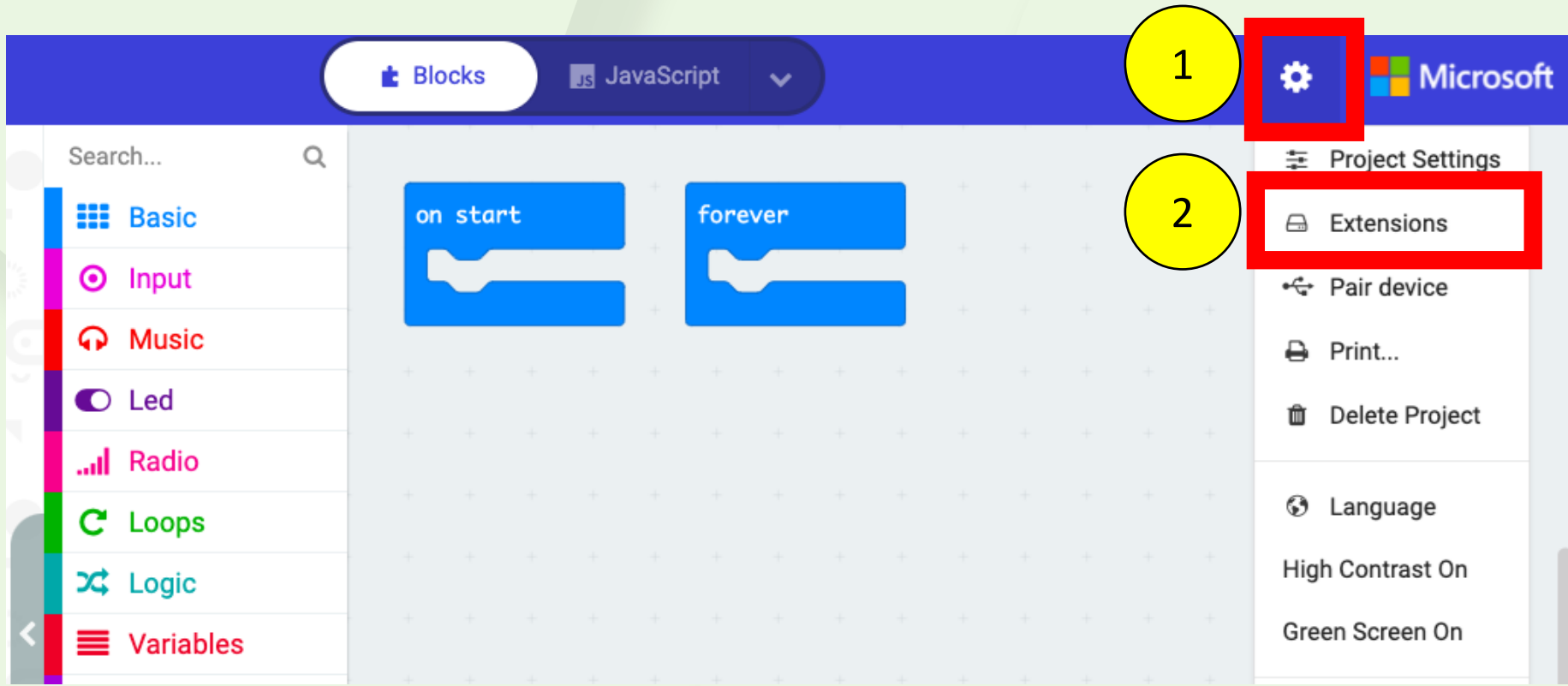


2. 輸入名稱 Face following

3. 按 



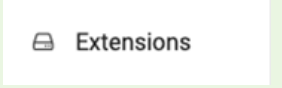
## 準備 CUHK-JC-iCar 模組

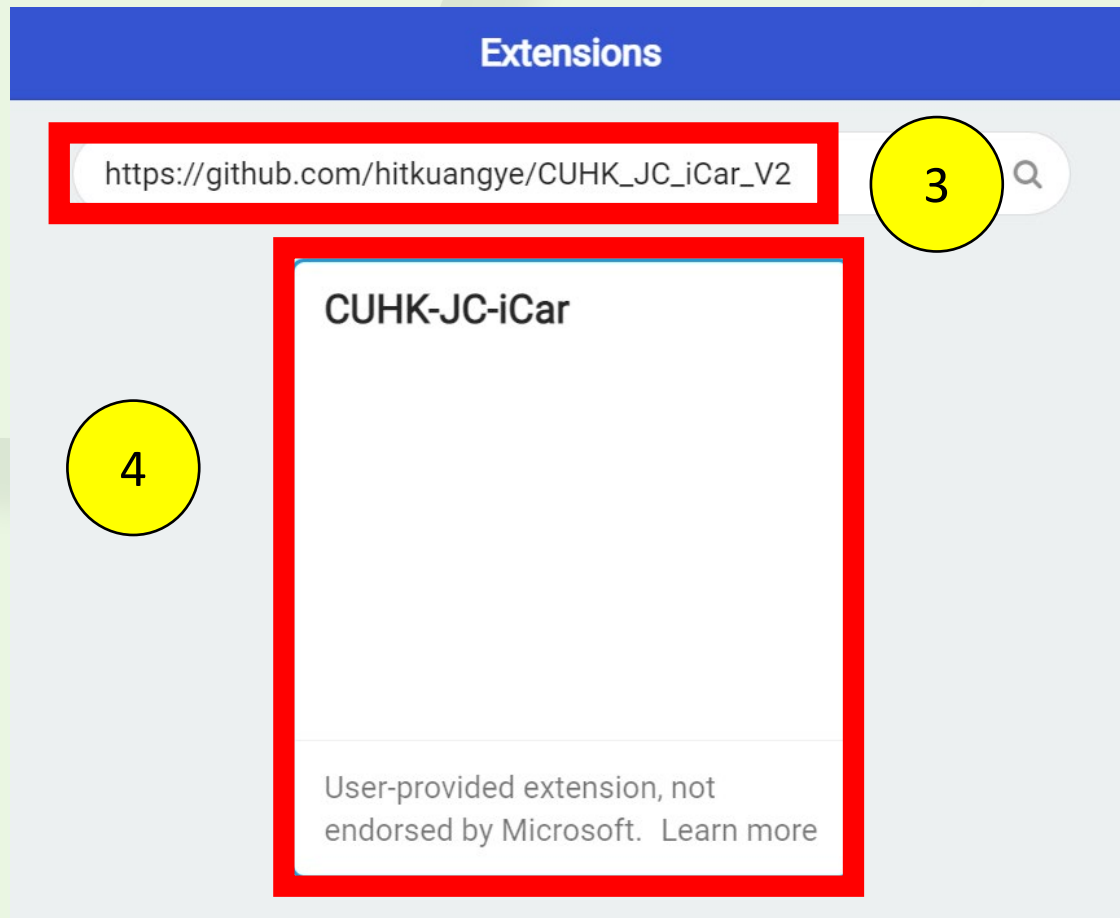


1. 按



2. 按

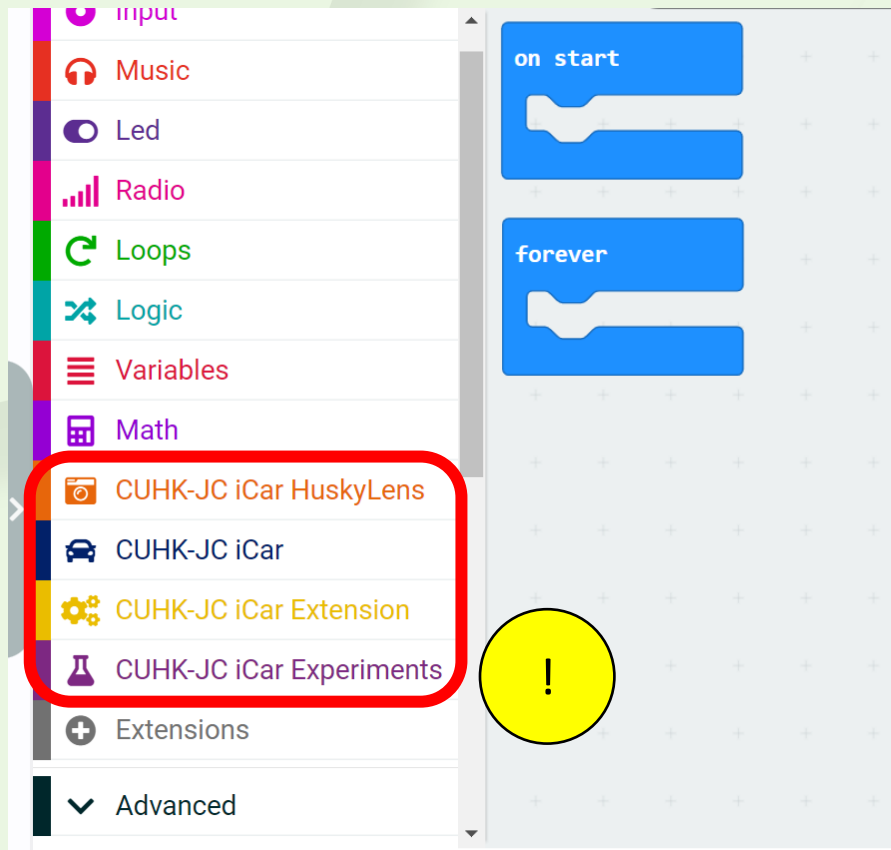




3. 貼上以下網址：

[https://github.com/hitkuangye/CUHK\\_JC\\_iCar\\_V2](https://github.com/hitkuangye/CUHK_JC_iCar_V2)

4. 按 CUHK-JC-iCar 模組



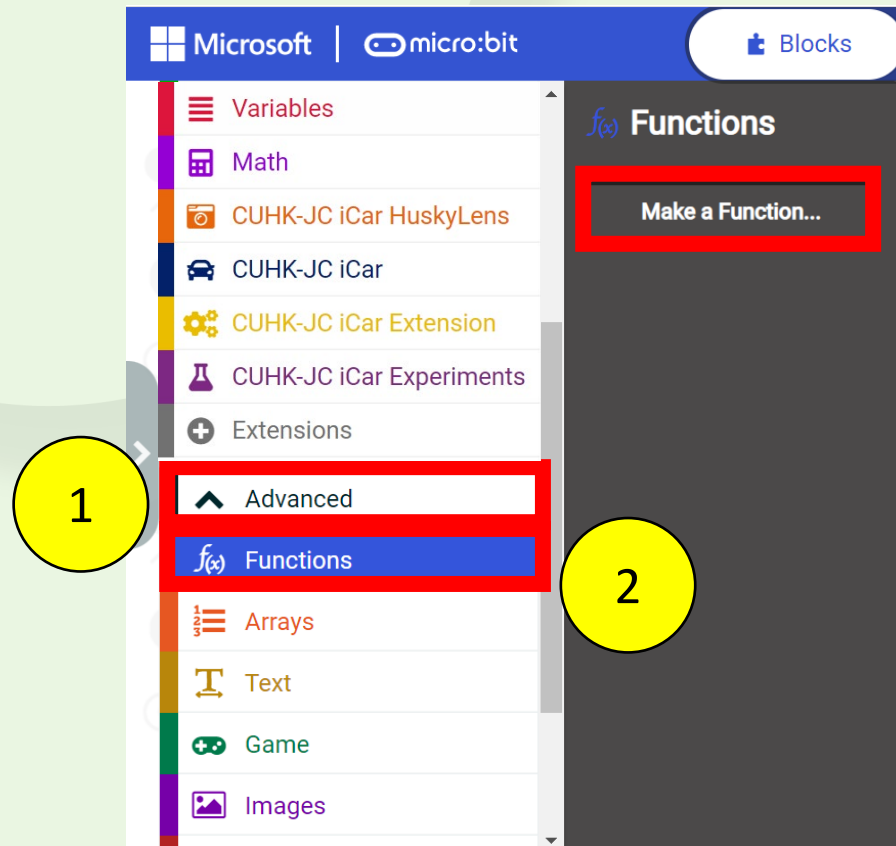
CUHK-JC-iCar 模組已加入到模組表


# micro:bit 基本入門



在編寫程式前，我們來認識一下 micro:bit 的模組吧！

# Functions 模組



1. 拉到模組表最底，按  Advanced

2. 按  Functions

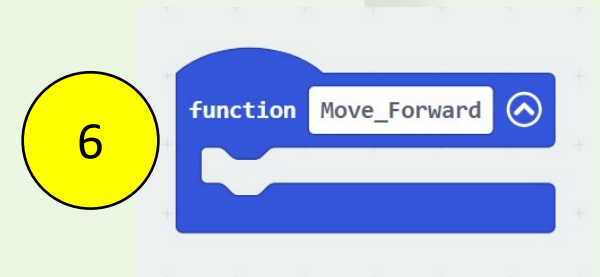
3. 按  Make a Function...



4. 改名為 Move\_Forward

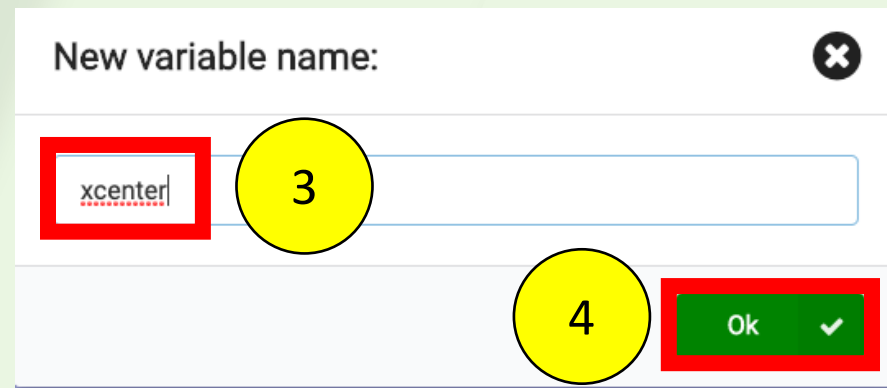
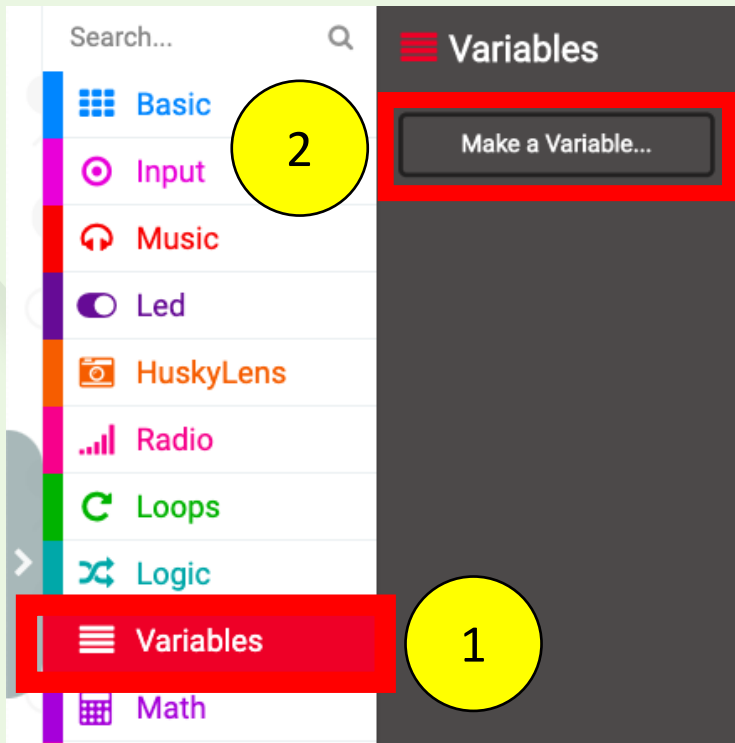
5. 按 


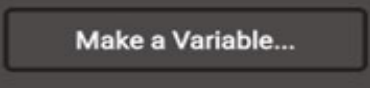
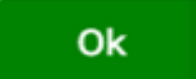
6. 視窗會出現一個 Move\_Forward 的 function



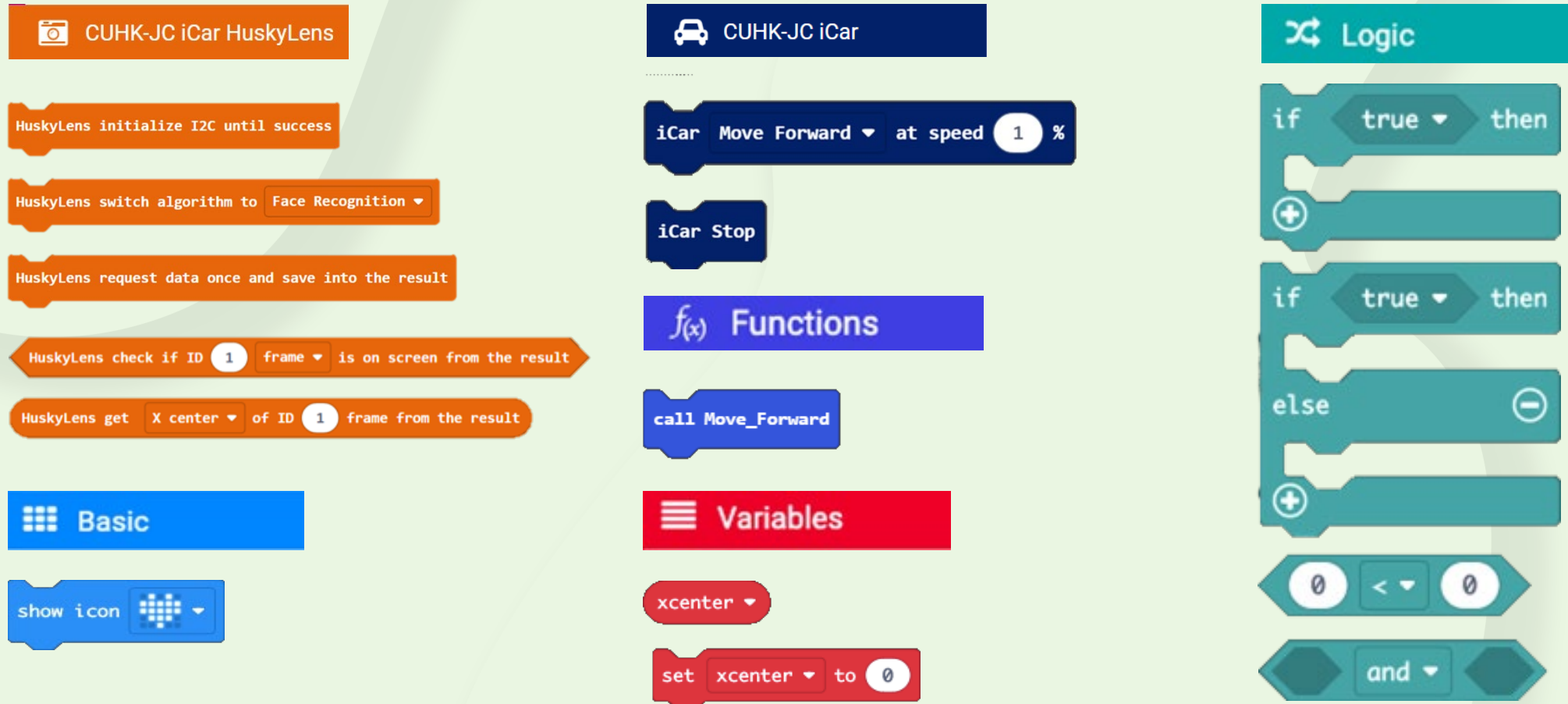


# Variables 模組



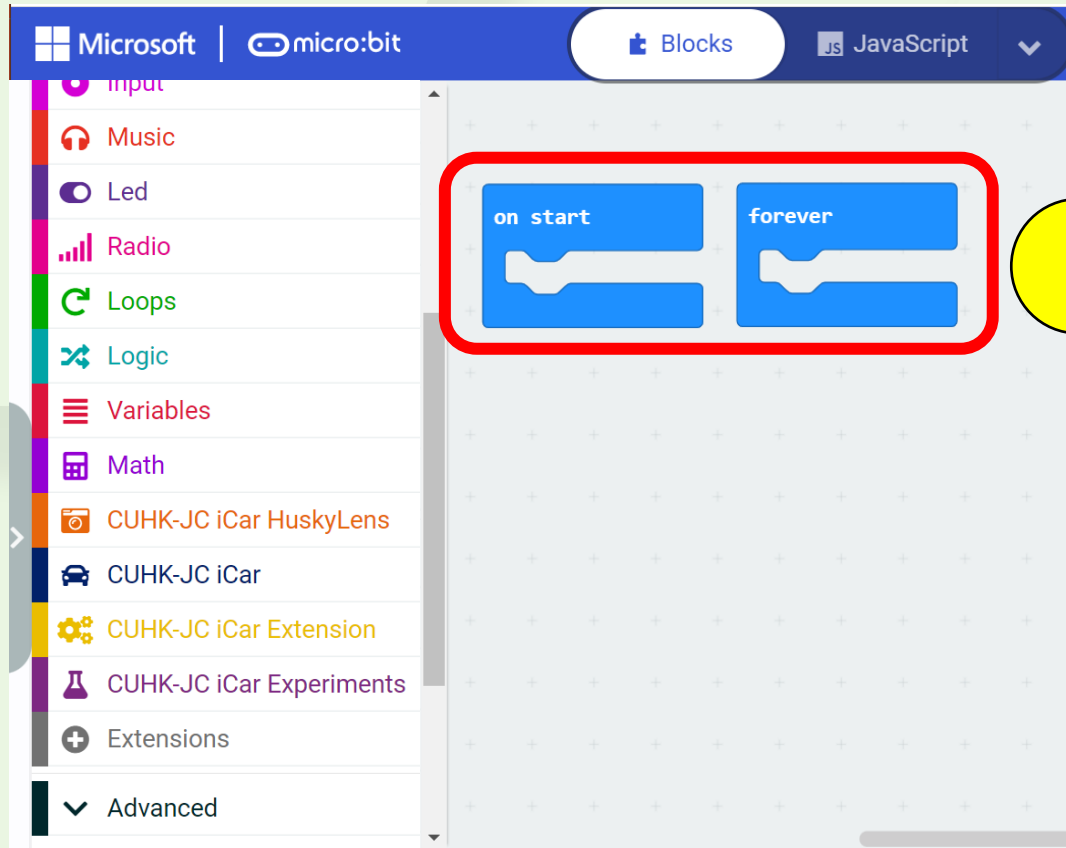
1. 按  Variables
2. 按 
3. 命名為 xcenter
4. 按 
5. 模組表會出現與 xcenter 相關的編程積木

# 其餘模組





開始編寫程式！

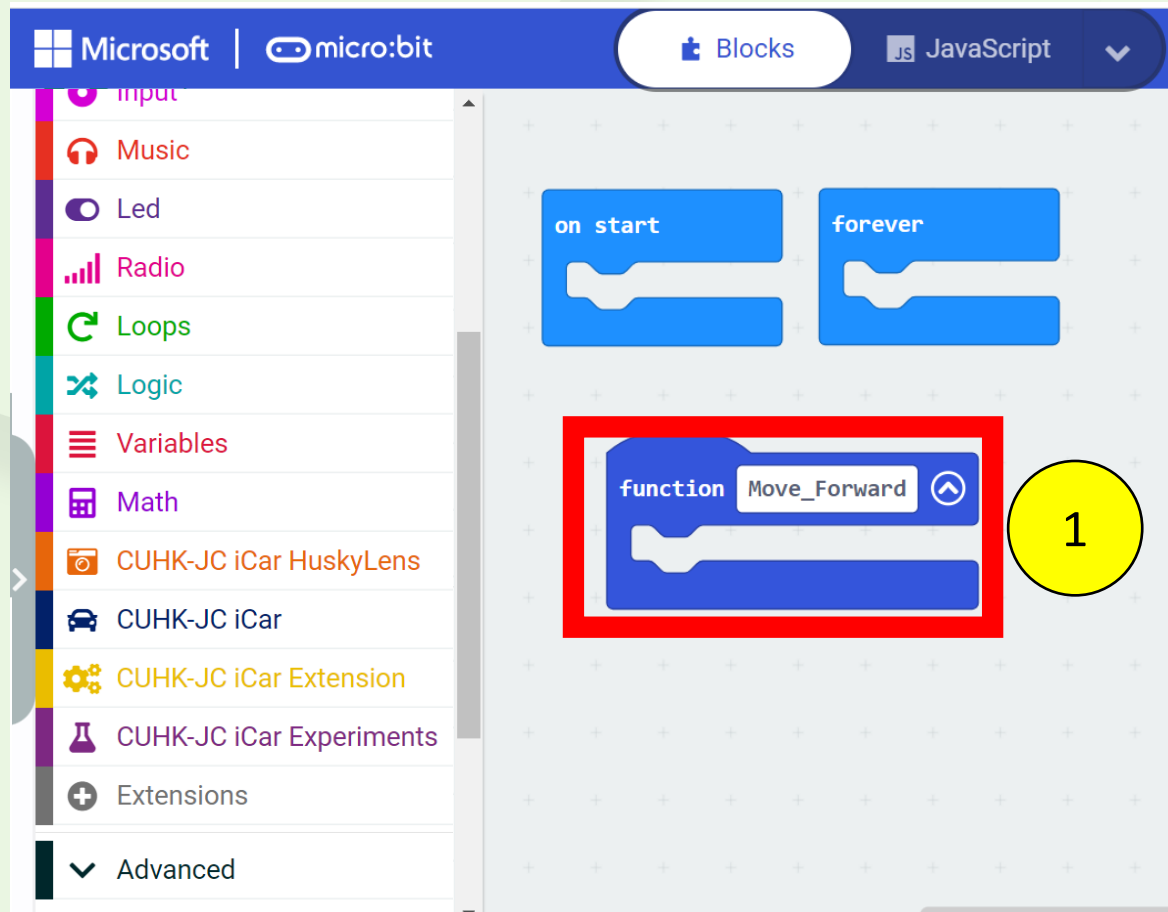


一開始，程式會預先設定  
on start 及 forever

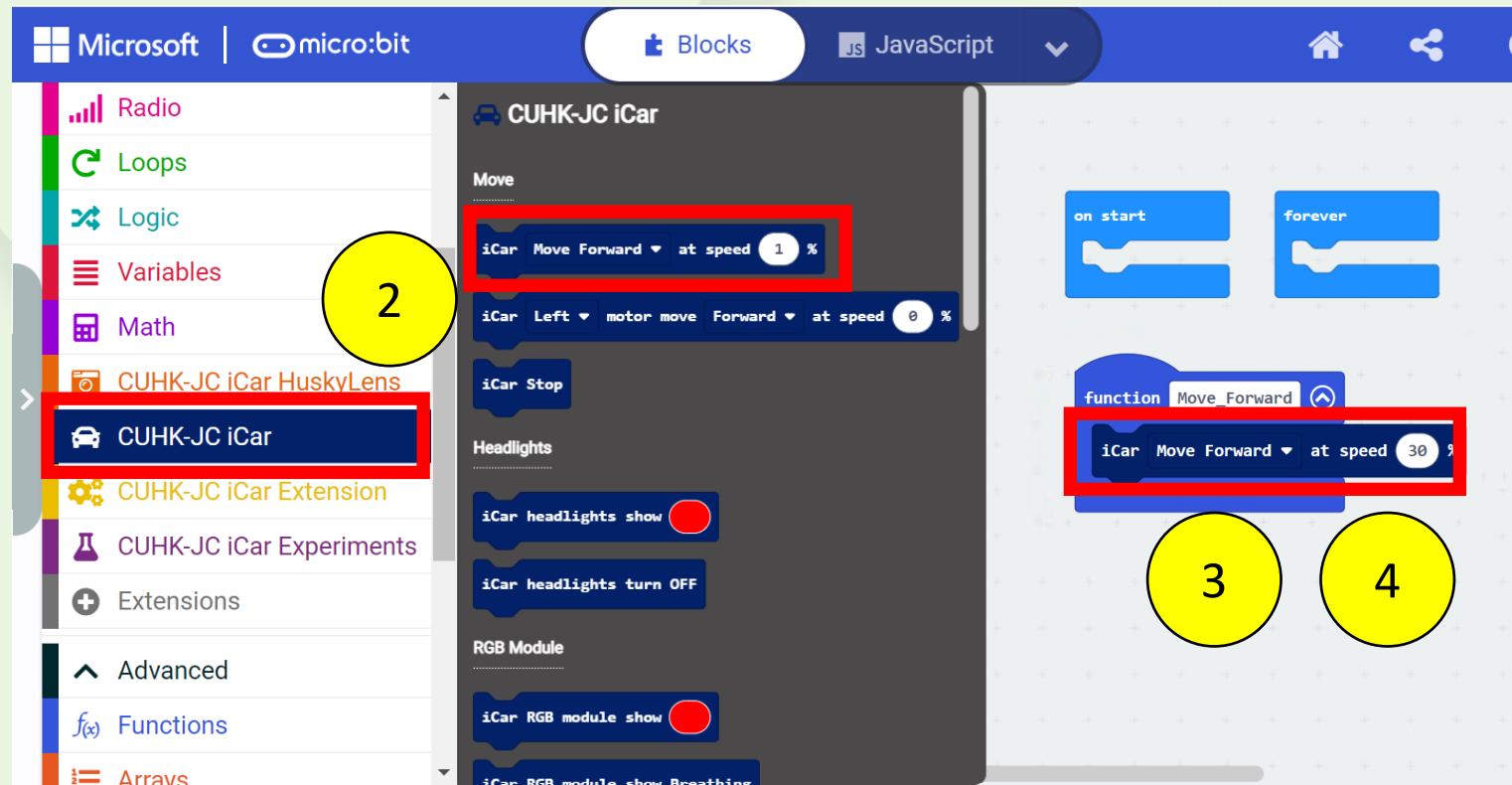
我們可以先不用理會，  
請不要刪除！



## 第一步：建立 Move\_Forward 的 function



### 1. 建立 Move\_Forward 的 function



2. 在 **CUHK-JC iCar** 模組中

拖拉 **iCar Move Forward at speed 1 %** 到 Move\_Foward 的 function 內

3. 在 **iCar Move Forward at speed 1 %**

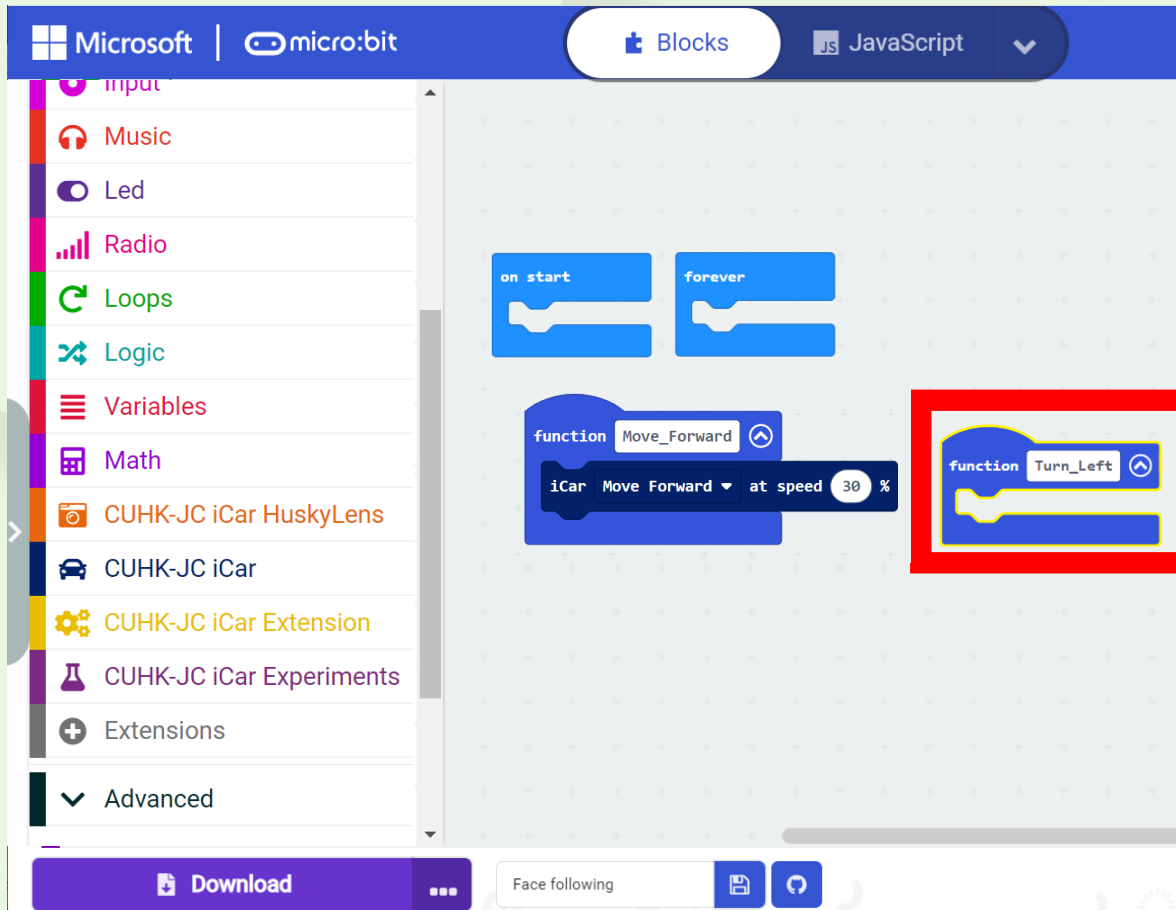
內選取 Move Forward

4. 在 **iCar Move Forward at speed 1 %**

speed 輸入 30



## 第二步：建立 Turn\_Left 的 function



### 1. 建立 Turn\_Left 的 function



2. 在 **CUHK-JC iCar** 模組中

拖拉 **iCar Move Forward at speed 1 %**  
到 Turn\_Left 的 function 內

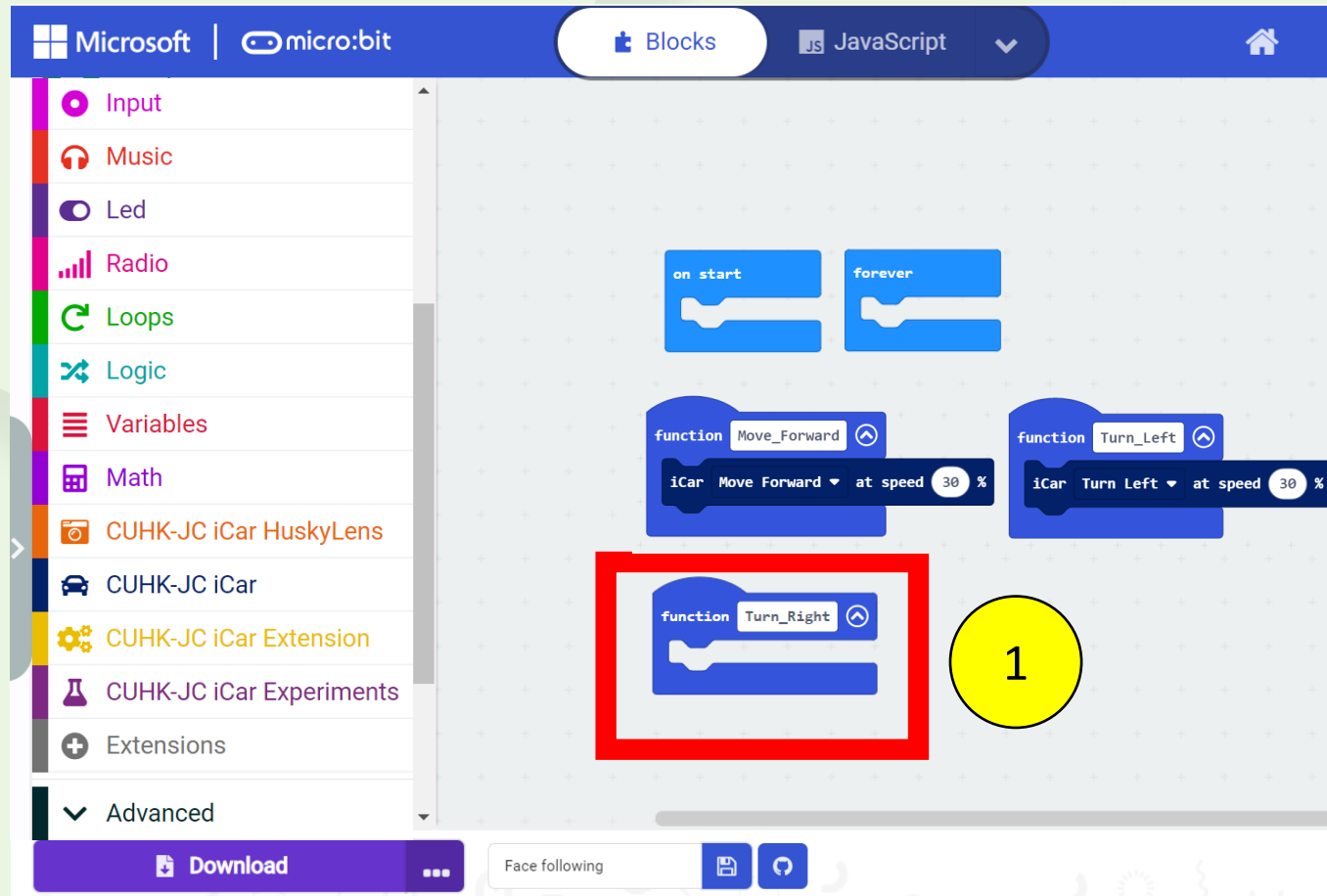
3. 在 **iCar Move Forward at speed 1 %**  
內選取 Turn Left

4. 在 **iCar Move Forward at speed 1 %**  
speed 輸入 30

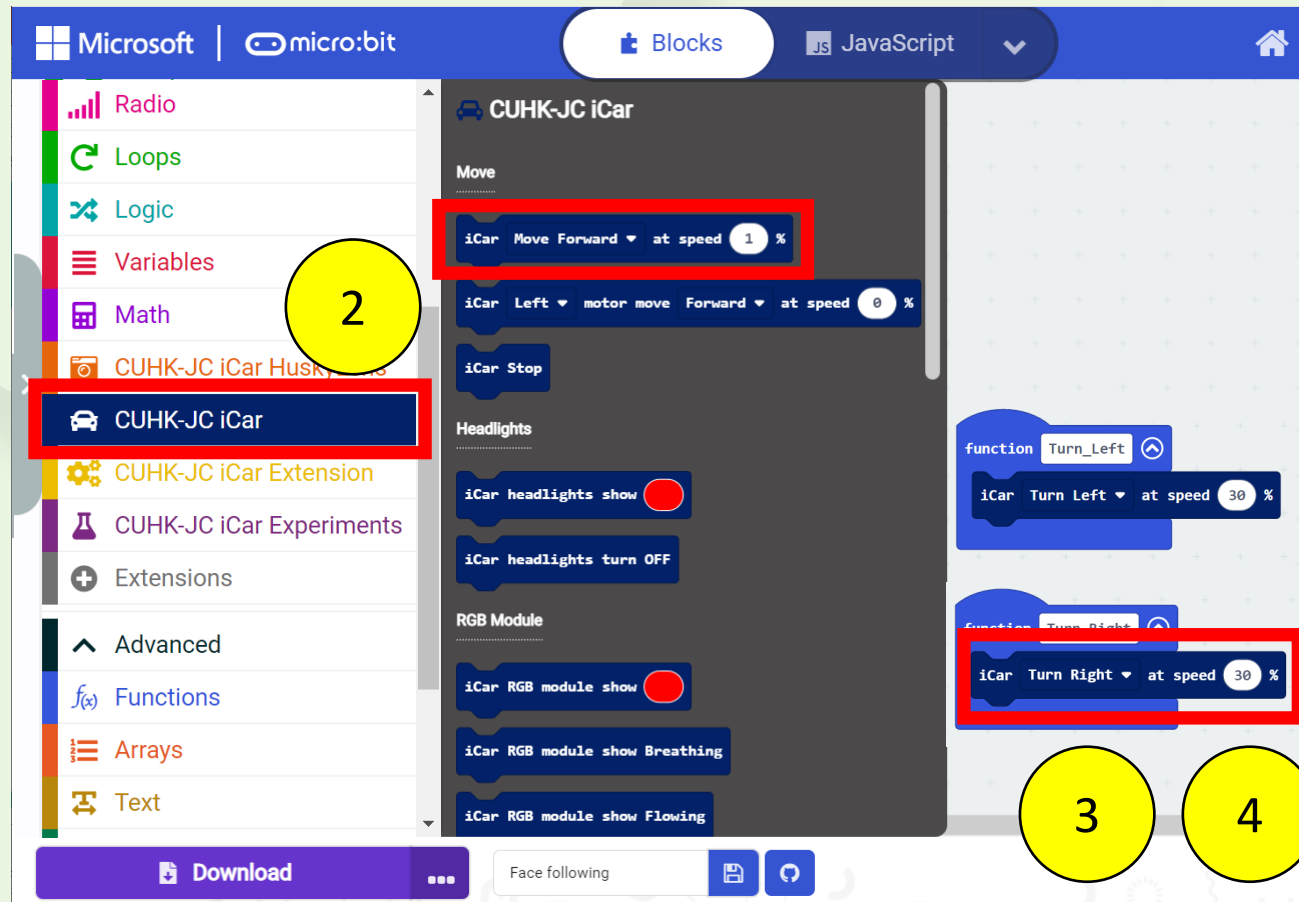




## 第三步：建立 Turn\_Right 的 function



### 1. 建立 Turn\_Right 的 function



2. 在 **CUHK-JC iCar** 模組中

拖拉 **iCar Move Forward** at speed 1 %  
到 Turn\_Right 的 function 內

3. 在 **iCar Move Forward** at speed 1 %

內選取 Turn Right

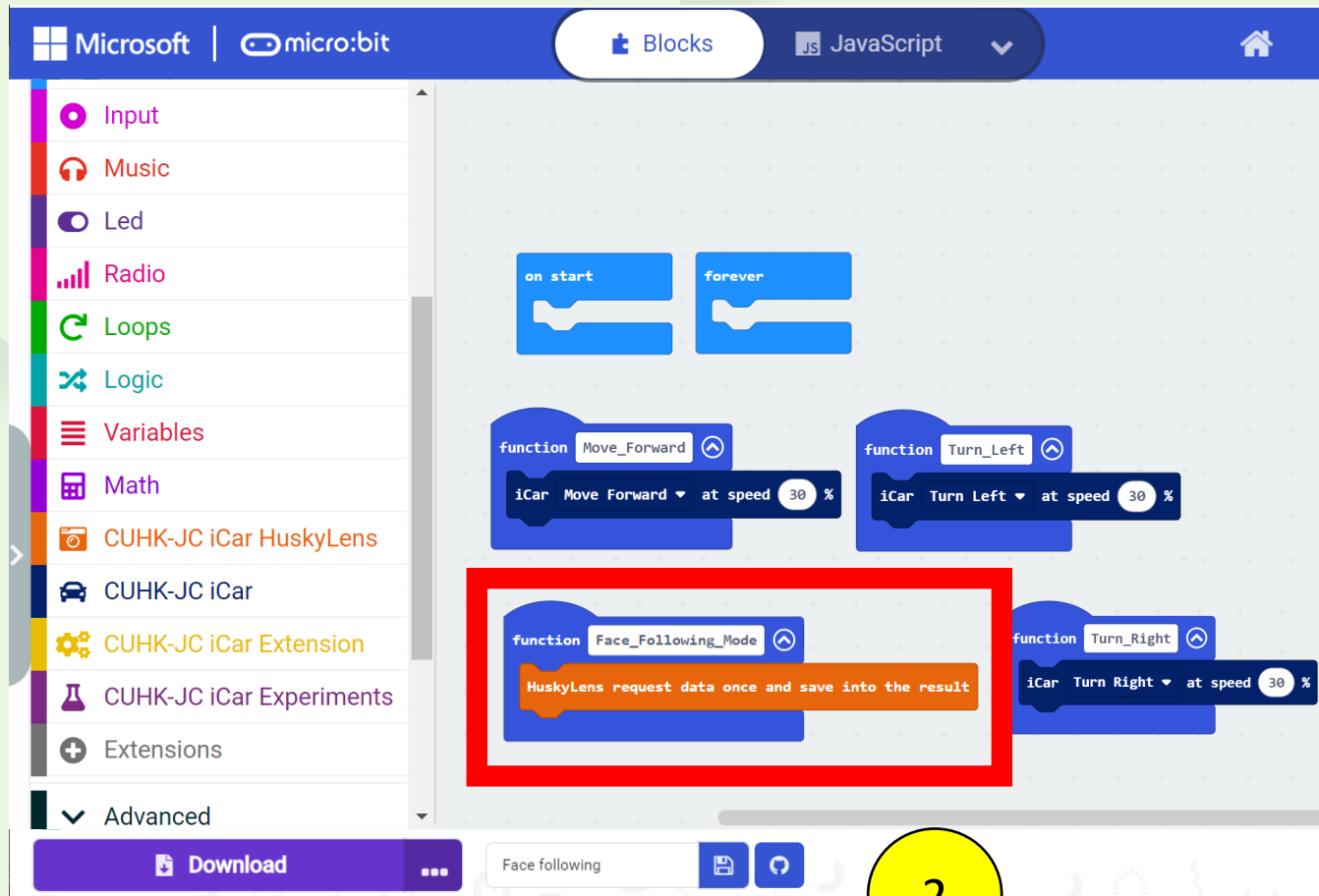
4. 在 **iCar Move Forward** at speed 1 %  
speed 輸入 30



## 第四步：建立 Face\_Following\_Mode 的 function



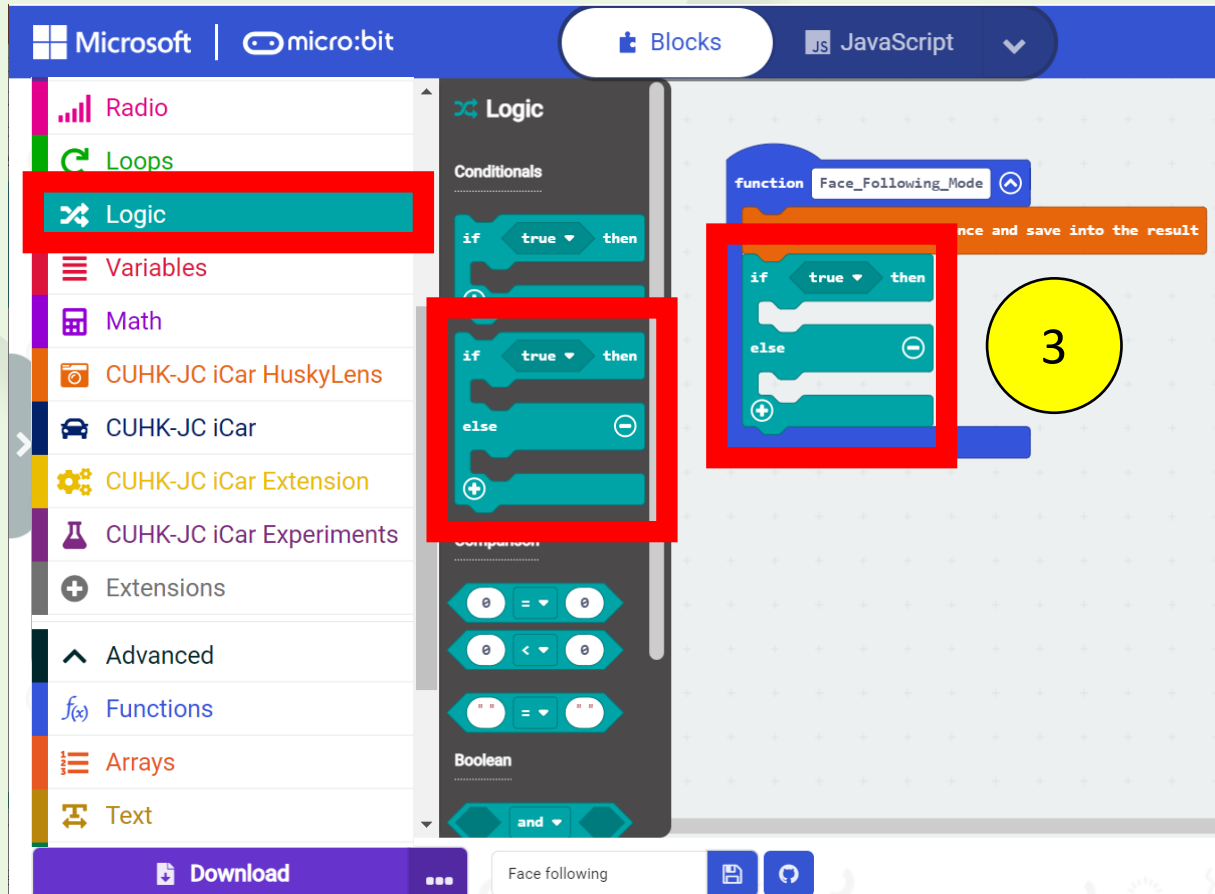
### 1. 建立 Face\_Following\_Mode 的 function



2. 在 CUHK-JC iCar HuskyLens 模組中拖拉

HuskyLens request data once and save into the result

到 Face\_Following\_Mode 的 function 內



3. 在 **Logic** 模組中拖拉



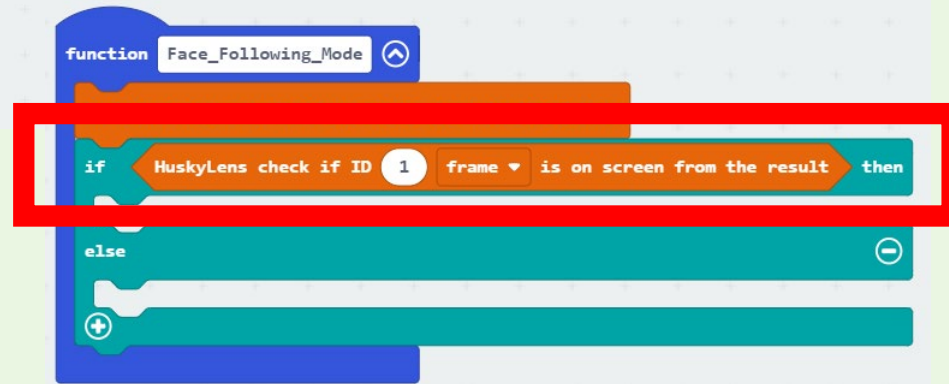
到 Face\_Following\_Mode 的 function 內

4. 在 CUHK-JC iCar HuskyLens 模組中拖拉

HuskyLens check if ID 1 frame is on screen from the result

到 if then else 內

4



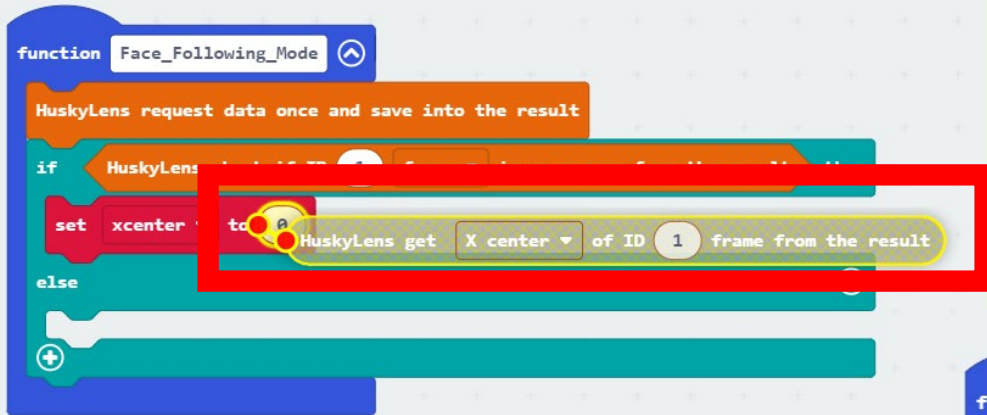


5. 在 **Variables** 模組中拖拉



到 if then else 內



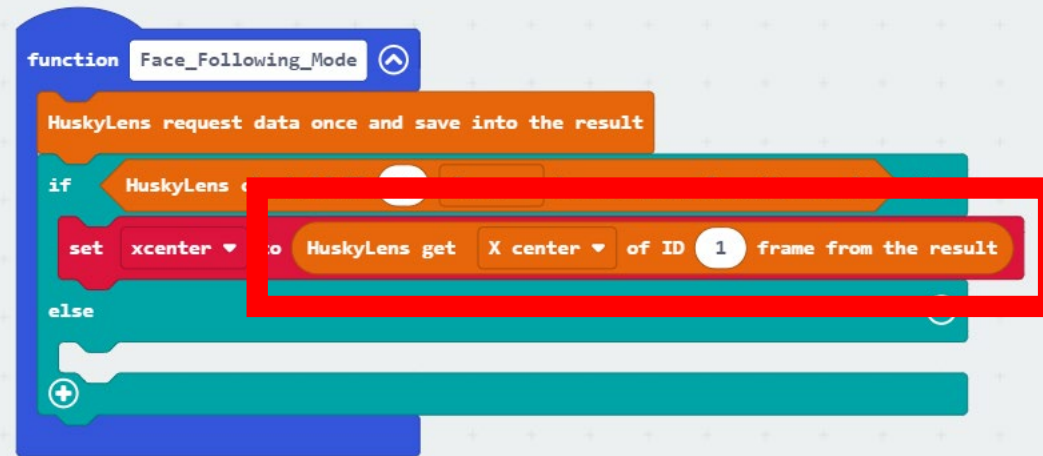


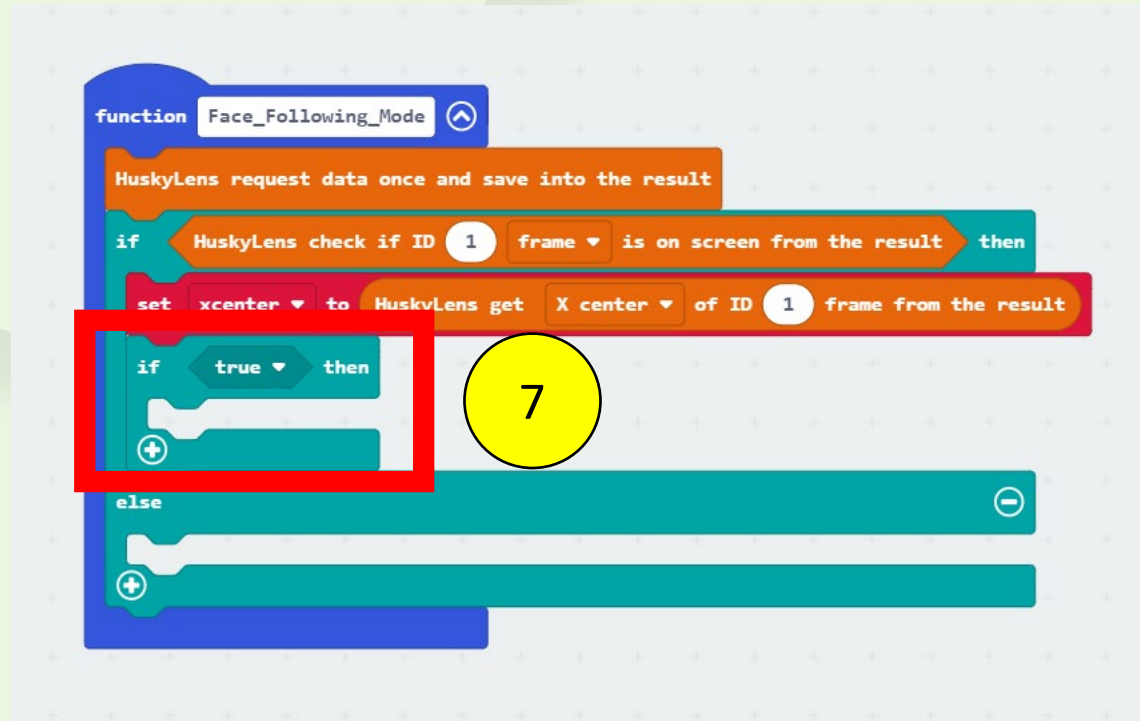
6

6. 在 CUHK-JC iCar HuskyLens 模組中拖拉

HuskyLens get X center of ID 1 frame from the result

到 set xcenter to 0 內

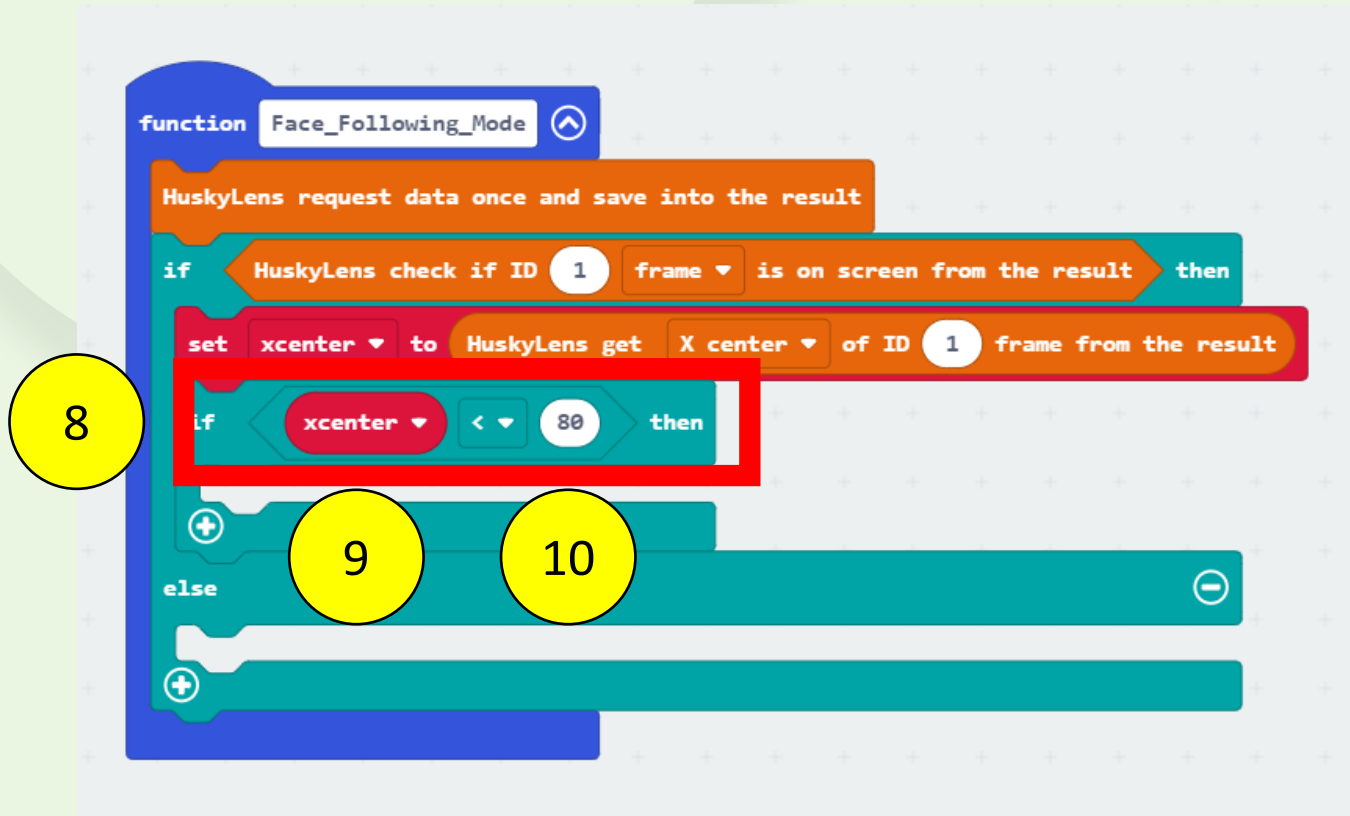




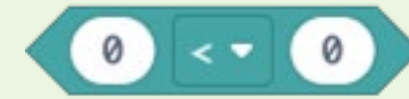
7. 在 Logic 模組中拖拉



到 if then else 內



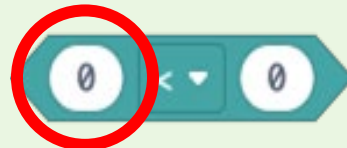
8. 在 **Logic** 模組中拖拉

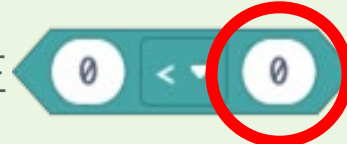


到 if then 內

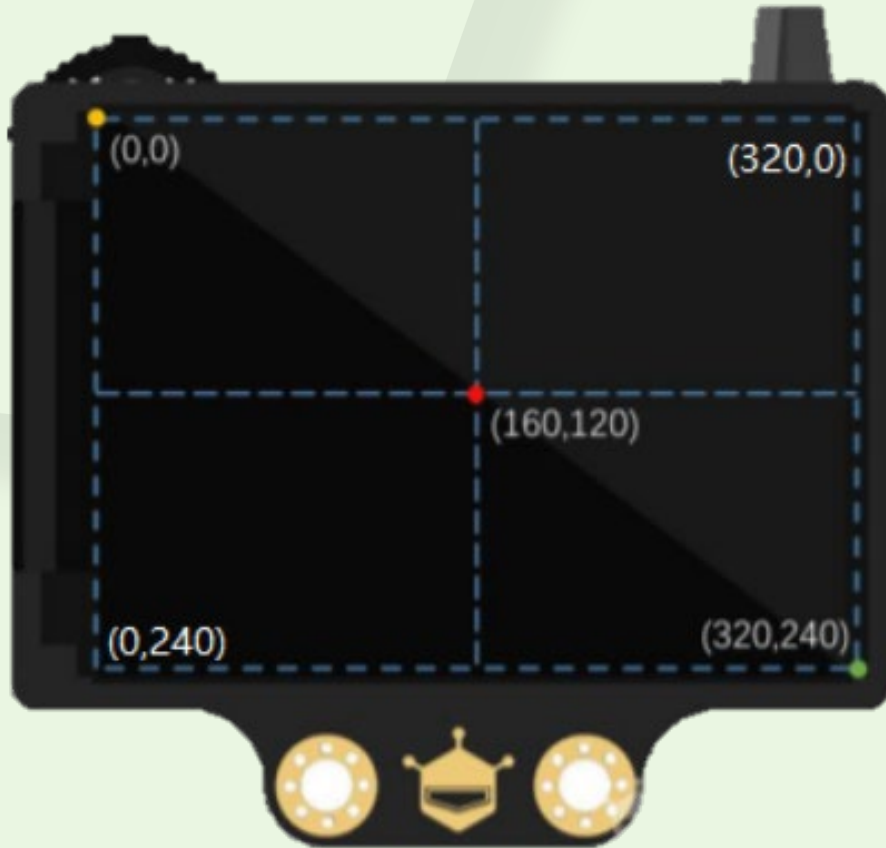
9. 在 **Variables** 模組中拖拉



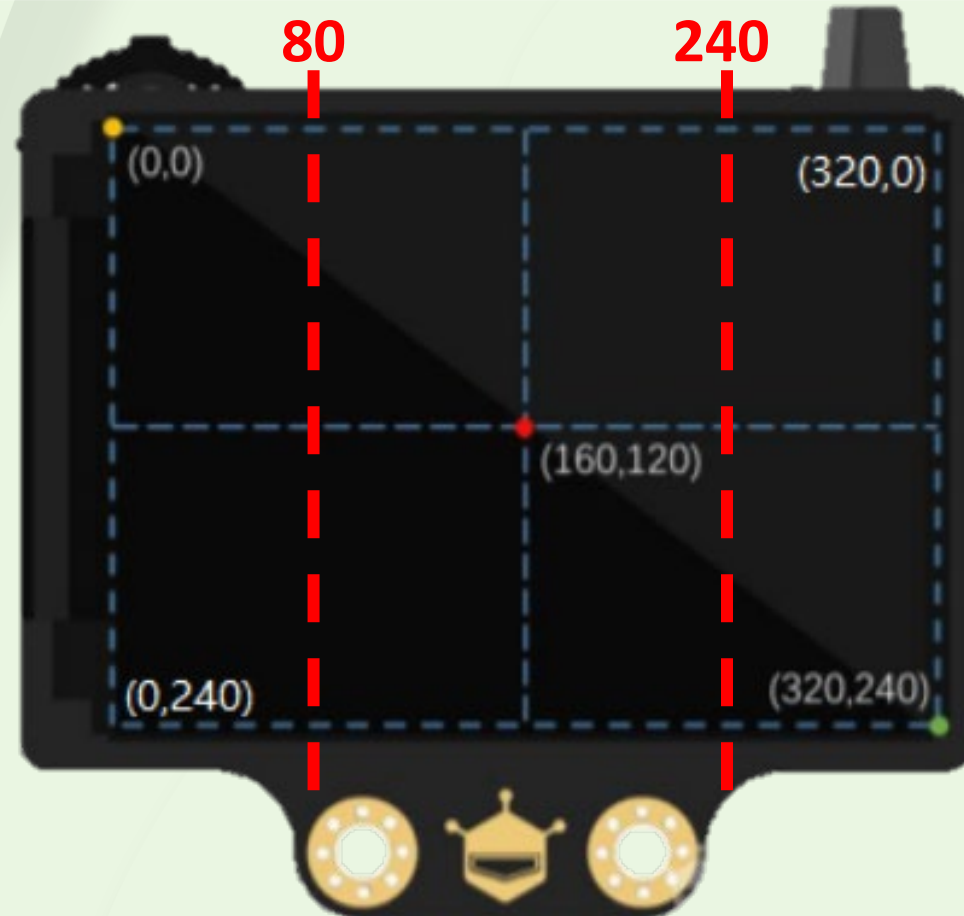
到  內

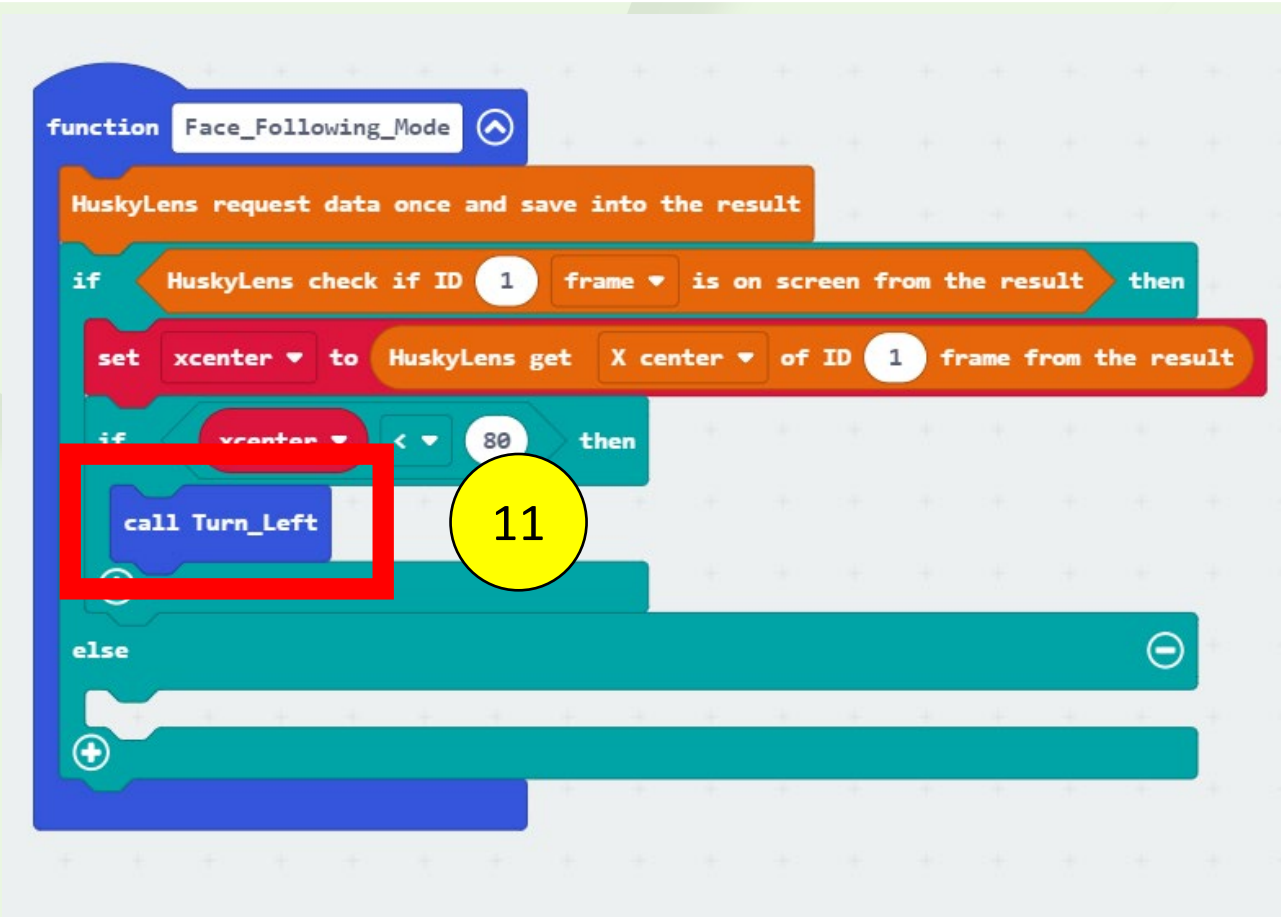
10. 在  輸入數值為 80

# 備註：HuskyLens 的坐標



- 當 HuskyLens 識別到物件時，目標的坐標會被輸出
- 形式： $(x, y)$

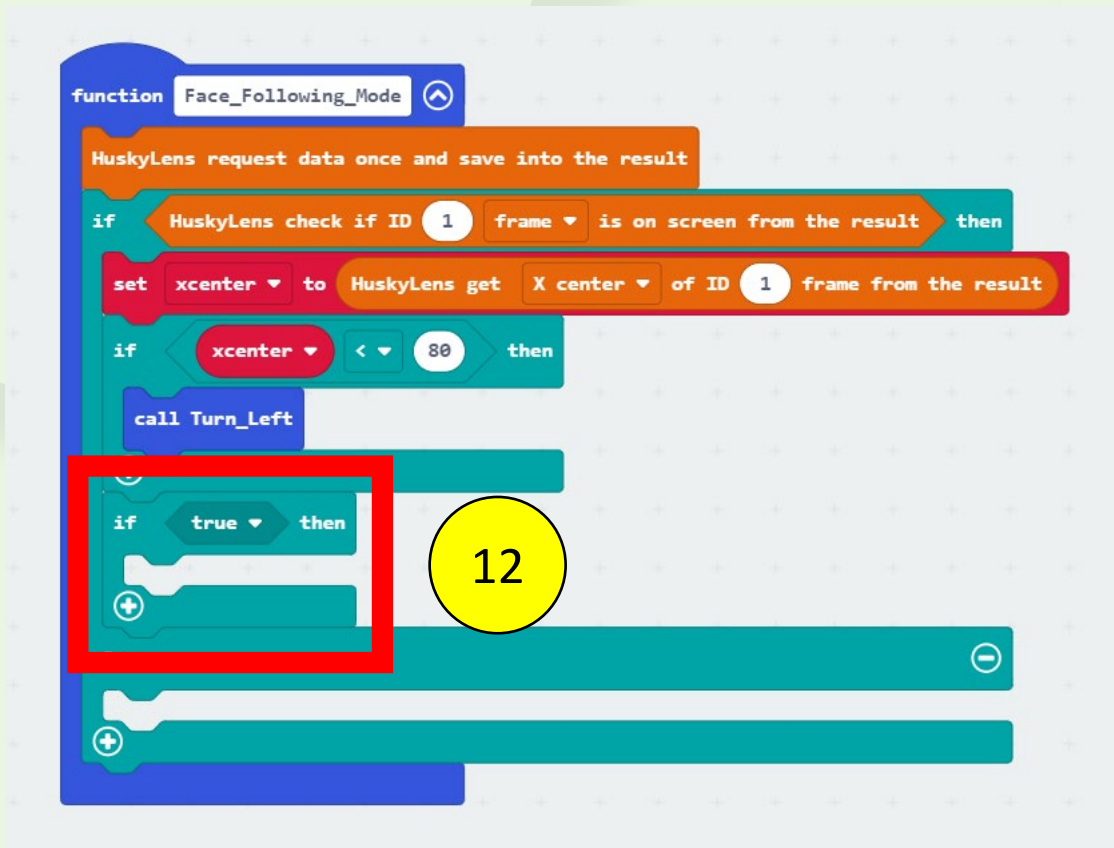




11. 在  $f(x)$  Functions 模組中拖拉



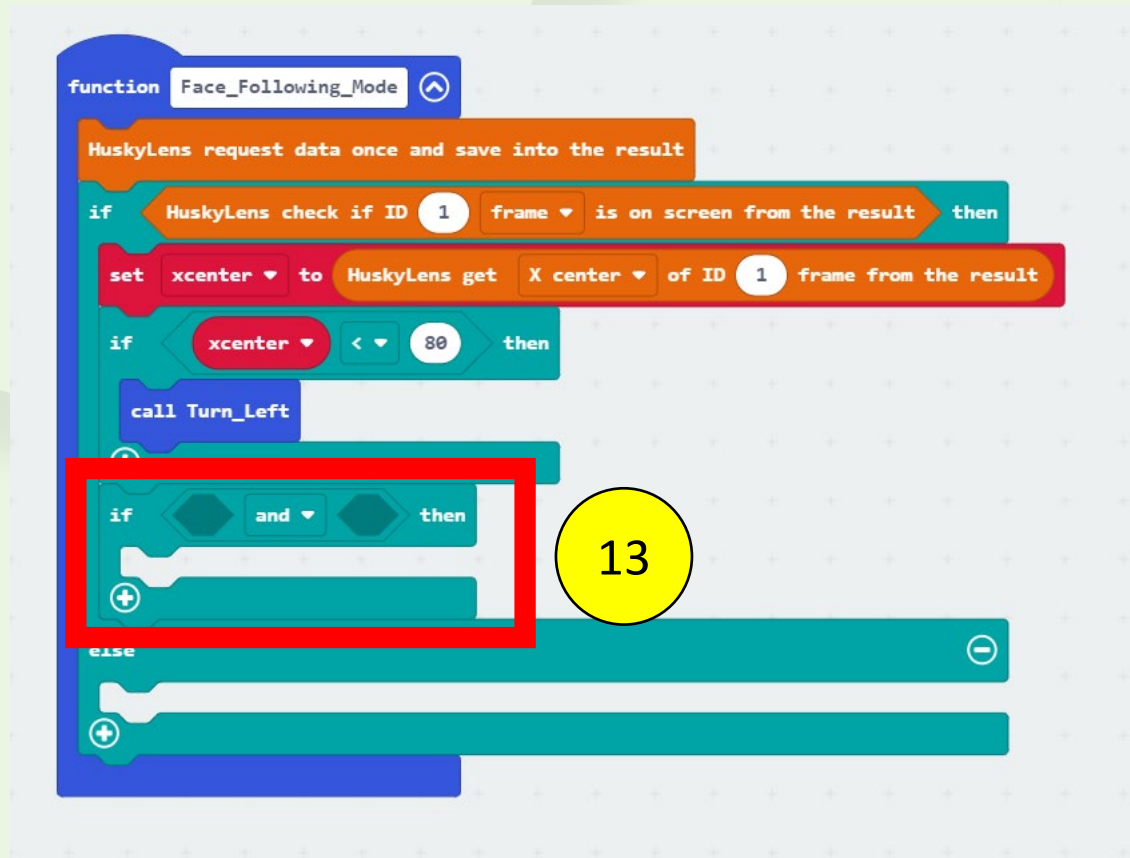
到 if then 內



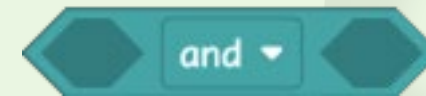
12. 在 Logic 模組中拖拉



到 if then else 內

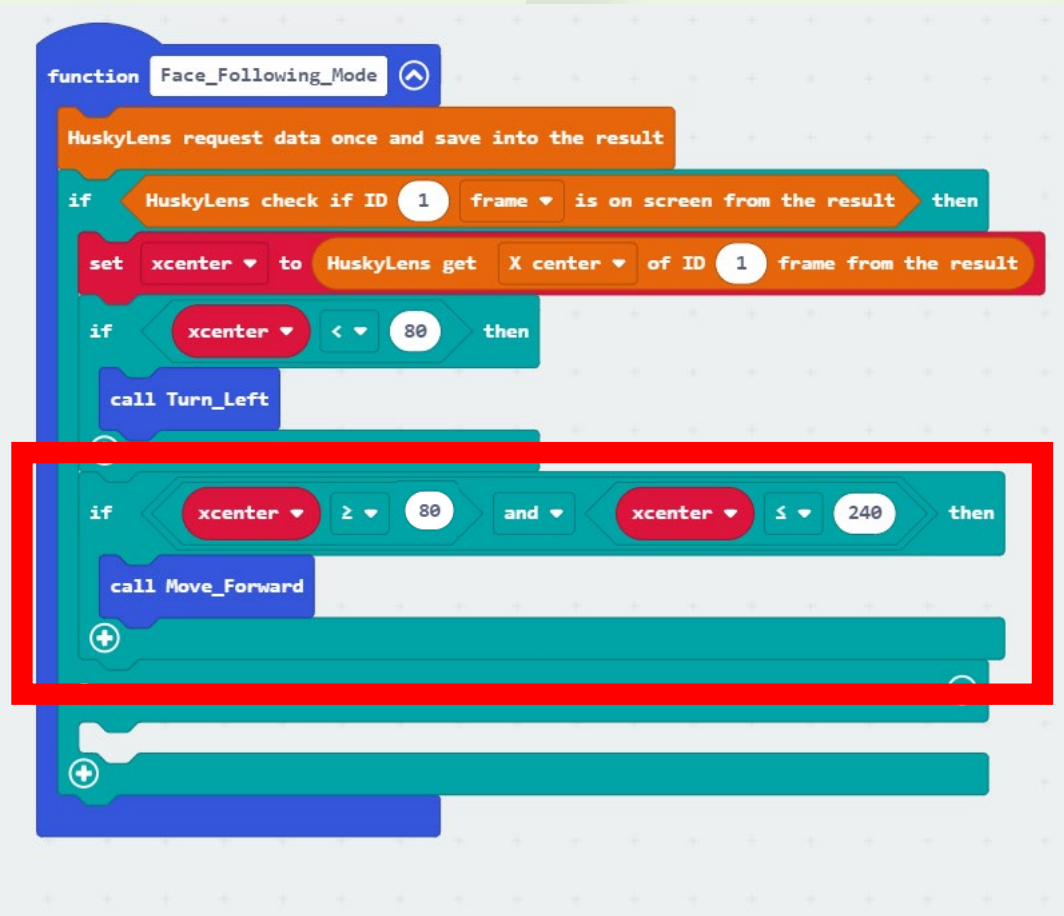


13. 在 Logic 模組中拖拉



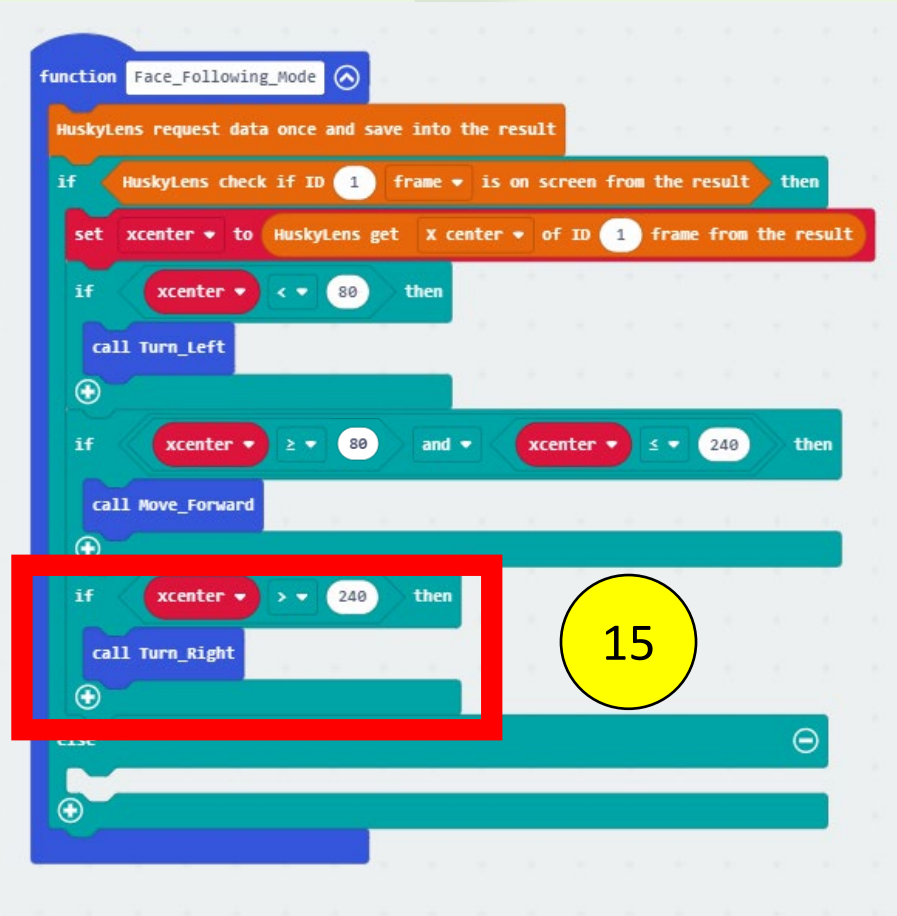
到 if then 內



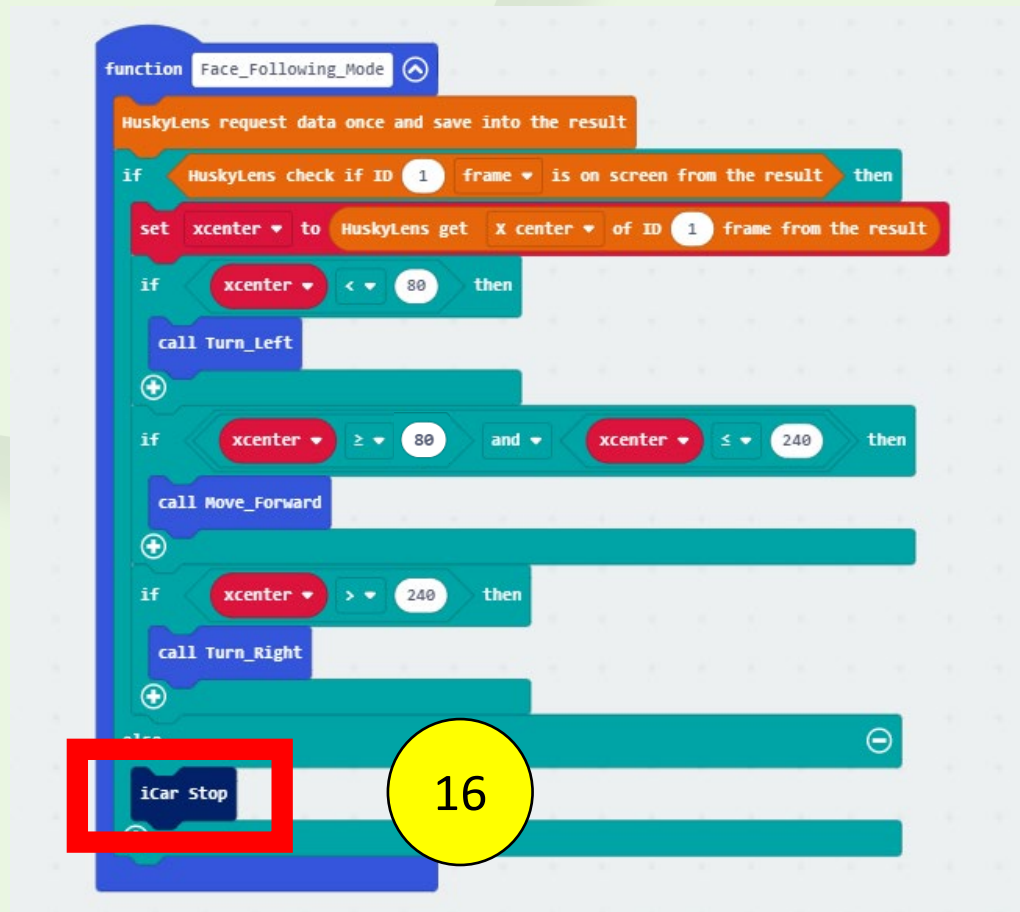



14. 設定如圖示

14



15. 設定如圖示



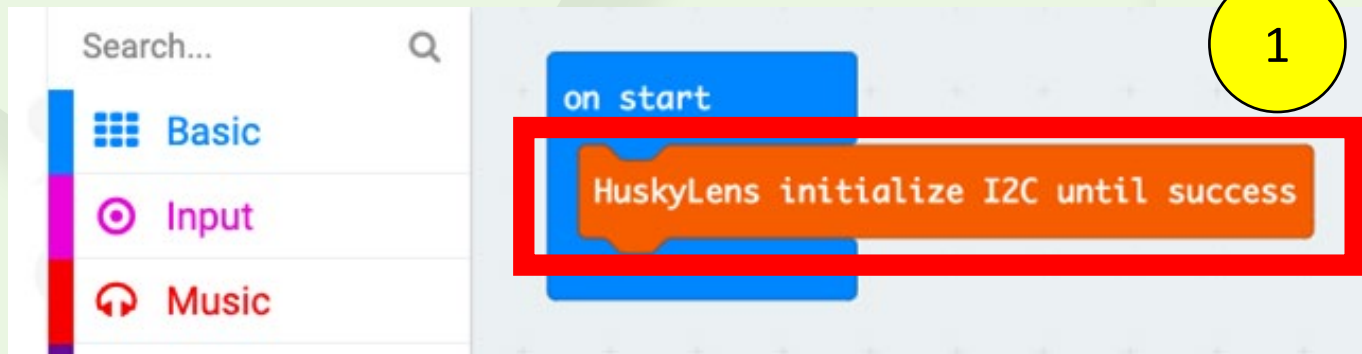
16. 在  CUHK-JC iCar 模組中拖拉



到 if then else 內



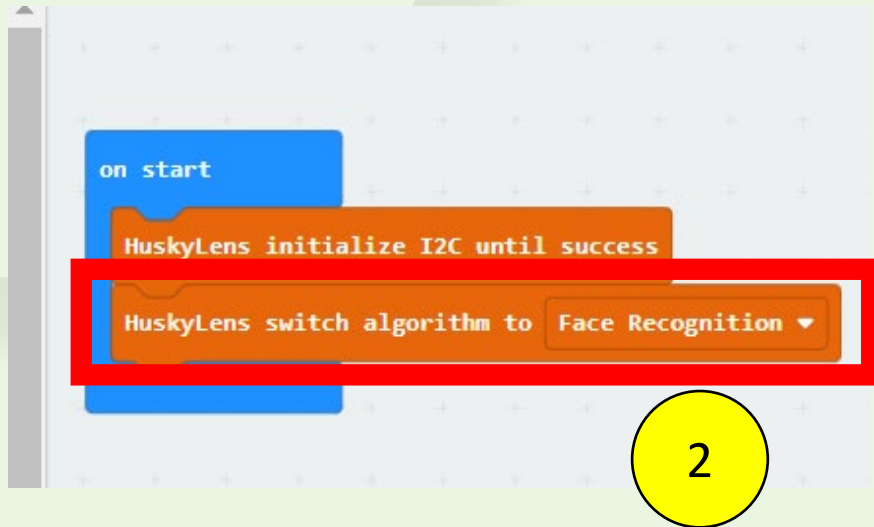
## 第五步：設定 on start



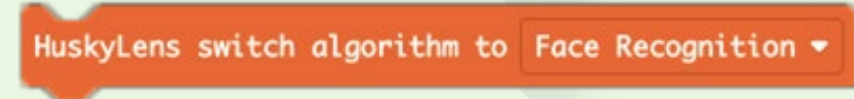
1. 在  CUHK-JC iCar HuskyLens 模組中拖拉

HuskyLens initialize I2C until success

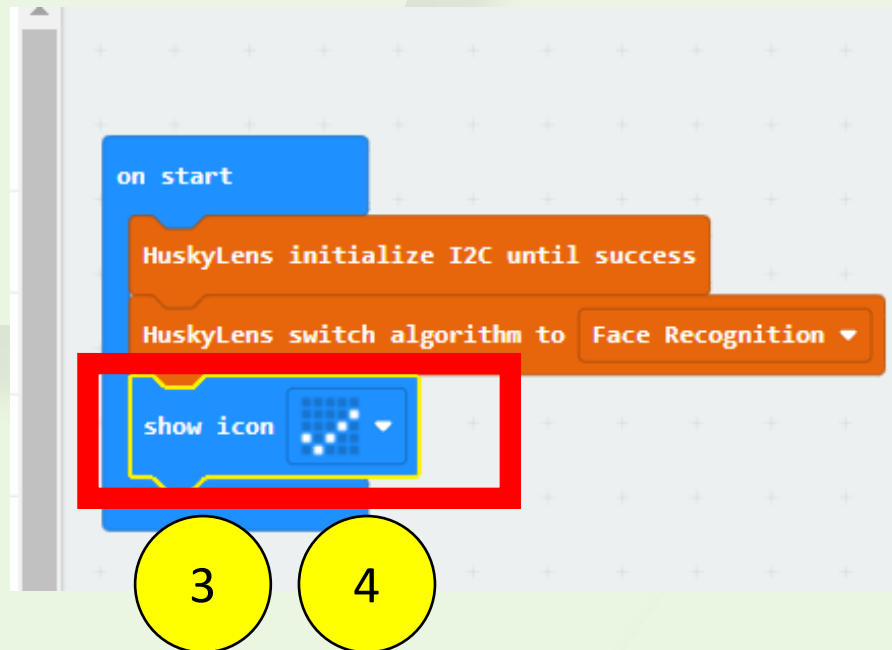
到 on start 內



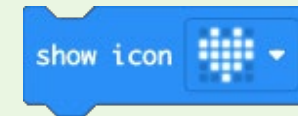
2. 在  CUHK-JC iCar HuskyLens 模組中拖拉



到 on start 內



3. 在 Basic 模組中拖拉



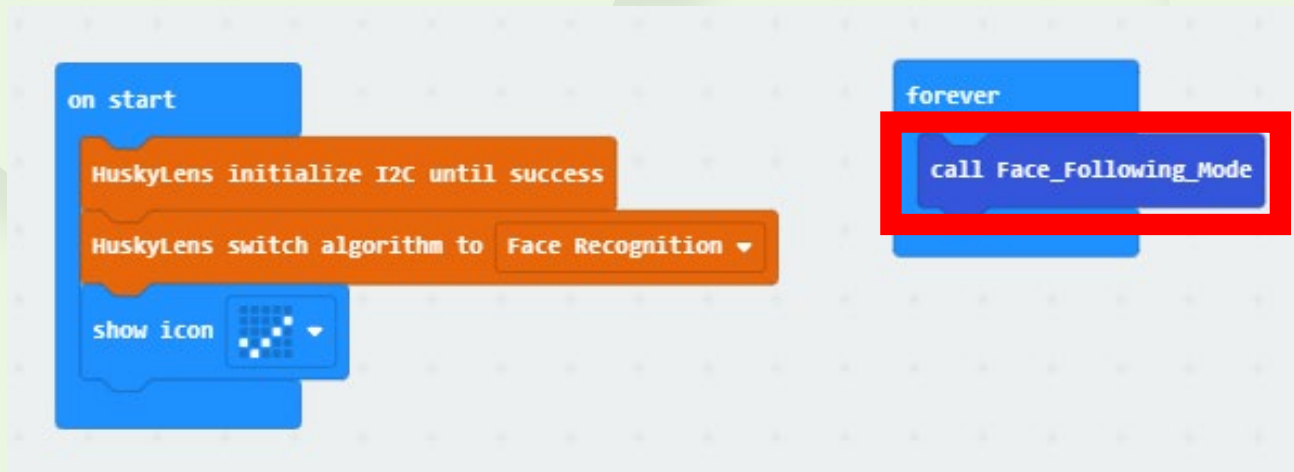
到 on start 內

4. 選取





## 第六步：設定 forever



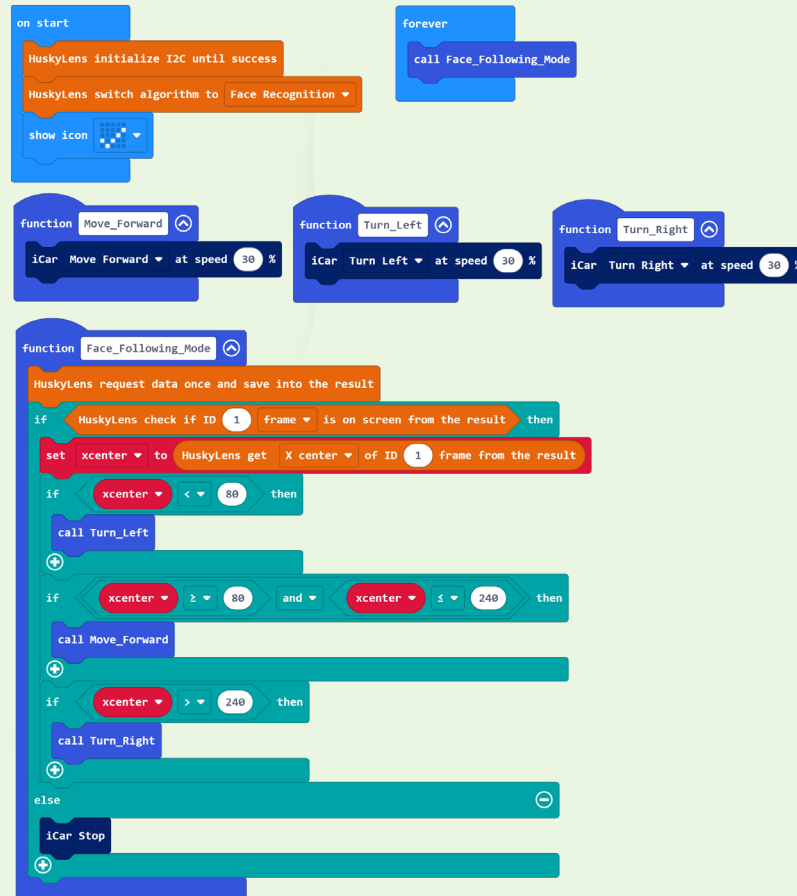
1. 在  $f(x)$  Functions 模組中拖拉



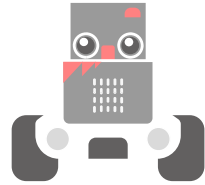
到 forever 內



完成編寫程式！





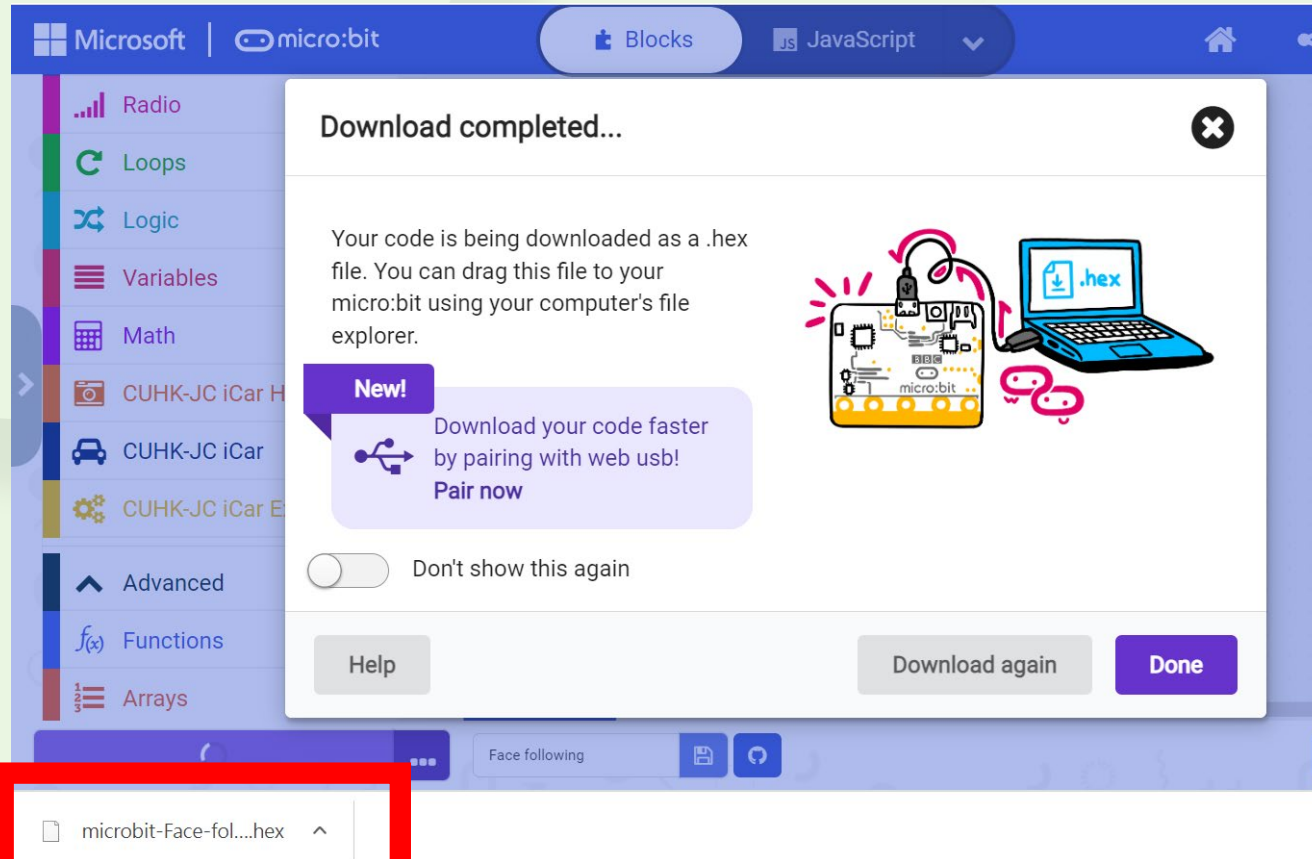


## 第七步：下載程式到 CUHK iCar

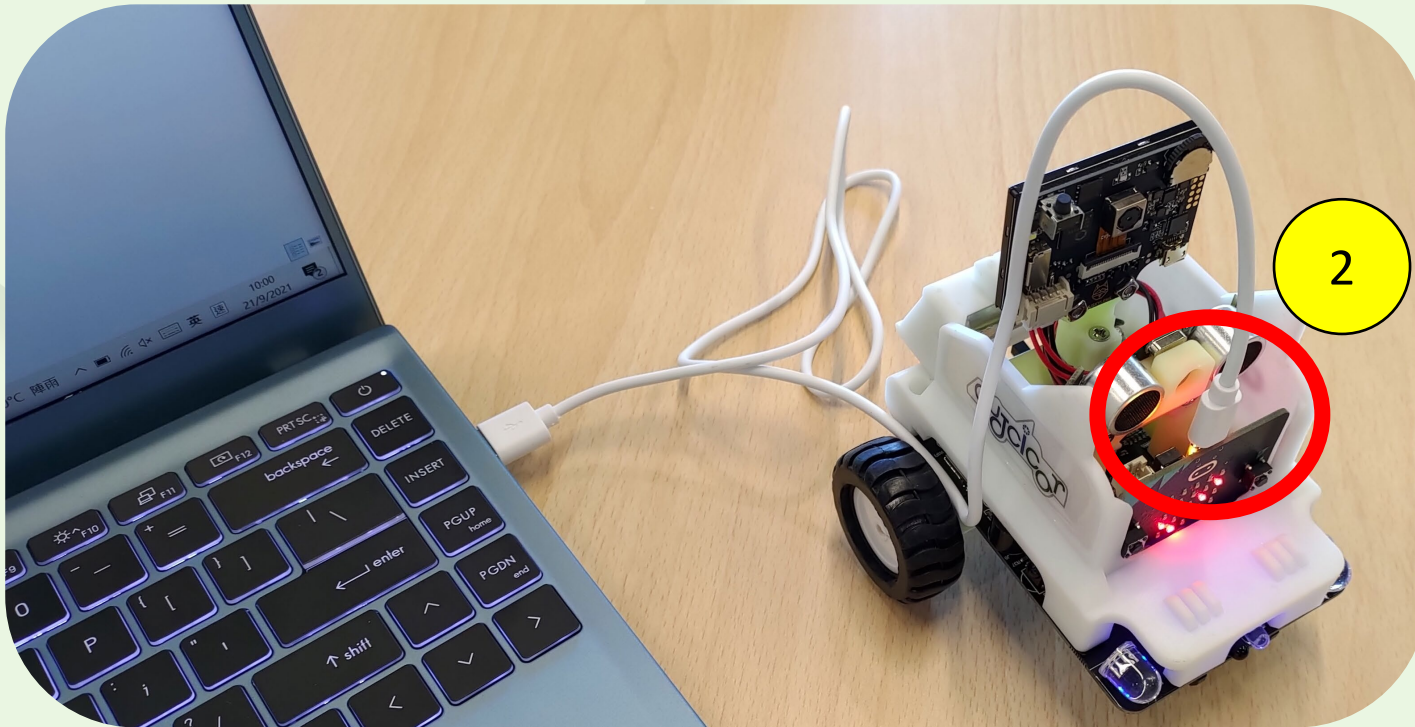
The screenshot shows the Microsoft micro:bit Blocks editor interface. On the left, the 'Basic' category is selected in the sidebar. The main workspace contains a script starting with 'on start' and 'forever' loops. The 'forever' loop contains a 'call Face\_Following\_Mode' block. Below this, there are three function blocks: 'Move\_Foreward', 'Turn\_Left', and 'Turn\_Right'. The 'Face\_Following\_Mode' function is expanded, showing a sequence of blocks: 'HuskyLens request data once and save into the result', an 'if' statement checking if ID 1 is on screen, 'set xcenter' to the X center of ID 1, and conditional logic to call 'Turn\_Left', 'Move\_Foreward', or 'Turn\_Right' based on the xcenter value. At the bottom of the workspace, a 'Download' button is highlighted with a red box. A yellow circle with the number '1' is placed next to it.

1. 按

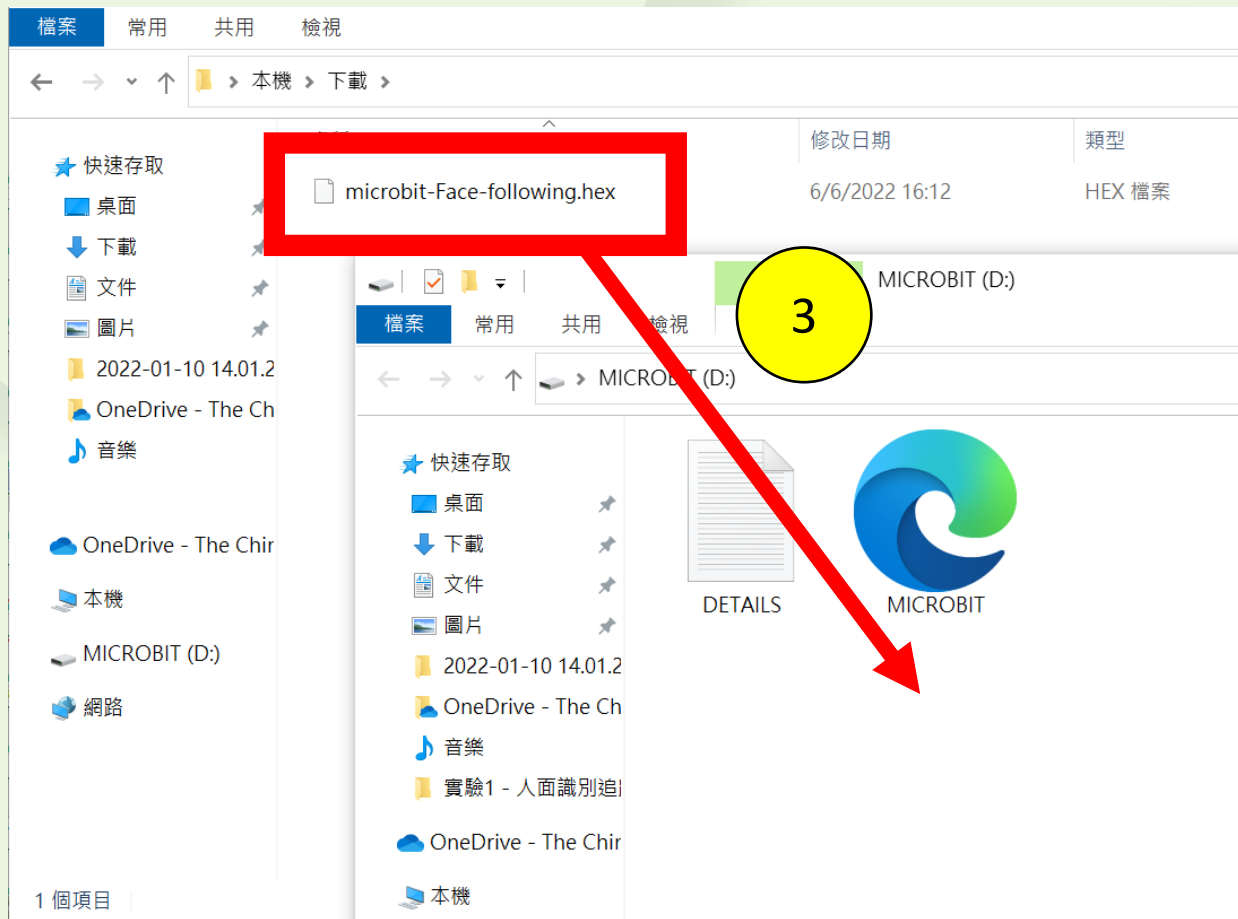
Download



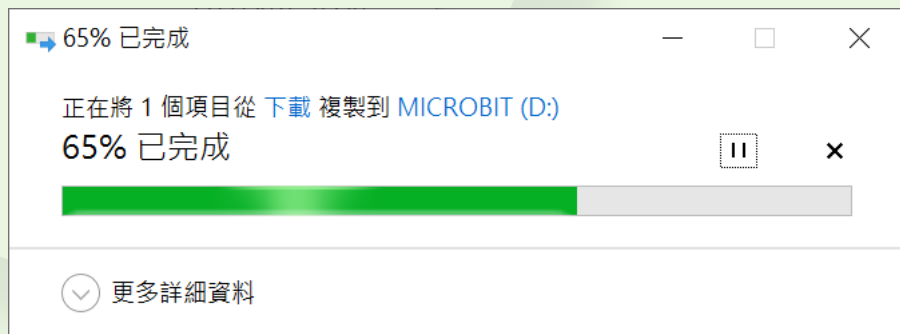
已下載程式的 hex 檔！



2. 使用 micro USB 線連接  
micro:bit 到電腦



3. 將已下載的程式拖拉到 micro:bit 內



#### 4. 等待複製完成

##### 注意：

- micro:bit 視窗有機會在複製完成後消失
- 複製完成後，micro:bit 視窗內不會顯示程式的 hex 檔

#### 5. 中斷 micro:bit 的連接

# 如不能連接 CUHK iCar 到電腦：

- 重新啟動電腦
- 嘗試另一個 USB 埠
- 換 micro USB 線

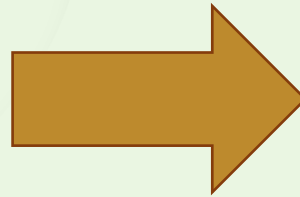


# 現在你已經完成 程式編寫及燒錄

可以進行實驗了！

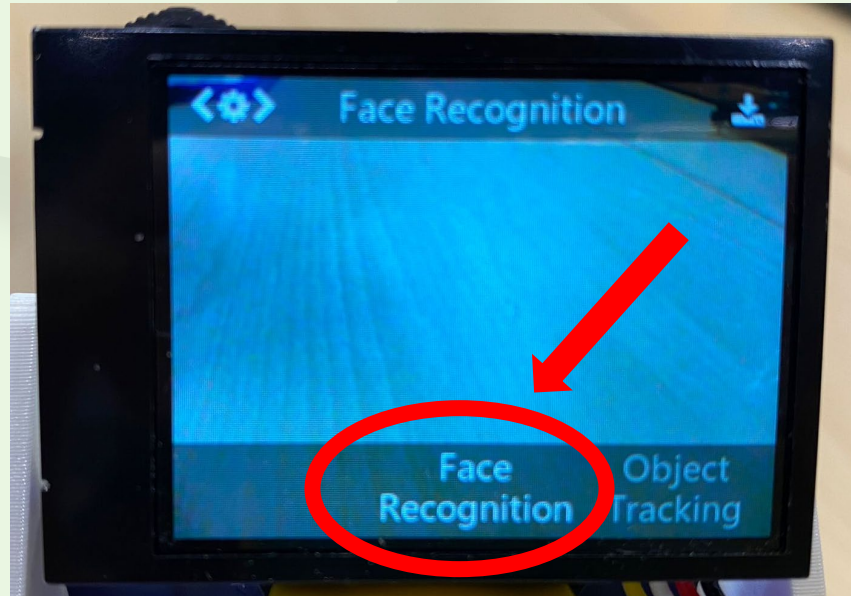


# 啟動 CUHK iCar





啟動 CUHK iCar 後，  
HuskyLens 會自動調校至  
人臉識別 (Face Recognition) 模式。

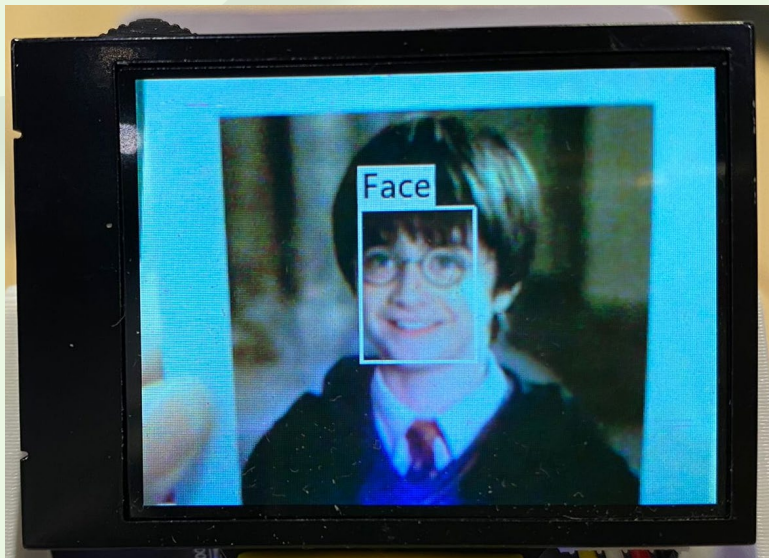


# 清除舊有數據

按一下學習鍵，  
待確認框出現後再按一下，  
忘記舊有數據。

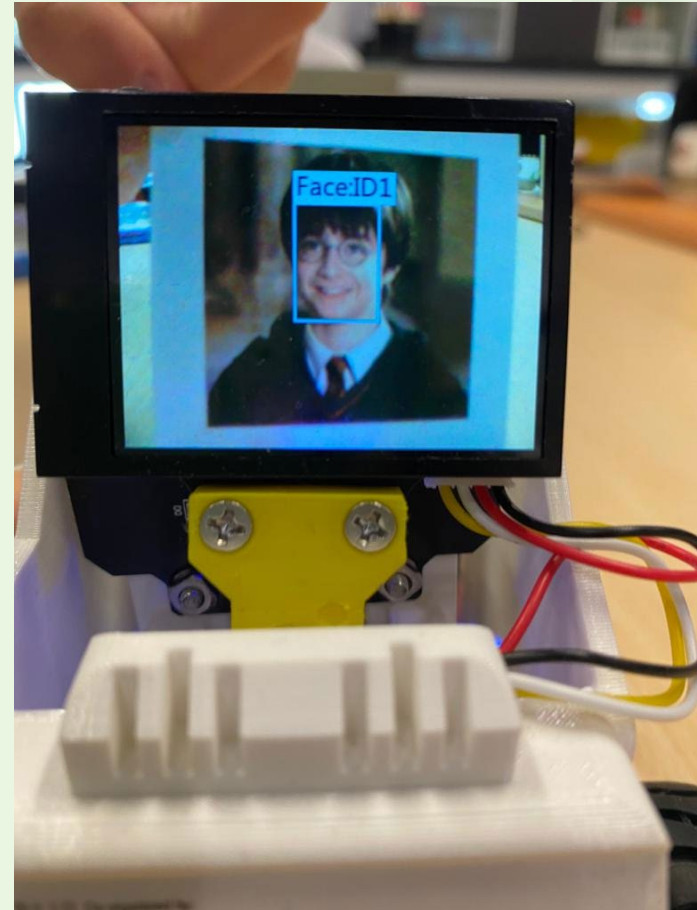


將人像照片放置 HuskyLens 前，  
屏幕會出現白色方框標示人臉位置。



按一下學習鍵，HuskyLens 便會記錄人臉，白色方框會變為藍色，並標示為 ID1。

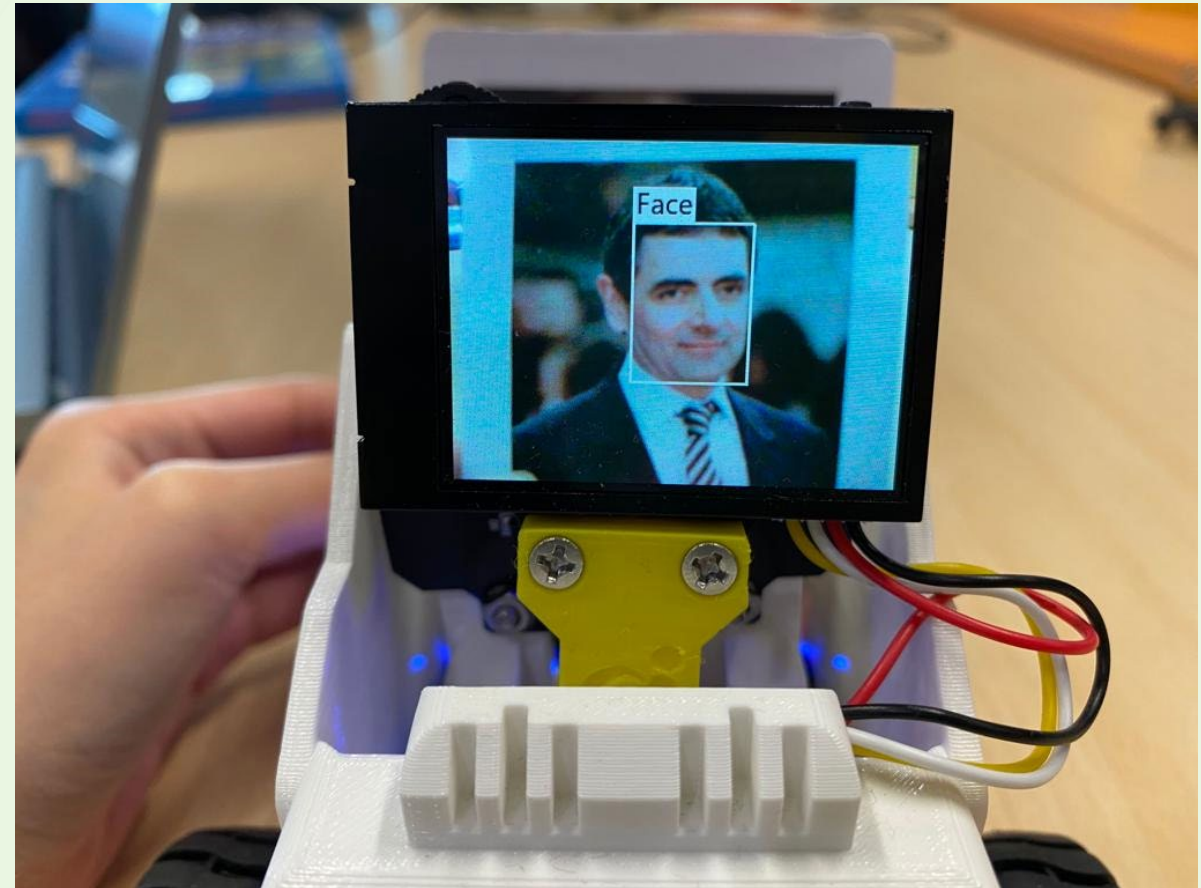
此時 CUHK iCar 會自動跟隨人像 ID1 前進。





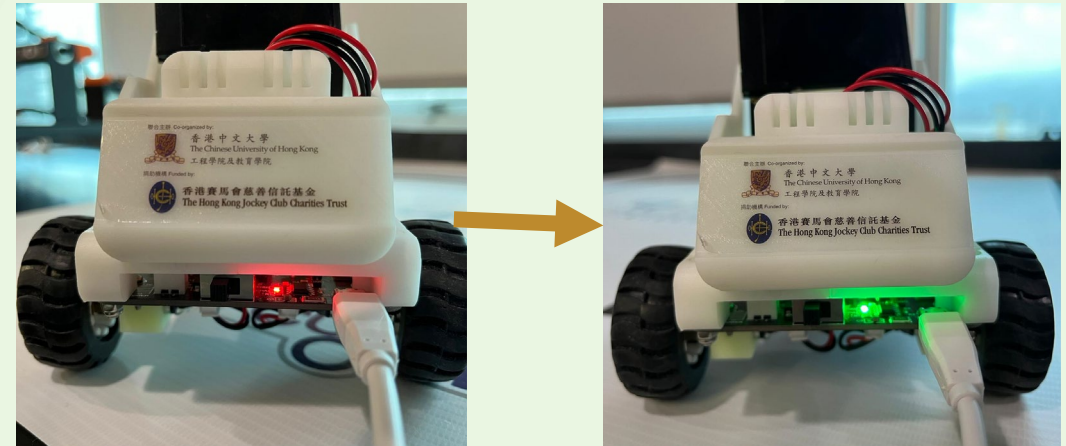
如放置另一個人臉在 HuskyLens 前面，因該人臉未曾被學習，HuskyLens 無法辨認，故會以白色方框標示。

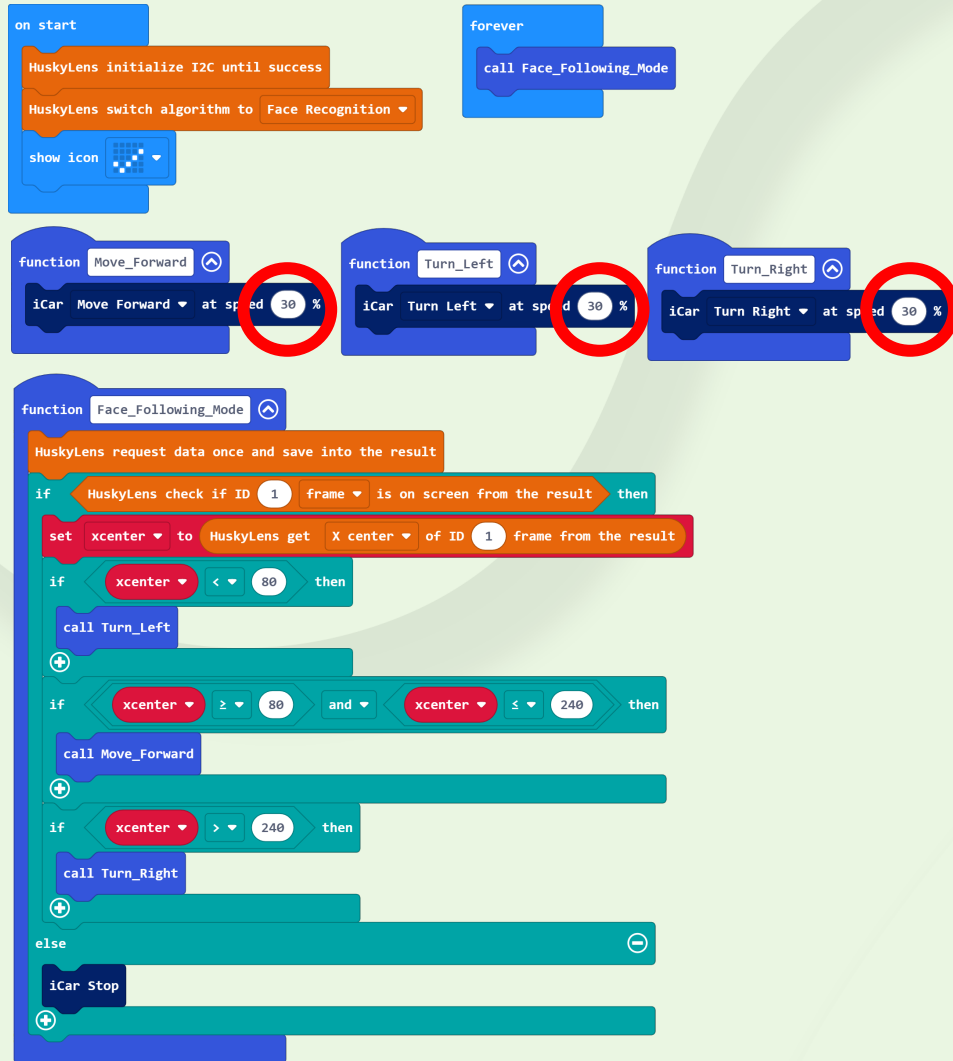
若需追蹤另一個人臉駕駛則需連按兩下學習鍵，忘記原有記憶。



# 如 CUHK iCar 未能理想運作：

- 可以嘗試將 CUHK iCar 充滿電
- 如果 CUHK iCar 仍未能理想運作，可以嘗試自己改寫程式，詳情可見下頁





紅圈標示的數字為電池被充滿下的建議速度

- 請因應電池量或電池老化程度自己以  $\pm 5$  逐步調節速度，並重新輸入數值到紅圈標示的位置
- 完成調節後請再次下載程式至 micro:bit，詳情可參考投影片第 46 張