

## Chapter 1

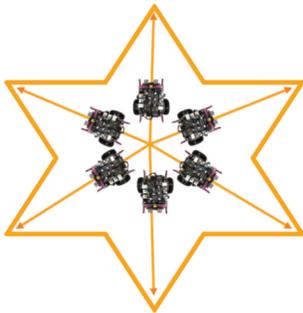
## mBot 競速賽車



1.1 認識 mBot.....	2
1.2 認識 mBlock 5 .....	4
1.3 mBot 連接電腦 .....	6
1.4 mBlock 5 連接方式.....	9
1.5 手機遙控 mBot .....	12
1.6 紅外線遙控器遙控 mBot .....	14
1.7 mBot 競速賽車 .....	17
課後練習 .....	20

## Chapter 2

## mBot 星星之舞學習機



2.1 mBot 星星之舞學習機元件規劃.....	22
2.2 mBot 基本功能 - 馬達.....	23
2.3 mBot 按鈕與蜂鳴器.....	27
2.4 mBot 蜂鳴器唱小星星 .....	31
2.5 mBot 馬達跳星星之舞 .....	34
2.6 mBot 星星之舞學習機 .....	37
課後練習 .....	38

## Chapter 3

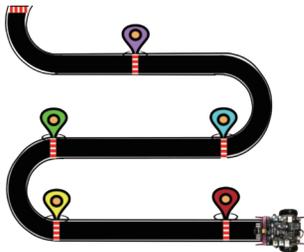
## mBot 避障機



3.1 mBot 避障機元件規劃.....	40
3.2 超音波感測器 .....	42
3.3 RGB LED .....	45
3.4 等待與判斷障礙物距離的積木.....	48
3.5 mBot 判斷障礙物距離 .....	50
3.6 mBot 播放警示聲與警示燈.....	52
3.7 mBot 避障機 .....	54
課後練習 .....	55

## Chapter 4

## mBot 智能送餐機



4.1 mBot 智能送餐機元件規劃.....	58
4.2 循線感測器.....	60
4.3 馬達動力與 mBot 運動.....	67
4.4 mBot 循黑線前進.....	69
4.5 mBot 智能送餐機連線循黑線前進.....	71
4.6 mBot 循白線前進.....	74
4.7 mBot 智能送餐機.....	76
課後練習.....	77

## Chapter 5

## mBot AI 智能辨識



5.1 人工智慧 (AI).....	80
5.2 mBot AI 智能辨識元件規劃.....	80
5.3 mBot AI 智能語音辨識.....	82
5.4 mBot AI 智能人臉年齡辨識.....	89
5.5 mBot AI 智能人臉情緒辨識.....	92
5.6 mBot AI 智能中文印刷文字辨識.....	94
5.7 mBot AI 智能英文手寫文字辨識.....	97
課後練習.....	99

## Chapter 6

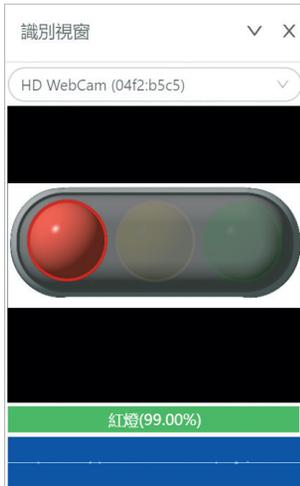
## mBot 氣象播報機



6.1 物聯網 IoT.....	102
6.2 mBot 氣象播報機元件規劃.....	103
6.3 表情面板.....	104
6.4 表情面板顯示文字跑馬燈.....	109
6.5 角色 Panda 說天氣資訊.....	111
6.6 mBot 氣象播報機.....	115
課後練習.....	119

## Chapter 7

## mBot 智能學習機



7.1 機器深度學習 .....	122
7.2 mBot 智能學習機元件規劃.....	124
7.3 訓練模型 .....	127
7.4 檢驗機器深度學習 .....	130
7.5 mBot 智能學習機.....	134
課後練習 .....	136

## Chapter 8

## mBot 娛樂機

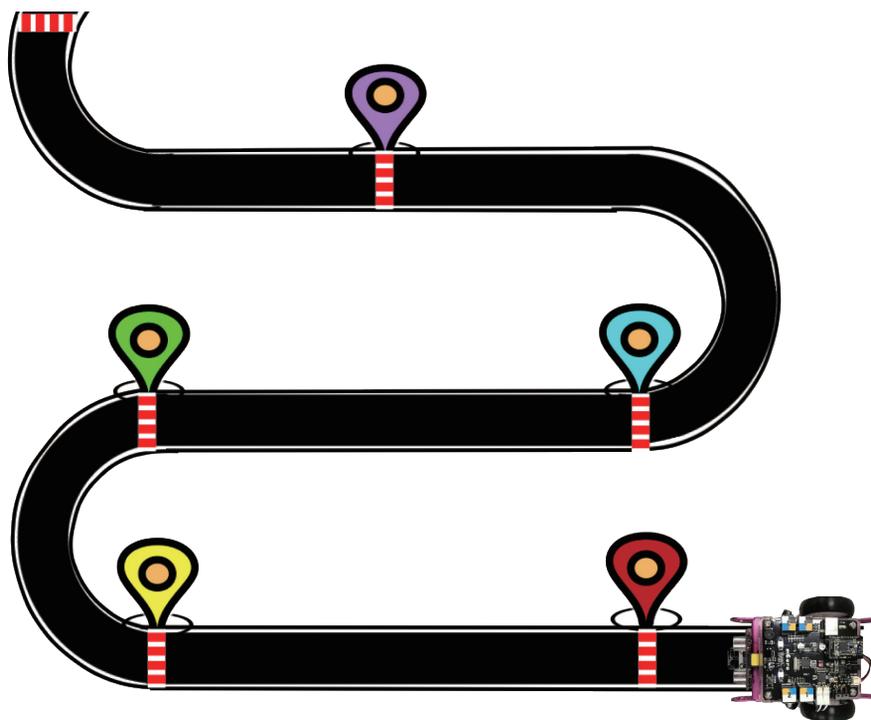


8.1 mBot 娛樂機元件規劃 .....	140
8.2 設備傳遞感測器數值給角色.....	142
8.3 循線感測器控制角色移動 .....	145
8.4 角色重複由上往下移動 .....	147
8.5 角色偵測碰到角色得分 .....	150
8.6 角色廣播 .....	151
課後練習 .....	154

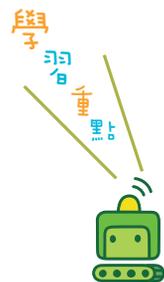
CHAPTER

# 04 mBot 智能送餐機

mBot 智能小尖兵現在要變身為智能送餐機，依據送餐地圖循線將餐點送達目的地。

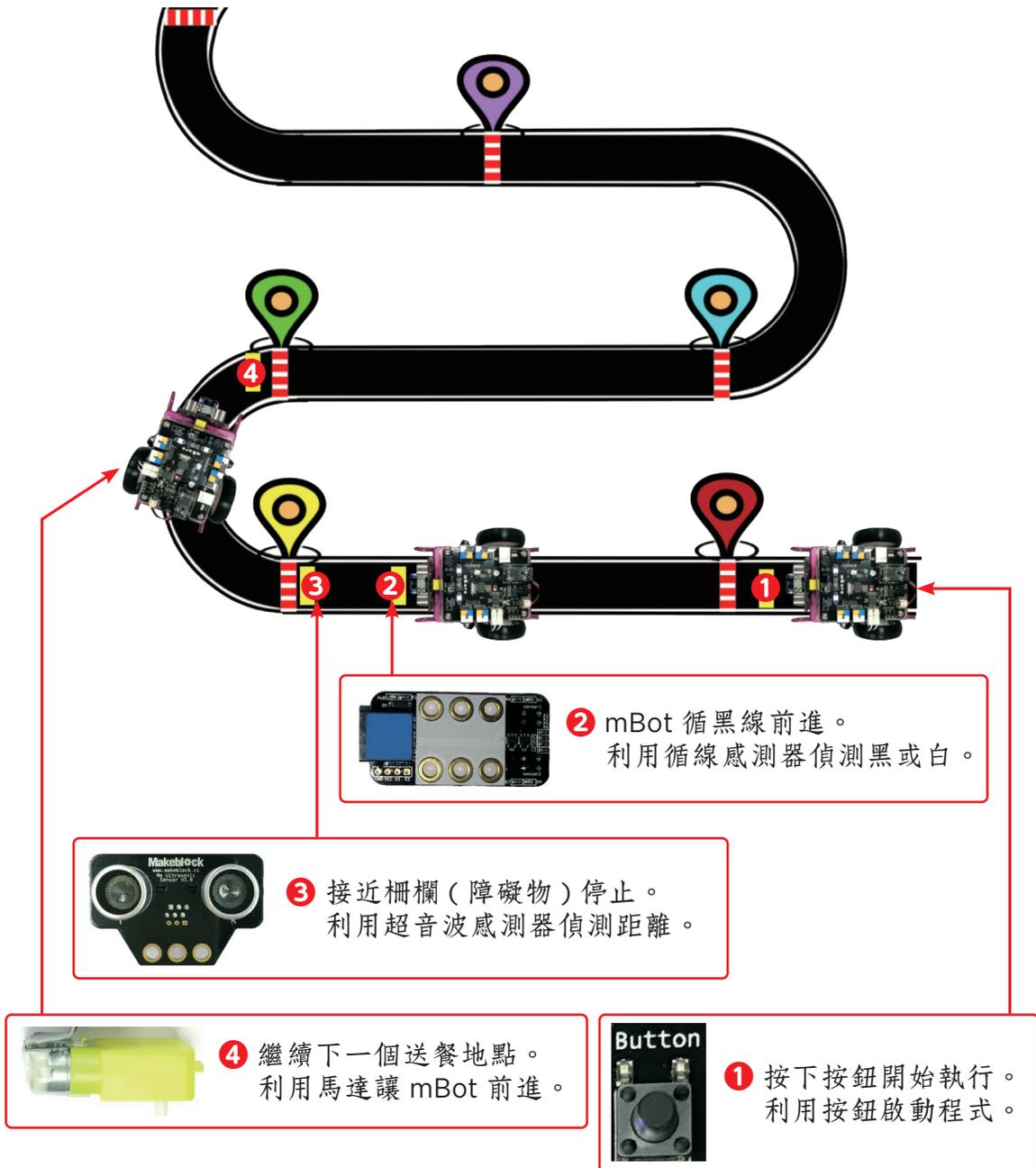


- 1 理解 mBot 的循線感測器與 mBlock 積木。
- 2 應用循線感測器設計 mBot 循黑線前進。
- 3 應用循線感測器設計 mBot 循白線前進。
- 4 應用 LED 設計 mBot 轉彎時亮燈。
- 5 應用循線感測器設計 mBot 智能送餐機。

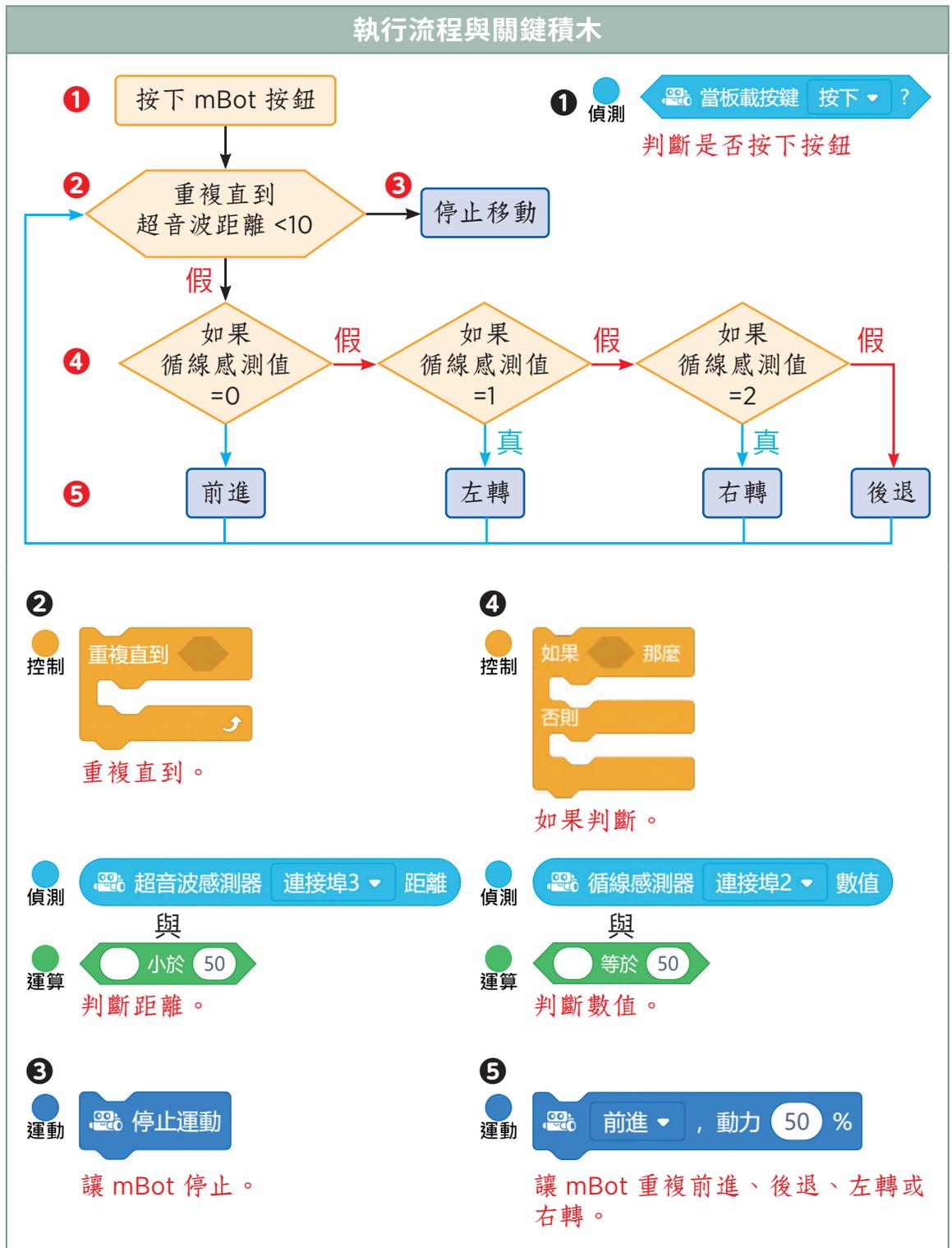


## 4.1 mBot 智能送餐機元件規劃

本章將設計智能送餐機，當按下 mBot 的「按鈕」時，mBot 循著黑線前進，當抵達送餐地點前的柵欄時停止，開始送餐，再繼續下一個送餐地點。mBot 智能送餐機在送餐流程中使用的元件與功能如下圖所示。



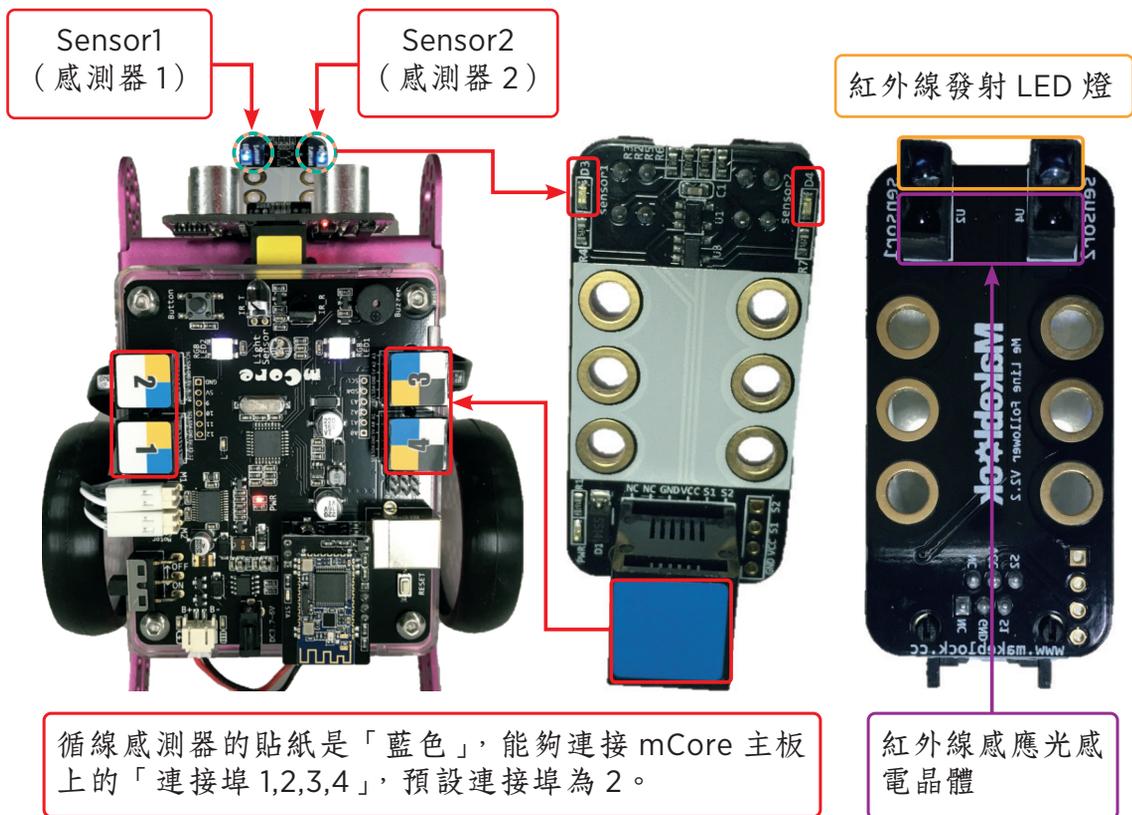
## 一、mBot 智能送餐機循黑線前進執行流程與關鍵積木



## 4.2 循線感測器

### 一、循線感測器

循線感測器上有兩組感測器，每個感測器上包含一個紅外線發射 LED 燈和一個紅外線感應光感電晶體。mBot 利用感測器的訊號在白底背景循著黑色的線前進或在黑底背景循著白色的線前進，感測器偵測到黑色時傳回 0，偵測到白色時傳回 1。循線感測器的連接方式如下圖：



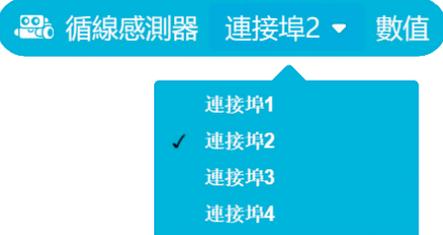
### 學中思

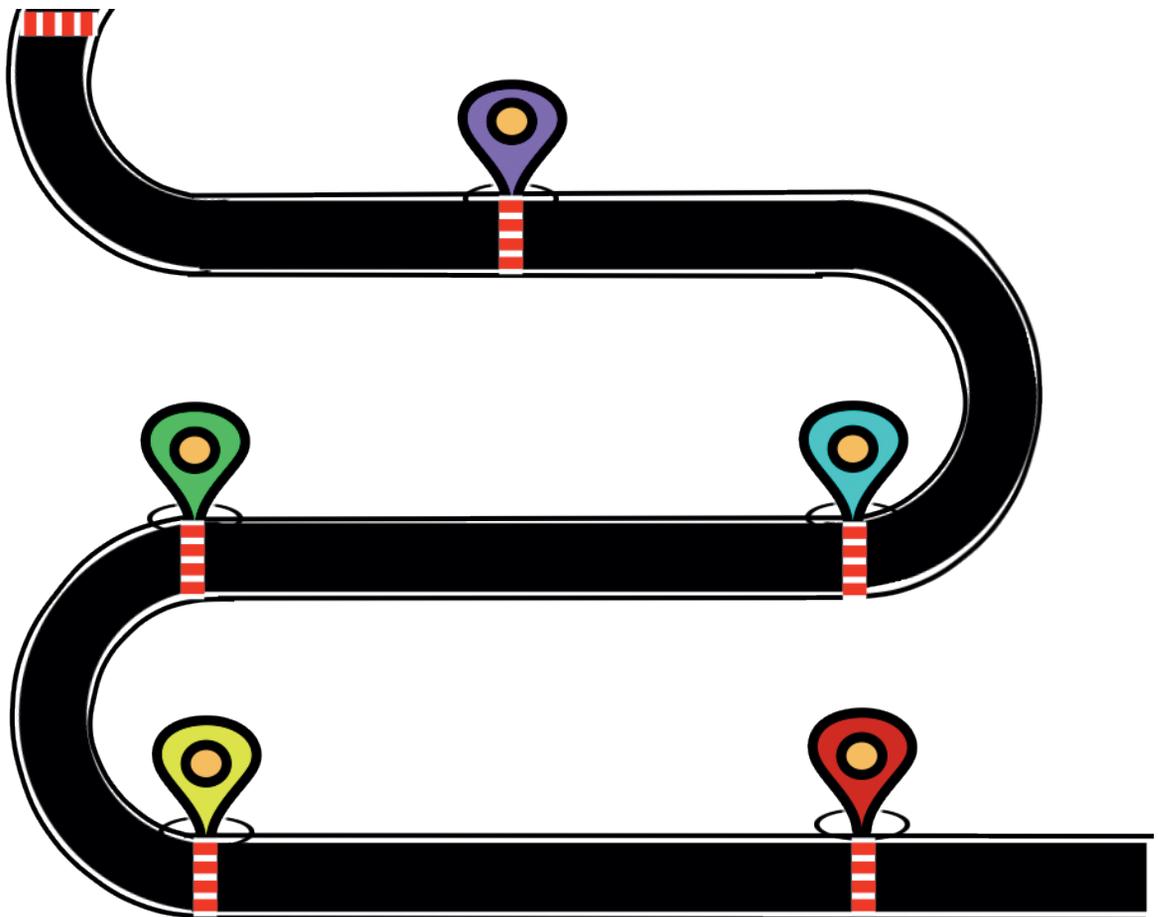
利用手機或紅外線遙控器遙控 mBot 時，原廠程式預設循線感測器與 mCore 主板的連接埠是 2。

## 二、循線感測器 mBlock 積木

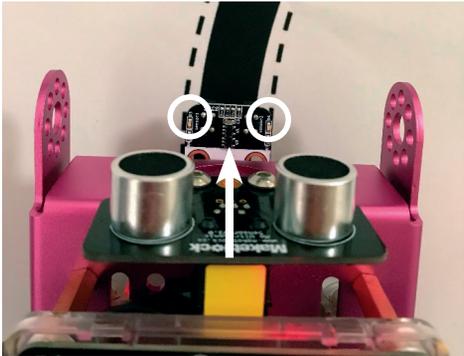
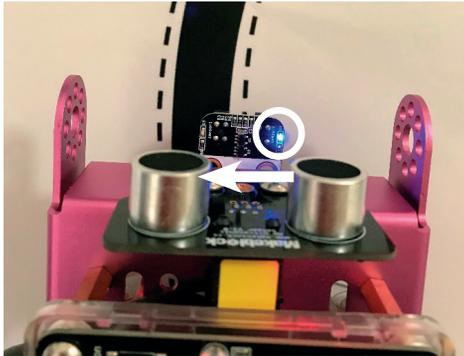
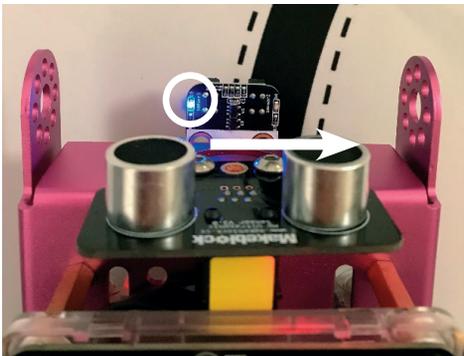
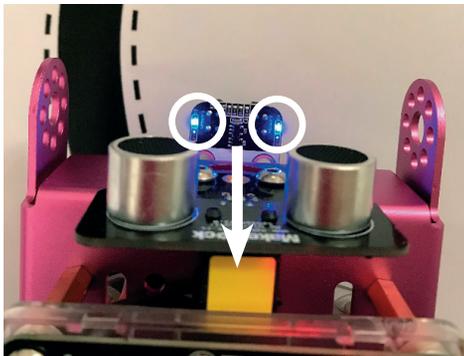
### 1 傳回循線感測器數值

循線感測器積木主要功能在傳回感測器偵測的數值或判斷循線感測器偵測到的黑或白。

功能	積木	說明
傳回循線感測器偵測值	 <p>循線感測器 連接埠2 ▾ 數值</p> <ul style="list-style-type: none"> <li>連接埠1</li> <li>✓ 連接埠2</li> <li>連接埠3</li> <li>連接埠4</li> </ul>	傳回連接埠 1~4 中，循線感測器偵測的數值，連接埠預設值為 2。



循線感測器偵測的數值包括：0,1,2,3，分別代表的訊息如下。

偵測值	數值 = 0	數值 = 1
亮燈	感測器 1，2：皆不亮燈	感測器 2 亮燈
圖例		
位置	在黑色線上	右偏右邊在白色
動作	前進	左轉
偵測值	數值 = 2	數值 = 3
亮燈	感測器 1 亮燈	感測器 1，2：皆亮燈
圖例		
位置	左偏左邊在白色	完全偏離全部在白色
動作	右轉	後退



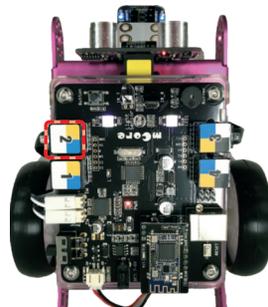
## 做·中·學

### ① 循線感測器偵測黑白數值

- 將連線方式設為 **上傳** **即時**，檢查循線感測器與 mBot 的连接埠，並勾選连接埠。

- 连接埠 1       连接埠 2  
 连接埠 3       连接埠 4

- 按 **偵測**， 勾選「循線感測器连接埠 2 數值」。在舞台顯示循線感測器即時的偵測數值。



③ 顯示 0~3 數值。

② 勾選在舞台顯示循線感測器的數值。

The screenshot shows the mBot software interface. At the top, the connection mode is set to '即時' (Real-time). On the right, the '偵測' (Detect) block is selected, and the '循線感測器' (Line Sensor) block is configured to '連接埠2' (Port 2) and '數值' (Value). A red dashed box highlights the 'mBot: 循線感測器 連接埠2 數值 0' display on the stage. At the bottom, the '即時' (Real-time) mode is selected.

① 連線模式設為【即時】。

## 4.5 mBot 智能送餐機連線循黑線前進

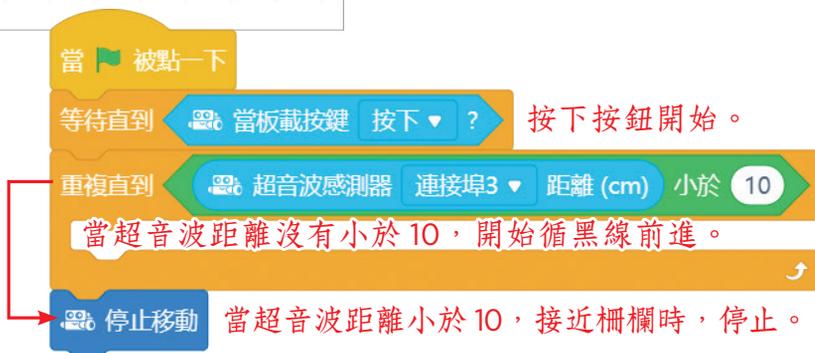
當按下 mBot 的「按鈕」時，mBot 循著黑線前進，當抵達送餐地點前的柵欄時停止，再繼續下一個送餐地點。

按下按鈕  
開始前進

循著黑線  
前進

接近柵欄  
停止

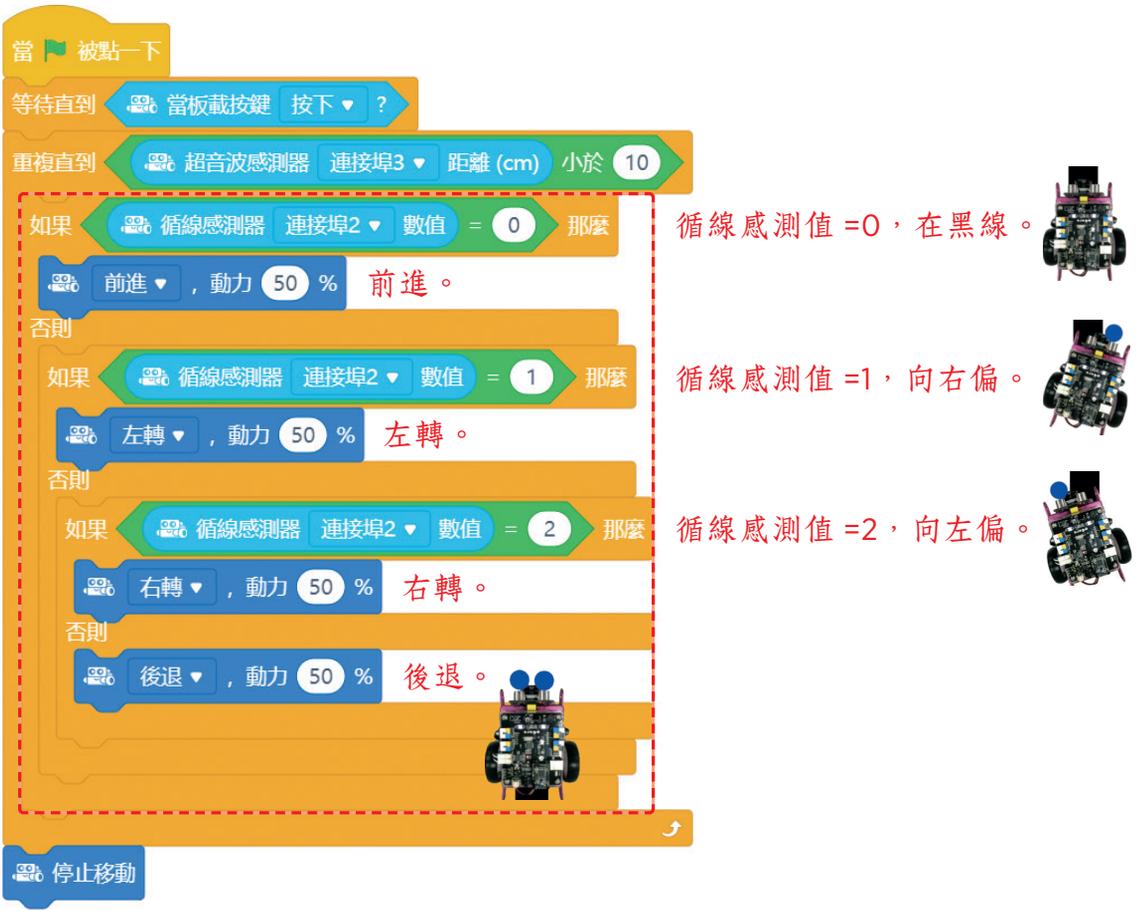
- 點選  檔案，【檔案 > 新建專案】，在「設備」按  添加，新增  【mBot】，點選【連接 > COM 值 > 連接】，將連線模式設定為【即時】。
- 點選  事件、 控制、 偵測、 運算 與  運動，拖曳下圖積木，當按下按鈕時，mBot 在接近柵欄障礙物時停止。



3 按 ，拖曳三個「如果 - 那麼 - 否則」判斷循黑線的狀態。



4 點選 、 與 ，拖曳下圖積木，讓 mBot 循著黑線前進。



- 5 將 mBot 放在循線地圖黑線上，點擊 ，檢查 mBot 是否循著黑線前進。
- 6 按 ，點選【新增積木指令】，輸入【循黑線 > 確認】，定義循黑線程式積木。



The screenshot shows the mBot programming software interface. On the left is a sidebar with various block categories: 顯示 (Display), 聲光 (Sound/Light), 運動 (Motion), 偵測 (Detection), 事件 (Events), 控制 (Control), 運算 (Math), and 變數 (Variables). The '自訂積木' (Custom Blocks) category is highlighted with a red dashed box. Below it is a '+' icon and the text '延伸集' (Extension Set).

The main workspace shows a custom block named '循黑線' (Follow Line) being defined. The block is a pink rectangle with a white border. It is connected to a '當 被點一下' (When clicked) event block. The main program area contains the following logic:

- 當 被點一下** (When clicked)
  - 等待直到** (Wait until) block: 當板載按鍵 按下 (When board button pressed)
  - 重複直到** (Repeat until) block: 超音波感測器 連接埠3 (Ultrasonic sensor port 3) 距離 (cm) 小於 10 (Distance (cm) less than 10)
  - 循黑線** (Follow Line) block: 當超音波距離沒有小於10，開始循黑線前進。 (When ultrasonic distance is not less than 10, start following the black line forward.)
  - 停止移動** (Stop moving) block
- 定義 循黑線** (Define Follow Line) block: 定義循黑線程式。 (Define the follow line program.)
  - 如果** (If) block: 循線感測器 連接埠2 (Line sensor port 2) 數值 = 0 (Value = 0) 那麼 (Then)
    - 前進** (Forward) block: 動力 50% (Power 50%)
  - 否則** (Otherwise)
    - 如果** (If) block: 循線感測器 連接埠2 (Line sensor port 2) 數值 = 1 (Value = 1) 那麼 (Then)
      - 左轉** (Turn left) block: 動力 50% (Power 50%)
    - 否則** (Otherwise)
      - 如果** (If) block: 循線感測器 連接埠2 (Line sensor port 2) 數值 = 2 (Value = 2) 那麼 (Then)
        - 右轉** (Turn right) block: 動力 50% (Power 50%)
      - 否則** (Otherwise)
        - 後退** (Backward) block: 動力 50% (Power 50%)

Red arrows and numbers 1 and 2 indicate the steps: 1 points to the '循黑線' block in the main program, and 2 points to the '定義 循黑線' block.

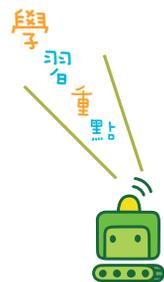
## CHAPTER

# 07 mBot 智能學習機

mBot 現在要變身為智能學習機。它能夠學習任何事物，例如：紅綠燈、行人、障礙物、鈔票或學生證等，再辨識學習的內容，依據辨識結果執行動作。



- 1 理解機器深度學習概念。
- 2 建立紅燈、黃燈與綠燈機器深度學習的模型。
- 3 應用機器深度學習，設計 mBot 智能學習機的互動模式。
- 4 設計 mBot 先判斷學習內容，再自動執行。



## 7.1 機器深度學習

### 一、機器深度學習

機器深度學習（Machine Learning，ML）是讓電腦學習東西，建立類似人類大腦的人造神經網路。例如：訓練電腦識別聲音、圖片、影片或文字等。



### 二、機器深度學習與人工智慧

人工智慧（AI）是指設計程式讓電腦具有類似人類的智慧。例如在第六章電腦能夠識別人臉年齡或語音等。機器深度學習與人工智慧的關係，就好像「學以致用」，讓電腦學習屬於「機器深度學習」、讓電腦將學習到的東西用出來就是「人工智慧」。



### 三、建立機器深度學習

機器深度學習分為訓練、檢驗與應用三階段。

#### (一) 訓練

訓練階段在訓練電腦建立模型，例如：訓練電腦建立眼睛、嘴巴、耳朵等特徵模型。

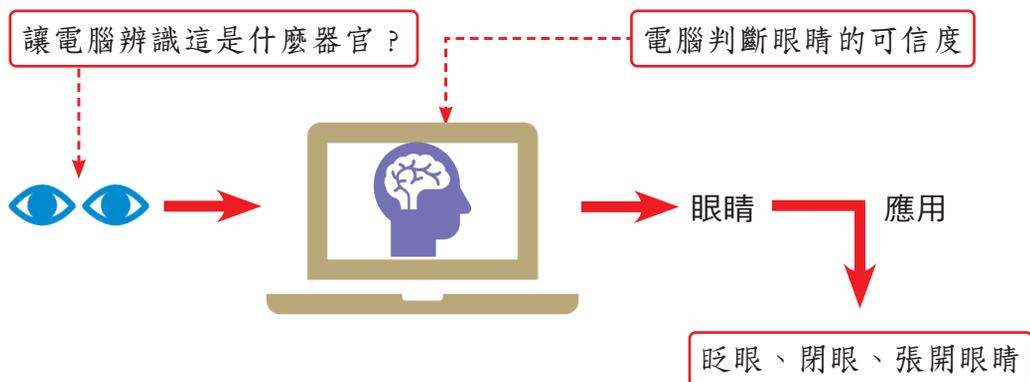


#### (二) 檢驗

檢驗階段在驗證電腦建立模型的可信度。例如：讓電腦辨識眼睛，電腦正確說出眼睛的可信度是多少？

#### (三) 應用

應用電腦判斷的結果。例如：電腦正確判斷眼睛之後，能夠將「眼睛」應用在判斷眨眼、閉眼或張開眼睛等。



## 7.2 mBot 智能學習機元件規劃

本章將設計 mBot 智能學習機。讓角色進行機器深度學習，學習紅綠燈，建立類似人類大腦的人造神經網路。再辨識紅綠燈，最後 mBot 執行停止、減速或前進的動作。mBot 智能學習機使用元件與執行流程如下圖所示。

### 一、角色機器深度學習



### 二、角色辨識紅綠燈



### 三、角色說辨識結果

紅燈請停止



- ① 辨識結果是紅燈，角色說出：「紅燈請停止」。
- ② 辨識結果是黃燈，角色說出：「黃燈請減速慢行」。
- ③ 辨識結果是綠燈，角色說出：「綠燈請通行」。

廣播

### 四、mBot 執行紅綠燈

- ① 紅燈時，mBot 停止。
- ② 黃燈時，mBot 減速。
- ③ 綠燈時，mBot 前進。



停止



減速



前進

## 一、mBot 與機器深度學習互動規劃

機器深度學習包含訓練模型、檢驗與應用三個步驟，在 mBot 智能學習機與 Panda 機器深度學習的互動規劃與關鍵積木如下圖所示。

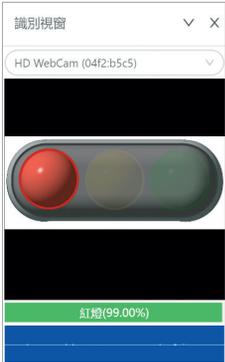
### 機器深度學習：訓練模型



- 訓練紅燈模型
- 訓練黃燈模型
- 訓練綠燈模型



### 機器深度學習：檢驗



- 以紅燈給角色辨識
- 以黃燈給角色辨識
- 以綠燈給角色辨識



### 機器深度學習：應用



- 如果辨識結果是紅燈，語音播放：「紅燈請停止」、角色同步說出文字、mBot 停止。
- 如果辨識結果是黃燈，語音播放：「黃燈請減速慢行」、角色同步說出文字、mBot 減速慢行。
- 如果辨識結果是綠燈，語音播放：「綠燈請通行」、角色同步說出文字、mBot 前進。

## 關鍵積木

機器深度  
學習

辨識結果

傳回辨識結果。

文字轉語音

說 hello

語音說。

運動



前進 ▾

,

動力 50 %

讓 mBot 重複前進、後退、左轉或右轉。

## 7.3 訓練模型

讓電腦學習紅燈、綠燈與黃燈，建立訓練模型。

- 1 點選 檔案，【檔案 > 新建專案】，在「設備」按 添加，新增 【mBot】，點選【連接 > COM 值 > 連接】，將連線模式設定為【即時】。
- 2 在角色新增機器深度學習積木。



- 點選 **機器深度學習**，按【訓練模型】，在「分類 1」分別輸入【紅燈】、【黃燈】與【綠燈】，訓練紅綠燈模型。



### 小提示

紅綠燈圖片請參閱課後習題。

### 思中創

利用生活中常見的具體範例建立機器學習模型，學習模型以圖片進行人工智慧比對，圖片範例的差異性愈大時，機器學習結果的可信度愈高。

- 4 開啟視訊攝影機，將紅燈放在視訊攝影機鏡頭前，長按【學習】，直到「樣本」照片超過 10 張，再放開「學習」按鈕，訓練辨識紅燈模型。

HD WebCam (04f2:b5c5) 模型訓練

1 紅燈放視訊鏡頭前。

11 樣本

紅燈

99.0%

學習

2 按住學習不放，直到樣本數超過 10，再放開。

0 樣本

學習

0 樣本

綠燈

3 顯示結果紅燈。

結果  
紅燈

新建模型 使用模型

- 5 重複上一步驟，學習「黃燈」與「綠燈」。最後點選【使用模型】，自動產生機器深度學習積木。

HD WebCam (04f2:b5c5) 模型訓練

1 黃燈與綠燈依序放視訊鏡頭前。

11 樣本

黃燈

學習

2 3 按住學習不放，直到樣本數超過 10，再放開。

10 樣本

綠燈

99.0%

學習

結果  
綠燈

新建模型 使用模型 4

## 7.4 檢驗機器深度學習

訓練模型建立成功之後，自動產生機器深度學習紅燈、黃燈與綠燈積木。

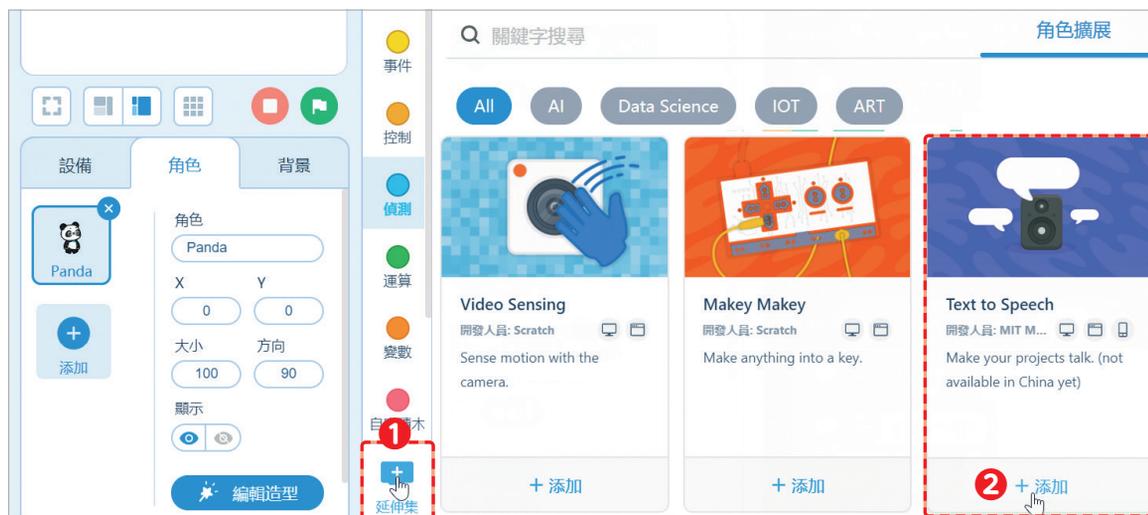
### 一、機器深度學習積木

功能	積木	說明
傳回結果		傳回辨識結果為紅燈、黃燈或綠燈。
可信度		傳回辨識結果的可信度。
判斷結果		判斷辨識結果是否為紅燈、黃燈與綠燈。

### 二、檢驗機器深度學習

以紅燈、黃燈與綠燈給角色辨識，語音說出辨識結果。

① 在角色新增文字轉語音 (Text to Speech) 積木。



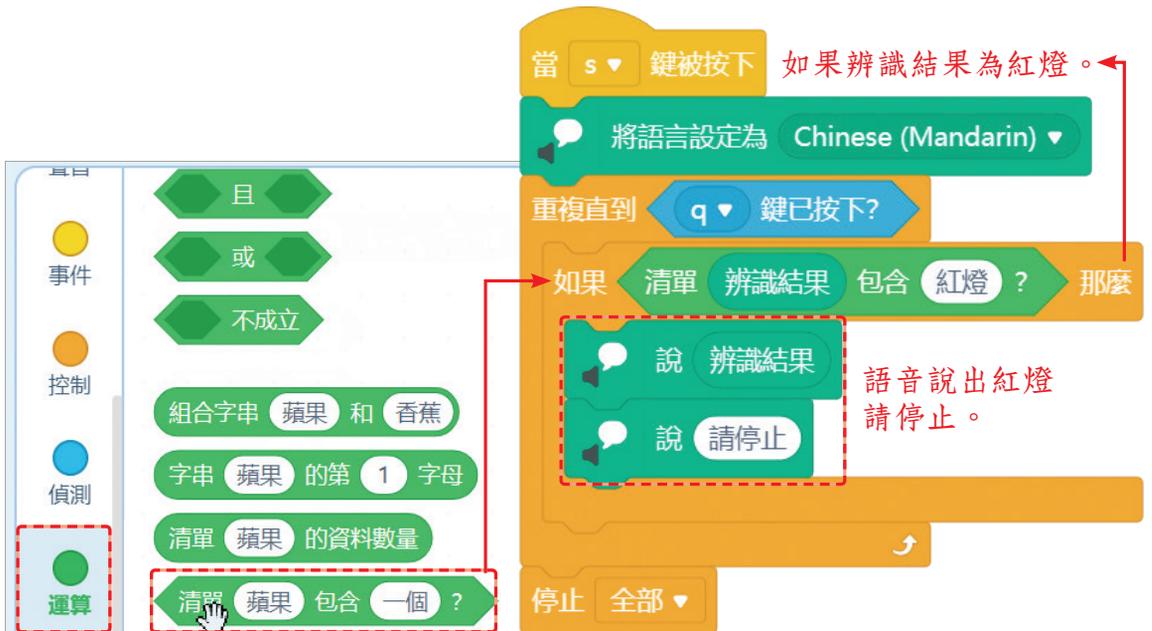
- 2 按  事件、 文字轉語音、 控制 與  偵測，將語音設定為中文 (Chinese Mandarin)、按下 s 鍵開始辨識、按下 q 停止辨識。



The screenshot shows the 'Text-to-Speech' (文字轉語音) module configuration. On the left, the 'Language' (語言) dropdown menu is open, with 'Chinese (Mandarin)' selected. The main workspace contains the following blocks:

- 當 s 鍵被按下 (When s key is pressed) - 按下 s 開始。
- 將語言設定為 Chinese (Mandarin) (Set language to Chinese (Mandarin))
- 重複直到 q 鍵已按下? (Repeat until q key is pressed?) - 按下 q 停止全部程式。
- 辨識紅綠燈。 (Recognize red and green lights.)
- 停止 全部 (Stop all)

- 3 按  事件、 運算、 文字轉語音 與  機器深度學習，語音說出辨識結果 (紅燈) 請停止。



The screenshot shows a more complex program using logic blocks. The workspace contains the following blocks:

- 當 s 鍵被按下 (When s key is pressed) - 如果辨識結果為紅燈。
- 將語言設定為 Chinese (Mandarin) (Set language to Chinese (Mandarin))
- 重複直到 q 鍵已按下? (Repeat until q key is pressed?)
- 如果 清單 辨識結果 包含 紅燈? 那麼 (If list 'Recognition Results' contains 'Red Light'?)
- 說 辨識結果 (Say Recognition Results)
- 說 請停止 (Say Please Stop)
- 清單 蘋果 包含 一個? (List 'Apples' contains one?)
- 停止 全部 (Stop all)



課

後

練

習

## 一、單選題

- ( ) 1. 如果想讓電腦能夠像人類一樣學習辨識紅綠燈，應該使用下列哪一種功能訓練電腦學習？
- (A) 人工智慧 (B) 物聯網  
(C) 機器深度學習 (D) 使用者雲訊息。
- ( ) 2. 下列關於機器深度學習的敘述，何者「正確」？
- (A)  傳回辨識結果
- (B)  判斷辨識結果是否為紅燈
- (C)  傳回判斷結果是紅燈的可信度值
- (D) 以上皆是。
- ( ) 3. 如果想要讓電腦傳回判斷的可信度，應該使用下列哪一個積木？
- (A)  (B) 
- (C)  (D) 
- ( ) 4. 下圖程式應該寫在哪一個位置？



- (A) 舞台 (B) 背景 (C) 角色 (D) 設備。