

# Code for Everyone (編程)技能的普及化

Implementing Computing Science in Hong Kong Primary Schools 2019 – 2020

Micro:bit x Scratch 3.0 課程



## 內容

Micro: bit 與 Scratch 3.0 的介紹 .....	5
1. 為甚麼要學習編程? .....	5
2. 運算思維 (Computational Thinking).....	5
3. 編程概念.....	6
4. 甚麼是 Micro:bit? .....	6
4.1 Micro: bit 能做甚麼? .....	7
5. 甚麼是 Scratch 3.0? .....	9
6. 如何連接 Micro: bit 至 Scratch 3.0 .....	10
6.1 關於你的電腦.....	10
6.2 關於你的 Micro: bit .....	10
6.3 連接 Micro: bit 與 Scratch 3.0 .....	10
你好 Scratch 3.0!.....	12
1. 你需要什麼?.....	12
2. 學習目標.....	12
3. 使用的方塊.....	13
4. 編程步驟.....	13
5. 結果.....	14
移動吧，猴子.....	15
1. 你需要什麼?.....	15
2. 學習目標.....	15
3. 使用的方塊.....	16
4. 設計步驟.....	16
5. 編程步驟.....	18
6. 結果.....	19
左與右的移動.....	20
1. 你需要什麼?.....	20

2. 學習目標.....	20
3. 使用的方塊.....	21
4. 設計步驟.....	21
5. 編程步驟.....	23
6. 結果.....	23
偷取蘋果.....	25
1. 你需要什麼?.....	25
2. 學習目標.....	25
3. 使用的方塊.....	26
4. 設計步驟.....	26
5. 編程步驟.....	29
6. 結果.....	33
心的跳動.....	35
1. 你需要什麼?.....	35
2. 學習目標.....	35
3. 使用的方塊.....	36
4. 設計步驟.....	36
5. 編程步驟.....	38
傾斜吉他.....	43
1. 你需要什麼?.....	43
2. 學習目標.....	43
3. 使用的方塊.....	44
4. 設計步驟.....	44
5. 編程步驟.....	47
6. 結果.....	52
搖動爵士鼓.....	54
1. 你需要什麼?.....	54
2. 學習目標.....	54

3. 使用的方塊.....	55
4. 設計步驟.....	55
5. 編程步驟.....	59
6. 結果.....	63
引腳電子琴.....	66
1. 你需要什麼?.....	66
2. 學習目標.....	66
3. 使用的方塊.....	67
4. 硬件設計步驟.....	67
5. 軟件設計步驟.....	69
6. 編程步驟.....	71
7. 結果.....	72

# Micro: bit 與 Scratch 3.0 的介紹

## 1. 為甚麼要學習編程？

現在的教學已經開始嘗試把重點放在發展創造力、邏輯思維能力、問題解決能力和合作能力。編程過程可以培養上述能力，它將成為我們的第三語言。

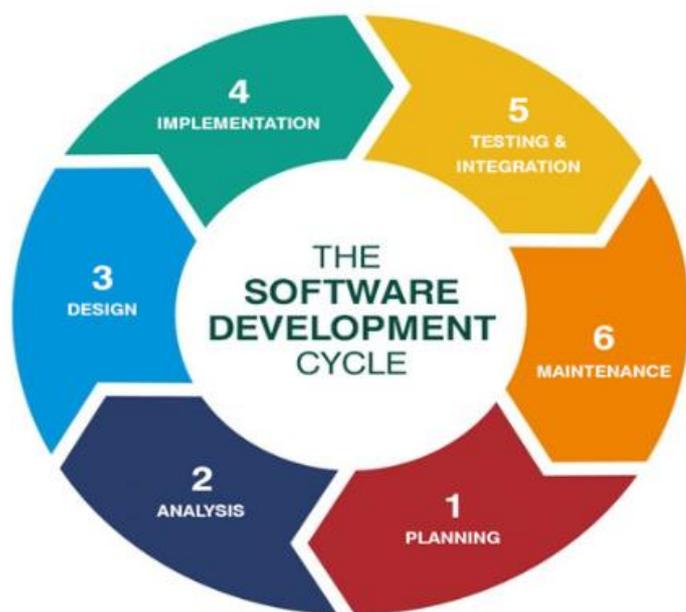
## 2. 運算思維 (Computational Thinking)

在編碼過程中，我們需要運算思維的能力，每個電腦計算都包括輸入，計算過程和輸出。



輸入 (x)	計算過程	輸出 (y)
2	x 2	4
4	x 2	8
6	x 2	12
8	x 2	16

### 3. 編程概念



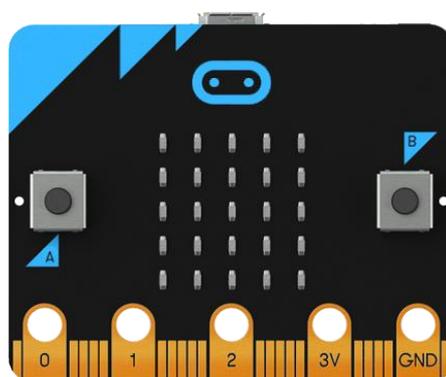
- Planning 規劃：利用編程達到目的
- Analysis 分析：需要利用的軟件及硬件，例如 micro: bit x scratch
- Design 設計：程式的設計及硬件的配合
- Implementation 實施：進行編程及組裝
- Testing & Integration 測試與組合：反覆測試及組合程式及硬件配搭，找出程式漏洞
- Maintenance 維護：定時檢閱程式，在有需要時作出修改，例如 Scratch 2.0 更新至 Scratch 3.0

### 4. 甚麼是 Micro:bit?

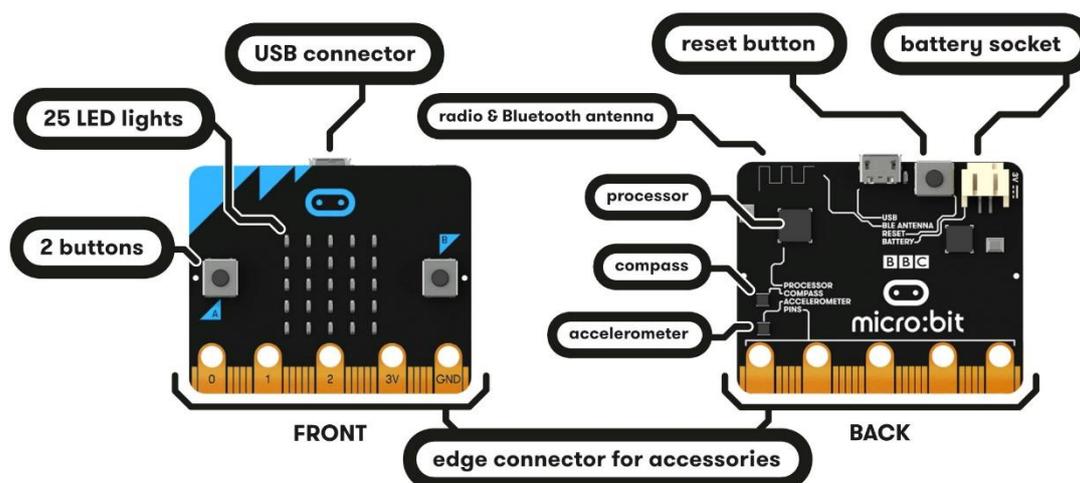
BBC micro: bit 給予學生機會進行創作，從樂器到機械人，提供無限的可能性。 Micro: bit 是一塊全程式電腦化的掌上微型控制版，BBC Micro: bit 相較於 1980 年代初用於學校教育的 BBC 微型電腦小七十倍及快十八倍。

Micro: bit 內置有不同功能，比如可以閃爍信息的 25 個紅色 LED 燈。 Micro: bit 有兩個可編程按鈕可用於控制遊戲或暫停和跳過播放列表中的歌曲。 Micro: bit 亦可以告訴你正在前進

的方向，及使用低能量藍牙連接其他設備和互聯網。同學亦可以瀏覽 Micro: bit 網站以獲取更多有關 Micro: bit 的資訊及編寫程式。愈了解 Micro: bit 的組成，你可以創造更多可能性。



## 4.1 Micro: bit能做甚麼？



**LED:** LED 是發光二極管。Micro: bit 有 25 個可單獨編程的 LED，可以顯示文本，數字和圖像。



**Buttons:** Micro: bit（標記為 A 和 B）前面有兩個按鈕。你可以隨時按下這些按鈕，允許你觸發設備上的程式。



**Pins:** Micro: bit 的邊緣連接器上有 25 個外部連接器，我們將其稱為“引腳”。利用引腳連接其他額外的傳感器來控制代碼，例如發動機 (Motor)。



**Light Sensor:** 通過將屏幕的 LED 反轉作為輸入，LED 屏幕作為基本光傳感器工作，允許你檢測環境中的光。



**Temperature Sensor:** Micro: bit 內置溫度計功能，可以以華氏或攝氏的表達方式量度現時溫度。



**Accelerometer:** 加速計測量微量加速度。當移動時，加速計能測量其變化。加速計還可以檢測其他動作，例如搖晃，傾斜和自由落體。



**Compass:** 指南針檢測地球的磁場，讓你檢測 Micro: bit 所朝的方向。指南針必須先進行校準才能使用。



**Radio:** 無線電功能允許你在兩塊或以上的 Micro: bit 之間進行無線



通信。使用無線電將消息發送到其他 **Micro: bit**，構建多人遊戲等等！

**Bluetooth:** 藍牙（藍牙低功耗）天線允許 **Micro: bit** 發送和接收藍牙信號。允許 **Micro: bit** 與電腦、手機和平板電腦進行無線通信，因此你可以通過 **Micro: bit** 控制手機，並通過手機無線發送代碼到你的設備！



## 5. 甚麼是 Scratch 3.0?

Scratch 是一種編程語言及提供網上社區平台，讓孩子們可以與來自世界各地的人們一起編程和分享故事，遊戲和動畫等互動媒體。當孩子們用 Scratch 創造時，他們學會創造性思考，協同工作，並系統地推理。Scratch 由麻省理工學院媒體實驗室的終身幼兒園小組設計和維護。



Scratch 3.0 是新一代 Scratch，旨在通過 Scratch 擴展你可以創建的方式，內容和位置。它將包括許多新的角色，一個全新的聲音編輯器和許多新的編程塊。除了筆記本電腦或桌上電腦之外，你還可以在平板電腦上使用 Scratch 創建和播放項目。

Scratch 3.0 還包括新的擴展庫（角色，背景和聲音），新的漸變工具，以及連接到 micro: bit 等設備的能力。

## 6. 如何連接 Micro: bit 至 Scratch 3.0

### 6.1 關於你的電腦

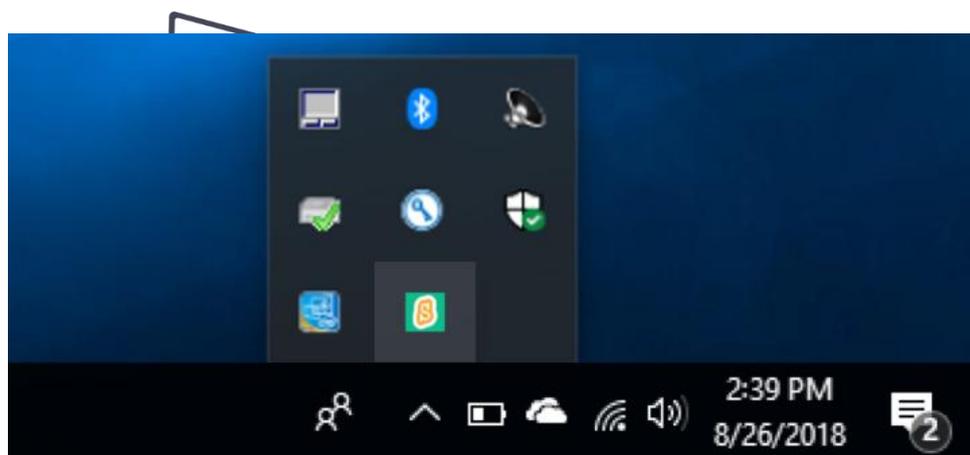
- 要使用 Scratch 3.0 連接 Micro: bit，你的電腦作業系統應該是 Windows 10 或 macOS 10.13 以上，並且需要配備藍牙 4.0 功能。
- 你的電腦需要下載並安裝 **Scratch Link**。  
Windows：<https://downloads.scratch.mit.edu/link/windows.zip>  
macOS：<https://downloads.scratch.mit.edu/link/mac.zip>



- 啟動 Scratch Link 並確保它正在運行，它應該出現在工具欄中。

### 6.2 關於你的Micro: bit

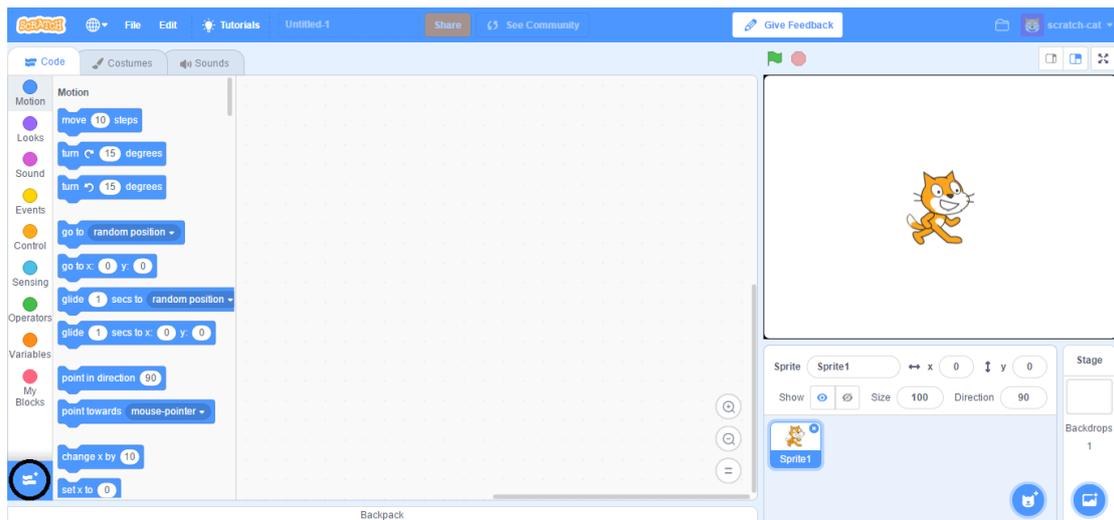
- 使用 USB 把 Micro: bit 連接在你的電腦



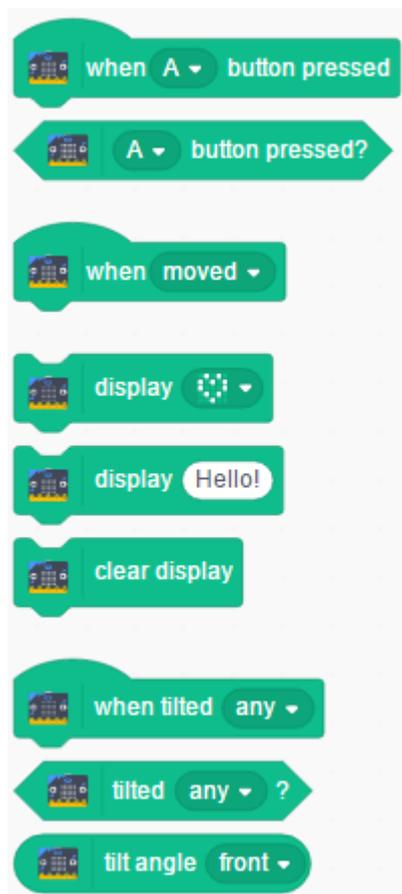
- 下載 Scratch micro:bit 檔案  
(<https://downloads.scratch.mit.edu/microbit/scratch-microbit-1.0.hex.zip>)  
至你的 Micro: bit

### 6.3 連接Micro: bit 與Scratch 3.0

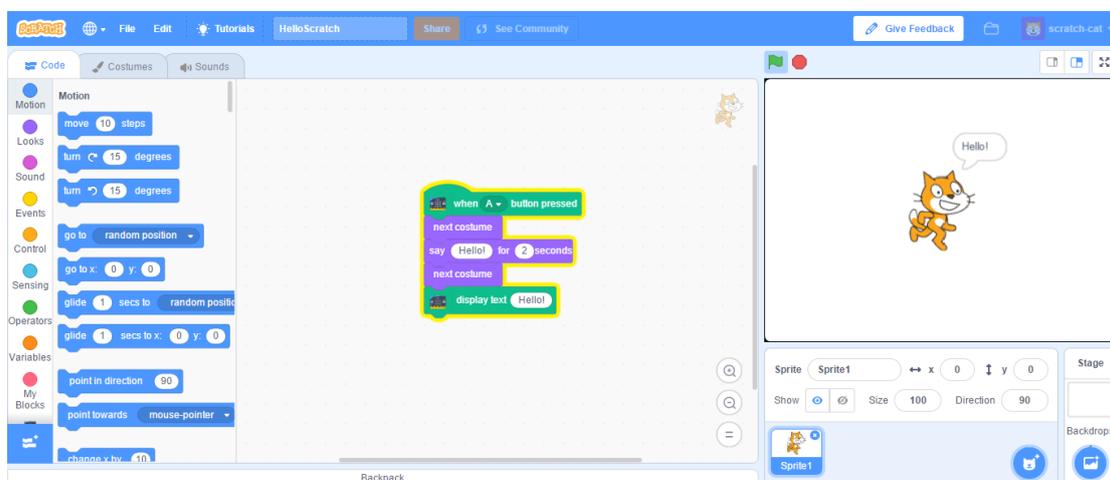
- 使用 USB 或電池為 Micro: bit 供電
- 前往 <https://scratch.mit.edu/>.
- 在左下角按“添加擴展”



- 選擇 micro: bit 擴展
- 選擇你想連接的 Micro: bit。每一塊 Micro: bit 都有它的名字，例如：gupaz。如果你是第一次使用 Scratch 3.0 連接 micro: bit，將會彈出一個窗口詢問: (1) Make sure you have Scratch Link installed and running; (2) Check that Bluetooth is enabled.
- 現在，你可以使用 Micro: bit 在 Scratch 3.0 中創建自己的程式。



# 你好 Scratch 3.0!



級別：初級

## 1. 你需要什麼?

- 1 x BBC micro:bit
- 1 x micro USB 線
- 1 x AAA 電池盒
- 2 x AAA 電池

## 說明:

使用 micro: bit 上的按鈕激活 scratch 3.0 和 micro: bit，並在 LED 顯示屏上顯示圖案。

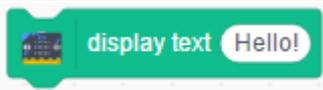
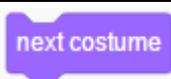
。

## 2. 學習目標

- 使用 micro: bit 和 Scratch 的基本功能完成第一個程式
- 學習使用 micro: bit 上的按鈕。

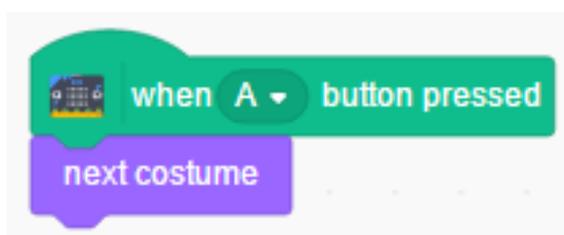
**目標:** 在第一課中，我們需要讓精靈說“Hello”並通過按下 micro: bit 上的按鈕來改變精靈的外貌。首先，來看看我們將在此活動中使用哪些方塊。

### 3. 使用的方塊

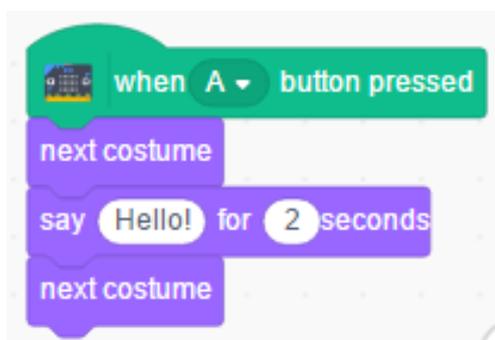
類別	塊	功能
Micro: bit		它是所有代碼的開始。它表示當按下 micro: bit 上的 A 按鈕時，程式將啟動。
		Micro: bit 將在 LED 顯示屏上顯示文本，一次顯示一個字符。如果字串適合屏幕（即一個字母），則不會滾動。
Looks		精靈附近顯示一個語音氣泡 2 秒鐘。在 2 秒後，繼續以下命令。
		如果你的精靈有多件服裝，你可以使用這個塊來改變下一個服裝。如果它顯示的是最後一件服裝，使用此塊將轉換為第一件服裝。

### 4. 編程步驟

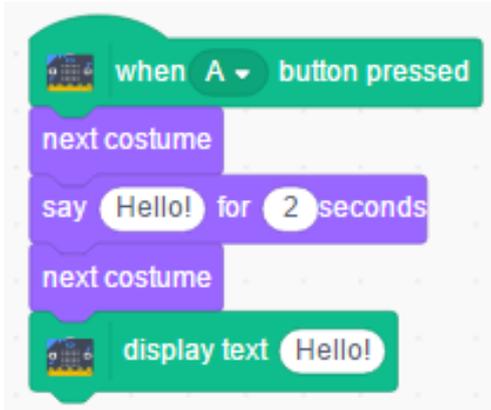
- 當我們按下按鈕 A 時，精靈貓將變為下一個服裝。



- 改變了精靈貓的外觀後，我們希望精靈貓說“Hello！”。
- 所以，加入“say Hello! for 2 seconds”在“next costume”塊之後。然後使用“next costume”塊更改至原本的外觀。

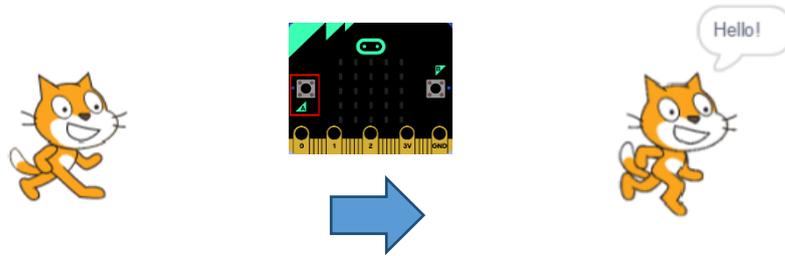


- 當 Scratch 完成所有步驟時，我們希望 micro: bit 使用“display text Hello！”塊顯示“Hello！”訊息。

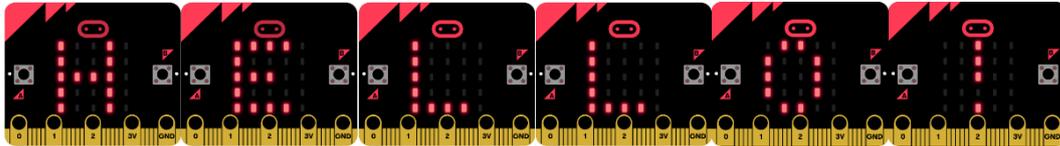


## 5. 結果

- 完成編程後，使用藍牙連接 micro: bit 來控制精靈貓，Scratch 的結果如下：

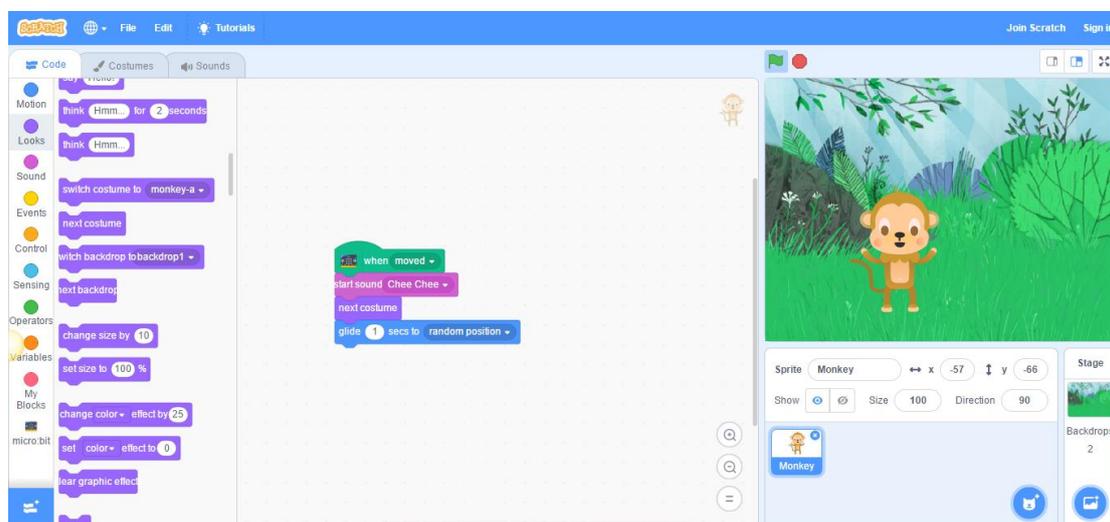


- Micro: bit 的 LED 顯示屏將會顯示如下：



只需按照步驟操作，你就可以學習如何使用 micro: bit 輕鬆控制 scratch 3.0。

# 移動吧，猴子



等級：初級

## 1. 你需要什麼?

- 1 x BBC micro:bit
- 1 x micro USB 線
- 1 x AAA 電池盒
- 2 x AAA 電池

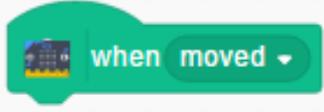
說明：使用 micro: bit 上的移動功能激活 Scratch 3.0

## 2. 學習目標

- 使用 micro: bit 和 scratch 3.0 的基本功能.
- 與角色的聲音互動
- 在 micro: bit 上學習移動功能的使用

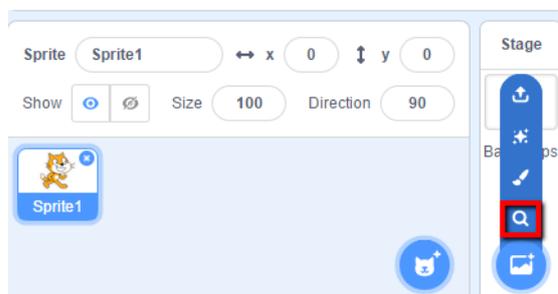
目標：在第二課中，我們通過移動 micro: bit 讓猴子發出“Chee Chee”的聲音，並同時控制猴子的移動。首先，來看看我們將在此活動中使用哪些塊。

### 3. 使用的方塊

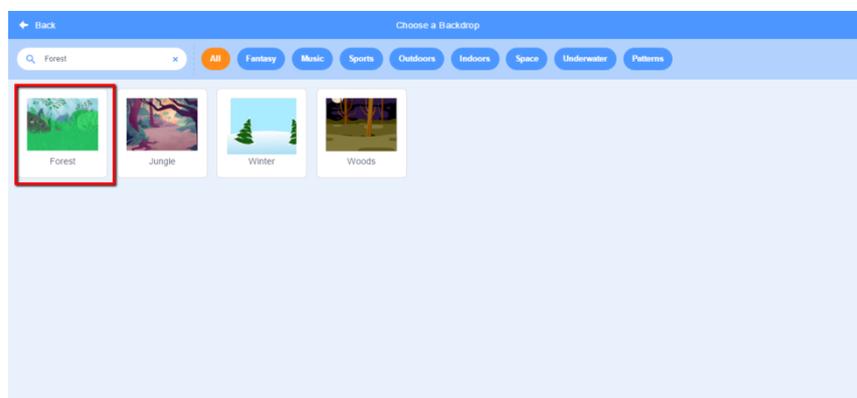
類別	塊	功能
Micro: bit		它是所有程式的開始。它表示當你移動 micro: bit 時，程序將啟動。
Sound		這可以播放 Scratch 中內置的聲音，也可以上傳或設計自己的聲音。

### 4. 設計步驟

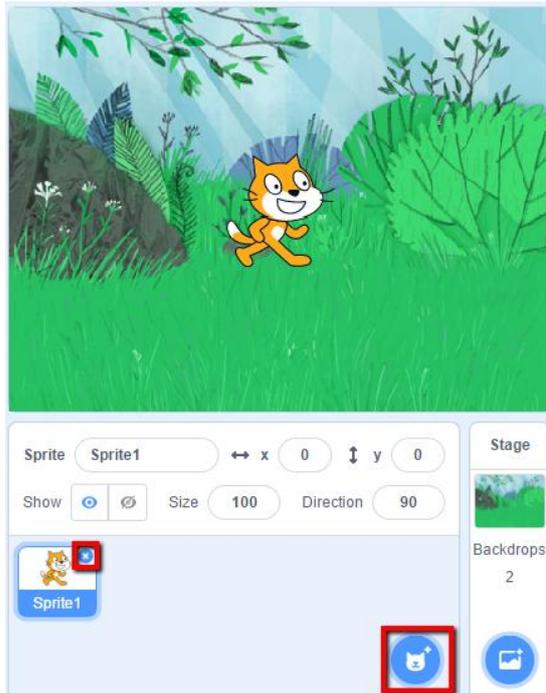
- 從上一課開始，你可能會覺得背景過於單調。因此，我們將在本課中添加背景元素。單擊右上角的按鈕，選擇“choose a background”。



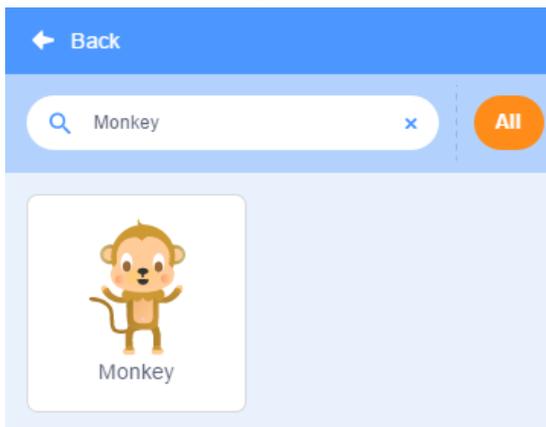
- 然後選擇你喜歡的背景，在這裡選擇“Forest”作為背景。



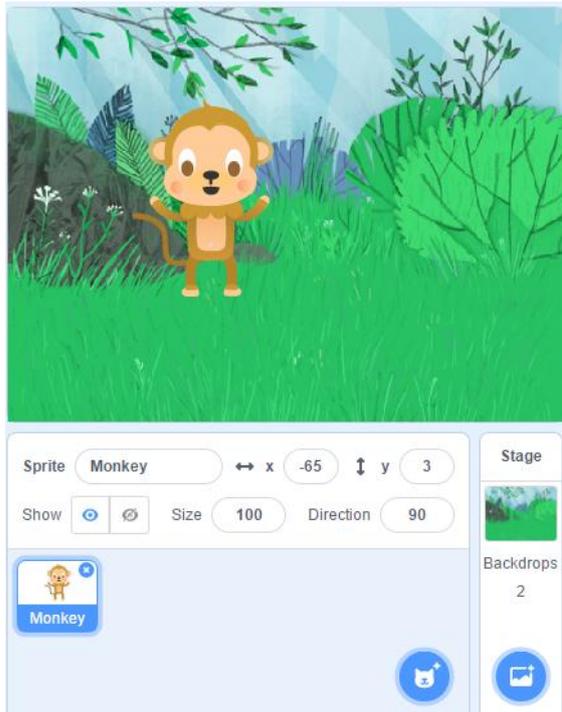
- 刪除原始精靈並創建一個新精靈。



- 選擇“Monkey”作為我們的新精靈。



- 屏幕上應該只有一個精靈。



## 5. 編程步驟

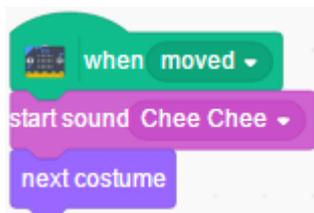
- 從“micro: bit”類別中拖動“when moved”塊。



- 組合“start sound Chee Chee”和“when moved”塊，可以從“sound”類別中找到“start sound”。



- 播放完聲音後，猴子的樣子會改變。從“Look”類別中拖出“next costume”。

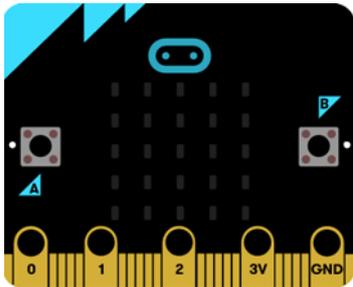


- 當 micro: bit 移動時，猴子會隨機移動。轉到“Motion”類別中拖出“glide 1 secs to random position”。



## 6. 結果

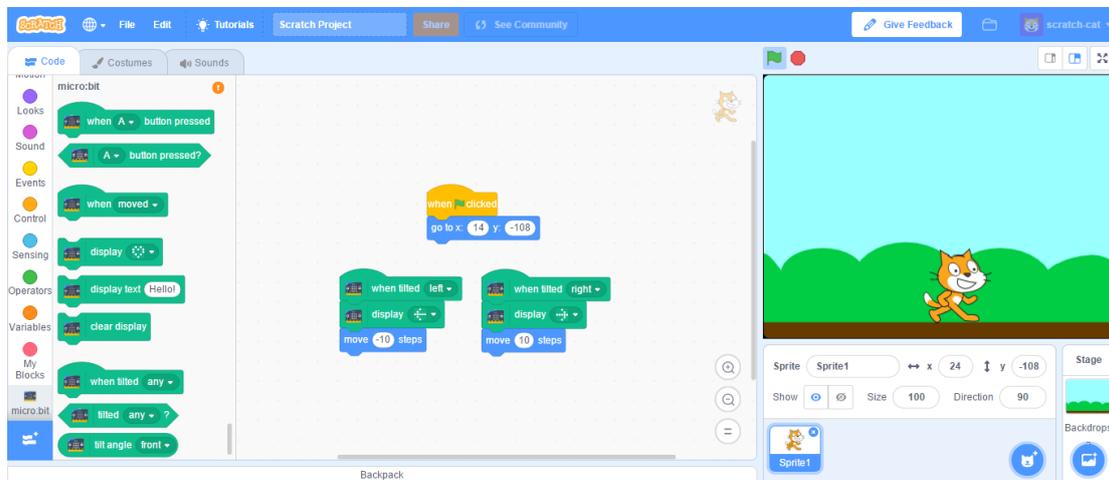
- 移動你的 micro: bit 。



- 猴子將會隨機地移動。



# 左與右的移動



程度: 初級

## 1. 你需要什麼?

- 1 x BBC micro: bit
- 1 x micro USB 線
- 1 x AAA 電池盒
- 2 x AAA 電池

## 說明：

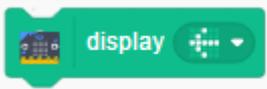
使用 micro: bit 上的傾斜功能來控制精靈貓的移動，並在 micro: bit 的 LED 顯示屏上顯示圖案。

## 2. 學習目標

- 了解 micro:bit 中加速度計與傾斜功能之間的關係
- 了解添加背景的功能

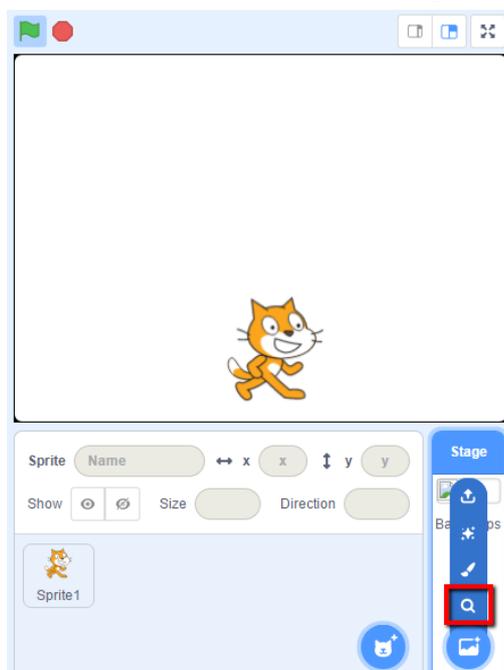
**目標：** 在本課中，我們需要選擇精靈貓的背景，並通過向左或向右傾斜 micro: bit 來控制精靈貓的移動。

### 3. 使用的方塊

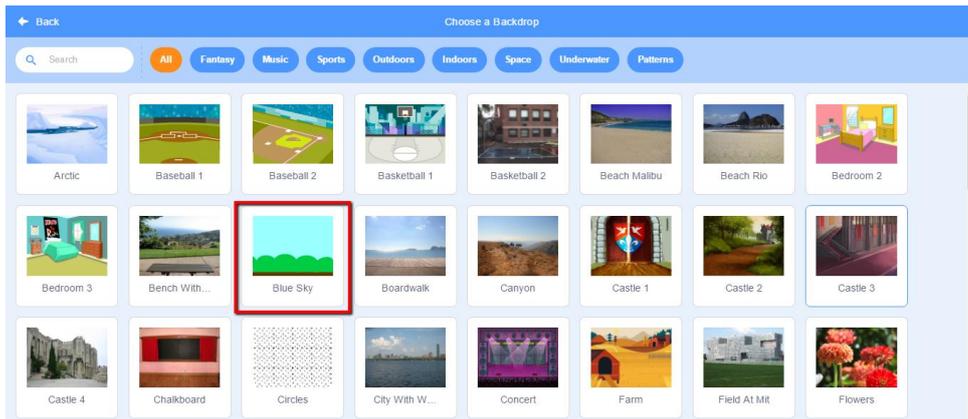
類別	塊	功能
Events		它是所有程式的開始。它代表按下“綠色標誌”時，程式將會啟動。
Motion		在相應的 x 軸和 y 軸上設置精靈。
		可以控制精靈移動的塊，方向取決於正負值。
Micro:bit		Micro: bit 將獲得以毫重為單位的加速度值（當 micro: bit 與屏幕平放時， $x = 0$ ， $y = 0$ 且 $z = -1024$ ）。你可以選擇 4 個移動方向：前傾，後退，左右傾斜。
		用於在 micro: bit 的 LED 屏幕上繪製圖像。

### 4. 設計步驟

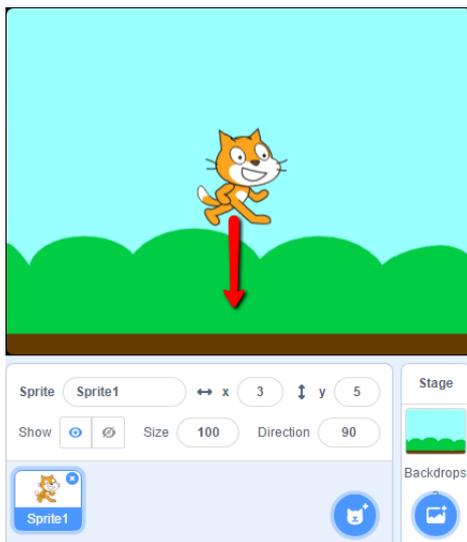
- 首先，點擊右上角的“choose a background”。



- 然後選擇你喜歡的背景，在這裡選擇“藍天”作為背景。



- 選擇背景後，將精靈拖動到適當的位置。

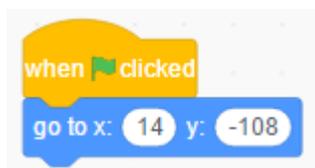


- 將精靈拖動到新位置後記錄坐標。

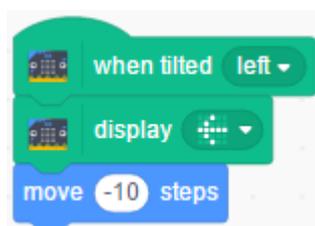


## 5. 編程步驟

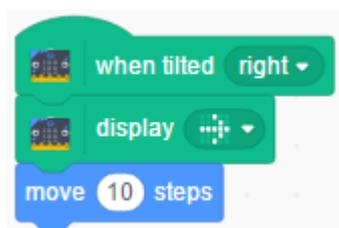
- 從“Event”類別中拖出“when flag clicked”塊在每次開始時初始化精靈的起始位置。從“Motion”類別中，將“go to x : y :”塊拖到編碼區域。坐標應該是你在上一步中記錄的數字。在這次示範中，它是 x 坐標是 14 和 y 坐標是-108。



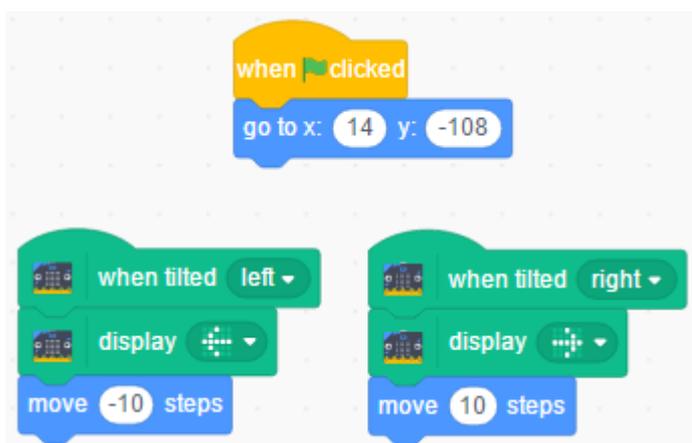
- 從“micro: bit”類別中拖出“when tilted left”，“display icon”塊，通過繪製圖形設計左箭頭圖標。然後，從“Motion”類別中拖出“move -10 steps”塊。通過向左傾斜 micro: bit 來控制精靈貓向左移動。



- 重覆步驟 2，但將“move -10 steps”更改為“move 10 steps”，通過將 micro: bit 向右傾斜來控精靈貓向右移動。



- 最後，你的程式應該是這樣：

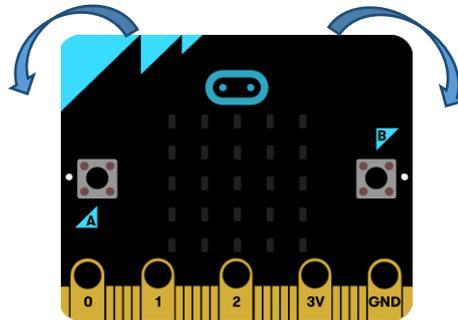


## 6. 結果

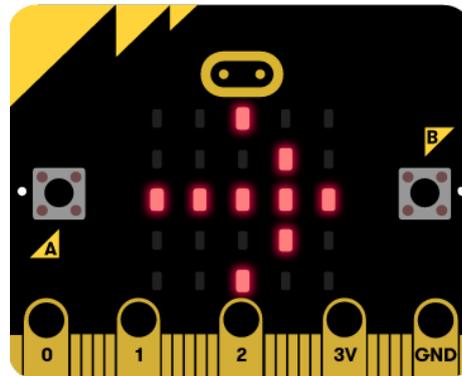
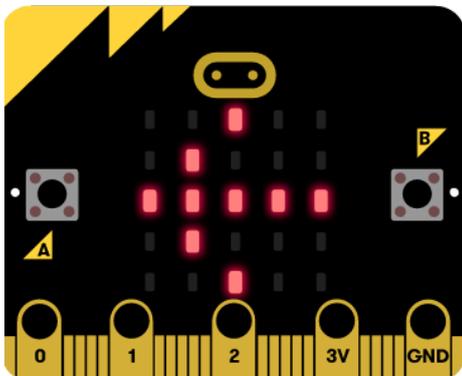
- 點擊綠色旗幟以開始將精靈貓設置為起點。



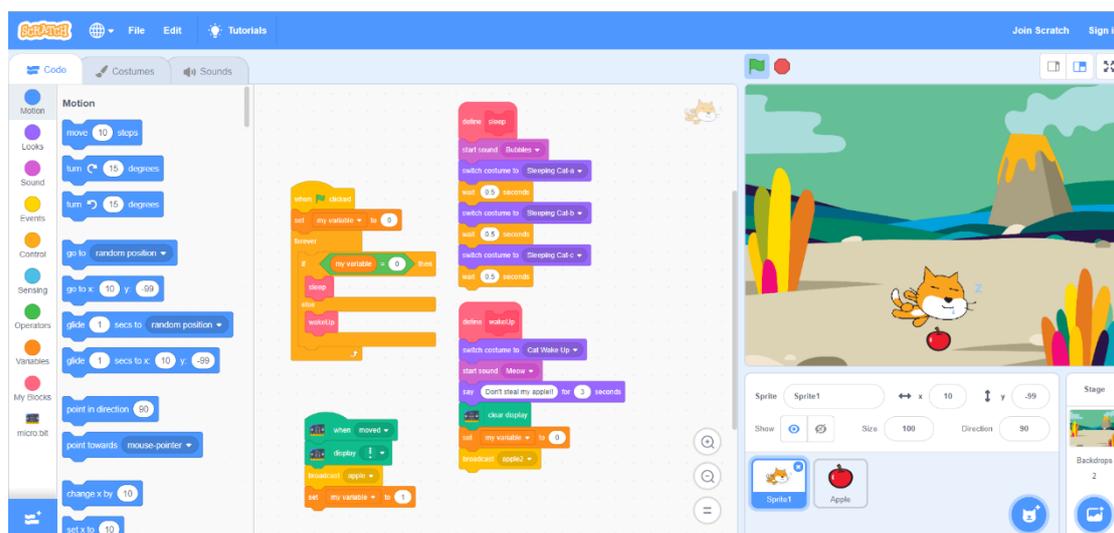
- 向左或向右傾斜 micro: bit 。



- 精靈貓將向左或向右移動，micro: bit 將顯示相應的箭頭。



# 偷取蘋果



級別：中級

## 1. 你需要什麼?

- 1 x BBC micro:bit
- 1 x micro USB 線
- 1 x AAA 電池盒
- 2 x AAA 電池

說明：

使用 Scratch 3.0 上的變數及邏輯條件控制，加上外觀的轉換，利用 micro: bit 移動功能與精靈貓進行互動。

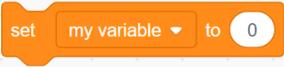
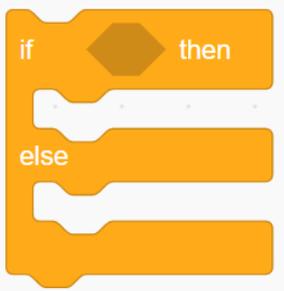
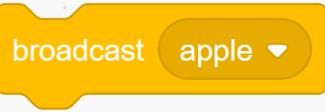
## 2. 學習目標

- 了解如何使用變數。
- 了解如何使用條件句作邏輯判斷。
- 了解如何使用函式積木。

目標：

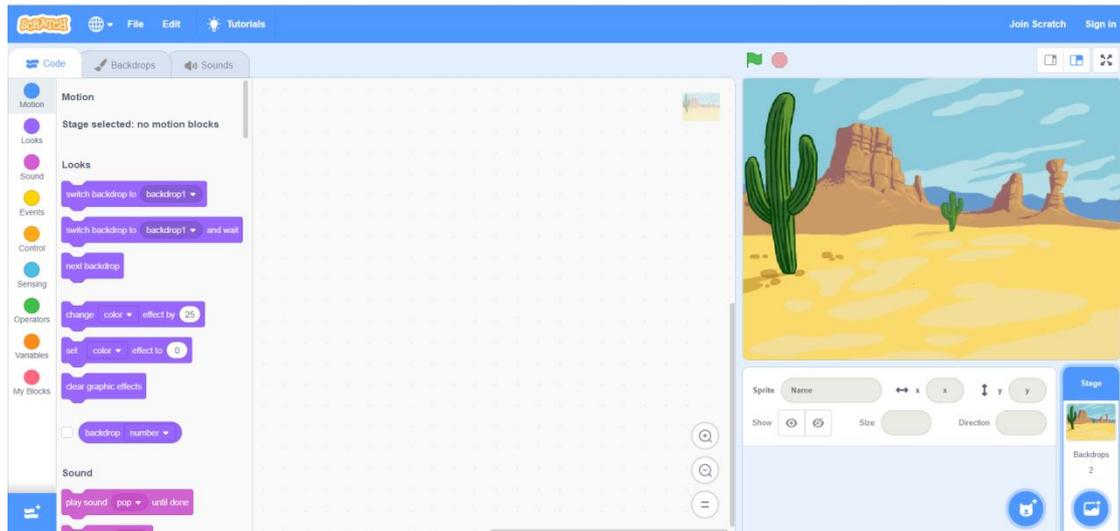
在這一章中，我們將學習如何使用變數及條件句模擬精靈貓看守蘋果的情況，加入函式積木及將精靈貓外觀轉換，配合前章的聲軌功課，利用 micro: bit 與 Scratch 進行互動。

### 3. 使用的方塊

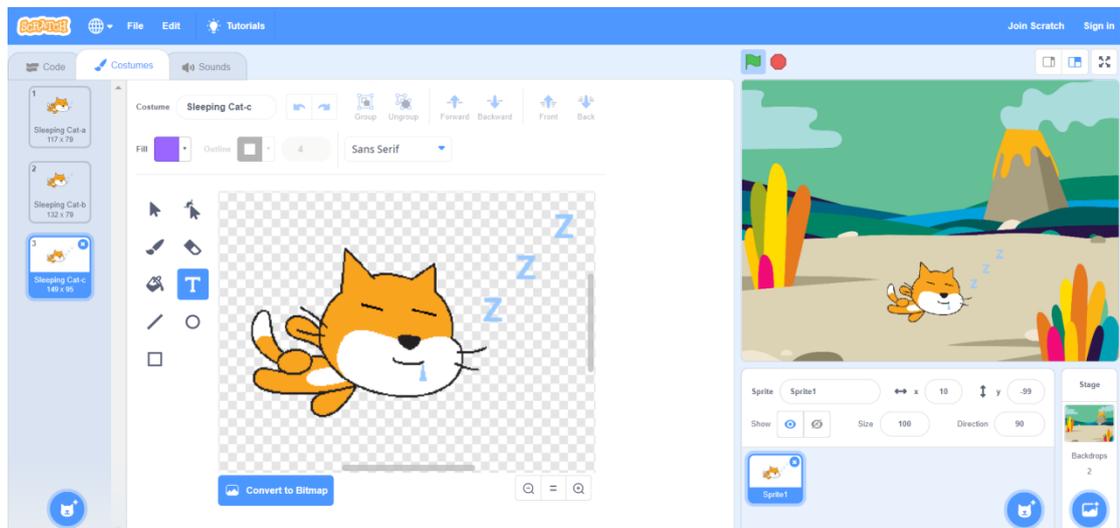
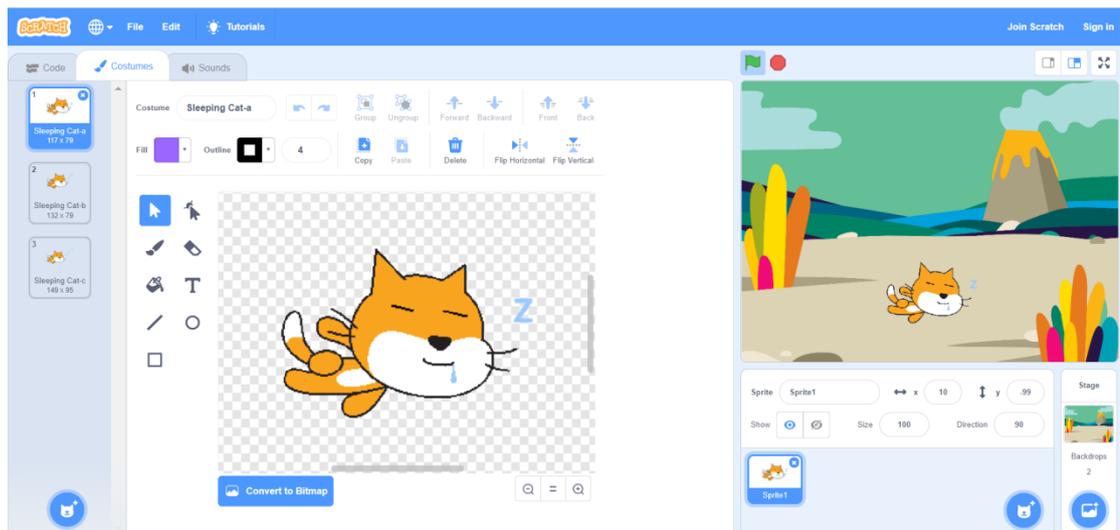
類別	塊	功能
Variable		設定變數 “my variable” 值為 0。
Control		設立多個判斷條件，如果符合 “if...then”內的條件將會進行以下的程式。如果不符合條件則進行 “else”內的程式。
Operation		設立邏輯條件，這個塊為等值條件。
My block		建立屬於自己的塊，把塊命名，將程式預設在塊內，在有需要時直接使用，減少重覆的程式。
		把預設的程式放在這塊內。
		將預設的程式呼叫出來並放在主程式內運行。
Event		廣播自定義訊息到 Scratch 並進行溝通。
		當 Scratch 收到相應的訊息時，進行下列的程式。

### 4. 設計步驟

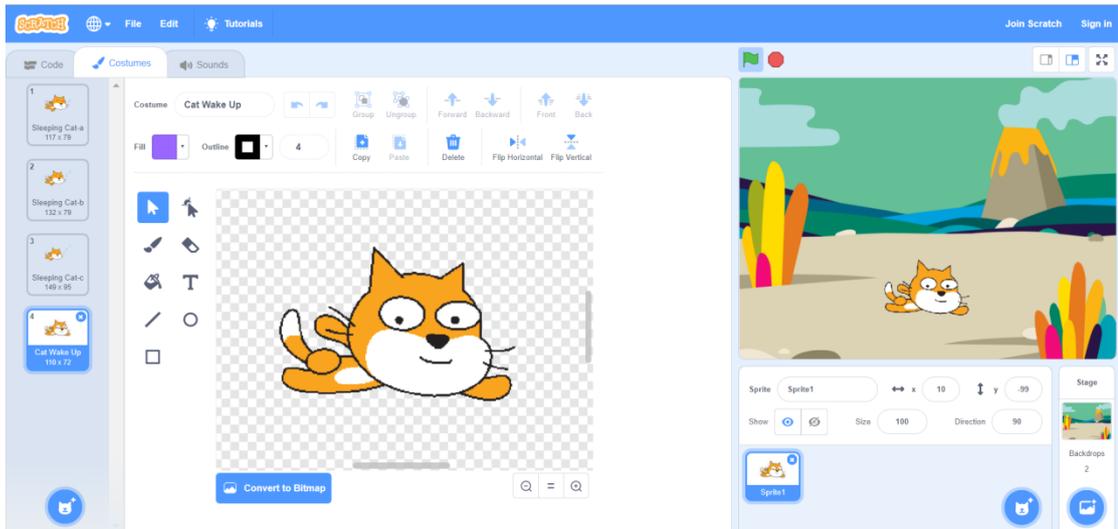
- 在本課中，我們將使用沙漠的背景。首先，點擊右下角的 “Choose a Backdrop”，然後揀選背景。



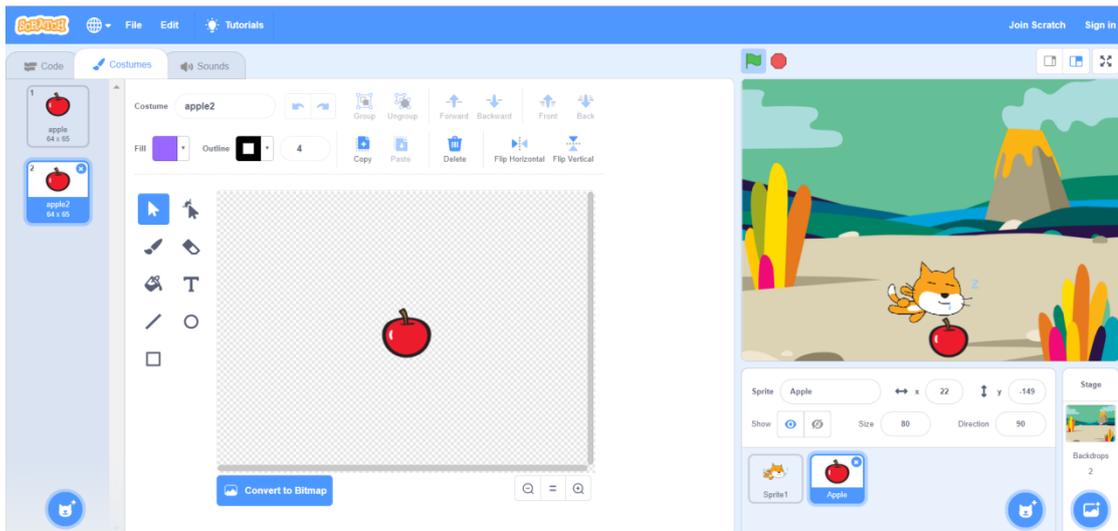
- 創建背景後，將精靈貓的身體移動至睡眠狀況，並複製三遍。單擊 “brush” 畫上精靈貓的唾液及單擊 “text” 輸入 Z，依次遞進輸入多一個 Z。



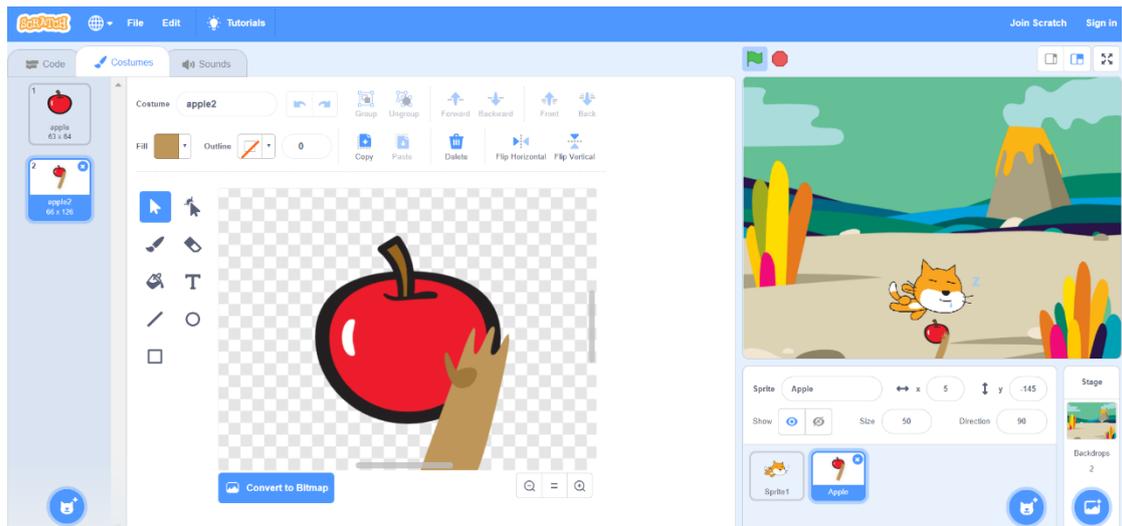
- 將精靈貓的雙眼打開為夢醒狀況。



- 創建精靈貓後，我們需要創建蘋果。單擊 “choose a sprite” 選擇蘋果，並複製一次備用。這次我們會建立兩個外貌，一個是原封不動的蘋果，一個是被偷的蘋果。把蘋果放到精靈貓的下方並調整大小。

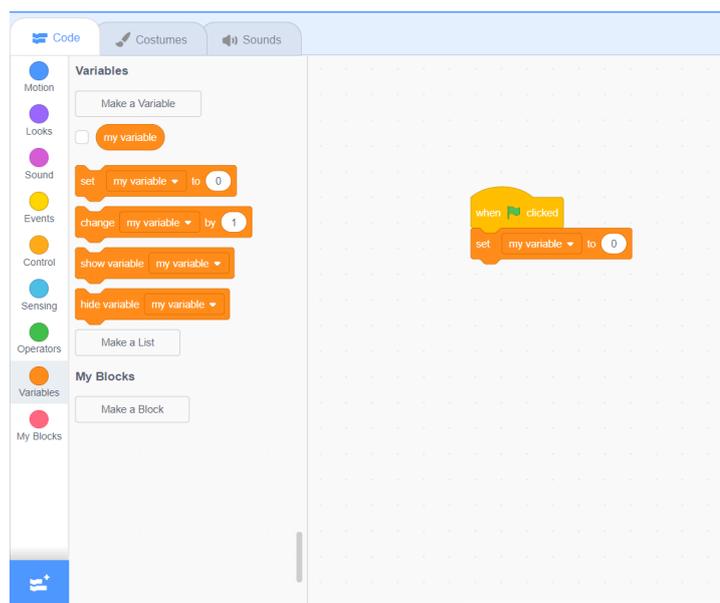


- 在第二個蘋果中，畫上一隻手拿取蘋果。設計步驟就完成，你應該有四個精靈貓的外貌，兩個蘋果的外貌及沙漠背景。

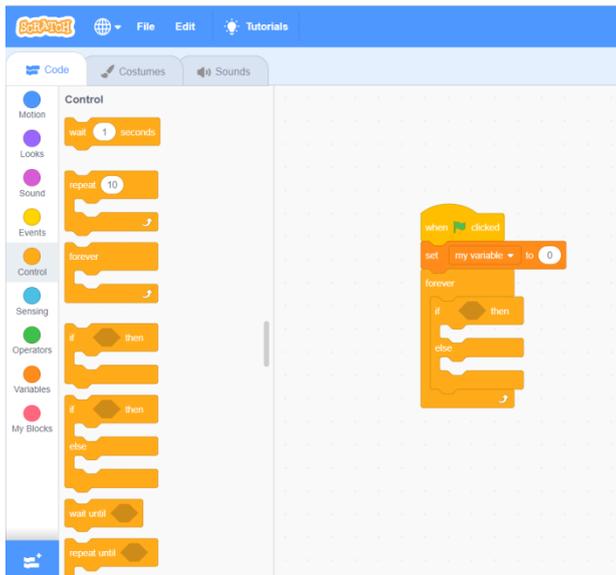


## 5. 編程步驟

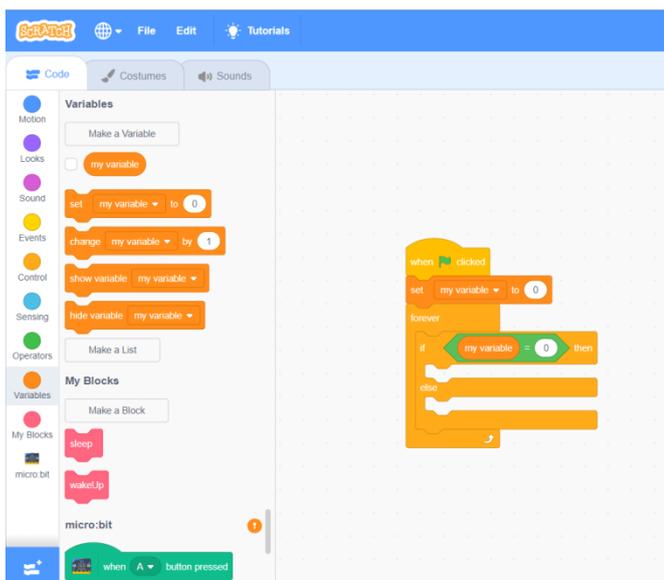
- 在開始時，先點擊 Sprite 進行編程，從 “Event” 類別中拖動 “When green flag clicked” 和 “Variable” 類別中拖動 “set my variable to 0” 塊。



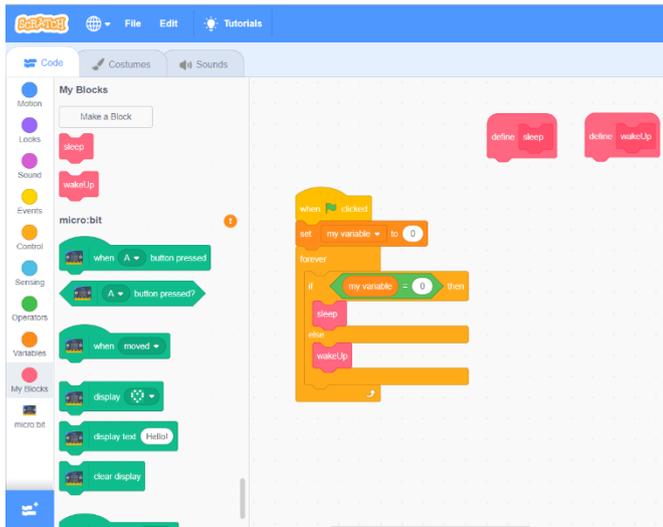
- 然後，從 “Control” 中拖動 “forever” 塊及拖動 “IF...THEN...ELSE” 塊。把 IF...THEN...ELSE” 塊放進 “forever”內。



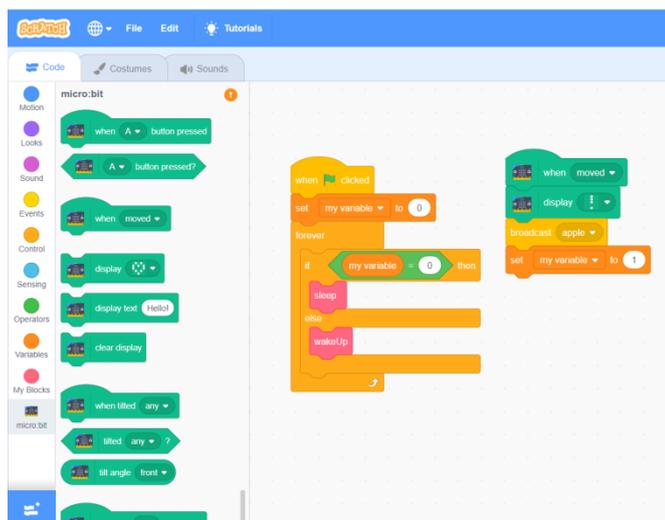
- 接下來，從 “Operation” 類別中拖動等值條件，放在 “IF...Then...Else” 內，並將 “my variable” 設定等如 0。



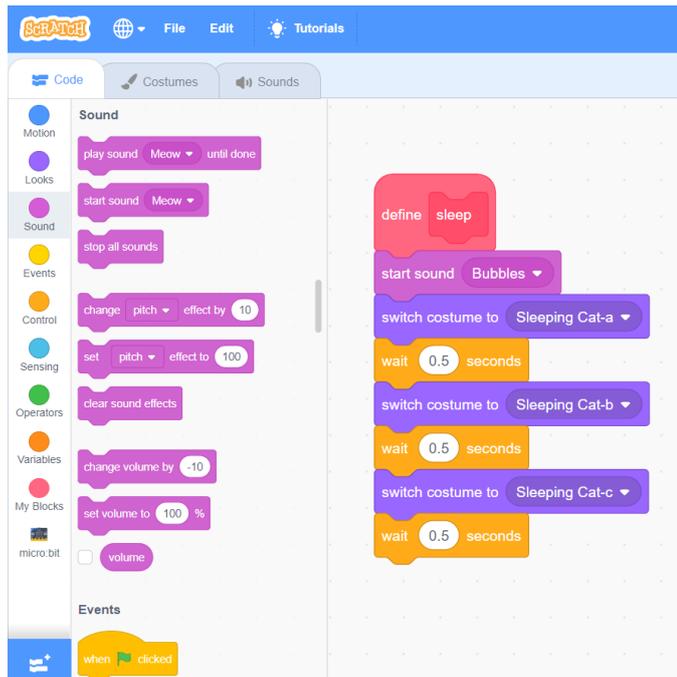
- 在 “My block” 類別中創造兩個新的塊， “sleep” 和 “wake up”。把它們放進在條件式內，編碼區域會出現兩個 “define” 的塊。



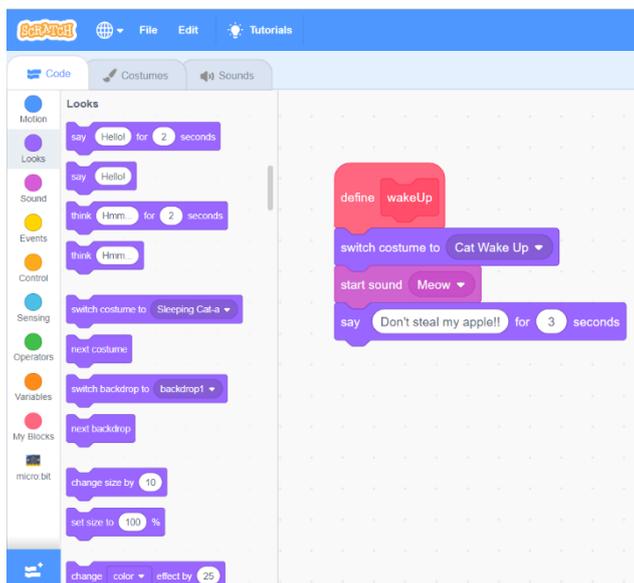
- 然後，在 “micro: bit”類別中拖動 “when moved”及 “display”顯示警告的圖案。及在 “Event”類別中發放 “apple” 訊息並於 “Variable”類別中設定 “my variable”為 1。



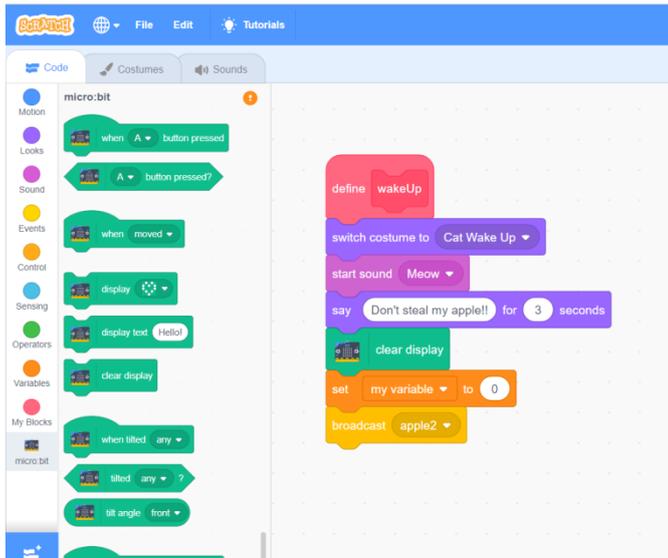
- 定義在 “sleep”內的程式指令，從 “Sound”類別中拖動 “start sound”並選擇 “bubbles”。然後在 “Looks”類別都選擇 “switch costume to”，並插入 “Control”類別中的 “wait seconds” 並設為 0.5 秒。重覆三次並依次選擇預先設定的三個外貌。



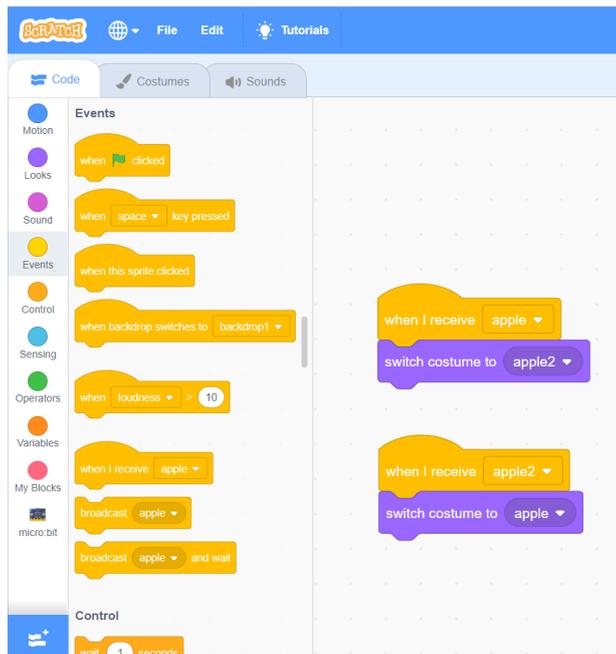
- 定義在 “wake up”的程式指令，從 “Looks”類別中拖動 “switch coustume to” 並選擇 “Cat Wake Up”，及於 “Sound”類別中拖動 “start sound” 並選擇 “Meow”。 然後從 “Looks”類別中拖動 “say for 3 seconds” 並輸入 “Don’t steal my apple!!”。



- 繼續定義 “wake up”，從 “micro: bit”類別中拖動 “clear display” 及從 “Variables”類別中拖動 “set my variable to 0”。最後從 “Event”類別中拖動 “broadcast”並設立訊息為 “apple2”。



- 最後，點擊右下角的蘋果進入蘋果的編碼區域。從“Event”類別中拖出“when I receive”塊兩次並分別選擇“apple”及“apple2”。在“when I receive apple”下插入從“Looks”類別中的“switch costume to”並選擇“apple2”。而在“when I receive apple2”下插入“switch costume to apple”。



## 6. 結果

- 在 micro: bit 靜止時，精靈貓咕嚕咕嚕熟睡。



- 當 **Micro: bit** 被移動時，偷取蘋果的手出現，精靈貓驚醒並警告不要偷牠的蘋果！

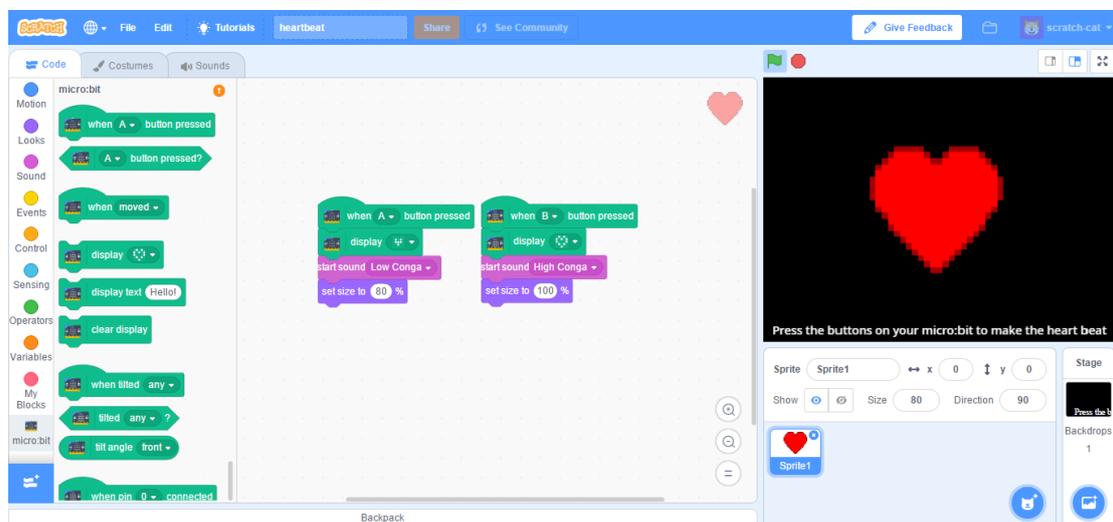


- 直到停止移動 **micro: bit** 時，懶惰的精靈貓又再次睡覺。



現在，你可以利用 **Micro: bit** 嘗試把精靈貓的蘋果拿走！

# 心的跳動



級別：初級

## 1. 你需要什麼？

- 1 x BBC micro:bit
- 1 x micro USB 線
- 1 x AAA 電池盒
- 2 x AAA 電池

說明：

使用 micro:bit 上的按鈕控制 Scratch 上圖像和聲音，並在 micro:bit 上的 LED 顯示屏上顯示圖案。

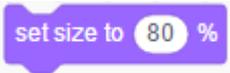
## 2. 學習目標

- 了解如何在 Scratch 上添加聲音。
- 了解如何改變精靈的大小。
- 嘗試設計自己的精靈。

目標：

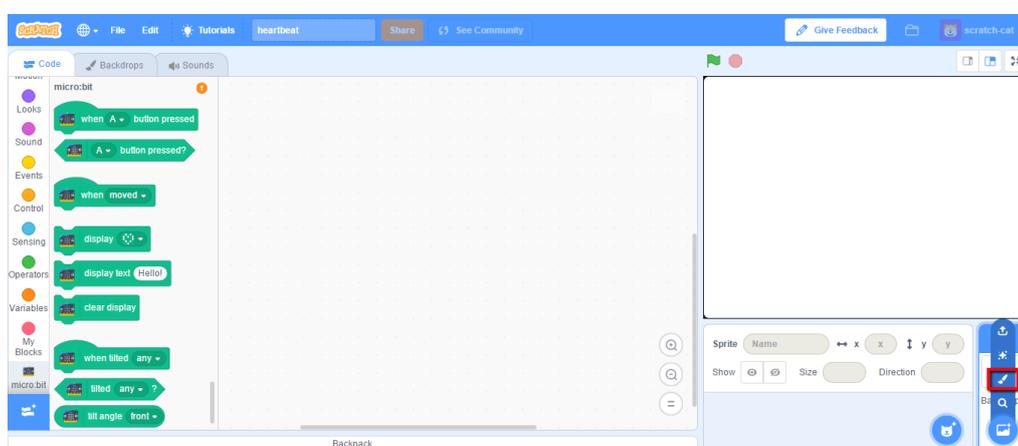
在這一章中，我們將學習如何把聲音軌道添加到 Scratch 上，並通過與 micro:bit 互動來播放。另外，與前一章不同，我們不再使用默認的精靈，新的精靈將由你自己設計。

### 3. 使用的方塊

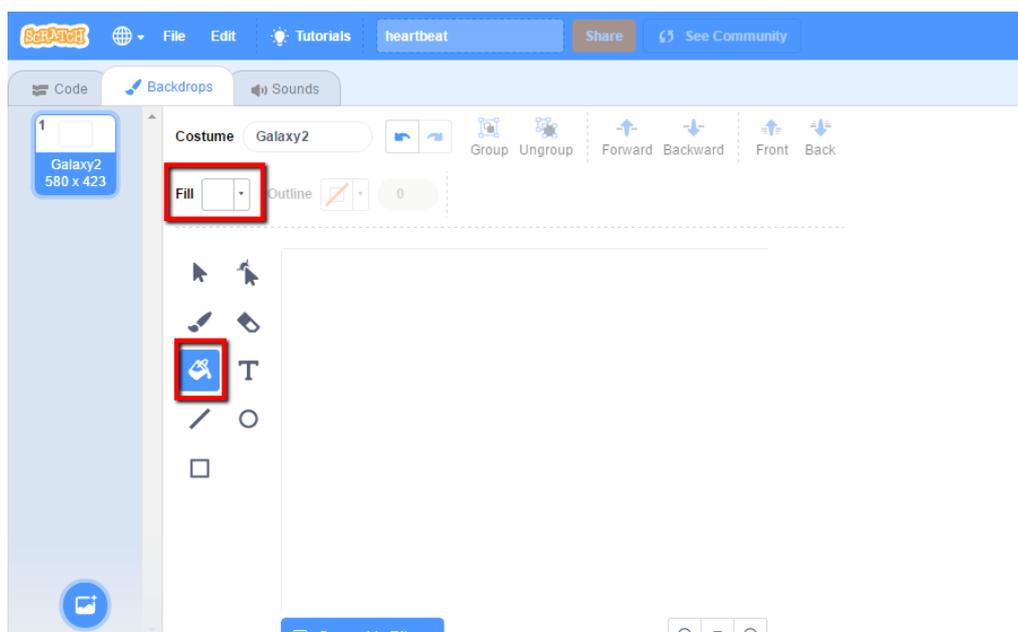
類別	塊	功能
Looks		“set size to”可以更改精靈的大小百分比。

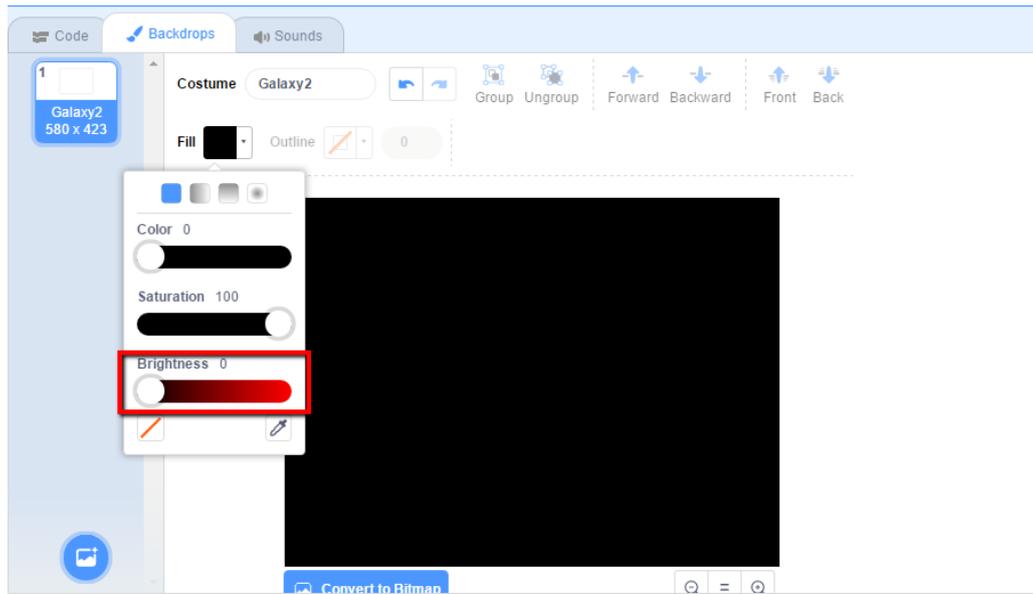
### 4. 設計步驟

- 在本課中，我們將使用黑色背景。首先，點擊右上角的“Choose a Backdrop”，然後點擊“Paint”。

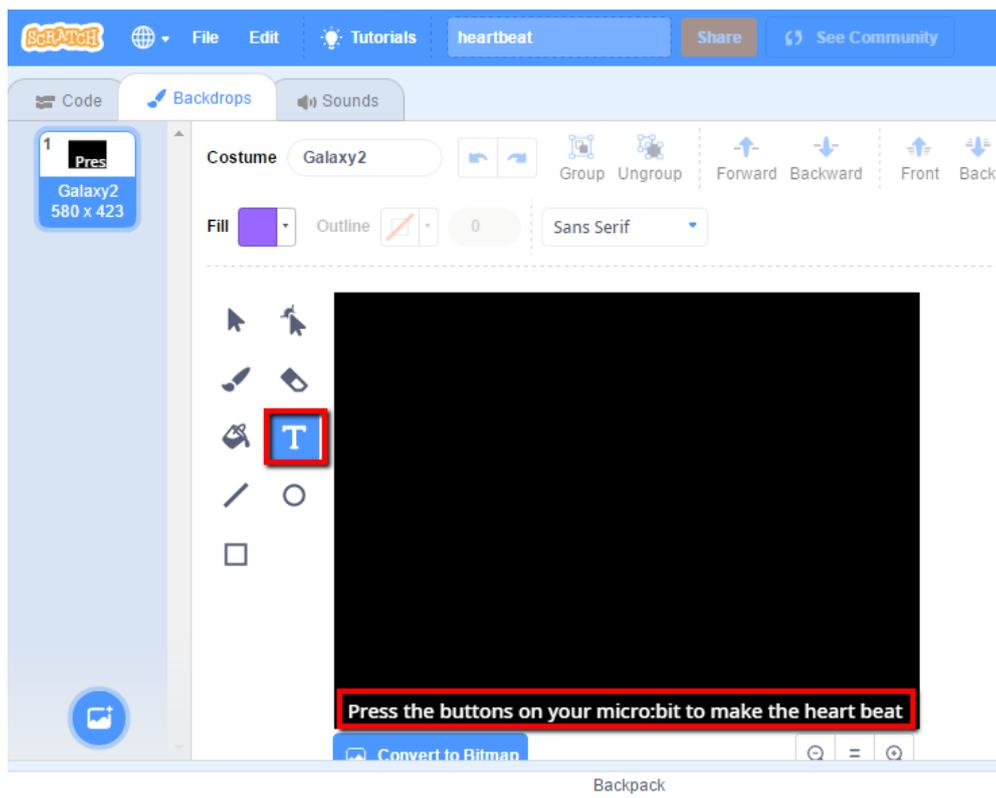


- 選擇“Fill”並選擇黑色。你可以在“Fill”中將亮度設置為0。

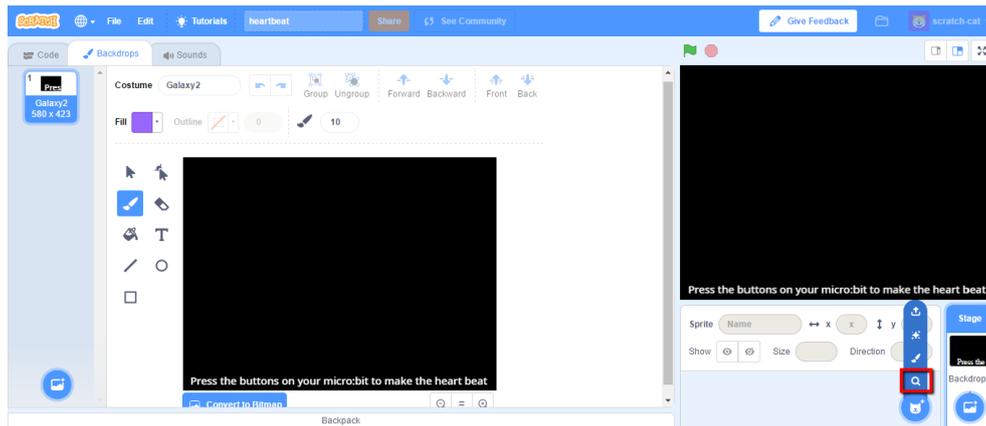




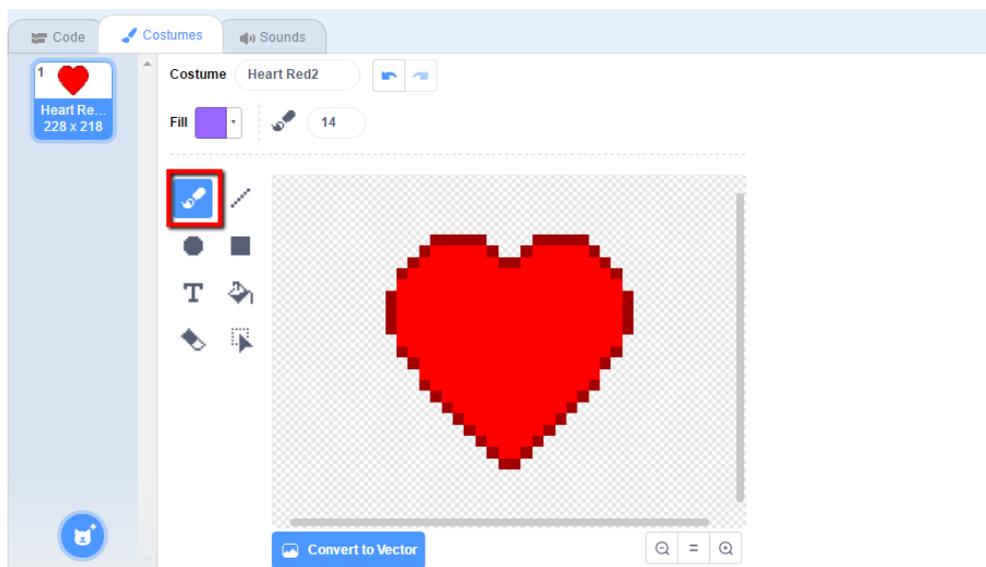
- 單擊 “Text” 並選擇一個地方來插入文字：“Press the buttons on your micro:bit to make the heart beat”。



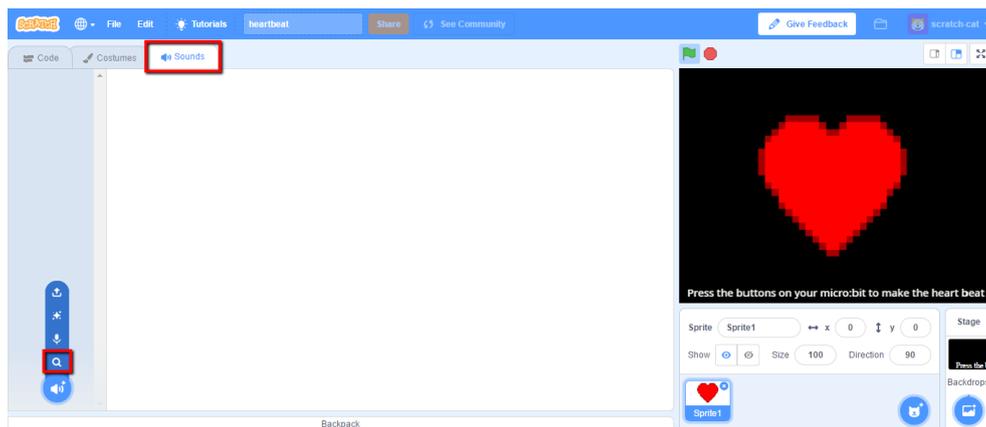
- 創建背景後，我們需要創建精靈。與前一章不同，我們不使用 scratch cat，我們將設計我們的新精靈。 點擊 “Choose a Sprite” 並選擇 “Paint”。



- 用“brush”畫出自己風格的心。



- 然後，我們要開始製作音軌。轉到“Sounds”頁面，然後點擊“Choose a Sound”。

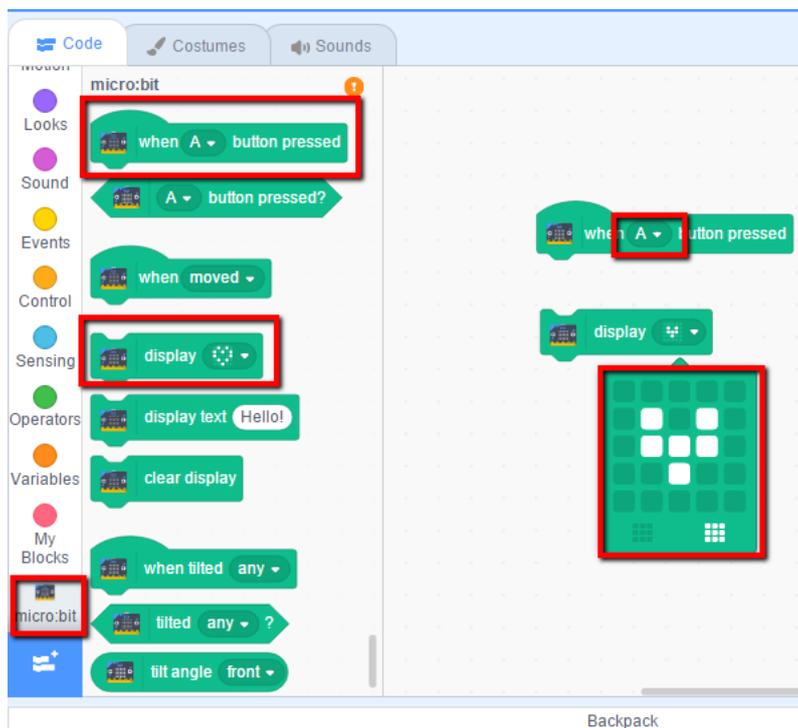


- 在“Choose a Sound”頁面中，搜索“Conga”並選擇“Low Conga”。然後再增加多一條音軌，但這次選擇“High Conga”。

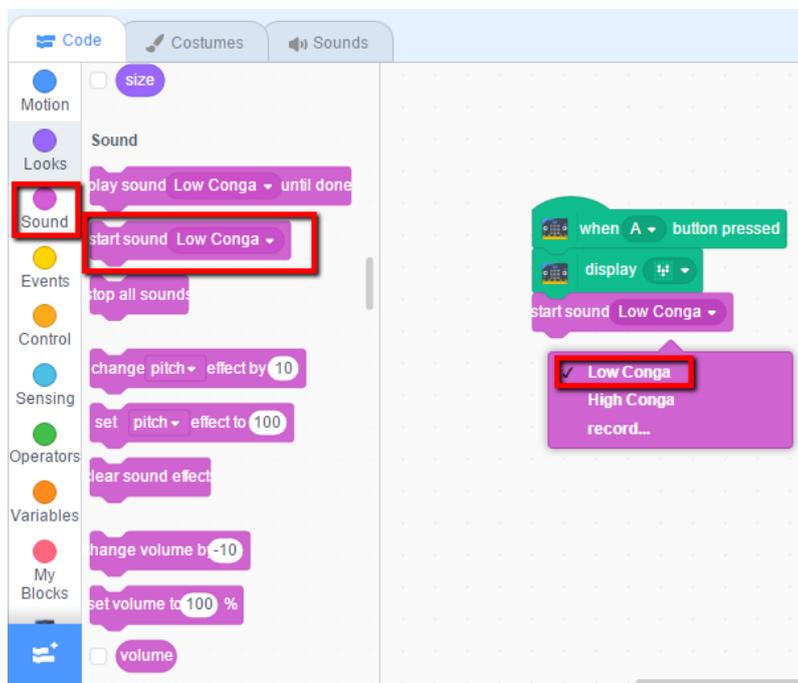
## 5. 編程步驟

- 在開始時，先點擊 Sprite 進行編程，從 micro:bit” 類別中拖動 “When A button

pressed” 和 “display” 塊。在 “display” 塊中，我們繪製一個小心心作為顯示圖案。



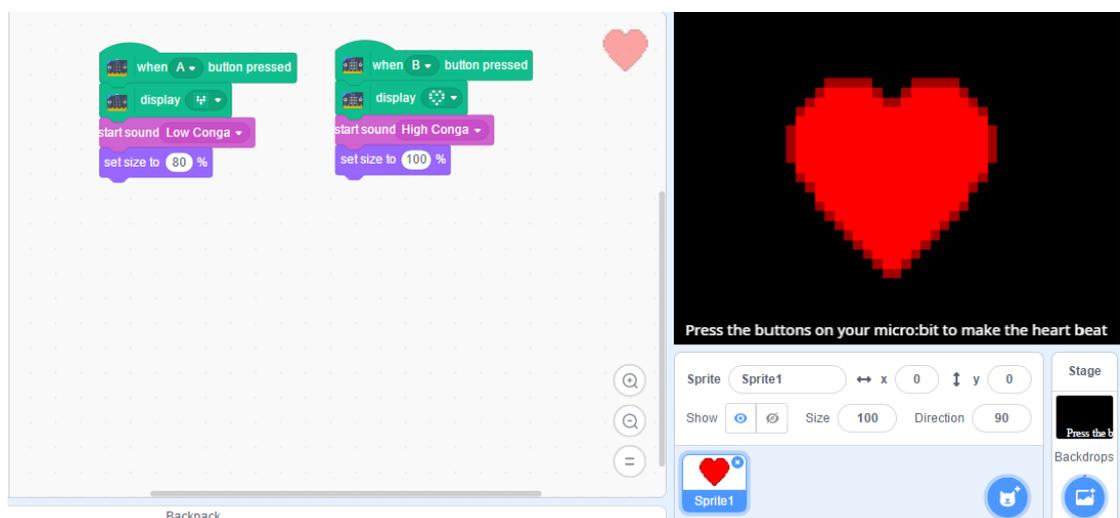
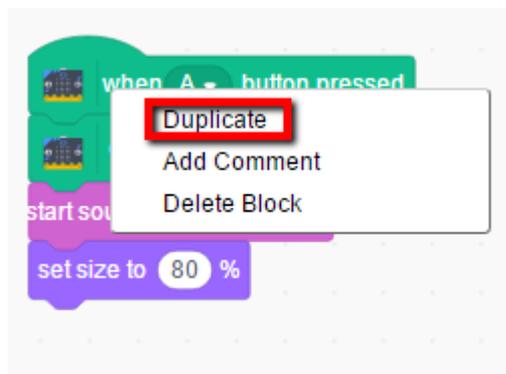
- 然後，從 “Sound” 中拖動 “start sound” 塊，在 “start sound” 塊中選擇 “Low Conga” 。



- 接下來，從 “Looks” 中拖動 “set size to %” ，將大小設置為 “80” %，運行此塊後，精靈的大小將變小。

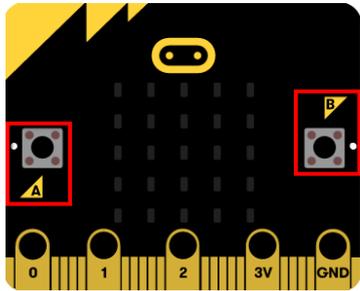


- 最後，右鍵單擊並選擇“Duplicate”以複製所有塊。然後將按鈕更改為“B”，顯示“大心心”，播放“High Conga”聲音並將大小設置為“100”%。

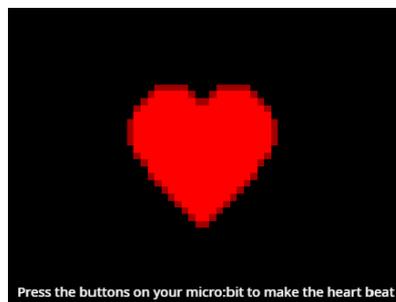
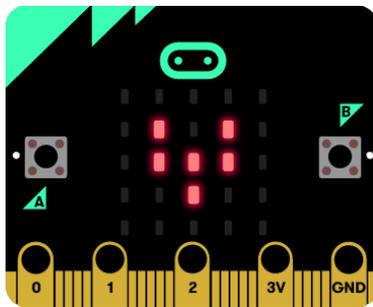
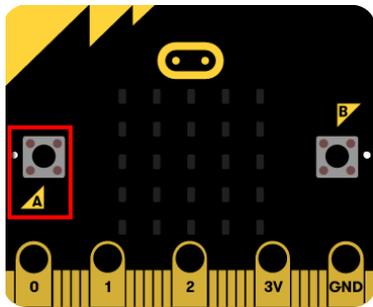


結果:

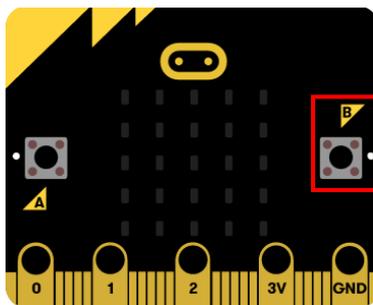
- 按下 micro:bit 上的 A 或 B 按鈕。

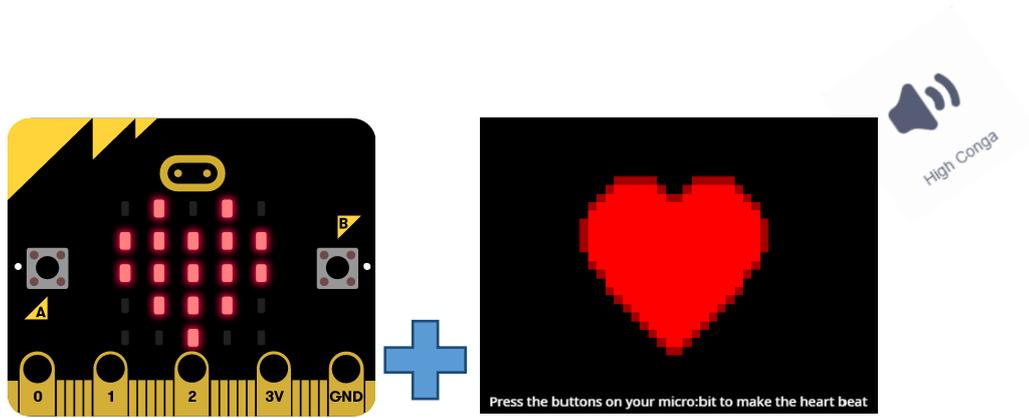


- 如果按下 A 按鈕，micro:bit 將顯示一個小心心，Scratch 3.0 將播放 “Low Conga” 聲音，精靈的大小將變小。



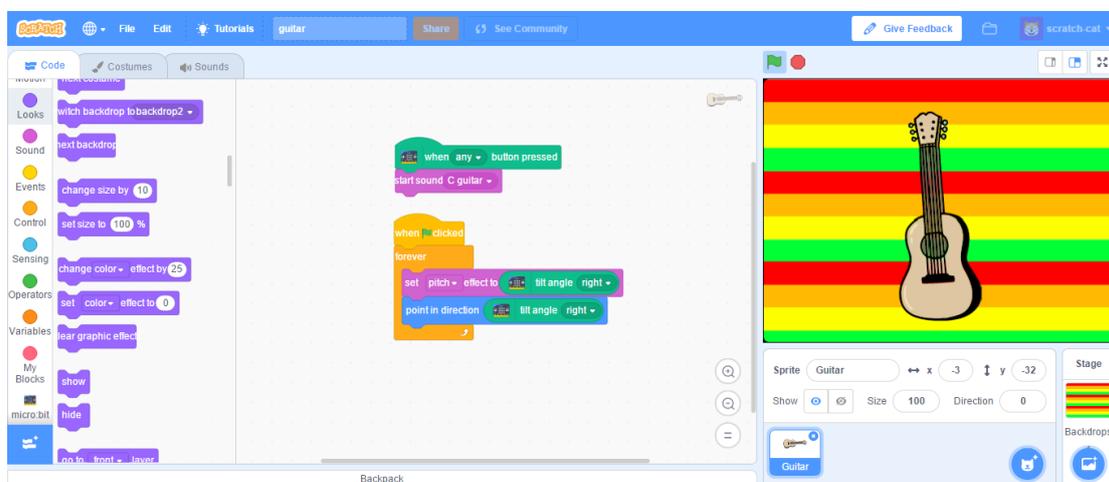
- 如果按下 B 按鈕，micro:bit 將顯示一個大心心，Scratch 3.0 將播放 “High Conga” 聲音，精靈的大小將變大





- 現在，你可以不停按 A 和 B 按鈕來做出心跳效果！

# 傾斜吉他



級別：中級

## 1. 你需要什麼?

- 1 x BBC micro:bit
- 1 x micro USB 線
- 1 x AAA 電池盒
- 2 x AAA 電池

說明：

使用 micro:bit 上的傾斜功能來控制背景顏色，聲音音高和 Scratch 上的移動。

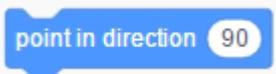
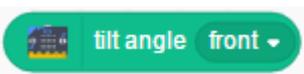
## 2. 學習目標

- 了解如何使用 micro:bit 更改音高和顏色。
- 了解迴圈(Forever loop)的使用。
- 掌握有關數值變化的知識。

目標：

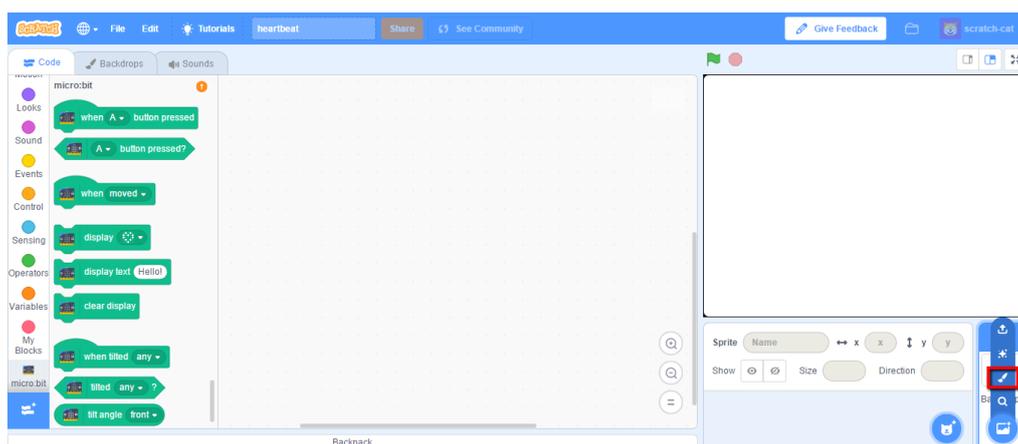
在這一章中，我們將學習如何根據你的控制製作出色彩繽紛，多變的背景和可變的音高吉他。因此，在本課中，你將使用傾斜角度作為可變值來控制顏色和聲音。

### 3. 使用的方塊

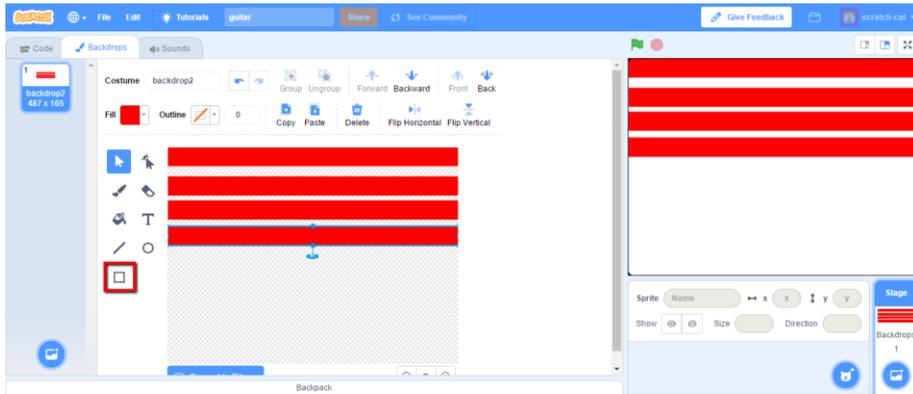
類別	塊	功能
Control		<b>Forever loop</b> ，是指把指令一直重複而不會結束，是“迴圈”類型之一。
Sound		可以根據值而改變音高，較大的數字表示較高的音高。
Motion		可以根據值使精靈指向適當的方向，值的數目等於角度， <b>90</b> 表示直指向前。
Looks		每種顏色都有自己的值，如果改變數值，顏色會發生變化，所以這個塊允許我們根據值而改變顏色。
Micro:bit		根據 <b>micro:bit</b> 中的 <b>x 軸</b> 和 <b>y 軸</b> 獲得傾斜角度值。

### 4. 設計步驟

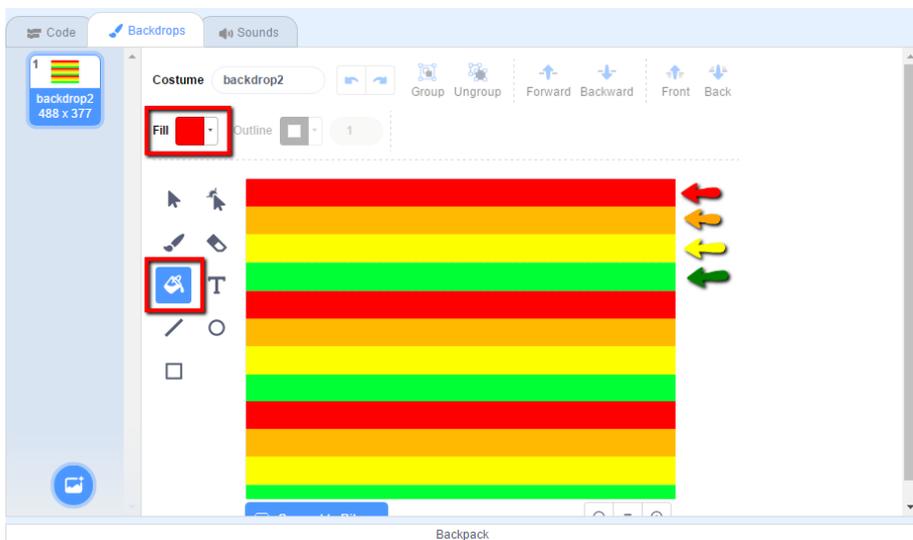
- 與之前相同，我們首先創建一個背景。單擊右下角的“Choose a Backdrop”，然後單擊“Paint”。



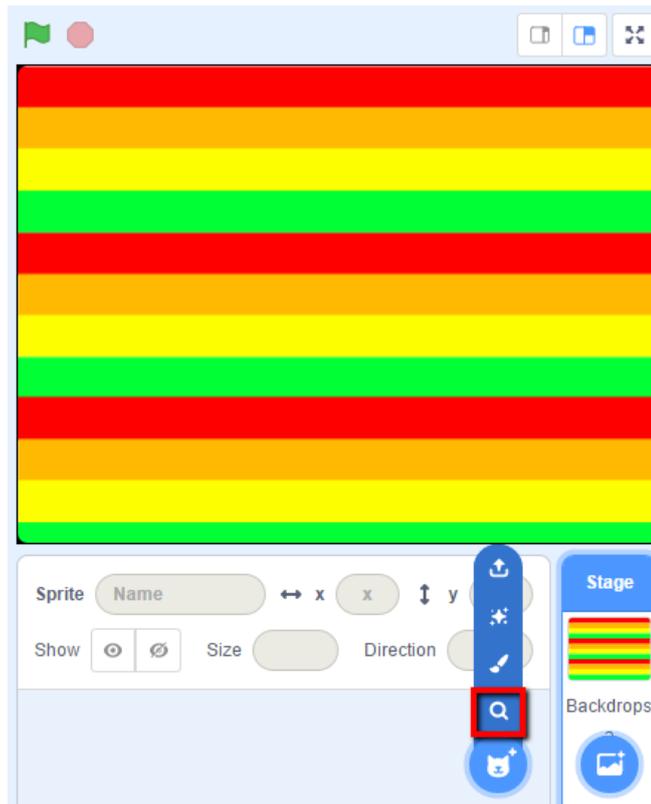
- 單擊“Rectangle”以先繪製直條。然後“複製並粘貼”以創建相同大小的條形並填滿整個背景。



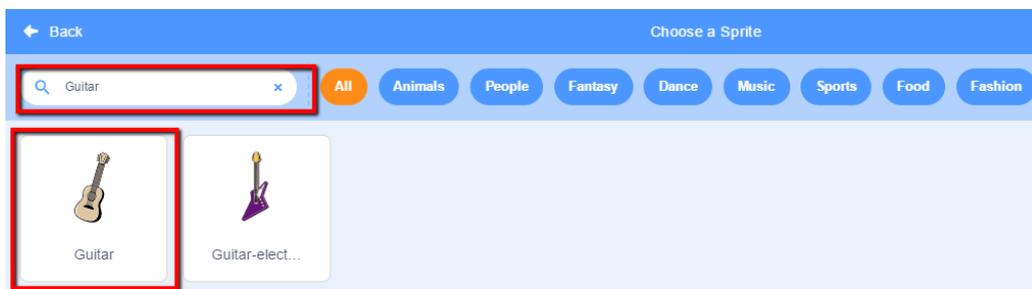
- 之後，你可以將每條長條更改為不同的顏色。（提示：嘗試為每個條形使用相似的颜色，例如：紅色>橙色>黃色>綠色，它會使它更漂亮。）



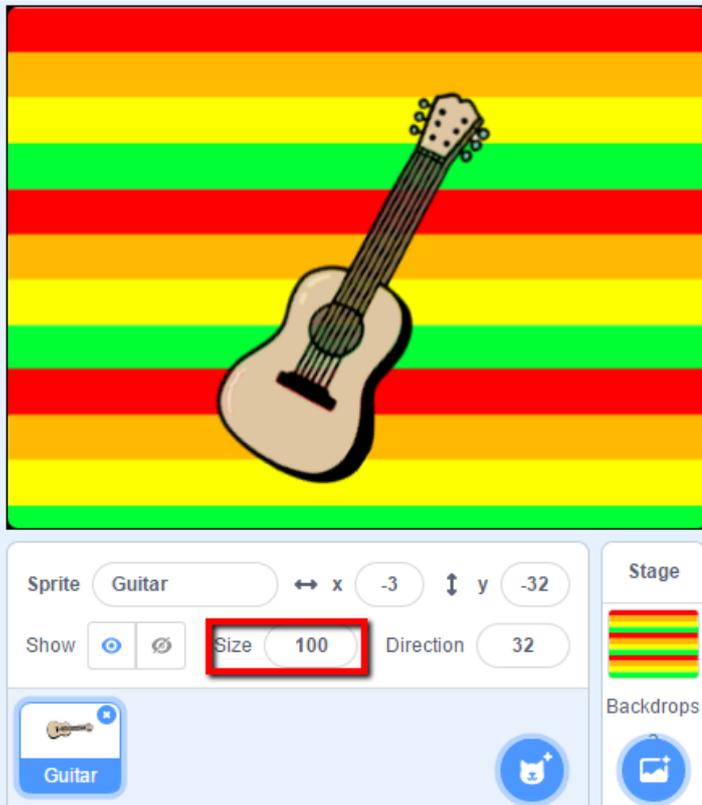
- 這次，我們將使用 Scratch 3.0 提供的默認精靈。 首先單擊“Choose a Sprite”。



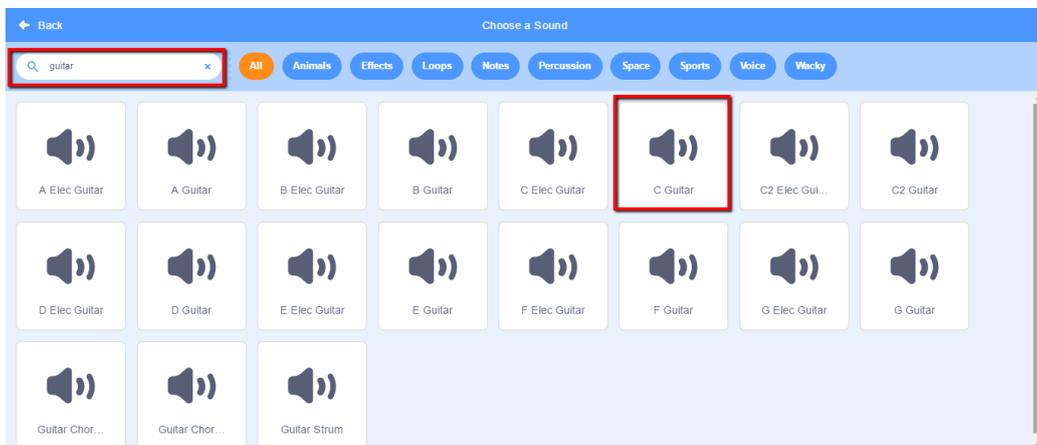
- 在搜索欄中搜索“Guitar”並選擇它。



- 設置精靈的大小並將其放在中心。



- 對於聲道，單擊“Sounds”頁面，然後單擊“Choose a Sound”。搜索“Guitar”並選擇“C guitar”。

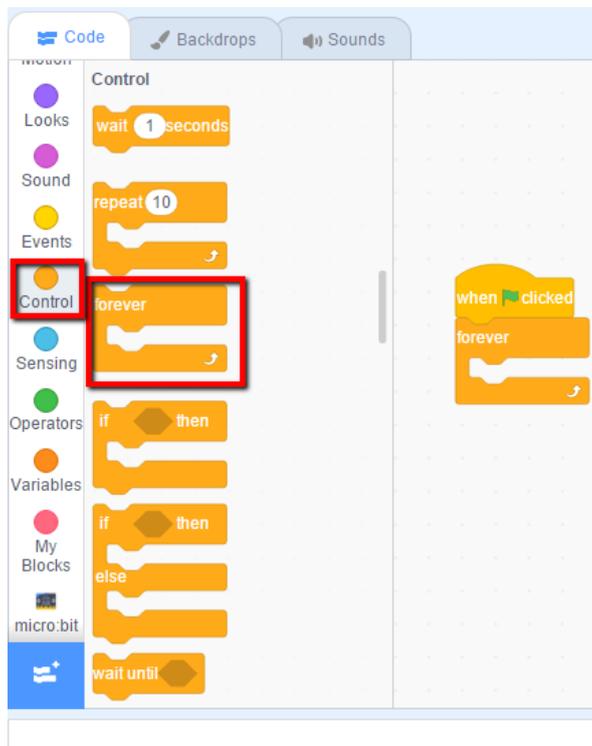


## 5. 編程步驟

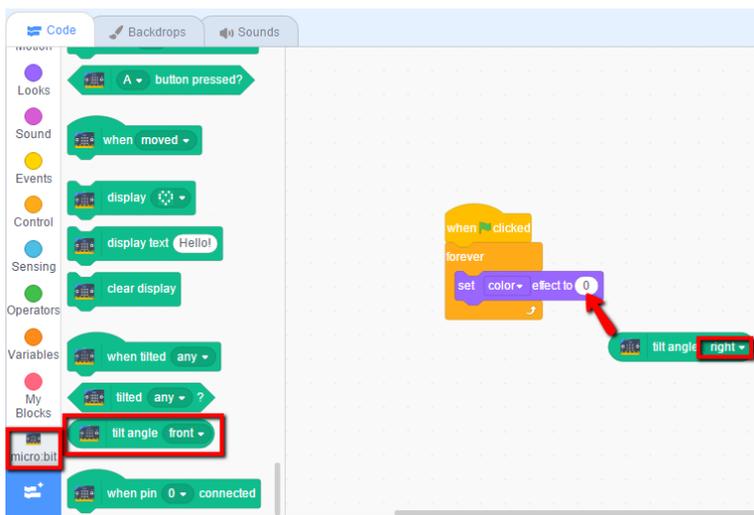
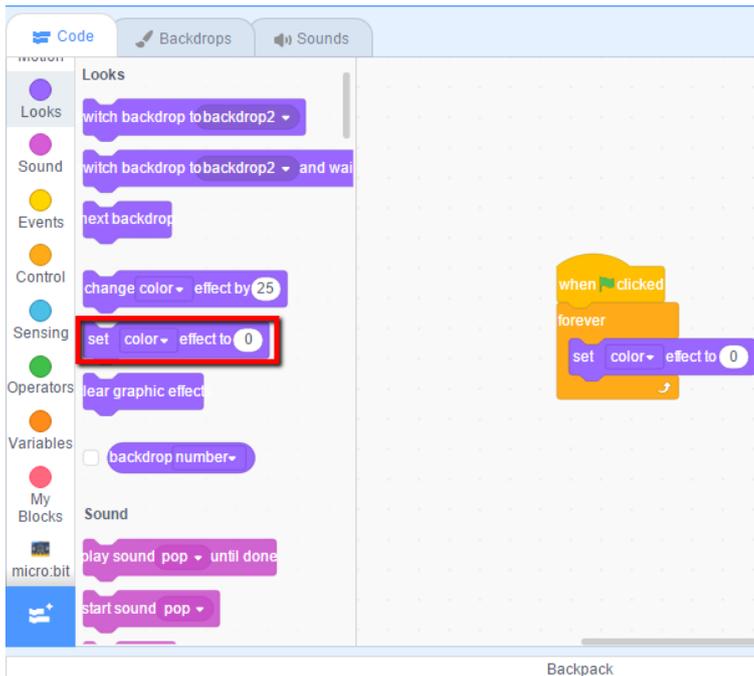
- 在本課中，我們需要編寫兩個部分：精靈和背景。對於背景，我們需要做變色效果。因此，我們首先到背景的編程區域並從“Events”中拖動“When Clicked”塊。



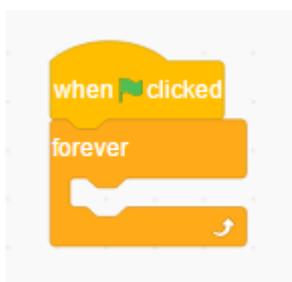
- 然後從“Control”中拖動“Forever”塊，以便在點擊綠色標記後，將啟動永久循環。



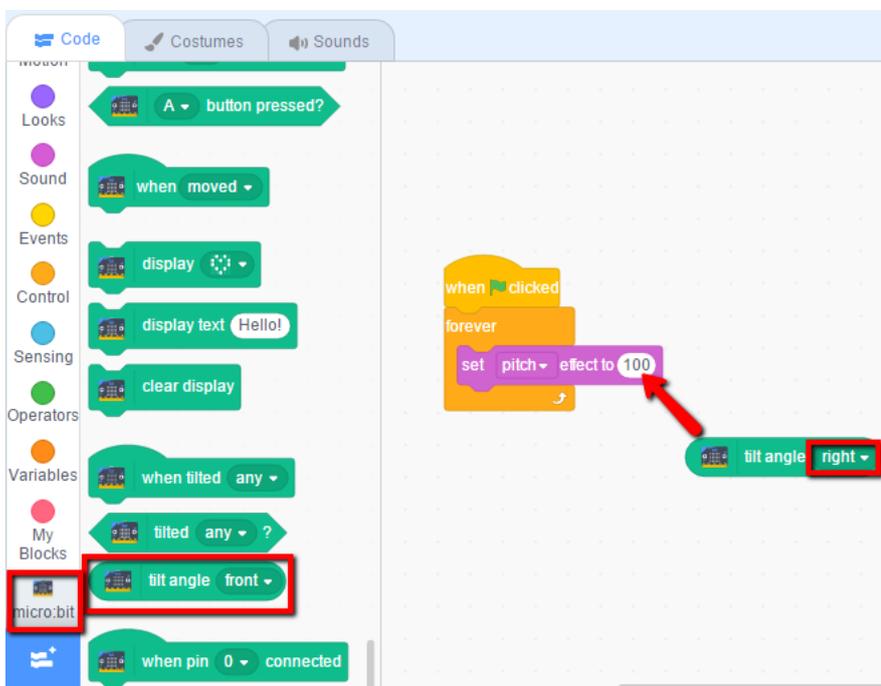
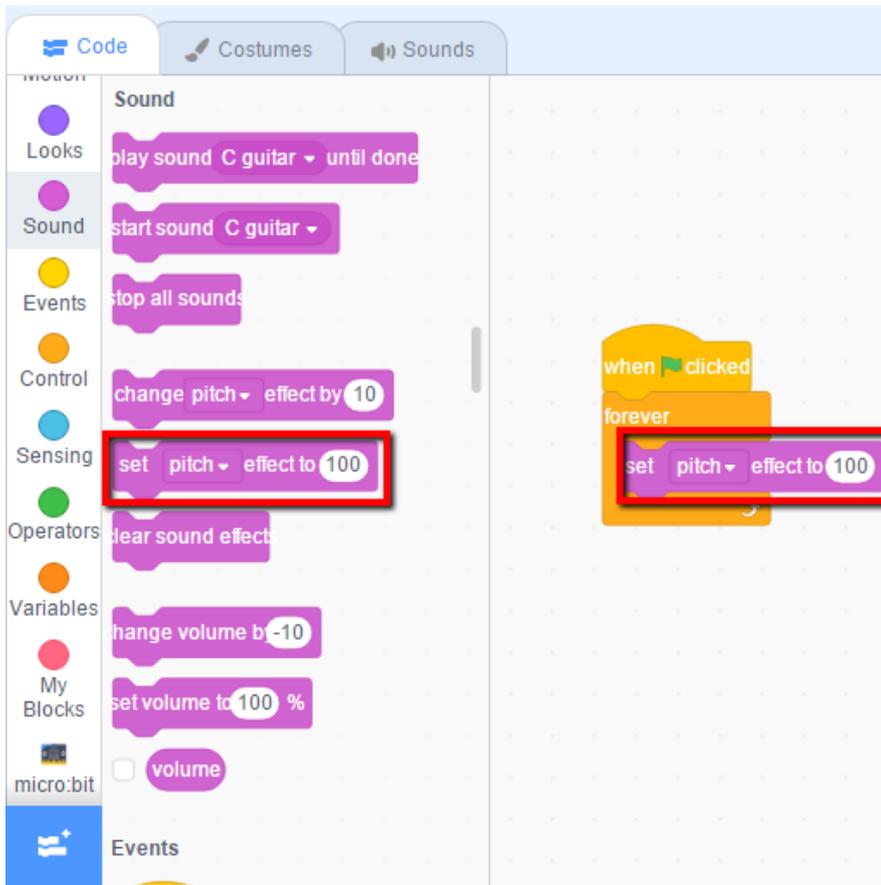
- 在永久循環中，我們使用“set the color effect to”來改變背景的颜色，把“tilt angle right”成為“set the color effect to”塊的值。因此，micro:bit 的角度變化將改變背景颜色。



- 對於吉他精靈，與背景相同，我們首先使用“when clicked”和“forever”。



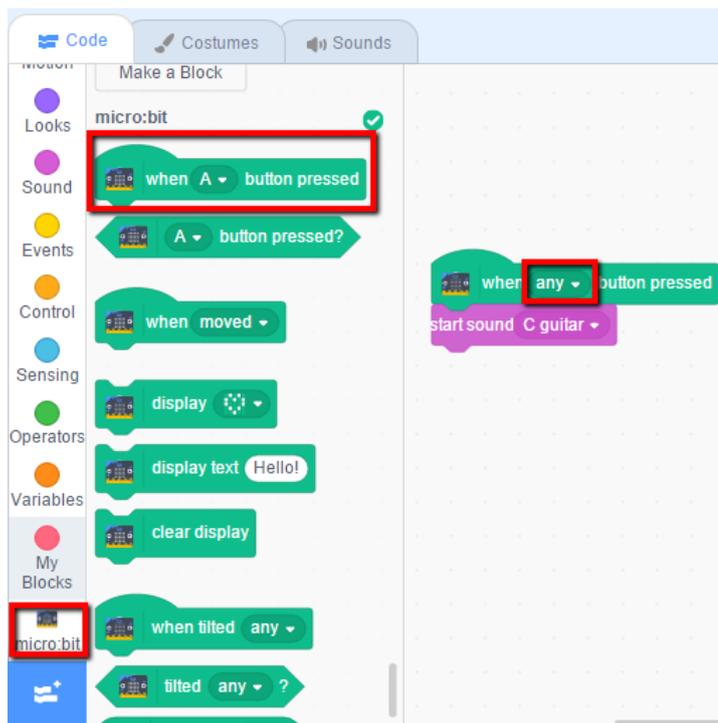
- 在迴圈中，我們使用“set pitch effect to”來改變音高，該值也將使用 micro:bit 的傾斜角。因此，我們將“tilt angle right”塊拖動到“set pitch effect to”塊的值。

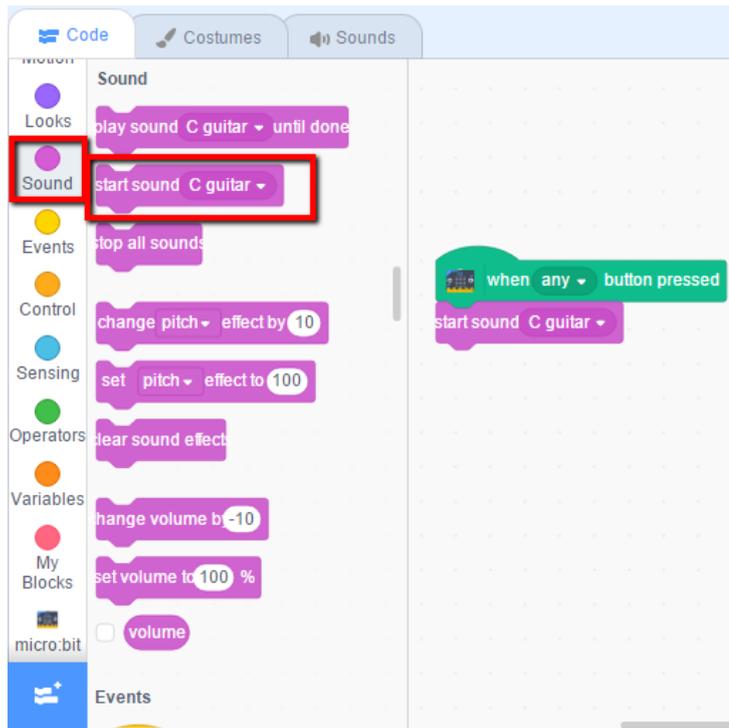


- 除聲音外，精靈的角度也受傾斜角度的影響。因此，我們使用來自“motion”的“point in direction”並把傾斜角度作為值。



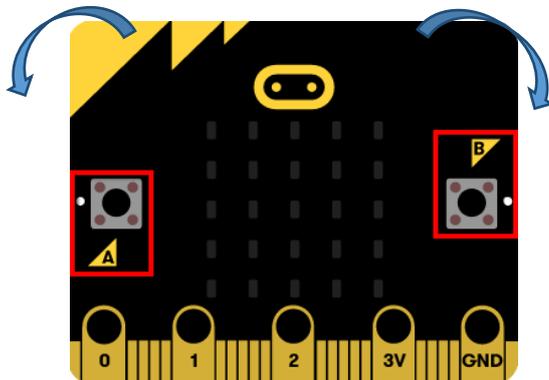
- 此外，當我們按下任何按鈕時，Scratch 3.0 將播放吉他聲音。因此，我們將“**When any button pressed**”和“**start sound**”拖動到編碼區域時，選擇聲音為“**C guitar**”。



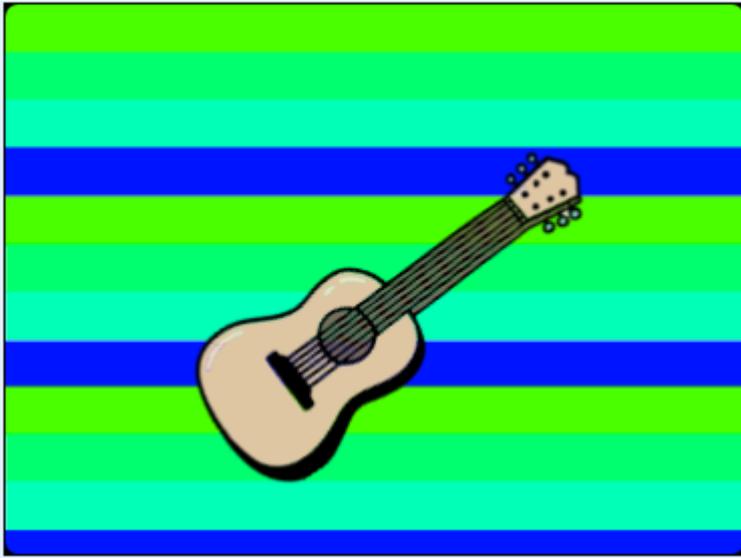


## 6. 結果

- 按下 micro:bit 上的任意按鈕，向左或向右傾斜 micro:bit。

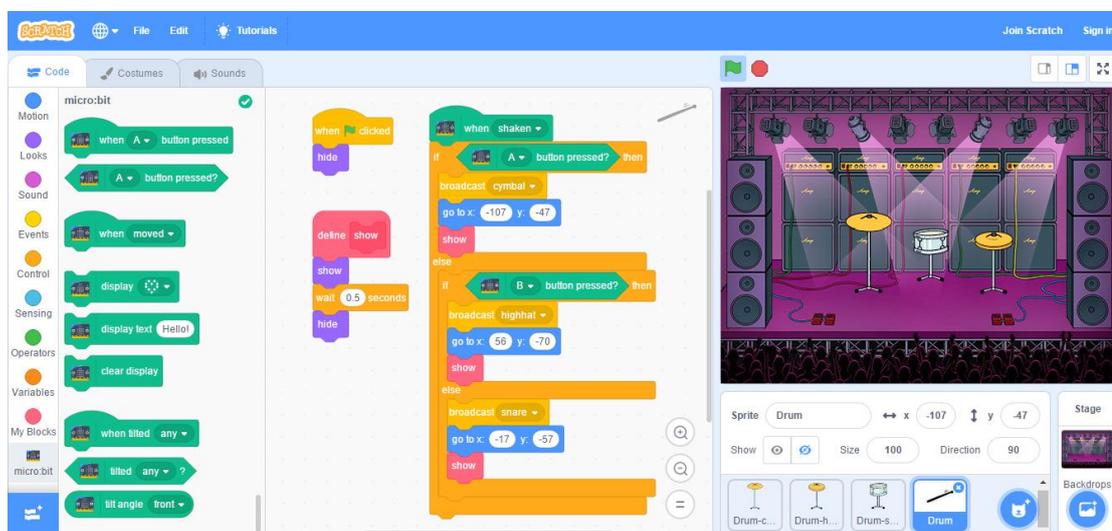


- 注意吉他聲音和背景顏色。你會發現音高和顏色都在不斷變化。此時，傾斜你的 micro:bit，成為一為吉他手！



- 現在，你可以傾斜並按下 `micro:bit` 上的按鈕來創建你的歌曲！

# 搖動爵士鼓



等級：高級

## 1. 你需要什麼？

- 1 x BBC micro:bit
- 1 x micro USB 線
- 1 x AAA 電池盒
- 2 x AAA 電池

## 說明：

使用搖動功能並按下 micro:bit 上的按鈕以控制外觀和聲音以及 Scratch 上的移動。

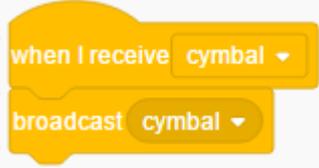
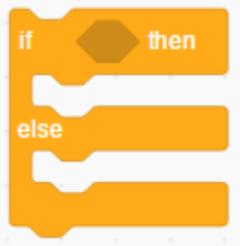
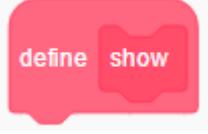
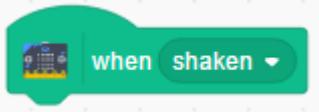
## 2. 學習目標

- 了解如何根據不同的條件隱藏和顯示精靈。
- 了解條件語句的使用。
- 掌握函數的知識。
- 了解廣播信息的使用。

## 目標：

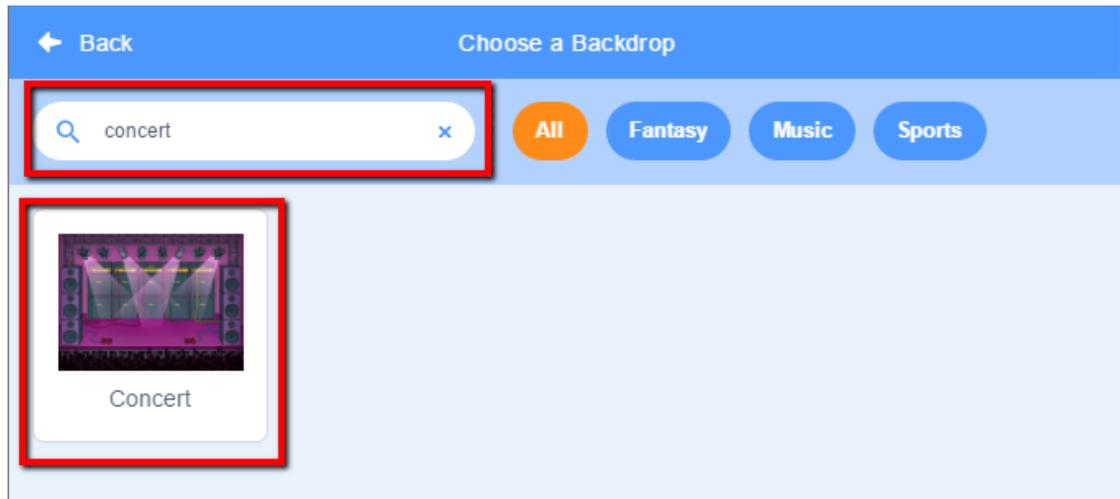
在這一章中，我們將學習如何使用三種不同的輸入方法(搖動，搖動+按鈕 A 和搖動+按鈕 B)更改 Scratch 中的外觀和聲音以模擬我們敲擊鼓時的實際情況。

### 3. 使用的方塊

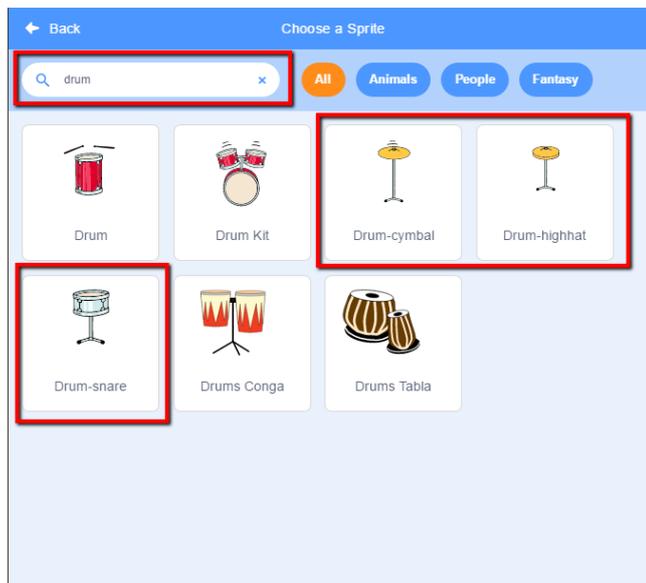
類別	塊	功能
Looks		“Show” 和 “Hide” 是一對控制精靈外觀的指令。使用 “show” 指令時，精靈將顯示在屏幕上。另一方面，“Hide” 會使精靈消失，直到你再次使用 “show” 。
Events		“When I receive” 和 “broadcast” 是另一對指令。你可以設計程式在某種情況下廣播信息。當其他精靈或舞台接收到廣播信息時，它將在 “when I receive” 塊後執行命令。
Motion		“if-else” 是一個條件語句，用於為不同的決策執行不同的操作。如果條件為真，則使用 if 指定要執行的代碼塊。如果條件為 false，則使用 else 指定要執行的代碼塊。
My Blocks		這個 “define” 塊在編碼中稱為函數。你需要先定義一個函數來執行特定任務。從這個意義上講，函數是一種過程。然後，在執行期間，你需要調用你定義的函數，它將運行你在函數中定義的所有塊。
Micro:bit		當 micro:bit 搖動時，將執行此塊之後的塊。

### 4. 設計步驟

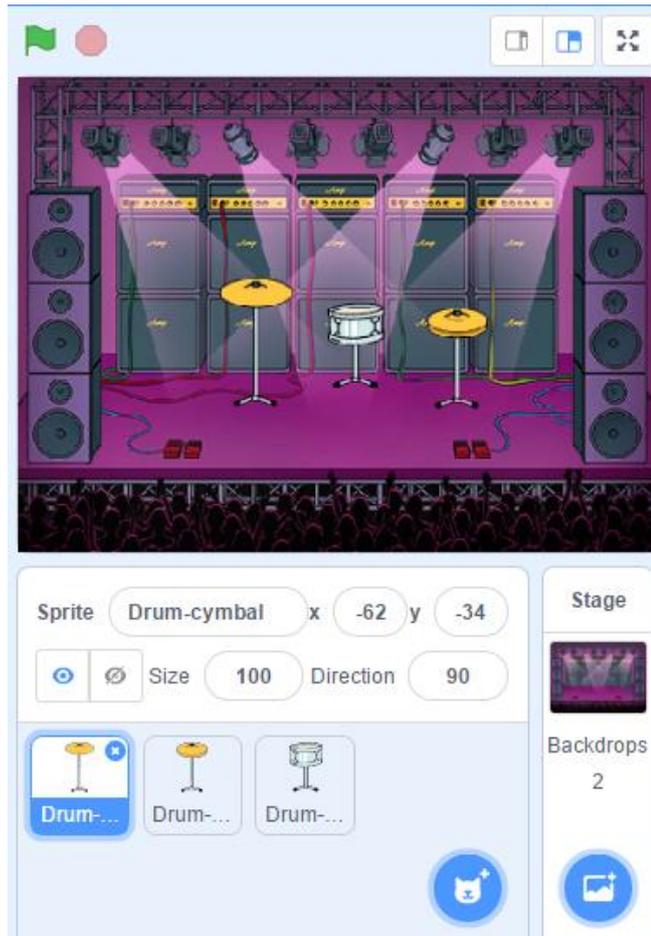
- 這次我們將使用默認背景 “Concert” 。



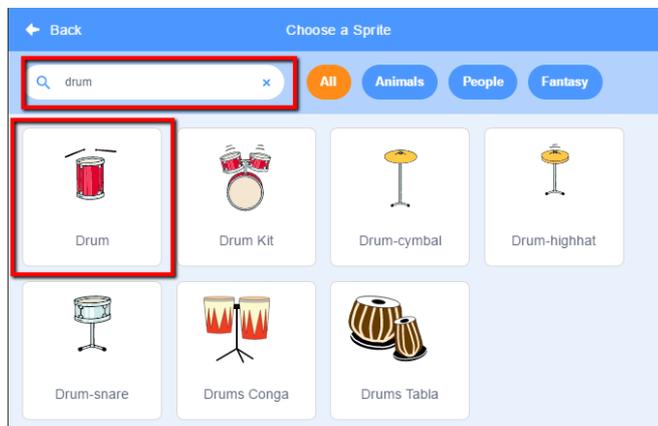
- 在這個項目中，我們需要 4 種不同的精靈，包括小鼓，腳踏鈸，鈸和鼓棒。對於頭 3 個精靈，我們可以簡單地創建和搜索“drum”，你會看到我們需要的精靈。



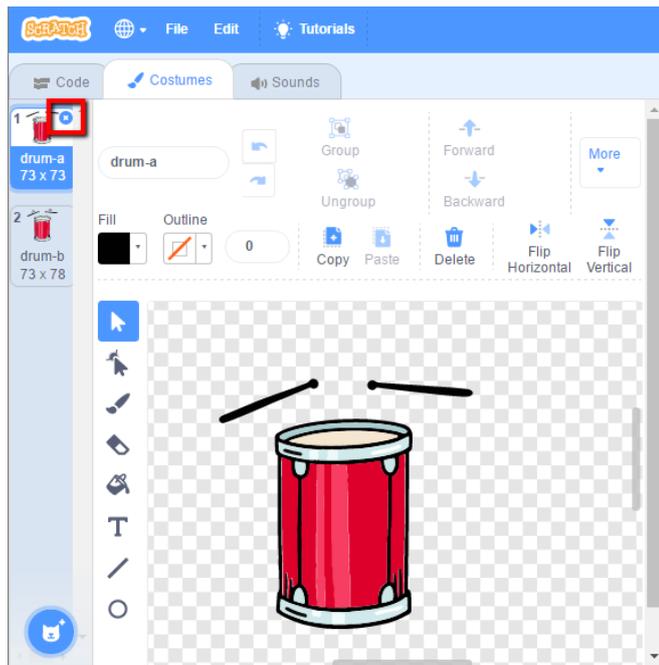
- 創建 3 個精靈後，可以調整每個精靈的位置，如下圖所示。



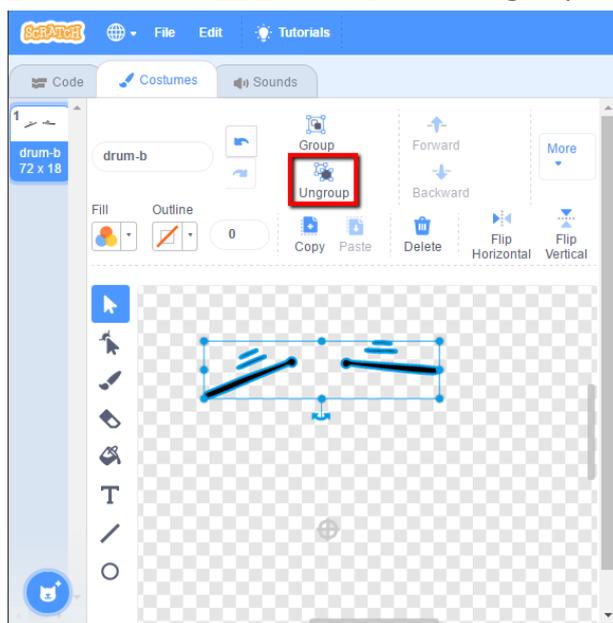
- 對於鼓棒精靈，我們搜索“drum”並選擇第一個。



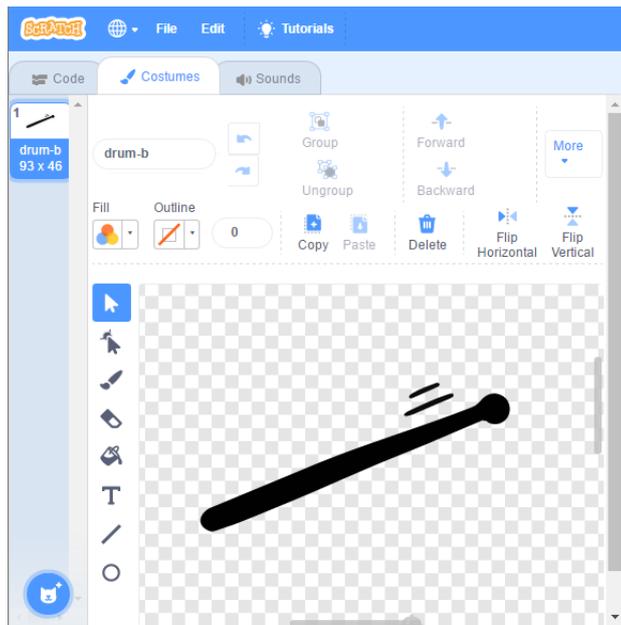
- 轉到“Costumes”頁面並刪除“drum-a”，因為我們只需要使用一個精靈。



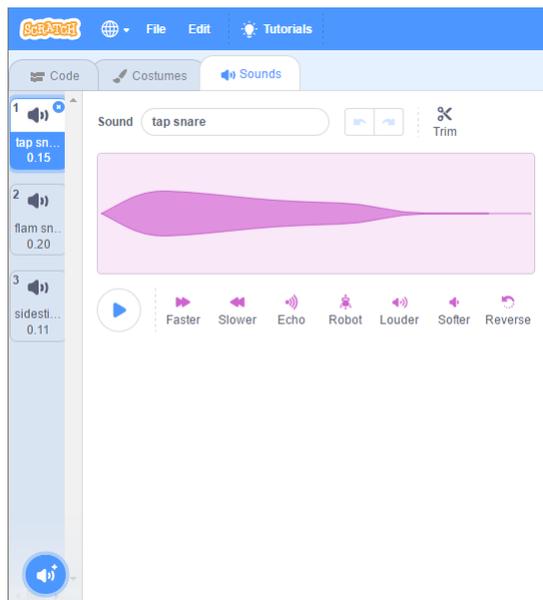
- 把鼓的部份刪除並留下鼓棍，按 “ungroup” 分開兩支鼓棍並刪除其中一支。



- 將鼓棒拖到中心並調整大小。

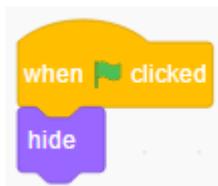


- 這次，我們不需要手動插入音軌，因為 Scratch 會在你創建精靈時自動生成相對應的音軌。因此，你只需要在編寫精靈時為每個精靈選擇你喜歡的聲音效果。



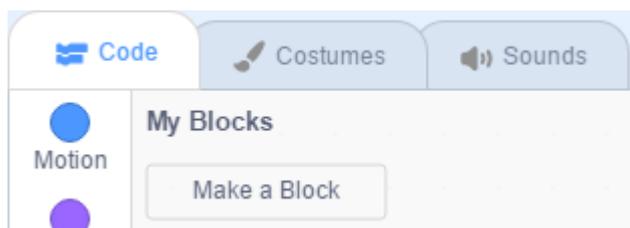
## 5. 編程步驟

- 首先單擊鼓棒精靈。由於我們只想在擊打爵士鼓時才顯示鼓棒，所以我們會在開始時隱藏鼓棒精靈。

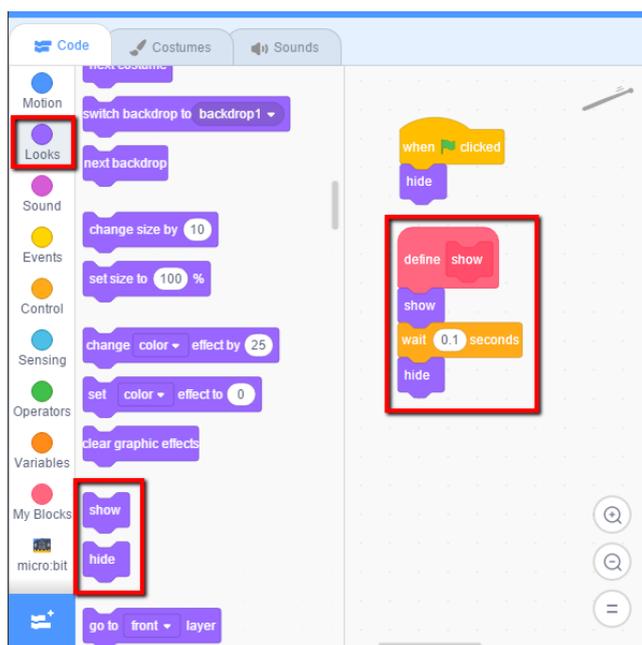


- 我們需要定義一個函數。我們先點擊“**My Blocks**”類別中的“**Make a Block**”

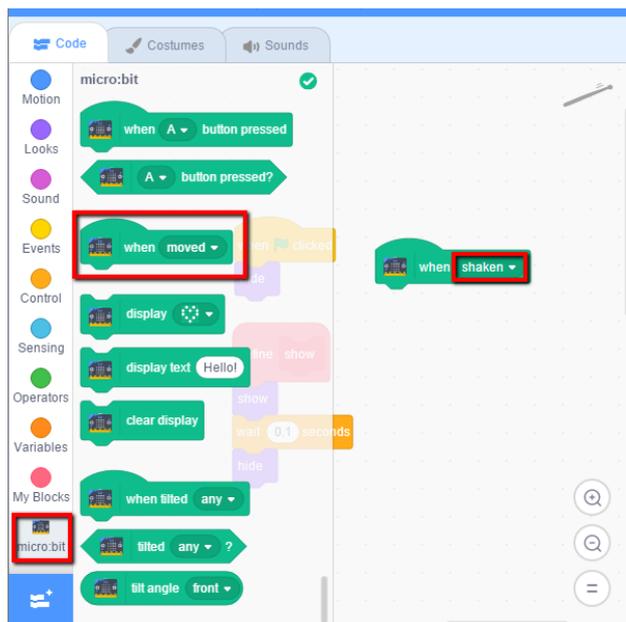
並將其命名為“show”。



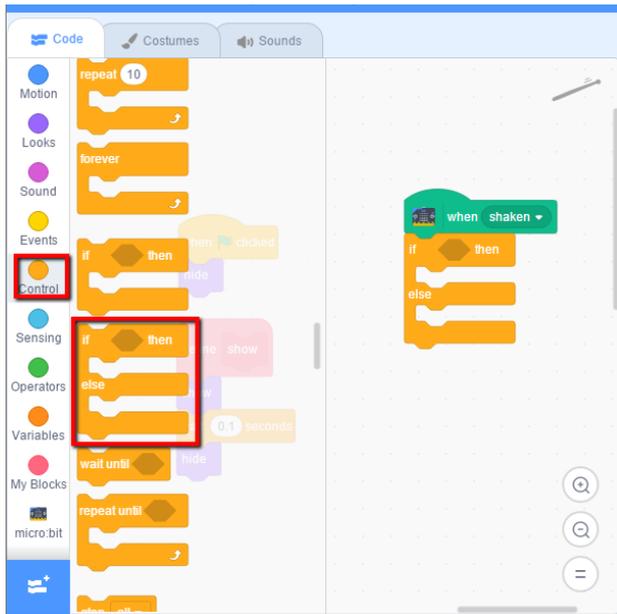
- 你將在編碼區域看到“define”塊。我們定義“show”函數，顯示精靈 0.1 秒然後隱藏。因此，我們從“Looks”類別中拖出“show”和“hide”塊，並在“show”和“hide”之間插入“wait 0.1 second”塊。



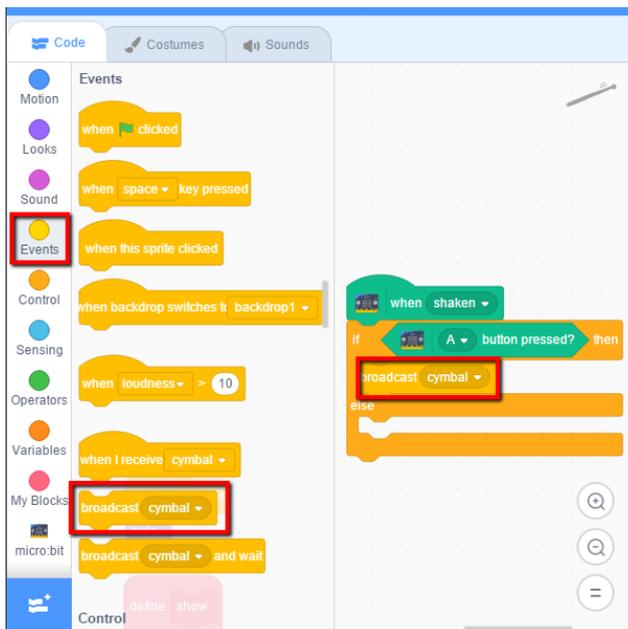
- 對於搖動動作，我們將“when shake”塊從“micro:bit”類別中拖出。



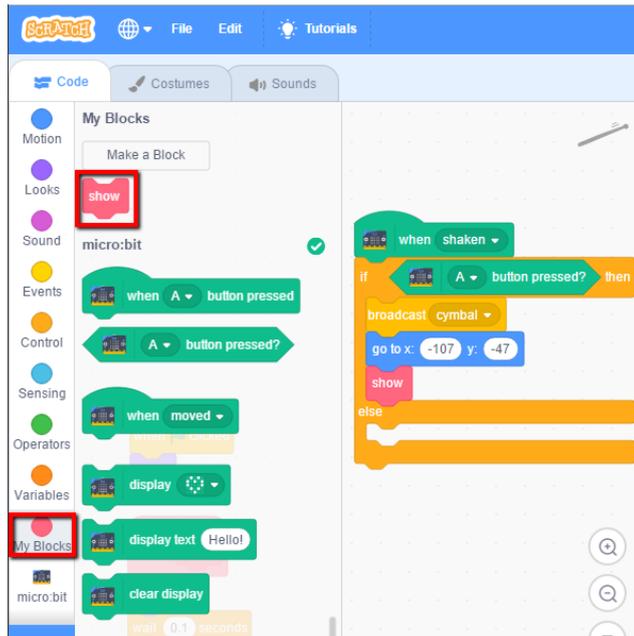
- 在“when shaken”塊後加上“if-else”塊。



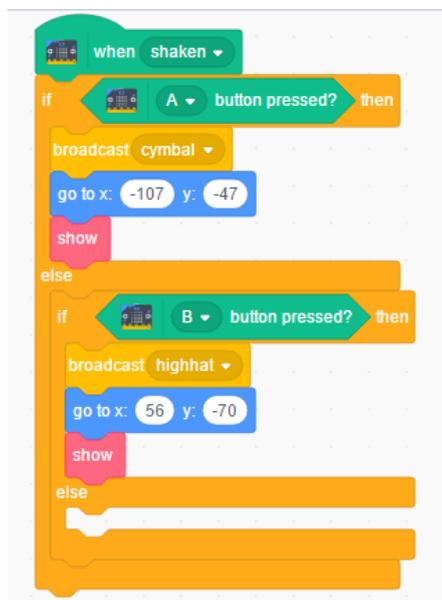
- “if-else” 條件語句幫助我們根據輸入而決定運行哪些編碼。當 micro:bit 被搖動並按下按鈕 A 時，將廣播一條消息 “cymbal” 並在鈸上顯示鼓棒。因此，我們需要先將 “A button pressed?” 塊拖到 “if condition” 中。然後拖 “broadcast” 塊並創建一個信息 “cymbal”。



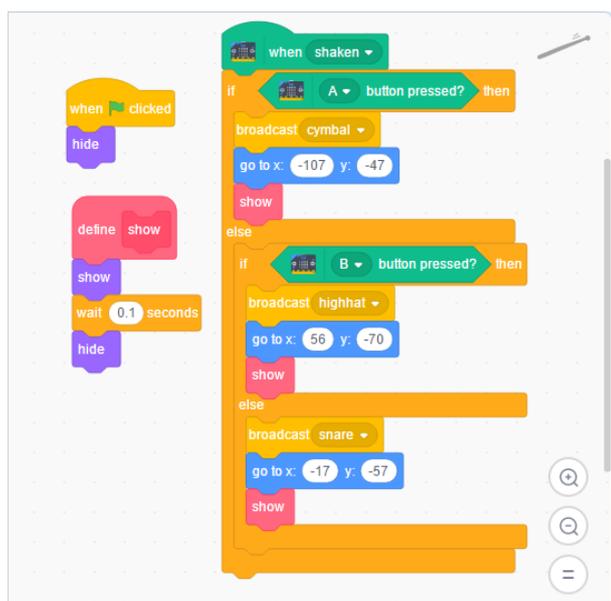
- 從 “Motion” 拖動 “go to x: y:”，將 x 和 y 坐標設置為鈸的位置，然後從 “My Blocks” 中拖動 “show” 來呼叫函數。



- 如果沒有按下按鈕 A，“if-else”條件將運行 else 的部分。因此，我們將另外一個“if-else”條件拖入 else 部分成為巢狀 if。在“if-else”條件中，它與前一個條件相似，但將條件設置為“按下 B 按鈕”，廣播另一條名為“highhat”的信息，並將 x 和 y 坐標設置為腳踏鉞的位置。



- 鼓棒精靈的最後一部分。如果沒有按下按鈕 A 和 B，這意味著擊中了小鼓，我們將廣播“snare”信息。



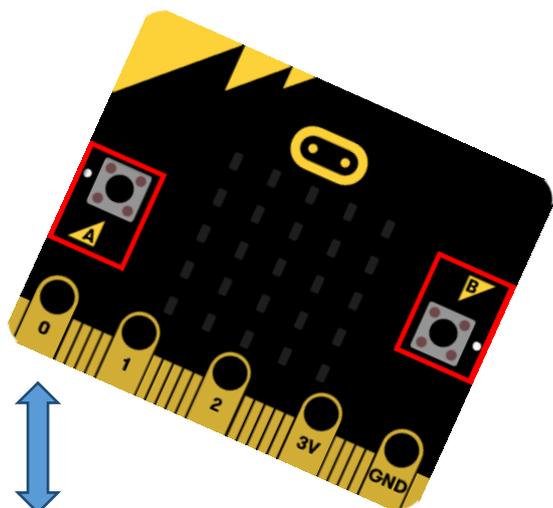
- 對於鈸，腳踏鈸和小鼓，編程非常簡單，幾乎相同。以小鼓為例，當精靈收到相應的廣播信息時，精靈會切換到下一個服裝並播放相應的音效然後切換到原來的服裝。



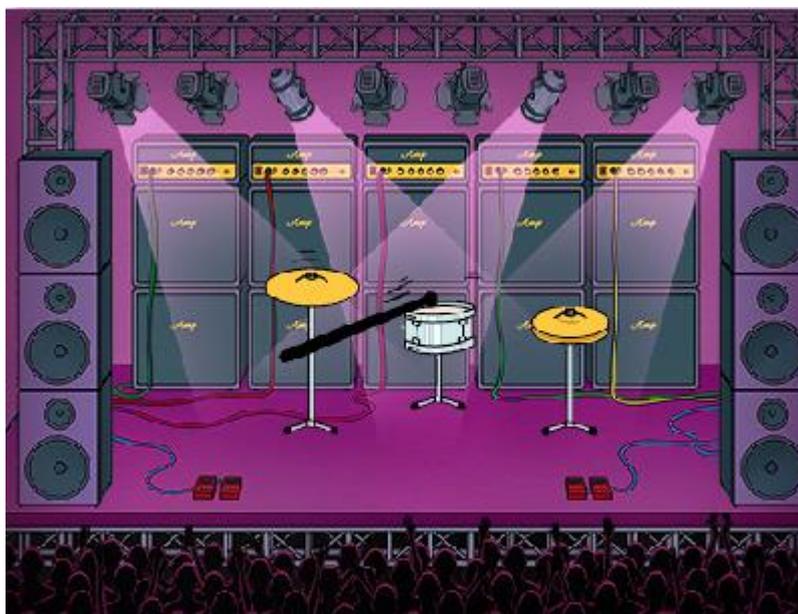
- 剩下的兩個精靈，你只需要改變其廣播信息和聲音。

## 6. 結果

- 在這個項目中，它有三種輸入方法，分別是搖動，搖動+按下按鈕 A 和搖動+按下按鈕 B.



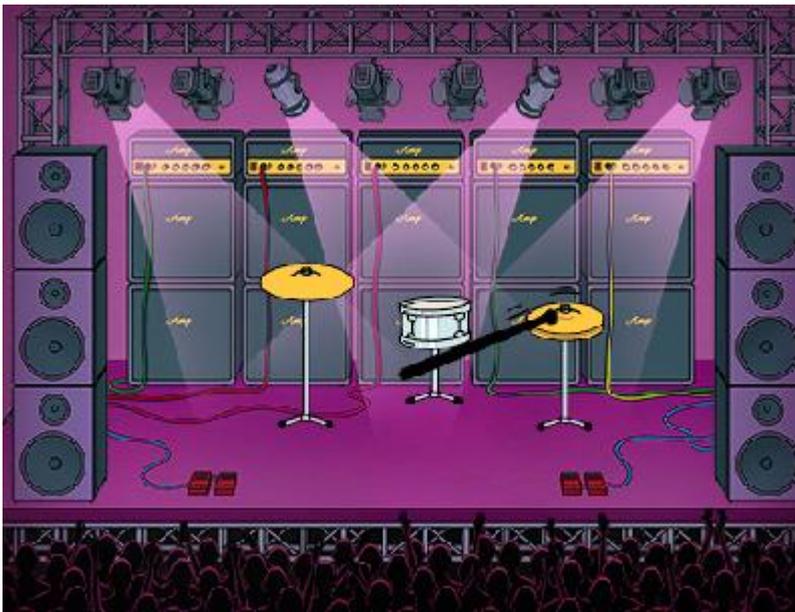
- 當你只搖動 micro:bit 時，小鼓將顯示被擊中的外觀並播放小鼓聲。



- 當你搖動並按下 micro:bit 上的按鈕 A 時，鈸將顯示被擊中的外觀並播放鈸聲。

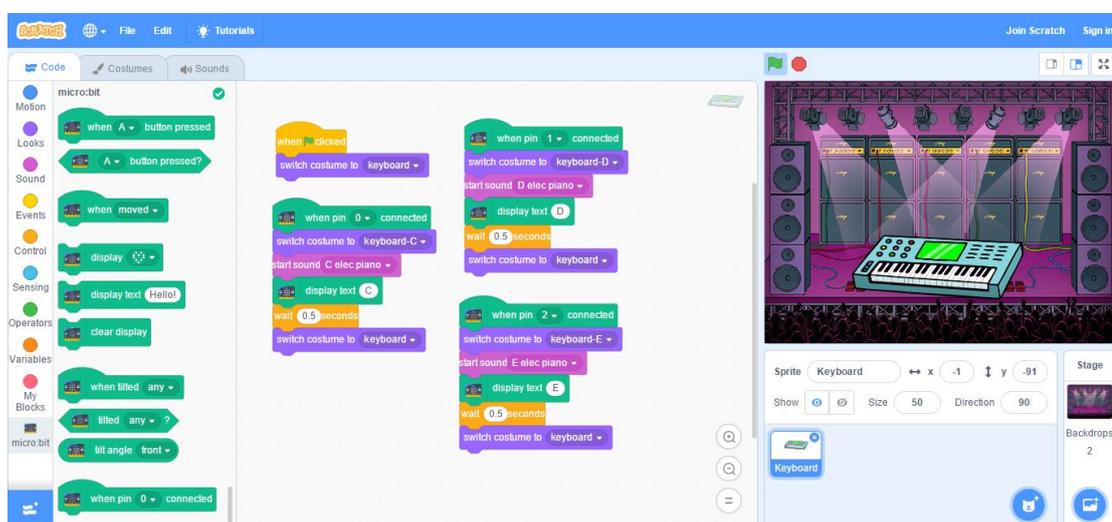


- 當你搖動並按下 micro:bit 上的按鈕 B 時，腳踏釵將顯示被擊中的外觀並播放腳踏釵的聲音。



- 現在，你可以搖動並按下 micro:bit 上的 A 或 B 按鈕來敲擊爵士鼓鼓，享受你的爵士鼓表演！

# 引腳電子琴



等級：高級

## 1. 你需要什麼？

1 x BBC micro:bit	4 x 鱷魚夾
1 x micro USB 線	銅貼
1 x AAA 電池盒	紙皮
2 x AAA 電池	顏色筆

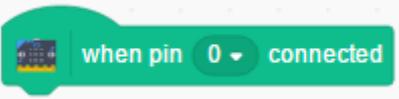
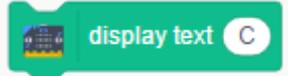
說明：連接 micro:bit 上的 PIN 以控制 Scratch3.0 的外觀和聲音。

## 2. 學習目標

- 了解如何在 micro : bit 上連接 PIN 。
- 嘗試使用 micro:bit 及紙皮來 DIY 屬於自己的小作品

目標：在第這一章中，我們將學習如何在 micro:bit 上連接 PIN 並使用紙皮創建紙皮電子琴。按下紙皮電子琴上的按鍵後，micro:bit 將顯示你按下的按鍵，並在 Scratch 上播放音調。

### 3. 使用的方塊

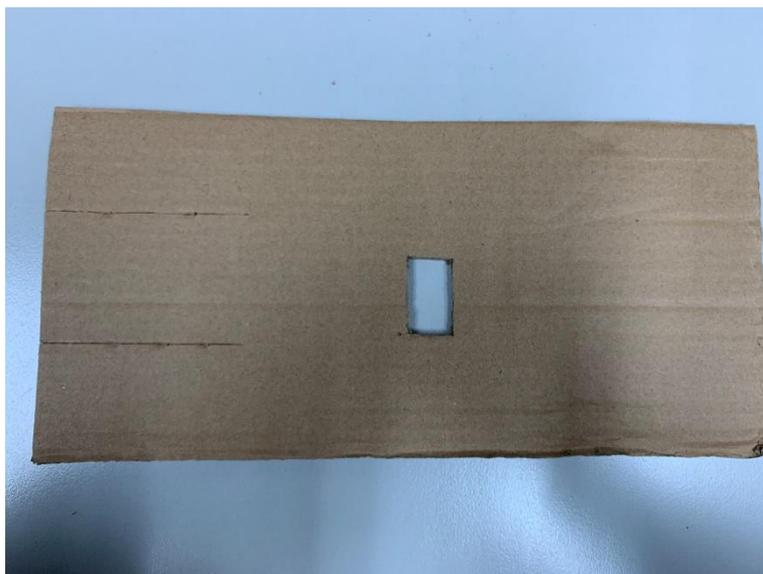
類別	塊	功能
Micro:bit		當 micro:bit 上的引腳 0/1/2 連接時，將執行此塊後的塊。
		此塊用於在 micro:bit 的 LED 屏幕上顯示文本。

### 4. 硬件設計步驟

- 準備一塊約 35cm x 15cm 的長方形紙皮。



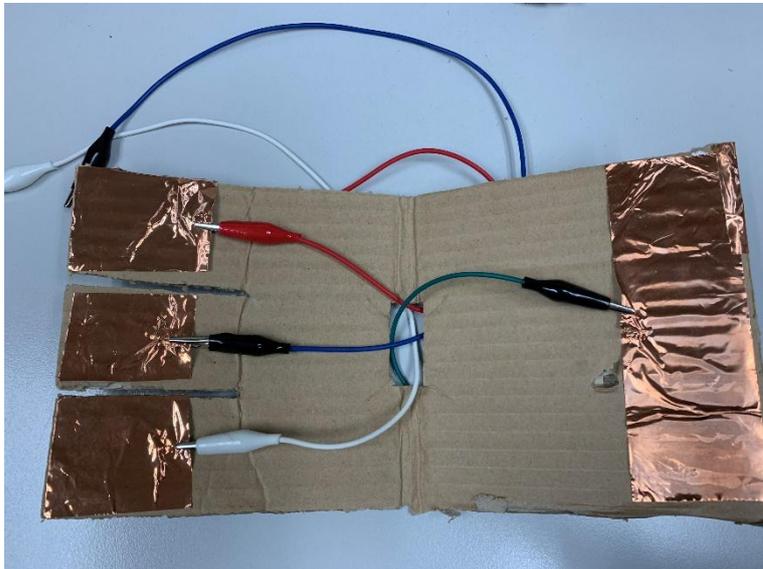
- 在紙皮的中間部分剪一個洞並在紙皮末端剪兩刀，形成 3 個琴鍵。



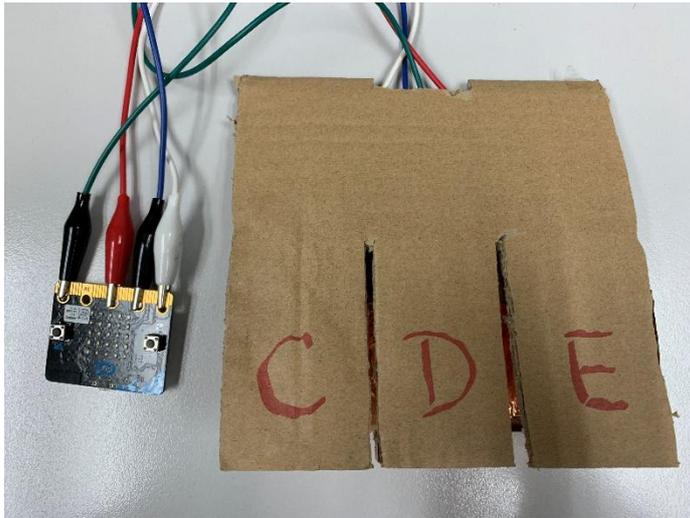
- 在琴鍵位置貼上 3 塊小銅貼，在另一端貼上長銅貼。之後，折疊紙皮。



- 將 4 條鱷魚夾連接到每個銅帶上，並將鱷魚夾放入中間的孔中。

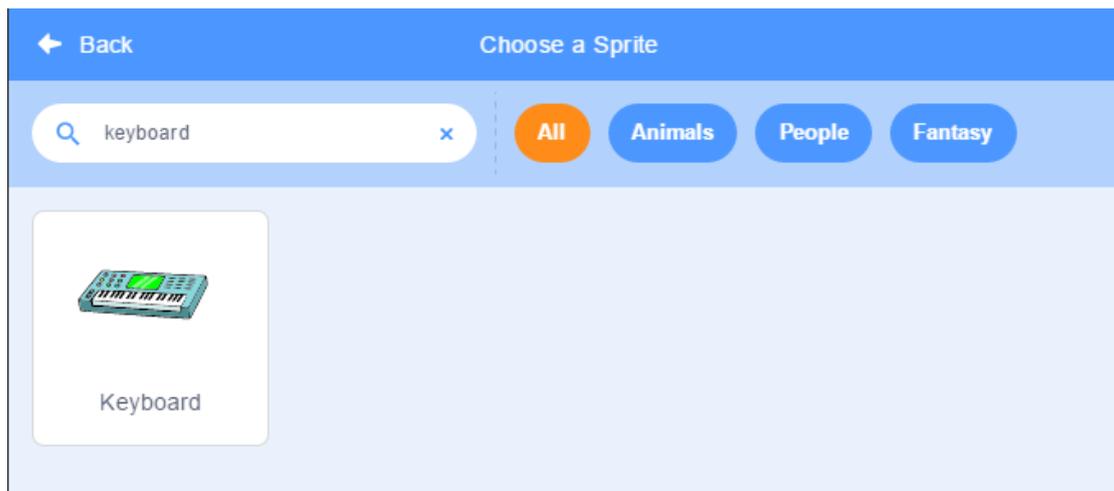


- 把音調寫在紙皮上，將鱷魚夾的所有末端連接到 micro:bit，如下所示：  
長銅貼 -> GND  
小銅貼 -> PIN 0/1/2

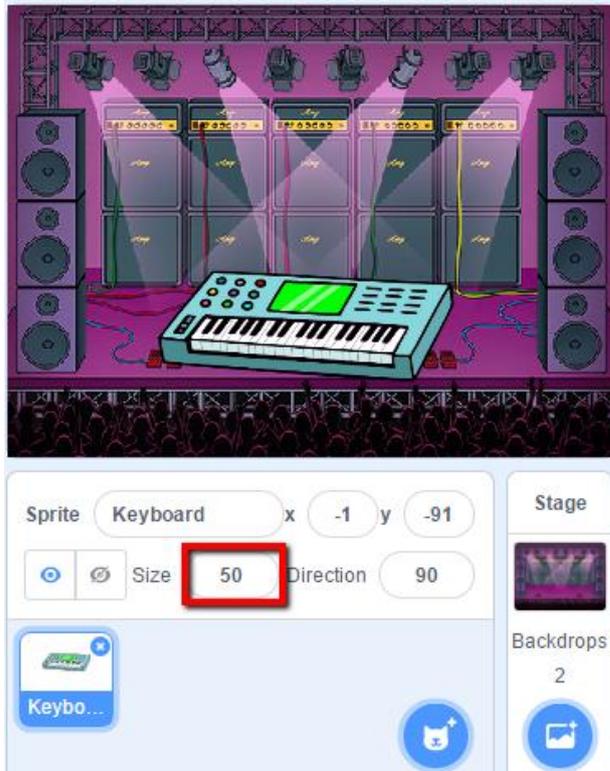


## 5. 軟件設計步驟

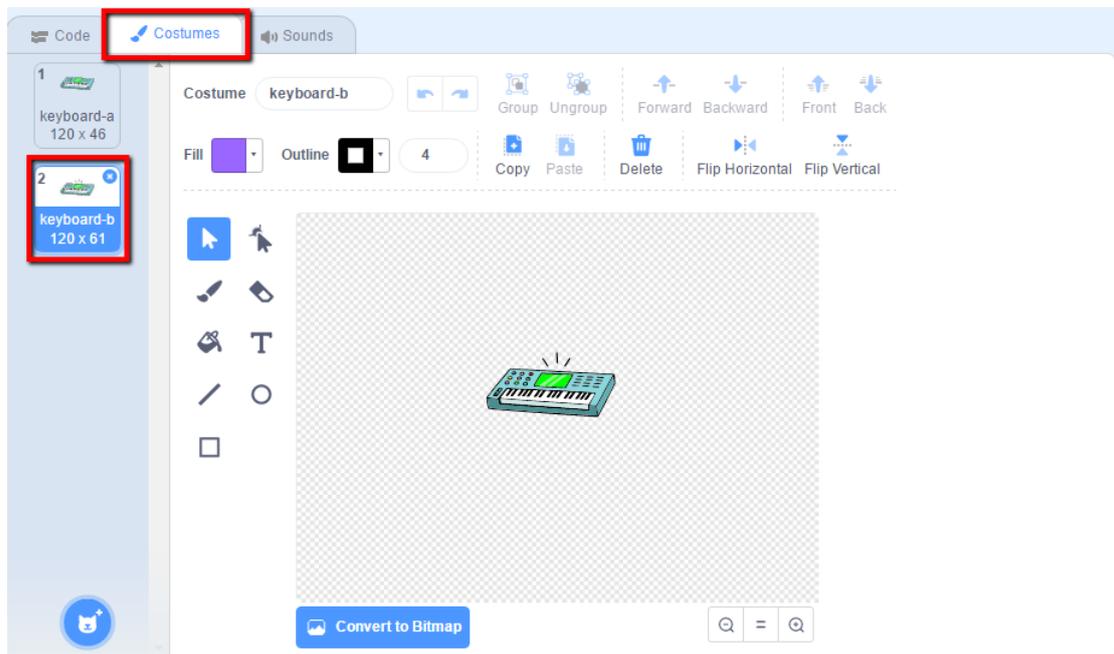
- 和前一章一樣，我們使用相同的背景。
- 選擇“Keyboard”作為精靈。



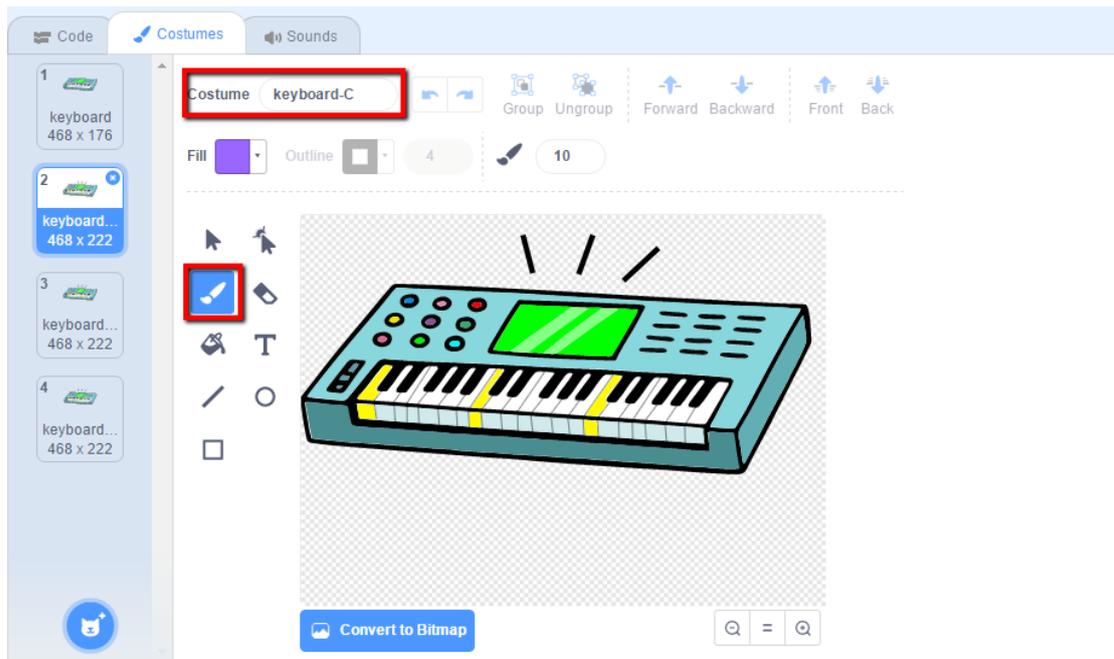
- 將大小設置為“50”並將精靈放在舞台上



- 轉到“Costumes”，你會看到“keyboard-b”，將“keyboard-b”複製兩次。

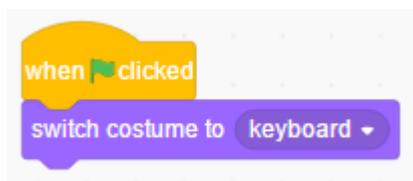


- 複製後，你可以將服裝的名稱更改為“keyboard-?”，“?”是你想要的音調，例如“keyboard-C”。然後，你可以使用畫筆突出相對的鍵。因此，你應該創建 4 種不同的服裝。

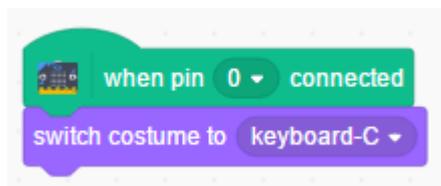


## 6. 編程步驟

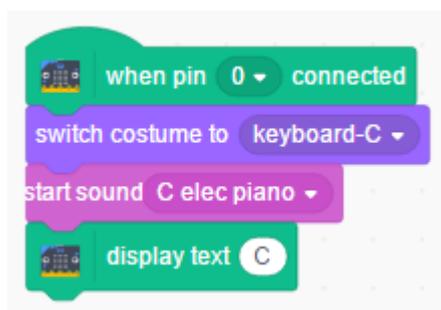
- 使用“When clicked”塊和“switch costume to”以確保精靈開始時的原始外觀。



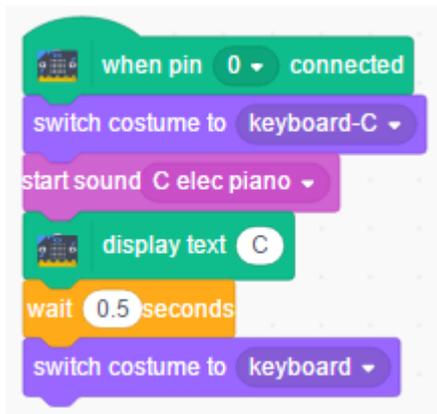
- 從 micro:bit 抽屜中拖出“when pin connected”，並與“switch costume to”塊結合使用。當連接 PIN 0 時，服裝將切換到“keyboard-C”。



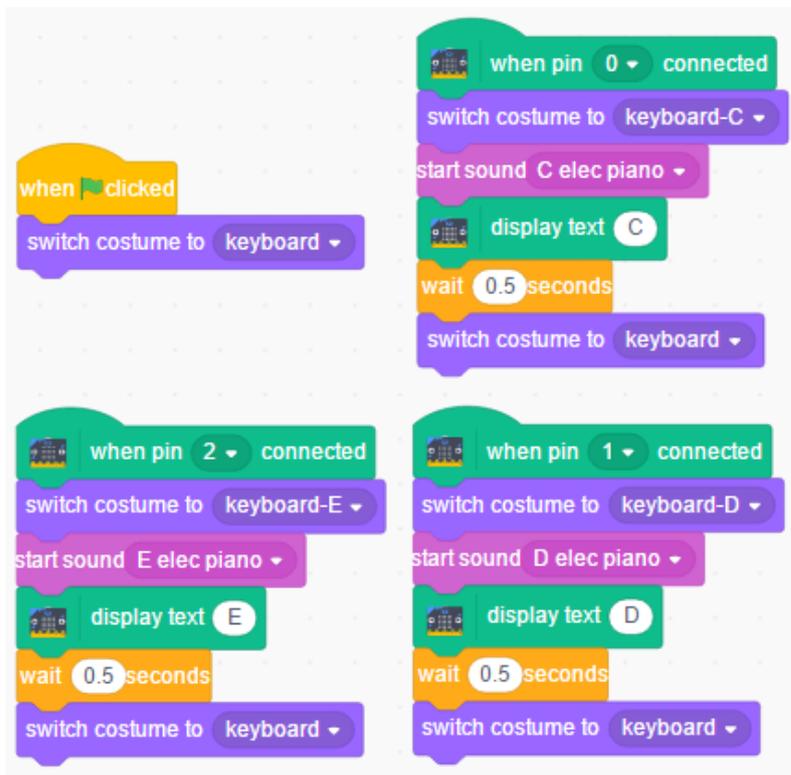
- 然後 Scratch 將播放 C 音調，並在 micro:bit 的 LED 顯示屏上顯示“C”字樣。所以我們拖動“start sound”塊，選擇“C elec piano”並從“micro:bit”抽屜中拖出“display text”塊。



- 在 micro:bit 上顯示字樣後，我們插入一個暫停並將服裝切換到原始外觀。

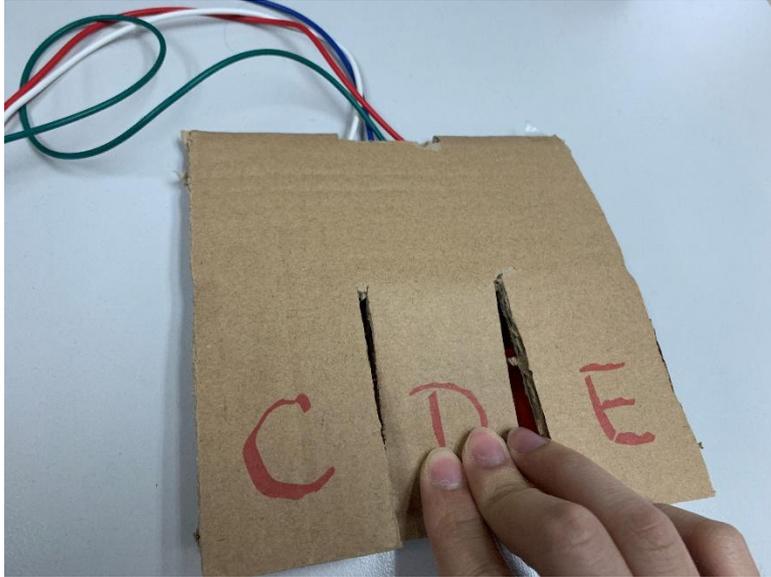


- 在剩下的部分中，我們只需完成其餘的 PIN，與第 4 步驟類似，但你需要更改 micro:bit 上的引腳編號，服裝，聲音和文字顯示。

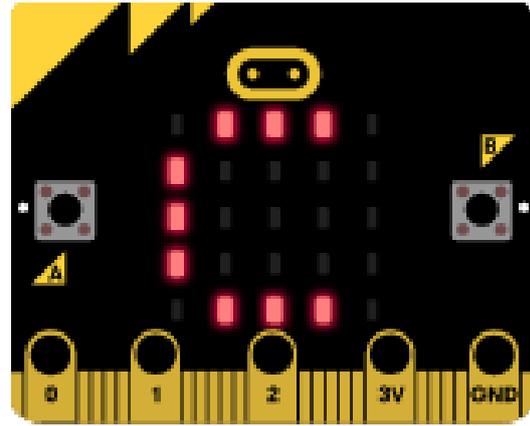


## 7. 結果

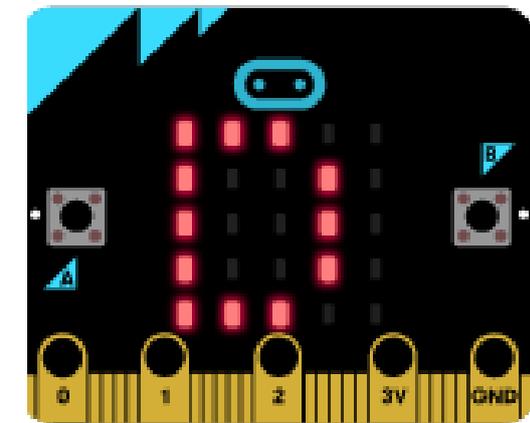
- 當你按下紙皮電子琴上的按鍵時，PIN 和 GND 的銅貼將連接並啟動程式碼。



- 如果按 C 鍵，Scratch 上的電子琴將點亮 C 鍵，播放 C 音並在 micro:bit 上顯示文字 “C” 。



- 如果按 D 鍵，Scratch 上的電子琴將點亮 D 鍵，播放 D 音並在 micro:bit 上顯示文字 “D” 。



- 如果按 E 鍵，Scratch 上的電子琴將點亮 E 鍵，播放 E 音並在 micro:bit 上顯示文字 “E” 。

