

---

# Scratch 蘋果捕手

2024/4/25

---

# 介紹

- 學習Scratch遊戲設計技巧
  - 音效與圖像效果
  - 使用變數
  - 使用亂數(隨機數字)產生器
  - 產生分身
  - 函數積木

# 遊戲情節介紹

在廣袤的沙漠中，有一位智慧而謙遜的僧侶，名叫卡維。他每天帶著簡樸的木碗踏上旅程，捕捉從天而降的蘋果，這象徵著對自然的感恩與奉獻。在常年累月的沙塵暴和嚴酷的自然環境中，卡維的旅途是一場精神追求，讓他與神聖的豐盈相連。

每當他捕捉到一顆蘋果，都代表著一份感恩之情，對宇宙賜予的祝福的認知。他的每一個優美動作都表達著對自然的敬意，將簡單的任務轉化為一種深刻的崇拜。對他而言，這段旅程不僅僅是捕獲蘋果；而是在最艱難的環境中尋找美麗和豐富。這是學習欣賞生活的簡單喜悅，並對宇宙提供的一切表示感謝。

當玩家加入卡維的追求時，他們也會學到這些寶貴的教訓。他們將共同努力，盡可能地捕捉更多的蘋果，不僅僅是為了贏得勝利，而是為了體驗感恩的喜悅和大自然豐盛的奇蹟。通過他們的行動，他們將明白真正的財富不在於物質財富，而在於心靈的豐盈和靈魂的美麗。

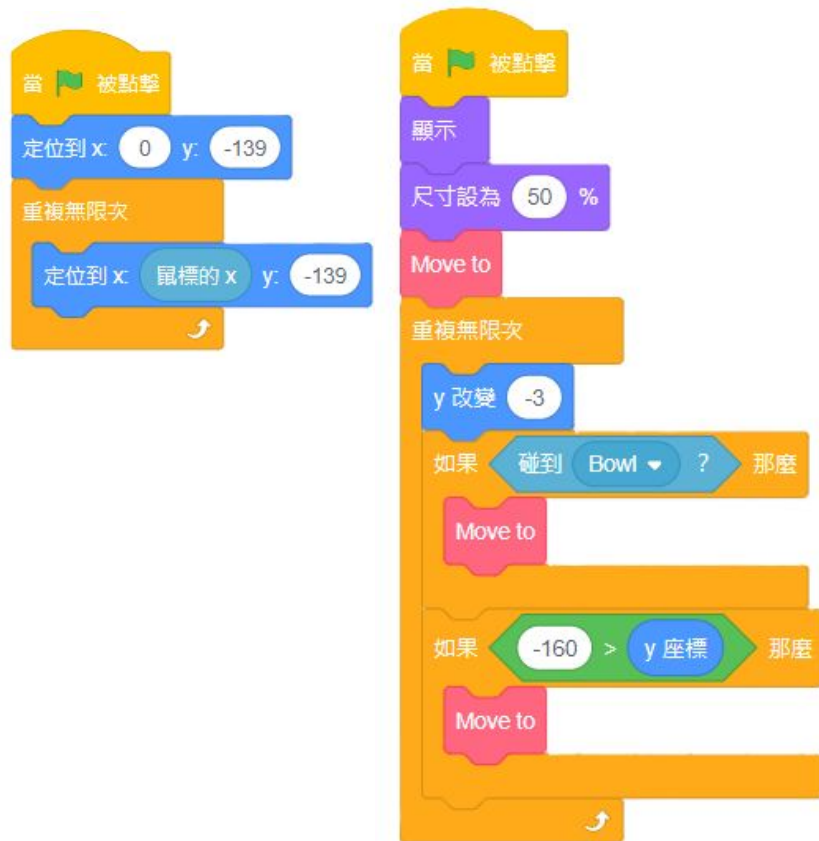
# 步驟一

- 舞臺背景: 例如沙漠背景
- 角色: 蘋果(Apple)與碗(Bowl)
- 碗:
  - 放在舞臺的下方位置
  - 可以利用鍵盤的向右與向左鍵往左右移動
- 蘋果:
  - 顯示並且調整蘋果大小
  - 定位到舞臺上方, 隨機的x位置
  - 讓蘋果由上往下掉落
  - 如果碗有接到, 讓蘋果回到最上方, 可以重新掉落
  - 如果沒接到, 也讓蘋果回到最上方, 可以重新掉落



## 步驟二

- 碗:
  - 可以直接利用滑鼠來控制碗左右移動位置
- 蘋果:
  - 將上步驟中“定位到舞臺上方, 隨機的x位置”的動作放到一個 函式積木(Move to)中, 這樣可以方便我們重複使用



# 步驟三

## 加入音效

- 舞臺:
  - 選擇一個背景音樂, 在遊戲過程當中一直重複播放
  - 隨著遊戲的進行, 漸漸加快音樂的音高
  - 也讓舞臺背景產生圖像顏色的變化特效
- 蘋果:
  - 當碗有接到蘋果, 則播放某一個音效
  - 當碗沒接到蘋果, 則播放另一個音效



# 步驟四

避免使用多個  ,改成使用廣播訊息

- 舞臺:
  - 將"綠色旗子被點擊"的事件積木留在舞臺中
  - 在裏面送出廣播訊息 (Start)
- 碗:
  - 將"綠色旗子被點擊"換成"當收到訊息 Start"
- 蘋果:
  - 將"綠色旗子被點擊"換成"當收到訊息 Start"



# 步驟五

## 產生蘋果分身

- 蘋果：
  - 共產生6個分身





# 步驟六

## 使用變數

- 舞臺:

- 建立適用於所有角色的變數:
  - time: 剩餘遊戲時間
  - score: 得分
  - apples: 蘋果分身數量
- 設定初始值
  - time: 60
  - score: 0
  - apples: 6
- 顯示time, score變數



- 我們需要倒數計時, 讓變數time每一秒鐘後減少1
- 當time變數等於0, 我們要結束遊戲



# 步驟六

## 使用變數

- 蘋果:
  - 建立一個僅適用當前角色的變數 **speed**, 代表蘋果掉下的速度
  - **speed** 初始值設為-3, 我們可以在**當收到訊息Start**或**當分身產生**中設定, 這裏我們放在當分身產生之中
  - 積木中使用**apples**與**speed**變數
  - 當接到蘋果時, 要將score加1



# 步驟七

- 蘋果:
  - 將speed變數的數值, 從固定的-3, 改成**隨機取數-3 到 -6**
  - 當蘋果回到最上方, 重新掉落時, 也重新設定speed數值
  - 我們定義一個新的函式積木, 用來取代之前的**Move to**積木, 並包含了前面設定speed變數的動作
    - 我們替這個函式積木加入一個數字參數 **y-position**, 代表蘋果出現的y座標, 所以我們可以調整蘋果出現的 y座標
    - 我們把這個函式積木命名為 **Setup Apple with y-position=**



# 步驟八

- 蘋果:

1. 當碗沒有接到蘋果時, 我們產生一個爛蘋果影像的特效, 再讓蘋果回到最上方
  - 製作第二個造型(apple2) - 爛蘋果
  - 利用畫筆的蓋章功能, 在舞臺最下方產生這個爛蘋果影像
  - 我們利用一個函式積木 **smashed**, 來完成這個動作
  - 記得要在遊戲剛開始時, 將筆跡全部清除



# 步驟八

- 蘋果:
  2. 製作蘋果消失的動畫
  3. 當碗接到或沒有接到蘋果, 我們都顯示這個消失的動畫, 然後再讓蘋果回到最上方
    - 我們利用一個函式積木 **vanish**, 來完成這個動作
    - 利用圖像效果中的 **像素化**及**幻影**特效



# 步驟八

- 蘋果:
  4. 製作蘋果**出現**的動畫
  5. 當蘋果回到最上方, 要重新顯示時, 顯示這個**出現**的動畫
    - 我們利用一個函式積木 **appear**, 來完成這個動作
    - 利用圖像效果中的 **顏色**特效, 產生不同顏色的蘋果
    - 利用圖像效果中的 **像素化**及**幻影**特效, 讓蘋果顯現



# 步驟九

- 當遊戲結束時, 產生Game over的文字動畫
- 舞臺:
  - 當time計時器倒數等於0時, 將原本的**停止全部**, 改為:
    - 廣播訊息**stop**
    - 停止這個物件的其他程式 (用來停止當綠色旗子被點擊中的重複無限次積木)
    - 停止這個程式 (用來本身中的重複無限次積木)
- 蘋果及碗
  - 當收到廣播訊息**stop**時, 停止這個物件的其他程式





# 步驟九

- 新增一個Mask的角色, 用來當作一個遮罩
  - 製作造型, 為一個純白色的背景, 涵蓋整個舞臺
  - 當遊戲開始時, 先定位到舞臺中心, 然後將它的圖層移到最上層, 然後隱藏
  - 當遊戲結束時, 先顯示, 然後調整它的透明度, 從全透明(幻影=100), 漸漸變為半透明(幻影=50)





# 步驟九

- 新增一個GameOver的角色,
  - 製作造型, 包含**"Game over!"**文字
  - 當遊戲開始時, 先定位到舞臺左邊中間位置, 然後隱藏
  - 當遊戲結束時,
    - i. 將它的圖層移到最上層, 然後顯示
    - ii. 選擇一段結束音效, 在播放的同時
    - iii. 將Game Over! 文字移動到舞臺中心位置
    - iv. 結束這個物件的其他程式

