

# 從 **Scratch** 進入 程式思維 的18堂課

橘子蘋果程式設計學苑 OrangeApple 編著

在創用CC(BY-NC-SA)授權下發行



## 目錄

[目錄](#)

[前言](#)

[首先，為什麼是18堂課？](#)

[這門課教的是什麼？](#)

[特別感謝](#)

[本課程的核心 - Computational Thinking](#)

[程式設計概念 Concepts](#)

[程式設計技巧 Practices](#)

[程式設計觀點 Perspectives](#)

[Reference:](#)

[Lesson 1 - 初探 Scratch](#)

[學習元素](#)

[課程教案](#)

[Lesson 2 - 牛刀小試](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 3 - 指令與序列](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 4 - 事件&平行化](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 5 - 互動設計](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 6 - 動畫](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 7 - 角色的溝通](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 8 - 場景設計](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 9 - Remix](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 10 - 認識變數](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 11 - 更多的互動設計](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 12 - 創造分身](#)

[學習元素](#)

[課程教案](#)

[Lesson 13 - 專案發想](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 14 - 專案實作\(1\)](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 15 - 專案實作\(2\)](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 16 - 檢視專案](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 17 - 專案實作\(3\)](#)

[學習元素](#)

[課程教案](#)

[Homework](#)

[Lesson 18 - 期末發表](#)

[學習元素](#)

[課程教案](#)

## 前言

### 首先，為什麼是18堂課？

因為每學期大約是18週，於是我們將來自哈佛教育研究所製作的「[Creative Computing](#)」課程，經過適當的刪減，並依我們長久來的教學經驗，編製成適合台灣教育現場的教案(每堂課80分鐘，剛好是國小兩節課的時間)，使用情境如下：

1. 每週的社團課、資優課程、課後才藝課，每次上課80分鐘，共一學期
2. 隔週上兩節的資訊課，每次上課80分鐘，共兩學期

對於國中小的程式設計教育，我們強力建議每次上課至少能有80~120分鐘的上課時間，因為程式設計是一個不斷創作的過程，如果僅讓孩子上40分鐘就打住，往往還沒進入狀況就下課，更別提團體討論以及作業的發表了。

### 這門課教的是什麼？

身為一群教育從業人員，我們深知一個深入、完整的課程有多不容易開發。

2013年，我們秉持著「不土法鍊鋼、不重造輪子」的原則，將這門由麻省理工學院多媒體實驗室製作的「[Creative Computing](#)」課程引進台灣，藉由一路以來的教學經驗，不斷地調整課程，變成較適合台灣教育現場的形式及進程。

而在美國，這個專案也同時在進化，到了2014年，這個專案由麻省理工轉交哈佛教育研究所進行維護，並進行了一次的大改版。本課程便是在這次的改版之後，所做出的最新版本。

在這門課中，學生將會學到：

1. **程式的基本概念：**  
幾乎所有的程式語言，都有共通的幾項核心基本概念，例如：序列、條件式、迴圈、變數、運算等等，學會了這些概念，對於未來的程式之路，將能打下良好的基礎。
2. **如何除錯：**  
程式設計師並非萬能，總是偶爾會有出錯的時候。因此，如何有效率、有系統的除錯，在程式設計上就是一件很重要的能力了。
3. **程式的互動性：**  
從APP到網頁，現在的軟體越來越精美、奪目，「使用者體驗」也成了現今軟體開發的一門顯學，攸關的不只是使用者的觀感，有時更直接影響到軟體公司的銷售量及營業額。
4. **程式設計的技巧：**  
在這門課中，學生將透過一個又一個的教學活動，以及專案的實作，體驗到一些開發的技巧，如：迭代式開發、混合(Remix)、回饋的收集...等等。讓學生能夠有方法、有效率的完成專案。

## 特別感謝

另外，本教案的誕生，要特別感謝吳嘉玲、呂永鈞、黃佑仁等資深程式設計老師。沒有這些年輕人的參與，橘子蘋果今天在台灣推動程式設計的過程絕對不會這麼順利。也希望當全世界先進國家的孩子都在學程式設計時，也能讓程式設計進入台灣的教學現場，早日為台灣培養下一代的科技人才。

如果您現在正在使用這份文件在學校推廣程式教育，我們誠摯的感謝您。也歡迎加入我們的「[橘子蘋果 Creative Computing 開源教案 - 智囊團](#)」臉書社團，及時反應教學上的問題及意見，我們也將依據老師們的意見，不斷編修、維護這份教材。

**讓這份「開源」的教學手冊，能夠透過群眾的力量不斷地進化，正是我們的初衷。**

橘子蘋果程式設計學苑 營運長  
東凱文

## 本課程的核心 - Computational Thinking

Computational Thinking（翻譯為：運算思維），是一種用電腦的邏輯來解決問題的思維。我們的目標不僅是教會學生使用電腦，抑或是在計算上的速度與精準度，而是在學習Scratch的過程中，建立基礎概念，從實作中學習，訓練學生「表達」、「溝通」、「提問」的能力。

Scratch是一個免費編寫程式的軟體，學生可以使用他盡情發揮創意，創建屬於自己獨特的故事、互動性遊戲和動畫，讓學生在學習Scratch的同時，也建立起程式設計的基礎概念。而Scratch的開發始祖 - 麻省理工多媒體實驗室 (MIT Media Lab)，在2011年提出了一套Computational Thinking的架構，分為“Concepts”，“Practices”，及“Perspectives”三個層面。

### 程式設計概念 Concepts

在學生學習程式設計的過程中，可區分為七個非常基本的大概念，以下將講述七個概念，並提供範例。學生學習了這幾個概念後，不只適用於Scratch，未來用於其他的程式語言也非常有幫助。

- **Sequence**（序列）：  
一個特定的動作能被細分並表示成各個步驟或指令，依序執行，例如轉一圈後說Hello!，可由下圖所示的指令達成。

*Identifying a series of steps for a task.*



- **Loops**（迴圈）：  
對同一個序列或動作重複執行，我們將上述旋轉一圈的指令利用重複4次表示，可將原本的八行指令縮減為三行，使指令更簡潔。

*Running the same sequence multiple times.*



- **Parallelism** (平行) :

電腦可以一邊播放音樂，一邊瀏覽社群網站，多個程式同時被進行。

平行就是多個指令同時被執行。以下範例是同時移動、改變特效及發出聲音，同時執行使角色有跳舞的感覺。

*Making things happen at the same time.*



- **Events** (事件) :

事件，指的就是一件事觸發一個/多個應對機制，例如：當角色被點下時，觸發「說 Hello 兩秒」。

舉個實務上的例子：當要登入facebook時，輸入完帳號密碼後，按下登入，就會觸發將帳號密碼送出。

*One thing causing another thing to happen.*



- **Conditionals** (條件) :

當電腦在播放音樂時，如果暫停鍵被按下，音樂就會暫停，等到播放鍵被按下才會繼續播放。

程式會依據設定的條件式，做出判斷（例如：如果...否則...）。

*Making decisions based on conditions.*



- **Operators (運算子)** :

當我們在計算機上依序按下  ,  ,  , 最後再按下  時, 計算機就會進行數學式的計算, 並得到結果 13 顯示在畫面上。

運算子即是做數學運算 (例如 : +、-、\*、/) 或邏輯運算 (例如 : <、=、>) 。

*Support for mathematical and logical expressions.*



- **Data (資料)** :

當我們在玩遊戲時, 會有時間倒數、得到的分數、獲得的寶物...等, 用來記錄遊戲過程中的資料。可以是單一個數值, 如「剩餘時間」, 也可以是放在一起的多個資料, 如「寶箱內的寶物」。

Data 就是用來記錄, 可做儲存及修改。Scratch可用變數 (用來儲存單一數字或字串) 以及清單 (可以儲存多個數字或字串) 來存資料。

*Storing, retrieving, and updating values.*



## 程式設計技巧 Practices

在帶領學生實作Scratch專案，我們設計了四種活動，利用活動進行的過程，來提高學生的參與度、激發創造力、訓練主動思考。未來不只是在用在開發程式專案，各個領域的發展皆會有幫助。

- **Experimenting and iterating**（實驗和迭代）：

「迭代」在程式設計的過程中，扮演了舉足輕重的角色。在面對難解、複雜的需求時，我們往往無法一次到位，一下就做出期望的功能。但是我們可以透過一種漸進開發、逐步測試的過程，有系統、有方法地一步步邁向目標功能。

在 Lesson 7 的教案中，我們會先拋出一個想法 "jump"，問學生該如何讓角色跳呢？再讓學生去嘗試，慢慢的實驗，找出方法，待有初步的結果(角色能向上移動再向下移動)，接著再引導學生開發更多，如讓角色有重力的感覺。

*Developing a little bit, then trying it out, then developing more.*

- **Testing and debugging**（測試和除錯）：

當程式執行時，出現問題或是沒按照預想中的執行，試著發現和解決它，可以依循著以下的步驟。

*Making sure things work — and finding and solving problems when they arise.*

- 發現問題
- 閱讀並了解所有指令的意思
- 實驗看看
- 嘗試寫出正確的指令
- 找到正確的答案

- **Reusing and remixing**（重用和混合）：

在已有的專案中進行加強或是加入更多的想法，我們可以在Scratch的網頁上找到許多有創意的專案，這可以幫助學生發想，引用他人的專案再加入自己的點子，改造成一個新的專案。

*Making something by building on existing projects or ideas.*



- **Abstracting and modularizing**（抽象化和模組化）：

將大的區塊，切開成多個小的模組，模組化讓他人方便閱讀與理解，在測試除錯上也

更好管理。如下範例，第一個模組設定時間由60秒倒數，第二個模組設定分數，第三個模組則是設定造型的改變。

*Exploring connections between the whole and the parts.*



## 程式設計觀點 Perspectives

學習此套課程後，我們希望學生在表達、溝通以及提問上有更顯著的進步。透過學習Scratch，更為了解自己、增進團隊合作的能力。

- **Expressing** (表達) :  
生活中有越來越多能夠具體發揮創意及想像力的工具，例如修圖軟體 Photoshop，可以將照片中的人物修得美美的，再加上創意文字，讓圖片更加吸引人。  
透過學習Scratch，讓學生了解到電腦是創造的媒介，能利用電腦表達出自己的想法。  
*Realizing that computation is a medium of creation, "I can create."*
- **Connecting** (溝通) :  
子曰：「三人行，必有我師焉」，讓學生透過與團體的溝通互動，多多學習他人的優點，汲取他人的意見，有助於創造出更棒的作品。  
*Recognizing the power of creating with and for others, "I can do different things when I have access to others."*
- **Questioning** (提問) :  
更能精準、明確的提出問題，幫助個人或團隊快速鎖定目標並解決問題。  
*Feeling empowered to ask questions about the world, "I can (use computation to) ask questions to make sense of (computational things in) the world."*

## Reference:

- <http://scratched.gse.harvard.edu/ct/defining.html>
- [http://web.media.mit.edu/~kbrennan/files/Brennan\\_Resnick\\_AERA2012\\_CT.pdf](http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf)

## Lesson 1 - 初探 Scratch

### 學習元素

- 如何註冊/登入scratch帳號
- 如何記錄專案想法
- 認識Scratch基本操作界面

### 課程教案

時間	課程	學習評估	附註
0:00~ 0:05	老師自我介紹、同學簡單自我介紹 (姓名、年紀)		可將學生之英文名字寫在白板上，幫助老師記憶及抽問
<b>Introducing to Scratch - 藉由影片及範例介紹創意編程 (15 mins)</b>			
0:06~ 0:10	詢問學員：「你平常都用電腦做什麼？」、「有哪些動作是能表達創意的？」，並請學生寫在學習單上。		
0:11~ 0:20	<ol style="list-style-type: none"> <li>1. 播放這個影片： <a href="http://youtu.be/-SjuiawRMU4">http://youtu.be/-SjuiawRMU4</a></li> <li>2. 詢問學員在剛剛的影片中，有沒有看到哪個專案是比較吸引人，自己有興趣的。</li> <li>3. 詢問學員：「如果你今天很會寫程式了，你想寫怎樣的程式？」</li> </ol>		
<b>Scratch Account - 在 MIT Scratch 網站創建帳號，並探索線上社群 (15 mins)</b>			
0:21~ 0:35	<ol style="list-style-type: none"> <li>1. 帶學生創建自己的帳號，並將帳號寫在學習手冊上。必要時，也請將密碼寫在手冊上，避免遺忘。</li> <li>2. 已經完成的學生可以請他們到這個網址自由探索： <a href="http://scratch.mit.edu/studios/137903">http://scratch.mit.edu/studios/137903</a></li> </ol>		可提醒學生資安的重要性
<b>Design Journal - 刺激學生想法，讓他寫下來，並與他人交流想法 (30 mins)</b>			
0:36~ 0:45	詢問學生，你會用什麼方式記錄點子？並示範以下開發日誌： <ul style="list-style-type: none"> <li>● <a href="http://lol.garena.tw/news/news_info.php?nid=2430">http://lol.garena.tw/news/news_info.php?nid=2430</a></li> <li>● <a href="http://waeroblog.blogspot.tw/2015/01/vo1.html">http://waeroblog.blogspot.tw/2015/01/vo1.html</a></li> </ul>		可提醒學生使用以下方式記錄創意想法：word, ppt, 紙張, Blog 等等
0:46~ 1:05	<ol style="list-style-type: none"> <li>1. 詢問學生：如果要你像同學解釋什麼是Scratch，你會怎麼說？請學生寫下來。</li> <li>2. 請學生在學習手冊上試著用文字</li> </ol>	訓練學生的文字表達能力。 訓練學生的規劃能力。	

	或圖畫表達他想做的作品，至少一個以上。		
<b>Scratch Surprise</b> - 使用 MIT Scratch 網站，創建第一個專案 (15 mins)			
1:06~ 1:20	帶領學生在MIT Scratch網站中新創一個專案，給予一個吸引人的專案名稱，再進行簡單的創作。 可以先引導學生試著讓小貓動起來，說一句話。		

## Lesson 2 - 牛刀小試

### 學習元素

- 認識Scratch工作室
- 如何使用Scratch工作室管理專案
- 了解專案開發過程
- 學習除錯的過程及策略

### 課程教案

時間	課程	學習評估	附註
<b>Scratch Studio - 學習 Studio 功能 (15 mins)</b>			
0:00~ 0:15	<ol style="list-style-type: none"> <li>帶領學生使用「工作室」功能，並把剛剛的專案加入其中。</li> <li>請學生在這個工作室中找一個喜歡的專案，在下面留言： <a href="http://scratch.mit.edu/studios/460431/">http://scratch.mit.edu/studios/460431/</a></li> <li>回答學習單上的問題</li> </ol>	認識到Scratch工作室的用途，並瞭解如何用它幫助管理專案。	
<b>10 Blocks - 只用10種積木創作Project (30 mins)</b>			
0:16~ 0:25	要求學生只能使用學習手冊上列出的10種積木做出一個有趣的專案	透過此活動初步了解Scratch專案的開發過程	
0:26~ 0:45	詢問學生以下問題，並將想法寫在學習單上： <ol style="list-style-type: none"> <li>在規定只能使用特定10種積木的限制下，有什麼好處？</li> <li>在規定只能使用特定10種積木的限制下，有什麼壞處？</li> <li>在創造這個作品的過程中，你的思考方式有什麼不同？為什麼？</li> </ol>		
<b>Debug it! - 釋出5個專案讓學生除錯 (35 mins)</b>			
0:46~ 1:20	展示以下專案，並敘述面臨的問題，請學生試著提出解決方案 <ol style="list-style-type: none"> <li><a href="http://scratch.mit.edu/projects/10437040">http://scratch.mit.edu/projects/10437040</a> 問題：只有一個角色會跳舞</li> <li><a href="http://scratch.mit.edu/projects/10437249">http://scratch.mit.edu/projects/10437249</a> 問題：第二次執行程式時，主角沒有動作</li> <li><a href="http://scratch.mit.edu/projects/10437366">http://scratch.mit.edu/projects/10437366</a> 問題：對空白鍵沒有反應</li> <li><a href="http://scratch.mit.edu/projects/">http://scratch.mit.edu/projects/</a></li> </ol>	讓學生體會到「 <b>除錯</b> 」的過程與策略	如果都沒有學生主動回答，老師要抽問學生，並給予一點提示。

	<p><a href="http://scratch.mit.edu/projects/10437439">s/10437439</a> 問題：有時會上下顛倒</p> <p>5. <a href="http://scratch.mit.edu/projects/10437476">http://scratch.mit.edu/projects/10437476</a> 問題：聲音以及說話框沒有同時出現</p>		
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--

## Homework

- 繼續完成“10 Blocks”專案

## Lesson 3 - 指令與序列

### 學習元素

- 學習指令 (Instruction) 的概念
- 學習序列 (Sequence) 的概念

### 課程教案

時間	課程	學習評估	附註
0:00~ 0:05	檢查一下作業，並注意有沒有值得分享或鼓勵的專案		
<b>Programmed to Dance</b> - 藉由指揮舞蹈動作的活動，讓學生體會指令的重要 (35 mins)			
0:06~ 0:30	<p>請學生志願擔任「指揮者」或「被指揮者」，被指揮者背對螢幕，老師播放影片，請指揮者指揮臺上的「被指揮者」做影片的動作：</p> <ul style="list-style-type: none"> <li>● <a href="http://vimeo.com/28612347">http://vimeo.com/28612347</a></li> <li>● <a href="http://vimeo.com/28612585">http://vimeo.com/28612585</a></li> <li>● <a href="http://vimeo.com/28612800">http://vimeo.com/28612800</a></li> <li>● <a href="http://vimeo.com/28612970">http://vimeo.com/28612970</a></li> </ul> <p>時間不夠的話只完成其中幾個舞蹈也行。</p>		指揮者千萬不可以使用動作引導被指揮者
0:31~ 0:40	<p>詢問學生以下問題，並完成課本上的學習單：</p> <ol style="list-style-type: none"> <li>1. 當你遇到較強勢的隊友時，有什麼特別難或特別簡單的地方呢？</li> <li>2. 當你遇到較聽話的隊友時，有什麼特別難或特別簡單的地方呢？</li> <li>3. 你覺得當一個旁觀者，有什麼特別難或簡單的地方呢？</li> <li>4. 這個活動跟我們操作Scratch有什麼關係？</li> </ol>	讓學生了解到：要有效的指揮對方，首先要能夠給出清楚的指令	
<b>Step-by-Step</b> - 藉由創作「跳舞的貓」專案，理解 Sequence 概念 (35 mins)			
0:41~ 1:15	<ol style="list-style-type: none"> <li>1. 跟著畫面右方提示視窗中「開始使用Scratch」的指示完成作品</li> <li>2. 讓學生自由加入積木，創作屬於自己的專案</li> </ol>	學生能藉由此活動理解到：指令在程式中是一個接著一個被執行的 (Sequence)	

### Homework

- 完成 “My Studio” 活動

## Lesson 4 - 事件&平行化

### 學習元素

- 學習事件 (Event) 的概念
- 學習平行 (Parallelism) 的概念

### 課程教案

時間	課程	學習評估	附註
0:00~ 0:05	檢查一下作業，並注意有沒有值得分享或鼓勵的專案		
<b>About me - 實作「自我介紹」專案 (50 mins)</b>			
0:05~ 0:55	<ol style="list-style-type: none"> <li>1. 先到“About me”的 Studio： <a href="http://scratch.mit.edu/studios/475470">http://scratch.mit.edu/studios/475470</a>，放幾個專案給學生看，幫助學生構想自己的專案。並提醒學生觀察這些專案的互動性。(5 mins)</li> <li>2. 老師先透過為角色加入「點擊後會唱歌」和「點擊後會跳舞」等功能，示範如何為角色加入互動性。(3 mins)</li> <li>3. 讓學生自由創作，此時老師需在教室走動，給予學生靈感並幫助學生除錯。</li> <li>4. 挑一兩個學生分享自己的作品，並由老師利用他的作品網址展示給其他學生 (8 mins)</li> </ol>	讓學生了解到：“ <b>互動性</b> ”能夠讓一個程式更加吸引人。	
<b>Performing Scripts - 用實體活動引導學生學習 Event 和 Parallelism 概念 (20 mins)</b>			
0:55~ 1:05	<p>挑選兩個志願者：</p> <ul style="list-style-type: none"> <li>● 請第一人開始踏步向前，如果遇到牆壁或是無法向前，則折返。重複動作直到老師說「歸零」（停止所有動作）</li> <li>● 接著，請第一人同時做兩件事，例如踏步向前並唱歌或說話</li> <li>● 加入第二人，讓這個人開始唱歌或說話，第一個人的動作仍繼續</li> <li>● 當第一人踏步向前且遇到第二人時，第一人說：「Hi, 你好嗎?」，接著，第二人說：「Hello, 我很好」。對話結束後，原本的動作繼續進行。</li> </ul> <p>並詢問學生，在剛剛的活動中，學生觀察到了哪些現象。引導學生認知到動作可以由事件來觸發，並了解到「平行化」的概念。</p>	讓學生了解「 <b>事件</b> 」與「 <b>平行化</b> 」的概念	

1:05~ 1:15	讓學生在Scratch中探索「事件」與「接收」機制，試著做出具有「事件觸發」與「平行化」機制的小程式。 (使用兩隻以上的角色)		
---------------	--------------------------------------------------------------------	--	--

## Homework

- 繼續完成具有「事件觸發」與「平行化」機制的小程式。

## Lesson 5 - 互動設計

### 學習元素

- 複習事件 (Event)、迴圈 (Loop)、平行 (Parallelism)
- 學習使用「外觀積木」及「繪圖編輯器」

### 課程教案

時間	課程	學習評估	附註
0:00~ 0:05	檢查一下作業，並注意有沒有值得分享或鼓勵的專案		
<b>Performing Scripts</b> - 用實體活動引導學生學習 Event 和 Parallelism 概念 (10 mins)			
0:06~ 0:15	讓學生"複習"上週在Scratch中探索「事件」與「接收」機制，試著做出具有「事件觸發」與「平行化」機制的小程式。 (使用兩隻以上的角色)	讓學生複習「事件」與「平行化」的概念	
<b>Build-a-Band</b> - 音效的整合應用，並複習 Sequence, Loop, Parallelism 概念 (30 mins)			
0:16~ 0:45	<ol style="list-style-type: none"> <li>1. 到這個網頁： <a href="http://scratch.mit.edu/studios/475523">http://scratch.mit.edu/studios/475523</a>，選幾個專案讓學生看，幫助學生構想自己的專案 (5 mins)</li> <li>2. 給學生時間自行創作具有互動性的樂團專案，提醒學生可在素材庫中找所需素材，並參考課本上的範例 (20 mins)</li> <li>3. 挑一個學生分享自己的作品，並由老師利用他的作品網址展示給其他學生 (5 mins)</li> </ol>		
<b>Orange Square, purple circle</b> - 探索“外觀積木”及“繪圖編輯器” (30 mins)			
0:46~ 1:15	<ol style="list-style-type: none"> <li>1. 先到“Orange Square, Purple Circle”的 Studio： <a href="http://scratch.mit.edu/studios/475527">http://scratch.mit.edu/studios/475527</a>，放幾個專案給學生看，幫助學生構想自己的專案。並提醒學生觀察這些專案用了哪些「特效」或「動作」。(5 mins)</li> <li>2. 讓學生自由創作，此時老師需在教室走動，給予學生靈感並幫助學生除錯。(15 mins)</li> </ol>	認識“特效”、“動作”等積木。	

### Homework

- 繼續完成“Build-a-Band”專案



## Lesson 6 - 動畫

### 學習元素

- 學習新增角色
- 學習改變角色外觀
- 了解角色與外觀的不同
- 學習除錯的過程及策略
- 如何結合「動作」和「音樂」

### 課程教案

時間	課程	學習評估	附註
0:00~ 0:05	檢查一下作業，並注意有沒有值得分享或鼓勵的專案		
<b>It's alive!</b> - 了解“角色”與“外觀”的不同，並能活用外觀創造生動的專案 (20 mins)			
0:06~ 0:25	<ol style="list-style-type: none"> <li>1. 先到 “It's Alive!” 的 Studio : <a href="http://scratch.mit.edu/studios/475529">http://scratch.mit.edu/studios/475529</a>，放幾個專案給學生看，告訴學生：動畫就是一系列的圖片連續播放。(8 mins)</li> <li>2. 讓學生自由創作，此時老師需在教室走動，給予學生靈感並幫助學生除錯。(12 mins)</li> </ol>	了解到動畫是由一幅幅靜態的圖案不斷切換造成的	
<b>Debug it</b> - 釋出5個專案讓學生除錯 (25 mins)			
0:26~ 0:50	<p>展示以下專案，並敘述面臨的問題，請學生試著提出解決方案：</p> <ol style="list-style-type: none"> <li>1. <a href="http://scratch.mit.edu/projects/23266426">http://scratch.mit.edu/projects/23266426</a> 在這個專案中，小橘貓想要跳一支舞給你看。當你點他時，他應該要跟著鼓聲一起跳舞。但是現在小橘貓只會踏下一步，且當他踏完步時，鼓聲也沒有停下來！我們該如何修正這個程式呢？</li> <li>2. <a href="http://scratch.mit.edu/projects/24268476">http://scratch.mit.edu/projects/24268476</a> 在這個專案中，當綠旗被按下時，Pico會朝向 Nano走去，當Pico碰到Nano時，Pico應該要說 “Tag, you're it!” 然後 Nano 說 “My turn!”。但是程式出了一些問題，導致 Pico 沒有向 Nano 說任何話，我們該如何修正這個程式呢？</li> <li>3. <a href="http://scratch.mit.edu/projects/24268506">http://scratch.mit.edu/projects/24268506</a></li> </ol>	讓學生更加熟悉除錯的策略	

	<p>這個專案原本會畫出一個笑臉，但是出了一些差錯，使得這個笑臉的眼睛和嘴巴給連在了一起！我們該如何修正這個程式呢？</p> <p>4. <a href="http://scratch.mit.edu/projects/23267140">http://scratch.mit.edu/projects/23267140</a></p> <p>在這個專案中，當綠旗被點下時，會開始播放一朵花成長、開花的動畫。但是程式出了一點差錯，使得動畫不停的從頭開始播放。我們該如何修正這個程式呢？</p> <p>5. <a href="http://scratch.mit.edu/projects/23267245">http://scratch.mit.edu/projects/23267245</a></p> <p>在這個專案中，當綠旗被點下後，會開始播放生日快樂歌，當生日快樂歌放完後，應該會出現「請點我吹熄蠟燭」的提示。但是出了一些問題，使得歌還沒放完，便出現了提示。我們該如何修正這個程式呢？</p>		
<p><b>Music Video - 結合音效及動畫，自製一個MV (25 mins)</b></p>			
<p>0:51~ 1:15</p>	<ol style="list-style-type: none"> <li>1. 先到“Music Video”的 Studio： <a href="http://scratch.mit.edu/studios/475517">http://scratch.mit.edu/studios/475517</a>，放幾個專案給學生看，幫助學生構想自己的專案。並提醒學生觀察這些專案如何結合「動作」和「音樂」。(5 mins)</li> <li>2. 讓學生自由創作，提醒學生可參考課本上的提示。此時老師需在教室走動，給予學生靈感並幫助學生除錯。(5 mins)</li> <li>3. 讓學生停下手邊動作，互相觀摩左右同學的作品(5 mins)</li> <li>4. 讓學生繼續創作(10 mins)</li> </ol>		
<p><b>下回預告 - 介紹下一階段課程 (5 mins)</b></p>			
<p>1:16~ 1:20</p>	<p>介紹第二階段課程： 學生們在第一階段學到的是關於創意、設計的培養，也就是UI的部份 (User Interface)，下一階段叫「Game Designer」，著重於邏輯的養成，在這個階段，我們將會教學生如何利用第一階段學到的互動設計結合「程式邏輯」及「資料的儲存」做出一個內外兼具的遊戲程式。</p>		<p>這個階段部分很重要，請務必將下一階段要傳授的內容向學生交代清楚</p>

## Homework

- 繼續完成“Music Video”專案



## Lesson 7 - 角色的溝通

### 學習元素

- 學習使用「定義積木」
- 了解迭代開發 (Iterating) 的概念
- 學習使用「等待」
- 學習使用「廣播」
- 了解「等待」與「廣播」的不同

### 課程教案

時間	課程	學習評估	附註
<b>Characters</b> - 探索“定義積木”，並複習 Events, Parallelism, Iterating 概念 (35 mins)			
0:00~ 0:15	<ol style="list-style-type: none"> <li>1. 詢問一位學生從家裡到教室的詳細方式並寫在白板，試圖定義“上學”的過程。 假設得到的答案為： 「走路→捷運市政府站→忠孝新生轉車→古亭7號出口」</li> <li>2. 老師詢問學生：「如果同學問你周六下午做了些什麼？你會回答剛剛那串話呢？還是回答“上課”上課？」</li> <li>3. 接著，老師即可將上述「一長串的上課路程」答案定義成「上學」。接著問學生：「哪一種說法比較方便呢？」</li> <li>4. 老師須講解“廣播(事件)”與“定義”的差異與使用時機。               <ol style="list-style-type: none"> <li>a. <b>廣播(事件)</b>:                   <ol style="list-style-type: none"> <li>i. 可能會影響到其它角色 (例如:當A角色要廣播「跳躍」時，其他角色也可能會執行「跳躍」的程式碼)</li> <li>ii. 被廣播執行的程式碼會與接下來的程式「平行」執行</li> </ol> </li> <li>b. <b>定義</b>:                   <ol style="list-style-type: none"> <li>i. 在一個角色中被定義的積木只有他自己能使用，而且就算其他角色有一樣名稱的積木方塊，也不會互相影響。</li> </ol> </li> </ol> </li> <li>5. 如果有時間的話，可以展示此專案讓學生了解「廣播」與「事件」的差異： <a href="https://scratch.mit.edu/projects/67715818/">https://scratch.mit.edu/projects/67715818/</a></li> </ol>	<ul style="list-style-type: none"> <li>● 瞭解如何使用「<b>定義</b>」積木及其好處</li> <li>● 了解“事件”及“定義”的差別與使用時機</li> </ul>	這個部份其實是在教學生「 <b>抽象化</b> 」的概念
0:16~ 0:35	<ol style="list-style-type: none"> <li>1. 老師講解使用更多積木並創造一個積木名為“jump”</li> <li>2. 讓角色可以做一個往上跳的動作</li> </ol>	藉由實際操作，更清楚 <b>迭代開發</b> (Iterating) 的概念。	

	3. 開放學生自由創作		
<b>Conversation</b> - 探索“計時”和“廣播”兩種同步的策略，並複習 Events, Parallelsim, Reusing, Remixing 概念 (35 mins)			
0:36~ 1:10	<ol style="list-style-type: none"> <li>1. 講解如何使用"計時"讓兩個角色可以互相對話，並讓學生自由創作對話內容</li> <li>2. 講解如何使用"廣播"讓兩個角色可以互相對話，並讓學生自由創作對話內容</li> <li>3. 請學生在創作的對話內容的中間，再加入任一句話，體驗擴增專案時，兩者在修改上的難易程度</li> <li>4. 詢問學生：「你們覺得使用"計時"與"廣播"實作的差異為何？有什麼優缺點呢？」</li> </ol>	學習計時與廣播的功能，並了解兩者實作上的差異 <ul style="list-style-type: none"> <li>● 用計時來實作：需自行計算每個步驟所需時間</li> <li>● 用廣播來實作：動作結束後接著進行，在進行專案擴增上，較好維護</li> </ul>	

## Homework

- 無

## Lesson 8 - 場景設計

### 學習元素

- 學習場景的變換
- 學習除錯的過程及策略
- 學習如何蒐集、歸納點子

### 課程教案

時間	課程	學習評估	附註
<b>Scenes - 在實作中熟悉場景的運用 (25mins)</b>			
0:00~ 0:25	利用上一階段的對話程式，請學生加入場景，並要有變化		
<b>Debug it - 釋出5個專案讓學生除錯 (25 mins)</b>			
0:26~ 0:50	<p>展示以下專案，並敘述面臨的問題，請學生試著提出解決方案：</p> <ol style="list-style-type: none"> <li>1. <a href="https://scratch.mit.edu/projects/24269007/">https://scratch.mit.edu/projects/24269007/</a> 在這個專案中，Scratch貓要教Gobo貓如何喵喵叫。但輪到Gobo時卻沒發出聲音，我們該如何修正這個程式呢？</li> <li>2. <a href="https://scratch.mit.edu/projects/24269046/">https://scratch.mit.edu/projects/24269046/</a> 在這個專案中，Scratch貓應該要從1數到我們指定的數字，但每次Scratch都只會從1數到10，我們該如何修正這個程式呢？</li> <li>3. <a href="https://scratch.mit.edu/projects/24269070/">https://scratch.mit.edu/projects/24269070/</a> 這個專案是Scratch貓帮大家點名的程式，但一開始大家卻同時說話，我們需要Scratch貓點到人時，才給出回應，我們該如何修正這個程式呢？</li> <li>4. <a href="https://scratch.mit.edu/projects/24269097/">https://scratch.mit.edu/projects/24269097/</a> 在這個專案中，當Scratch貓說完"jump"時，Gobo貓要上下跳10下，但Gobo貓現在卻不會動。我們該如何修正這個程式呢？</li> <li>5. <a href="https://scratch.mit.edu/projects/24269131/">https://scratch.mit.edu/projects/24269131/</a> 在這個專案中，每當按下鍵盤右鍵時，就會切換背景，我們需要恐龍在背景為"表演廳"時才顯示</li> </ol>	讓學生更加熟悉除錯的策略	

	並跳舞，並在其他背景中影藏起來。我們該如何修正這個程式呢？		
<b>Dream Game List</b> - 腦力激盪提出各種遊戲，收斂，並討論他們共通的功能 (20 mins)			
0:51~ 1:10	小組討論後，詢問學生以下問題，並完成課本上的學習單： 1. 寫一份你最喜愛的遊戲清單 2. 這些遊戲有哪些相同的地方？ 3. 這些遊戲在設計上有哪些特色？ 4. 做一份自己夢想遊戲的設計元素清單。		2~3人一組

## Homework

- 無

## Lesson 9 - Remix

### 學習元素

- 學習Remix (混合)

### 課程教案

時間	課程	學習評估	附註
<b>Pass it on</b> - 藉由活動了解如何在MIT平台上Remix他人的Project (50mins)			
0:00~ 0:50	1~2人一組，至少分為三組，每人新建一個專案，自由創作，10分鐘後，順時鐘移一個位置，接著創作他人的專案，重複3次。	<ul style="list-style-type: none"> <li>● 讓學生瞭解到如何對專案進行 <a href="#">Remix</a></li> </ul>	請勿用自己過去做過的專案，必須是新開的
<b>Starter Games</b> - 從3個遊戲專案中，選擇一個 Remix (20 mins)			
0:51~ 0:59	要求學生先探索三個專案後，挑選一個喜歡的做remix。 <ul style="list-style-type: none"> <li>● Maze: <a href="http://scratch.mit.edu/projects/1414041/">http://scratch.mit.edu/projects/1414041/</a></li> <li>● Pong: <a href="http://scratch.mit.edu/projects/10128515/">http://scratch.mit.edu/projects/10128515/</a></li> <li>● Scrolling: <a href="http://scratch.mit.edu/projects/22162012/">http://scratch.mit.edu/projects/22162012/</a></li> </ul>		
1:00~ 1:10	詢問學生以下問題，並完成課本上的學習單： <ol style="list-style-type: none"> <li>1. 對你來說設計遊戲的挑戰是什麼？</li> <li>2. 你最得意的是什麼？</li> </ol>		

### Homework

- 繼續完成“Starter Games”專案

## Lesson 10 - 認識變數

### 學習元素

- 學習變數 (Data)
- 學習擴充功能 (可以加入遊戲)

### 課程教案

時間	課程	學習評估	附註
0:00~ 0:05	檢查一下作業，並注意有沒有值得分享或鼓勵的專案		
<b>Score</b> - 在現成專案中加入分數功能，藉以學習“變數”的概念(30 mins)			
0:05~ 0:35	在 Fish Chomp 中加入分數的功能。 Fish Chomp: <a href="http://scratch.mit.edu/projects/10859244/">http://scratch.mit.edu/projects/10859244/</a>	了解什麼是變數，並學會如何在Scratch中創造變數、設定變數、取得變數	
<b>Extensions</b> - 示範9個擴充功能，增加學生的思考廣度 (40 mins)			
0:36~ 1:05	請學生在studios中探索9個擴充功能中較重要的7個 1. 分數 <a href="http://scratch.mit.edu/projects/1940443">http://scratch.mit.edu/projects/1940443</a> 2. 等級 <a href="http://scratch.mit.edu/projects/1940453">http://scratch.mit.edu/projects/1940453</a> 3. 計時器 <a href="http://scratch.mit.edu/projects/1940445">http://scratch.mit.edu/projects/1940445</a> 4. 敵人 <a href="http://scratch.mit.edu/projects/1940450">http://scratch.mit.edu/projects/1940450</a> 5. 獎品 <a href="http://scratch.mit.edu/projects/1940456">http://scratch.mit.edu/projects/1940456</a> 6. 滑鼠 <a href="http://scratch.mit.edu/projects/25192659">http://scratch.mit.edu/projects/25192659</a> 7. 多人遊戲 <a href="http://scratch.mit.edu/projects/25192711">http://scratch.mit.edu/projects/25192711</a>		
1:06~ 1:10	詢問學生以下問題，並將想法寫在學習單上： 1. 有哪些不同方式可以用來增加遊戲的難度？		

	2. 加了哪些擴充項目到遊戲專案中? 3. 請描述一下你在遊戲中加入擴充功能的過程		
--	----------------------------------------------	--	--

## Homework

- 觀摩所有的“Extension”

## Lesson 11 - 更多的互動設計

### 學習元素

- 學習各種不同形式的互動模式
- 學習除錯的過程及策略
- 學習如何有效率地做知識管理

### 課程教案

時間	課程	學習評估	附註
<b>Interactions</b> - 藉由挑戰9個 Puzzles, 探索「互動性」的意義 (25 mins)			
0:00~ 0:10	1~3人一組, 共8組(第二題由老師示範), 將8個puzzle分配給各組, 先搶先贏, 9個puzzle皆為「互動式」專案, 角色與角色間的互動, 或是使用者與角色間的互動, 互動式使遊戲更佳有趣	了解 <b>互動性</b> 的意義, 並知道各種不同的互動模式	
0:11~ 0:20	請學生起立, 並將座椅靠上, 將作品開至全螢幕, 走動瀏覽別人的作品與解決方式		
0:21~ 0:25	詢問學生以下問題, 並將想法寫在學習單上: 1. 你選擇解決哪些難題? 2. 你解決這些難題的策略是什麼? 3. 哪些難題可以幫助你思考遊戲專案?		
<b>Debug it!</b> - 釋出5個專案讓學生除錯 (25 mins)			
0:26~ 0:50	展示以下專案, 並敘述面臨的問題, 請學生試著提出解決方案 1. <a href="http://scratch.mit.edu/projects/24271192/">http://scratch.mit.edu/projects/24271192/</a> 問題: 更新物品清單 2. <a href="http://scratch.mit.edu/projects/24271303/">http://scratch.mit.edu/projects/24271303/</a> 問題: 修改錯誤分數 3. <a href="http://scratch.mit.edu/projects/24271446/">http://scratch.mit.edu/projects/24271446/</a> 問題: 判斷錯誤 4. <a href="http://scratch.mit.edu/projects/24271475/">http://scratch.mit.edu/projects/24271475/</a> 問題: 被打到的次數增加1 5. <a href="http://scratch.mit.edu/projects/24271560/">http://scratch.mit.edu/projects/24271560/</a> 問題: 貓咪不穿牆  當反應不熱絡, 或是只有少數人會回	● 運用學生的觀察力及分析能力, 找出 <b>除錯</b> 的策略	如果都沒有學生主動回答, 老師要抽問學生, 並給予一點提示。

	答時，可以請學生「遙控」老師做除錯，一方面可以訓練學生的表達能力，一方面可以讓沒有想法的學生向其他學生學習。		
<b>Know What to Learn</b> - 回顧過去所學，思考自己「知道什麼」、「想學什麼」、「學了什麼」、「學習策略」 (20 mins)			
0:51~ 1:10	<ol style="list-style-type: none"> <li>1. 我知道什麼?(鼓勵學生分享目前的設計經驗)</li> <li>2. 我還想知道些什麼?(列清單)讓學生自由搜集資源來探索自己的學習清單。(20min)</li> <li>3. 我學到了什麼?(從上面的過程學到什麼?)</li> <li>4. 你使用什麼策略來探索自己想了解的知識呢?</li> </ol>	引起學生的學習興趣，並懂得如何尋找自己需要的知識	

### Homework

- 繼續完成“Interactions” 的挑戰
- 觀摩所有的“Extension”

## Lesson 12 - 創造分身

### 學習元素

- 學習專案擴充
- 學習使用「分身」

### 課程教案

時間	課程	學習評估	附註
0:00~ 0:05	檢查一下作業，並注意有沒有值得分享或鼓勵的專案		
<b>Round Two - 拿過去的專案做重新發想、設計、擴充 (40 mins)</b>			
0:06~ 0:45	<ol style="list-style-type: none"> <li>1. 引導學生最喜歡過去哪個專案?想如何擴充?(5min)</li> <li>2. 讓學生自由創作，拿過去的專案做重新發想、設計、擴充 (25min)</li> <li>3. 鼓勵學生向左右同學展示作品 (10min)</li> </ol>		
<b>Advanced Concepts - 學習 Video, Cloning 等進階技巧 (30 mins)</b>			
0:46~ 1:15	<ol style="list-style-type: none"> <li>1. 介紹分身：                     <p><a href="https://www.youtube.com/watch?v=vrlvGzxrflA">https://www.youtube.com/watch?v=vrlvGzxrflA</a></p> <p>藉由 Battle Cats 的遊戲影片說明分身的用途，接著教大家如何做出「我方城堡」派貓咪出來。而「敵方城堡」自動產生貓咪的功能要請學生自己做。</p> <p>如果有人問怎麼對打，在時間允許的狀況下再教，或是請他們回家動動腦。</p> <p>&lt;範例&gt;： <a href="https://scratch.mit.edu/projects/67718212/">https://scratch.mit.edu/projects/67718212/</a></p> </li> <li>2. 詢問學生：                     <p>「使用分身有什麼好處呢？」</p> <p>以及</p> <p>「為什麼不直接創造很多不一樣的角色就好呢？」，</p> <p>試圖引導他們說出以下優點：</p> <ol style="list-style-type: none"> <li>a. 彈性 - 能依不同狀況產生不同數量的分身</li> <li>b. 維護性 - 當同一類型的角色的程式碼或造型需要變動時，不用一一改動。</li> <li>c. 便利性 - 不用辛苦的一個</li> </ol> </li> </ol>	<ul style="list-style-type: none"> <li>● 瞭解到如何在Scratch中使用分身</li> <li>● 了解到使用分身以及多個同樣角色的差別</li> </ul>	

	個創建新角色		
--	--------	--	--

## Lesson 13 - 專案發想

### 學習元素

- 複習「分身」
- 了解不同類型的遊戲
- 學習規劃專案
- 了解自己將要開發的專案

### 課程教案

時間	課程	學習評估	附註
<b>Advanced Concepts - 複習 Video, Cloning 等進階技巧 (20 mins)</b>			
0:00~ 0:20	複習 Video, Cloning 等進階技巧	<ul style="list-style-type: none"> <li>● 複習如何在 Scratch 中使用 <b>分身</b></li> <li>● 確實了解到使用分身以及多個同樣角色的差別</li> </ul>	
<b>Project Pitch - 輪流發表自己的點子 (25 mins)</b>			
0:21~ 0:45	1. 先由老師拋磚引玉, Demo 以下: <ul style="list-style-type: none"> <li>● 塔防(battle cats): <a href="https://www.youtube.com/watch?v=2KyAlnQ2M8M">https://www.youtube.com/watch?v=2KyAlnQ2M8M</a></li> <li>● 故事動畫(賣火柴的小女孩): <a href="https://www.youtube.com/watch?v=8EIIIcx0hu4&amp;list=PLCCBB672D73A7D3E4">https://www.youtube.com/watch?v=8EIIIcx0hu4&amp;list=PLCCBB672D73A7D3E4</a></li> <li>● 格鬥遊戲(小朋友齊打交): <a href="https://www.youtube.com/watch?v=429BKs1Gp4g">https://www.youtube.com/watch?v=429BKs1Gp4g</a></li> <li>● 射擊: <a href="https://scratch.mit.edu/projects/16042635/">https://scratch.mit.edu/projects/16042635/</a></li> <li>● <a href="https://scratch.mit.edu/projects/37322706/#editor">https://scratch.mit.edu/projects/37322706/#editor</a></li> <li>● 貪吃蛇: <a href="https://scratch.mit.edu/projects/28193692/">https://scratch.mit.edu/projects/28193692/</a></li> <li>● Jetpack: <a href="https://scratch.mit.edu/projects/31920734/">https://scratch.mit.edu/projects/31920734/</a> (15mins)</li> </ul> 2. 請小朋友踴躍發表意見, 並寫下想法	藉由不同的遊戲或專案, 刺激學生的 <b>創意</b> 。	
<b>Project Planning - 有系統、有方法的規劃專案 (25 mins)</b>			
0:46~ 0:50	老師提出問題, 鼓勵學生多回答 1. 我最喜歡的專案(從以前到現在,		配合專案發想學習單1~3頁

	<p>我最喜歡的專案是...)</p> <p>2. 我的Hackathon專案點子(下一個專案, 你想做什麼)</p> <p>3. 我的專長和興趣(有那些知識、技能、天賦是你投入專案的?)</p>		
0:51~ 1:00	<p>老師提出問題, 請學生能越詳細寫下來越好</p> <p>1. 我的專案</p> <ul style="list-style-type: none"> <li>○ 請敘述一下你正要開發的專案</li> <li>○ 為了完成這個專案, 需要哪些步驟, 請寫下來</li> </ul> <p>2. 我的資源</p> <ul style="list-style-type: none"> <li>○ 你目前已經有哪些資源了</li> <li>○ 你還需要哪些資源, 來開發你的專案?</li> </ul>	<p>清楚了解自己將要開發的專案, 充分利用已有的資源, 學習找尋需要的資源</p>	
1:00~ 1:10	<p>專案草圖: 老師說明此頁, 並請學生將所有畫面應有的背景、角色、輸贏畫面...等, 詳細畫出。</p> <p>如果是<b>遊戲</b>, 應至少構想出:</p> <ol style="list-style-type: none"> <li>1. 起始畫面</li> <li>2. 遊戲畫面</li> <li>3. 勝利畫面</li> <li>4. 失敗畫面</li> </ol> <p>如果是<b>動畫</b>, 應至少構想出:</p> <ol style="list-style-type: none"> <li>1. 起始畫面</li> <li>2. 結束畫面 (The End)</li> </ol>	<p>學習專案規劃, 將自己的創意畫下, 表達想法。</p>	

## Homework

- 繼續實做自己的專案

## Lesson 14 - 專案實作(1)

### 學習元素

- 學習實作專案

### 課程教案

時間	課程	學習評估	附註
<b>Design Sprint - 動手實作專案 (70mins)</b>			
0:00~ 1:10	老師來回走動，解決問題、給予提示		

### Homework

- 繼續實做自己的專案

## Lesson 15 - 專案實作(2)

### 學習元素

- 學習實作專案

### 課程教案

時間	課程	學習評估	附註
<b>Project Sprint - 動手實作專案 (70 mins)</b>			
0:00~ 1:10	給學生時間實作，並提醒學生：等一下老師會需要收集學生的專案做 Review。		

### Homework

- 繼續實做自己的專案

## Lesson 16 - 檢視專案

### 學習元素

- 學習收集別人對自己專案的意見
- 學習對他人專案提出意見
- 了解自己專案的優缺點

### 課程教案

時間	課程	學習評估	附註
<b>Project Feedback - 有系統、有方法的收集別人對自己專案的意見 (40 mins)</b>			
0:00~ 0:40	<p>首先，請學生將自己的作品統一上傳到老師預先開好的工作室。</p> <ul style="list-style-type: none"> <li>● 人少時(約10人以下)，一一展示工作室中的學生作品</li> <li>● 人多時，詢問有沒有學生自願向大家展示自己的專案，沒有或不足額時，由老師挑選幾個專案做展示。</li> </ul> <p>在展示時，詢問各位學生：</p> <ol style="list-style-type: none"> <li>你們喜歡這個專案的什麼部分？(角色、背景、設計、互動性，或特別的功能) <b>(綠)</b></li> <li>你們有發現哪個部分是還沒被做好、或是壞掉的？ <b>(紅)</b></li> <li>你們覺得有哪個已經完成的部分，是可以被改進的？ <b>(黃)</b></li> </ol> <p>並請專案的製作者，將大家的Feedback記錄在「紅, 黃, 綠」學習單中。</p>	<ul style="list-style-type: none"> <li>● 學生是否能就事論事，針對作品的狀況提出讀到的見解？ &lt;分析能力&gt;</li> <li>● 學生是否能傾聽並接受同儕的建議 &lt;溝通能力(傾聽)&gt;</li> </ul>	<ul style="list-style-type: none"> <li>● 注意勿讓學生提出破壞性建議、或是針對個人進行攻擊。</li> <li>● 當學生的作品幾乎只被提出缺點(紅、黃)時，老師應該適時的提出這個專案的優點，建立孩子的自信心。</li> </ul>
<b>Project Check-in - 暫停開發，並細細檢視自己目前的專案作品 (30mins)</b>			
0:41~ 1:10	<p>請學生檢視Feedback學習單，並討論以下問題：</p> <ol style="list-style-type: none"> <li>專案進度             <ol style="list-style-type: none"> <li>目前為止,專案中的哪一個部分是你最喜歡的?</li> <li>有哪些部分是還需要改進的?</li> </ol> </li> <li>下一步             <ol style="list-style-type: none"> <li>接下來你要實作的是專案的哪一部分?</li> <li>為了讓你的專案更進一步,你還需要什麼協助?</li> </ol> </li> </ol>		

## Homework

- 繼續完成專案

## Lesson 17 - 專案實作(3)

### 學習元素

- 了解如何訪談、如何鎖定對象
- 學習精進專案

### 課程教案

時間	課程	學習評估	附註
<b>Unfocus Group -</b> 讓其他人試用自己的專案作品，觀察這些人操作的過程，並訪談這些人的想法 (40 mins/skip)			
skip	<ol style="list-style-type: none"> <li>1. 向學生敘述什麼是「訪談小組 (Unfocus Group)」</li> <li>2. 詢問學生以下問題，並記錄在「訪談小組」第一張學習單的「鎖定對象」中：               <ol style="list-style-type: none"> <li>a. 哪些人或許能夠對你的專案提供一些獨到的意見？</li> <li>b. 你打算向哪兩個人分享你目前的專案？</li> </ol> </li> <li>3. 請學生回家以後，訪問這些鎖定對象，並將此單元的學習單完成。</li> </ol>	了解如何有效的鎖定測試對象，收集有用的使用者回饋。	
<b>Showcase Prep -</b> 為成果展做準備，將自己的作品與發表變得更好 (70 mins)			
0:00~1:10	讓學生精進自己的作品，為「訪談小組」及成果展作準備。		

### Homework

- 找父母或朋友組 Unfocus Group，將作品展示給他們看，並取得使用者的feedback
- 為下週成果展做準備

## Lesson 18 - 期末發表

### 學習元素

- 學習發表專案，展示專案最好的一面
- 學習表達能力

### 課程教案

時間	課程	學習評估	附註
<b>Showcase Prep -</b> 為成果展做準備，將自己的作品與發表變得更好 (10 mins)			
0:00~ 0:10	讓學生實作期末專案，並提醒無論如何等等都要上台發表。並預先將學生的專案收集到studio中。		
<b>Showcase-</b> 成果發表，將自己作品最好的一面呈現出來 (60 mins)			
0:11~ 1:10	讓學生輪流發表各自專案。	● 學習如何適當的表達自己的專案 <表達能力>	<b>務必</b> 妥善控制時間，讓每個學生都能有發表的機會。
<b>老師講評時間 -</b> 針對專案特色及特點，鼓勵學生。(10 mins)			
1:11~ 1:20	1. 針對專案特色及特點，給予講評並鼓勵學生。 2. 當提早結束時，讓學生在自己的電腦上開啓專案，並離開自己的座位，到別人的電腦操作看看別人的專案。		