

使用 KODU 遊戲實驗室

設計你的專屬遊戲

輔導手冊

Microsoft

PRESENTED BY
 **ideaslab**
EXPLORING NEW POSSIBILITIES

目錄

簡介	4
一、 遊戲設計介紹& Kodu 遊戲實驗室	5
活動 1-1 自我介紹.....	5
活動 1-2 介紹 Kodu 遊戲實驗室	6
活動 1-3 什麼是遊戲?.....	7
活動 1-4 Kodu 遊戲.....	9
二、 Kodu 基本概念.....	10
活動 2-1 介紹 Kodu 的程式語言.....	10
活動 2-2 解析範例 Tutorials 2 and 3.....	12
活動 2-3 了解虛擬機器人與物件.....	17
活動 2-4 程式執行 Bots.....	18
活動 2-5 創造 2 bots	21
三、 遊戲規劃設計過程	22
活動 3-1 創意激發.....	22
活動 3-2 遊戲發想.....	22
活動 3-3 學校參考/個案探討	23
四、 創造遊戲世界.....	24
活動 4-1 設計遊戲空間.....	25
活動 4-2 遊戲世界的模板與指引.....	25
活動 4-3 遊戲空間版型.....	25
五、 遊戲設計模組	30



活動 5-1 顯示遊戲進度.....	30
活動 5-2 遊戲操作模式.....	41
六、 進階問題解決.....	49
活動 6.1 設計過程.....	49
活動 6.2 程式錯誤刪除，錯誤發現，開發工具.....	50
七、 如何融入學校課程.....	52



簡介

- 一． Kodu Game Lab 是一種以視覺化方式呈現的遊戲程式設計語言(Visual programming language)．使任何人即使沒有程式設計的先備知識也能獨自創造屬於自己的 3D 遊戲。Kodu 利用直覺式的圖示來操作以及修改角色及物件的各式屬性。Kodu Game Lab 主要是設計來幫助老師在學校課程中介紹遊戲設計時使用。所以接下來的課程將會介紹及示範 Kodu Game Lab 如何讓每一個人都可以創造很豐富、刺激的遊戲。
- 二． 此講義會涵蓋設計以及使用 Kodu Game Lab 來創造遊戲的能力，並且發展設計過程和 Kodu 如何在學校課程運用。經過設計以及建立他們自己的遊戲，使用者將會了解到運用
- 三． 利用簡單的技巧，也能創造強大有趣的遊戲

這本輔導手冊將引導你:

- ✓ 能夠運用其中的角色以及物件
- ✓ 能夠建立 3D 虛擬世界
- ✓ 介紹主要的遊戲設計工具
- ✓ 提供多樣化的編碼範例
- ✓ 提供多樣化的遊戲範例
- ✓ 提供更即時的幫助，讓使用者使用上更方便，更具信心

提醒您；

Kodu Game Lab 還在研究發展階段。這本指導手冊所列舉的範例以及工具是適用於 Kodu Game Lab 1.48 版。當您使用其他版本時，請參照 <http://media.planetkodu.com/workshop/resources.html>，裡面有詳細版本升級資訊以及輔導內容。



一、 遊戲設計介紹& Kodu 遊戲實驗室

活動 1-1 自我介紹

利用此活動讓組員認識彼此。組員利用十分鐘來建立各組的遊戲角色，然後再用十分鐘與各組分享。

如果你是遊戲角色，你會想成為什麼？

名字: _____

所具備的能力與特色: _____


那三件事情使你這個腳色顯得非常特別：

1.

2.

3.

在格子內畫出你的角色



5

活動 1-2 介紹 Kodu 遊戲實驗室

幾項溝通要點:

1. Kodu 是一個快速遊戲開發的介面及語言
2. Kodu 使用小圖示來作為開發語言的基礎
3. Kodu 適用於開發小型遊戲

在這項活動，我們將開始了解 Kodu Game Lab 的用途，以及觀摩其他玩家的作品(您可以到線上去搜尋或是參考我們的網站)。請使用以下空白，將與 Kodu Game Lab 有關的任何重要的想法記錄下來



活動 1-3 什麼是遊戲?

如果是一位開發者(設計師或是教師)您可以去詢問你的玩家(使用者或是學生)他們所想要玩的遊戲，並且鼓勵他們列出不同種類的遊戲，例如運動、棋盤遊戲、電腦遊戲等。並且請他們從中去發現，這些遊戲有什麼共同點，還有遊戲是由什麼所組成的。利用 Diamond 9 思考輔助圖表來歸納及思考組成遊戲的最重要因素

參加者應在活動之後互相分享他們的想法

提示重點:

- 遊戲是好玩的
- 遊戲是有規則的
- 遊戲必須具備一項任務/遊戲中有贏家也有輸家
- 遊戲需要玩家去做決定以及需要相關的遊戲技巧
- 遊戲是可以藉由經驗法則使其本身更有趣

1.

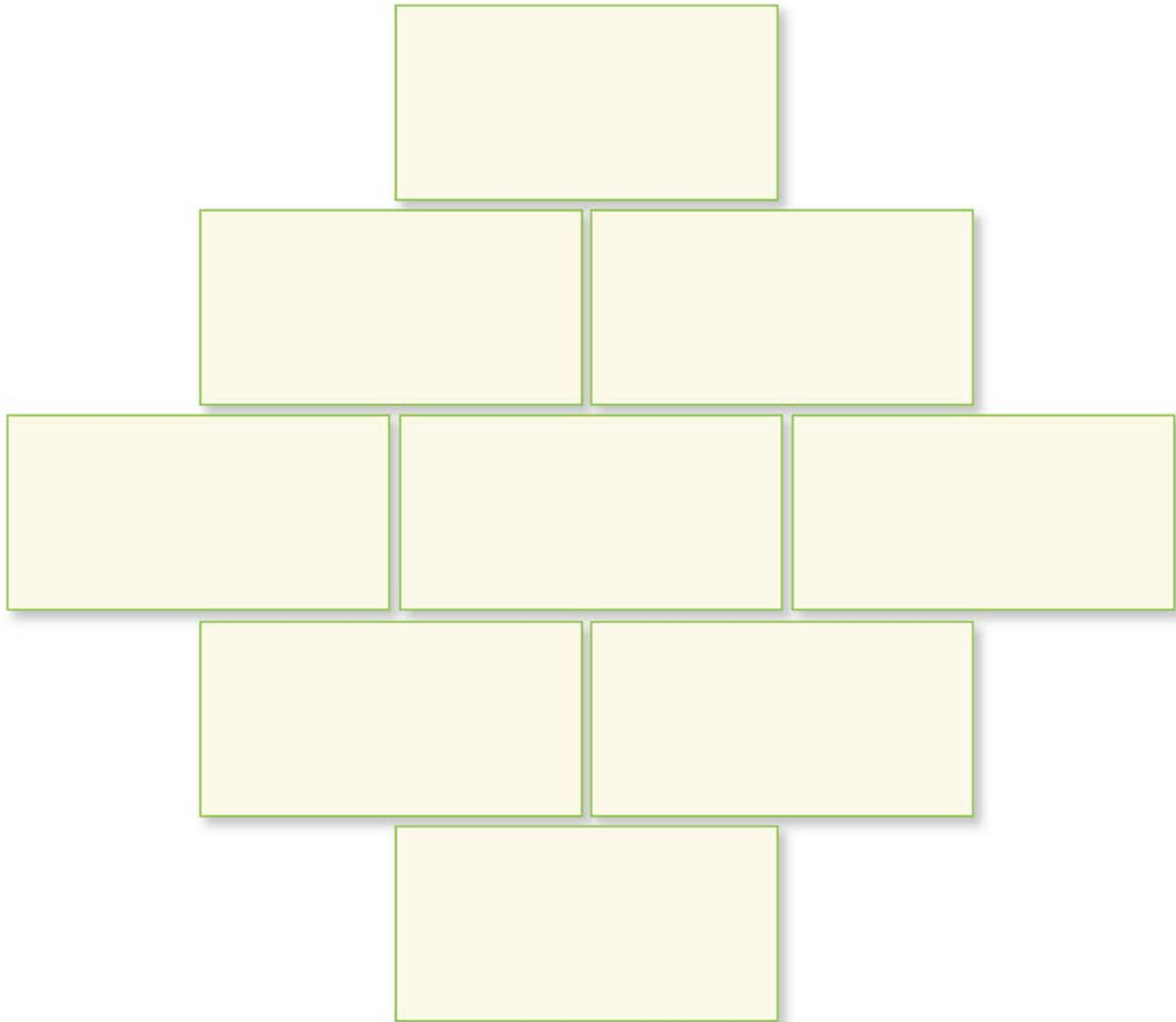
2.

3.

4.

5.

想著你列出的遊戲，列舉出該遊戲創作的**主要成分**，利用底下 Diamond 9 思考輔助圖表將遊戲的九項主要構成填入其中，並且將最主要的構成要素列在最上面的框格裡。



過來人怎麼說：

「或許你已經為一個軟體的子程式寫了超過十個以上的參數，但當你再回顧一次時，你將會發現你寫得還不夠完整！」 Alan Kay

「遊戲本身就是一個問題解決的歷程，而捷徑就是隨時保持一份玩樂的心！」 Jesse Schell

活動 1-4 Kodu 遊戲

這項活動目的是在於讓使用者去了解利用 Kudu Game Lab 會產生哪些種類的遊戲，並且探討是什麼原因讓他們如此有趣。時間分配如下：參與者將有 15 分鐘的遊戲時間，5 分鐘紀錄他們的發現，10 分鐘進行最後的討論。使用 Kodu Game 所產生的檔案去分享一些在 Xbox Kodu Community 建立的遊戲。

這個活動所使用的範例是 Bank Out 18，請參與者去討論遊戲的組成要件、遊戲規則以及為什麼它是有趣的。

遊戲探討(範例)：

遊戲名稱: Bonk-Out v18

主題: 擊倒對手的城堡同時保護你領地中的小塔。

規則: 一個按鈕將會啟動一個飛輪，推機器人也可以啟動飛輪，飛輪除了玩家之外，所有的東西都可以摧毀，玩家可以利用撞擊反彈控制冰球的方向。

好玩之處? 很容易就可以上手，因為玩法跟 Breakout 很像，移動很快速，如果遊戲時間可以延長些就更好玩。

你所設計的遊戲

遊戲名稱: _____

主題: _____

規則: _____

好玩之處? _____



二、 Kodu 基本概念

活動 2-1 介紹 Kodu 的程式語言

這章節裡我們會開始寫第一個 Kodu Game Lab 程式。Kodu Game Lab 內建了三個示範

示範 Tutorial 1 還有如何處理它

重點：腳色肢體動作及“ when” “do” 句子來表示程式語言

示範教學 1

- 1 打開 **Tutorial 1 world**

Kodu bot 會告訴你“ **I want to visit the castle** ”

2. 按鍵盤上的“ **ESC**” 鍵 或者 **start** (XBOX 360 的控制器)來編輯這個虛擬世界

3. 在工具欄上選擇

“Add and Program Objects”



圖示

4. 按左鍵來選擇角色，或者選擇 Y (遙控器)。編輯選項此時會出現，且選項會發黃光

5. 按右鍵打開 **Program**


這機器人的腳色的程式編碼將會出現，設計者有兩個個選擇，第一個是編輯已存在程式碼去編輯 Line 1，或者是編輯新的一行。以下所示範的是將既有的編碼重新編輯。第一行讀(「當(when)」被設計的腳色「看到(See)」「城堡(castle)」時，「做」「表現(express)」「喜愛(hearts)」的樣子，所以一開始當你打開這個範例的時候，執行時，KODU 腳色會顯得非常喜愛那個城堡。)



接著我們需要改變成



6. 要移除 Express 跟 heart 磚塊 按右鍵在那兩張圖上或者按 遊戲介面上的 **X** 圖示



7. 要增加程式碼，按住 **DO +** 左鍵  圖到“do” 磚塊的右邊(或者是在遙控器上按 **A**)

8. 選擇 move 

9. 選擇 Towards 



9. 按鍵盤上按住 Esc(或者在遙控器上按兩次 **B**)來結束此項目以及回到工具欄

10. 遊戲介面上點  來開啟程式， 或者用搖桿選擇  然後按 **A** 來啟動遊戲

11. Kodu bot 現在會移動到城堡


活動 2-2 解析範例 Tutorials 2 and 3

Tutorial 2 Solution

目標: 我們必須要從新啟動 cycle bot ，讓它能夠跳過河流

1. 打開 tutorial 2
2. 按 ESC(鍵盤) 或者 Start (遙控器)去編輯 world

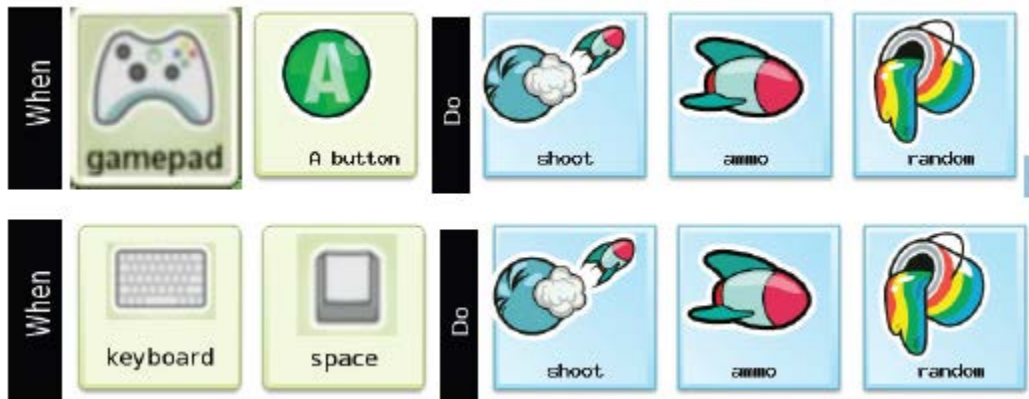
3. 在工具欄上選擇 **"Add and Program Objects"**  圖示

4. 按左鍵來選擇角色，或者游標到  (遙控器)。編輯選項此時會出現，且選項會發黃光

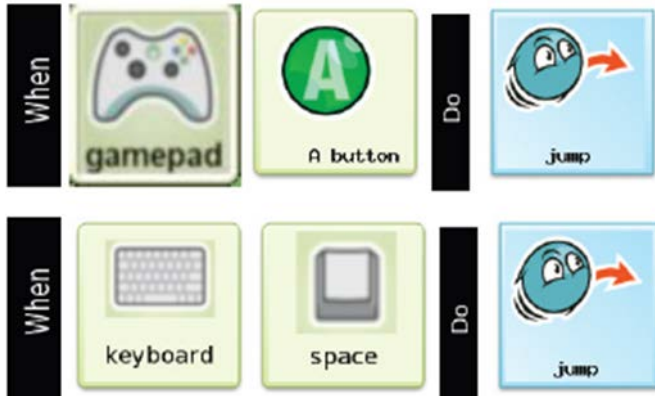
5. 按右鍵打開 **Program** 或者在遙控器選擇  按鈕

這 bots code 會出現，有兩個個選擇，第一個是編輯已存在程式碼 來編輯 Line3 或 line 4，或者是在程式碼中編輯新的行數。我們會編輯已存在的程式碼但是增加新的程式碼行數也是合適的選擇。

現在的程式碼為:





我們必須要改變他們為



6. 要增加新磚塊，按住 滑鼠左鍵在  圖上面到“ do” 編碼的右邊(或者是在遙控器上按 )



7. 利用滑鼠左鍵選擇  ，或者在遙控器上按 

注意:按住  Action(也就是動作)找尋  Jump(也就是跳躍)

程式碼現在變成為：



也就是當你使用遙控器按下 A 或是使用鍵盤按下空白鍵時，你的編輯腳色就會跳躍！


8. 按鍵盤上按住 Esc(或者在遙控器上按兩次 **B**)來結束此項目以及回到工具欄
9. 遊戲介面上點  來開啟程式 或者用搖桿選擇  然後按 **A** 來啟動遊戲
10. 按住空白建或者使用 **A**，Kodu 虛擬腳色將會進行跳躍的動作。


Tutorial 3 Solution

目標:我們必須要啟用 cycle bot ，讓它能夠撿起蘋果

1 打開 Tutorial 3

2.按 **ESC**(鍵盤) 或者 **Start** (遙控器)去編輯 world

3. 在工具欄上選擇 **"Add and Program Objects"**  圖示

4. 按右鍵來選擇角色，或者游標到  (遙控器)。編輯選項此時會出現，且選項會發黃光

5. 按左鍵打開 **Program** 或者在遙控器選擇  按鈕

第二行的程式碼會顯示



原本的程式碼：當腳色看到蘋果時，




腳色會顯示對話框

我們要改變它們為



6. 要增加新磚塊，在“+”圖示按住滑鼠左鍵到“Do”的右邊(或者是在遙控器上按

)


7. 選取 ，使用滑鼠左鍵或是在  來找 。注意是按取 holding group




程式碼現在變為



也就是當腳色看到蘋果時，進行抓取

的動作。

8. 按鍵盤上按住 Esc(或者在遙控器上按兩次 )來結束此項目以及回到工具欄

9. 遊戲介面上點  來開啟程式 或者用搖桿選擇  然後按  來啟動遊戲

10. 虛擬 Kodu 將會撿起正在成長的蘋果，越過橋

活動 2-3 了解虛擬機器人與物件

在此項活動，我們將會探討多項在 Kodu Game Lab 中的虛擬機器人及物品。我們將會探討各項不同虛擬 Bots 的功能與特性。

使用 Bots and Objects World 來示範在 Kodu Game Lab 裡可用的 bots 物件

重點整理:

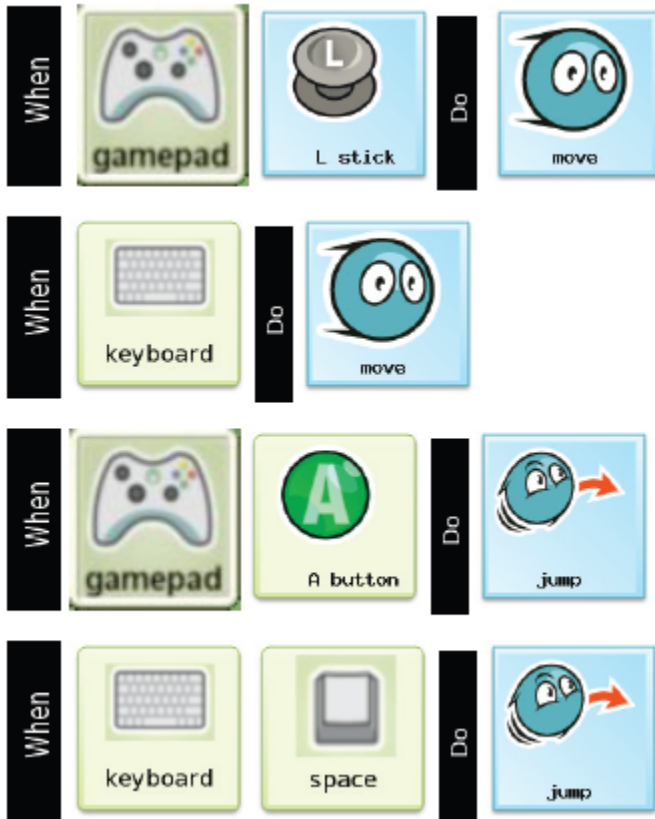
1. 不同的 bots 有不同的功能與特性
2. 使用不同顏色來去分不同特性的角色
3. 所有的 bots 與物件設定時是在遊戲建立時設定而不是在跑程式系統時設立



活動 2-4 程式執行 Bots

玩家掌握 bots

我們在以上的示範教學中看到我們利用了鍵盤或者是 XBOX 的搖桿來寫簡單的程式。我們也可以利用鍵盤上的按鍵或是搖桿上的特定按鈕來設定跳躍。



AI bots

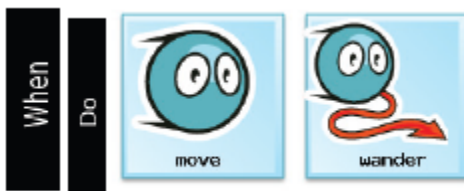
現在要開始看如何設計其他 Bot 與玩家的 bot 之間的互動關係



實際而言，我們常選擇“ see” 多過於“ hear”



如果只想要讓 bots “wander” (徘徊、閒逛) 多過於 follow，我們可以使用



“When” 肢體感應:

重要重點

1. See(看到),hear(聽到),bump(碰到).shot(射擊) hit(撞擊), health(體力), held by, on land(島上) and on water(水上) 都是 Bot and objects 的肢體動作
2. Timer(計時), scored(得分) gamepad(搖桿操控方式), keyboard(鍵盤動作) and mouse(滑鼠動作) 是 bots and object 外在指令



“DO” 肢體動作:



Bots and Objects 他們有特定的屬性來檢視他們的行為。當遊戲進行的時候，程式設計並不能更改這些 bot 的屬性，只有在設計階段裡才能做更改


活動 2-5 創造 2 bots


在這活動，我們會建立簡單版本的 two bots game

這個活動目的是在於建立簡單的 two bot game。此遊戲不應該太複雜且應該在一定時間內建立

1 打開 **Small World with Water**


2.按 ESC(鍵盤) 或者 Start (遙控器)去編輯 world




3. 在工具欄上選擇 **"Add and Program Objects"**  圖示

4. 按左鍵來選擇角色，或者按  (遙控器)。編輯選項此時會出現，且選項會發黃光

5. 按左鍵打開 **Program** 或者在遙控器選擇  按鈕

6. 增加程式碼

7. 按鍵盤上按住 Esc(或者在遙控器上按兩次 )來結束此項目以及回到工具欄

8. 遊戲介面上點  來開啟程式 或者用搖桿選擇  然後按  來啟動遊戲

9. 重複此步驟來增加第二個 bot

三、 遊戲規劃設計過程

活動 3-1 創意激發

請組員試想一下他們剛做好的遊戲或者試著讓他們去思考如何把遊戲做得更好
請組員把想法記錄下來

活動 3-2 遊戲發想你的遊戲

遊戲名稱: _____

主題: _____

規則: _____

Bots Used _____



活動 3-3 學校參考/個案探討

通過這項活動，我們要退一步想學校可以使用多少 Kodu Game Lab 在學習課程裡



四、 創造遊戲世界

Time 20 Min

KODU GAME DESIGN TEMPLATE

Score Board

Random Req and Sec Developme

36 + 36 = 44 + 44
156 + 156 =

Draw and label your Kodu world.

The drawing shows a hand-drawn game world with various elements and labels:

- Start**: A circular area on the left.
- Maze**: A circular area with a central point.
- Green**: A large area on the right.
- Trees** and **Forest**: A cluster of scribbled shapes on the right.
- Apples**: Several small circles scattered throughout.
- Bad guys**: A small figure in a box.
- Factory**: A structure with a chimney on the left.
- Cannons**: A structure with a barrel on the left.
- Castle**: A structure with a flag on the left.
- Finish**: A circular area on the right.
- Answer**: A circular area on the right.

Mathematical notes in the drawing include: 5×10 , $10 + 30$, and $10 + 30$.

<p>Describe the Game (and what do the players have to do?)</p> <ol style="list-style-type: none"> 1. Collect Red apples. 2. Answer the questions 3. Find Bonus stars 4. Don't get pink apples 5. get past bad guys 6. don't get shot by cannons 7. Get star in castle 8. Tele port to answer 	<p>List the Kodu objects and their purpose.</p> <p>Apples: Get points Pink apples: Lose points Stars: Bonus Points Trees: Forest Kodu: controller</p>	<p>Describe how a player wins/finishes the game, how a player scores and how long your game goes for.</p> <p>It goes for 20 Minits. you win by collecting stars and apples and you have to go throa every stage and to win fully you have to get to the big answer at the end of the game.</p>
--	---	--

活動 4-1 設計遊戲空間

這個活動將協助設計者去了解如何在設計遊戲時，去注意關於遊戲的資源分配、遊戲虛擬世界的設定，以及鏡頭的安排。

較大的遊戲虛擬世界必須盡可能地將這些大量的資源，有效的使用在受侷限的虛擬土地上。

活動 4-2 遊戲世界的模板與指引

Kodu 內建的模板以及相關指引，提供設計者擁有方便測試及易懂的步驟來解決程式設計過程問題。這個活動將會說明遊戲設計圖示如何幫助遊戲設計者利用圖示來解決一般遊戲設計的問題

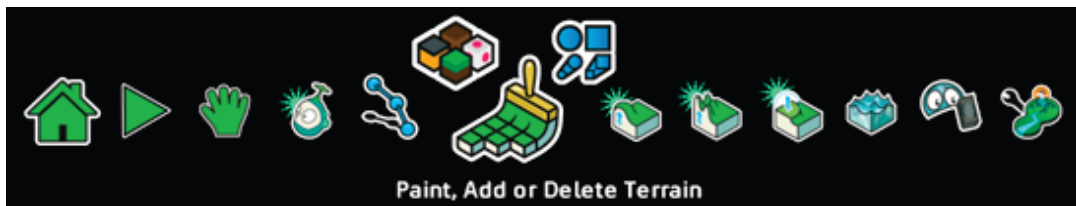
它會介紹程式碼種類的概念，也就是一般程式撰寫特定語言問題的解決方法

此活動會介紹 設計圖示以及程式碼種類的概念。

重點:設計模板大多是一些遊戲設計的標準的解決方式，並且提供解決方式的指引以及程式設計的範例。



此活動會介紹一些遊戲設計版型。**設計模板:Open World**

在 Kodu 中使用 open world 設計模板進行這個活動的體驗，該模板是一個允許設計者自由漫步在遊戲虛擬世界的模板，是一個非線性的遊戲設計範本(也就是沒有故事主軸，自由探索的遊戲腳本)。



	<p>使用 Ground Brush 去增加或移除地面。按住搖桿的  或者滑鼠左鍵來增加地面，而使用搖桿的  或者滑鼠右鍵來移除地面。</p> <p>多變形狀和不同尺寸的刷子可以在增加大型面積陸地時選擇，同時用於精細的設計中。地區的颜色也可以在很多樣式中選擇</p>
	<p>使用 Up/Down Brush 來創作小山丘以及山谷</p> <p>按住搖桿  紐或者滑鼠右鍵來填平選擇的陸地，左鍵是使其傾斜，不同的形狀及尺寸可以用來創造不同的效果</p>
	<p>使用 Water Brush 來增加或移除水色。按住搖桿的  或者滑鼠左鍵來增加水平面 in your world，而使用搖桿的  或者滑鼠右鍵來降低水平面</p>
	<p>使用 Flatten Brush 來製作平的面積 或者坡道。按住搖桿的  或者滑鼠左鍵來增加陸地高度，而使用搖桿的  或者滑鼠右鍵來降低選擇的陸地高度</p>
	<p>使用 Roughen Brush 來增加凸性面積。按住搖桿的  或者滑鼠左鍵來使選擇土地增加尖，而使用搖桿的  或者滑鼠右鍵來增加陡的程度</p>
	<p>使用 Delete Tool 來快速移除 bots and objects</p>

當編輯修改一個 world 時，使用 F3 可以打開 **Snap to Grid** 特性，刷子尺寸會維持相同尺寸，座標會在螢幕的左下角出現，協助您在設計時，可以有所依規。

按住  和  或者滑鼠滾輪來放大或縮小，放大將有助於發現你的游標，通常游標會比較難發現，因為它躲在陸地的後面

重要提示：

當創造你的你的世界時，注意資源測量儀(在畫面的右側，長得像是一支溫度計) 並且確認它不會進入到紅色地區(造成系統負載過大)。在創造大型虛擬世界時，這是很重要的一環。

設計指南；競賽

當對 bots 進行競賽對抗的設計時，最簡單的設計方式就是創造一個路徑，並且使 bots 依照著這個路徑來進行競速。

競賽遊戲需要玩家在跑道或者空間裡競賽。賽道內通常會設置檢核點，遊戲的最終目標通常是擊敗對手(比對方快速)或者在固定距離內使用更少的時間(破紀錄)。競速遊戲通常與現實生活相似，所以這一類型的遊戲，通常令人容易上手。

示範 Pure Plasma Racing game

重點：利用路徑去控制 bots 的行徑路線。

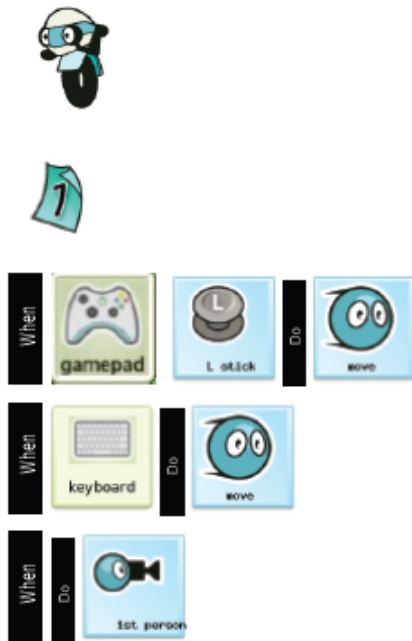


先在 world 裡面放置設計好的路徑，再將 bots 放置在路徑上後，再依上面的圖示進行編寫即可。

設計指引：迷宮遊戲

當創造一個迷宮的時候，比較簡單的作法是將整個迷宮的占地範圍的土地都升高然後降低迷宮通道所在的位置。如果你是利用建立單面牆來設計迷宮的話，在 kodu lab 裡面是比較困難的。

為了使迷宮遊戲更加有趣，我們通常不會將攝影視角跟往常一樣，選擇跟在 bots 的後面，迷宮遊戲建議使用第一人稱的攝影視角，這樣更能顯得具臨場感，設計者可以依照以下的步驟設定腳色的程式碼。



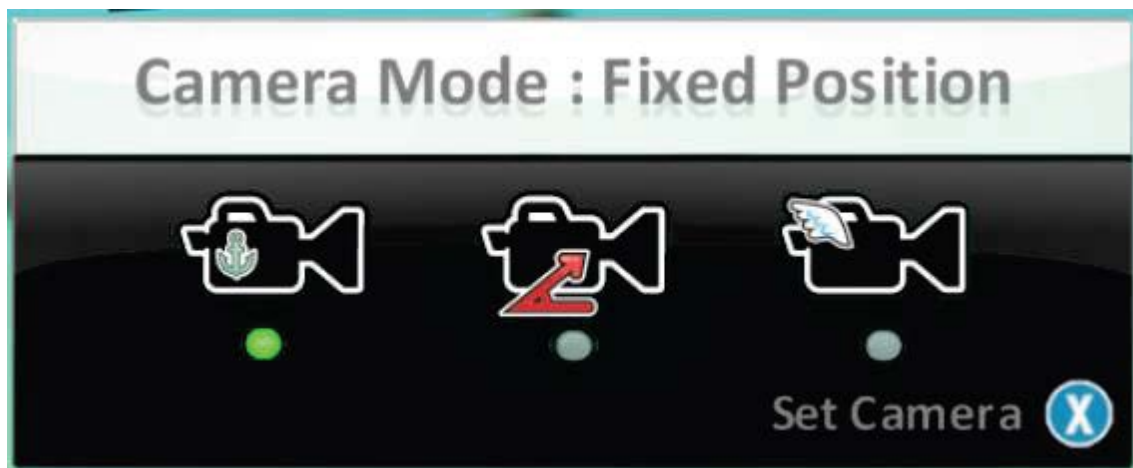
設計圖示；固定遊戲視角

一些遊戲是小的固定空間，這些遊戲空間是更適合多玩家且不必要拆開到不同視窗，固定的 camera 讓玩家有相同的方向及角度。Pakodu 是一個不錯的範例

方法:Fixed Game Board

我們需要鏡頭保持同樣的方向跟角度。要做到這樣我們必須要設定鏡頭的型態選項來固定住它的位置(fixed Position)

- 一. 選擇 World Setting
- 二. 選擇相機型態:Fixed Position
- 三. 按 **X** 按鈕來放相機鏡頭到它原來的位置，然後按住按鈕 **A** 來設置位置



五、 遊戲設計模組

活動 5-1 顯示遊戲進度

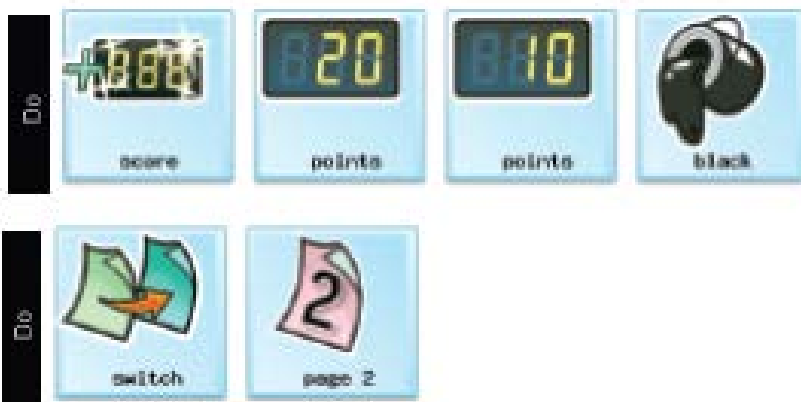
此活動介紹更高階的 kodu 程式進行概念，像是創造，程式碼分頁，使用腳色得分來當變數等。

設計指引：時間限制

限制玩家需要在一一定的時間裡去完成一項任務或者達到一個目標來贏得遊戲。倒數計時器通常都會顯示在螢幕上，有些遊戲會因為腳色在遊戲過程中完成某些任務來增加遊戲進行的時間，以利腳色完成所有的任務。

Kode 方法:時間限制

1. 確保你所擺置的計時器不會因為遊戲的進行而遭到破壞。
2. 可以去修改你計時器的顏色，但不建議修改計時器上數字的顏色。
3. 這個範例使用黑色碼錶並且設定遊戲進行時間為 30 秒，玩家將會在 30 秒後會贏得此遊戲。
4. 參考 Whack-a-Llama 這個範例，來看這個程式該如何設計。



2

When	 timer	 second	Do	 subtract	 point	 black
	 scored	 black		 point	Do	 switch

3

When	 timer	 second	Do	 win
------	--	---	----	--

設計模式：積分

方法

1. 這個方法每次打到飛行船就加 1 點到藍色的積分,打到飛碟就加 10 分,超過 100 分即可獲勝.
2. 把以下的設計指引加到 Bot 尚將使其具有射擊的能力。.
3. 這個方法適用於鍵盤以及 XBOX 遊戲搖桿
4. 詳請請見 Xevon v06 這個範例。



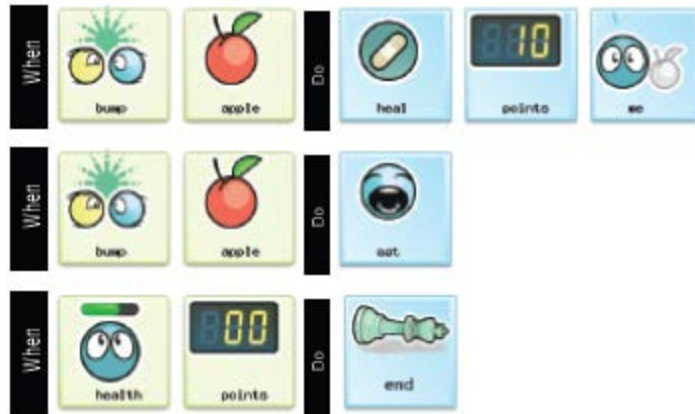
設計模式：體力值

玩家 BOT 的體力值也可以反應遊戲的進度,體力值表可以表示現在玩家的體力給予玩家最快的回應.

方法：體力值

1. 當 BOT 碰到金幣 10 分時,它會扣掉 BOT 的體力值，金幣會立即消失
2. 當 BOT 吃到蘋果 10 分時,他會增加 BOT 的體力值.
3. 當體力值歸 0 時，遊戲就結束.
4. Bot 的體力值可在 Bot 的設定當中去設置。數值為 0 到 1000 的值，可以以 5 為單位來增減.





設計模式：存檔點

存檔點在遊戲中很方便。通常在很難的關卡完成時，遊戲要可以自動存檔或者提供玩家可以自行存檔的選擇。就算玩家在下一個關卡輸了，還是可以直接跳回儲存點，這樣可以讓玩家不必再重玩簡單的關卡，減少他不想繼續玩下去的機率。

設計指引：存檔點

1. 玩家可自行創建自己的虛擬機器人
2. 紅色的分是用來記錄玩家的生命，1 代表玩家正在玩且未死，0 代表玩家需要從啟動。
3. 黑色的分是用來記錄玩家的儲存點，0 代表玩家剛進行遊戲，1 代表玩家已經玩到存檔 1 的地方，以此類推
4. BOT 必須要在每個場景以及時間點上都可以運作，在這個例子，我們所使用的是 STICK BOT，但不一定要指定使用這個虛擬腳色。
5. 在第三頁裡面我們將不設置任何與儲存點有關的編碼。
6. 儲存點的設計將會針對特定的 Bot 進行紀錄，以確保追蹤到最新的進度。
7. 不同顏色的分數可以用來追蹤和限制新生命的數量。此為小撇步。



The player controlled bot is a creatable



When	Do	score	red	point	once	
When	Do	gamepad	L stick	move		
When	Do	keyboard	move			
When	Do	bump	apple	subtract	red	point
When	Do	boom	no			

Starting point (save point 0)

When	Do	switch	page 2
------	----	--------	--------

{indent}代表的是，在程式編寫上需要進行縮排，請在數字上點一下後，出現上下箭頭後，直接拖拉往後，使其變成內縮。

Row handle
Use to cut and to paste rows. The numbering indicates the priority of the rules.

2



Save Point 1

1



2





Successive save point will need their black points incremented by 1

設計模式：多層關卡

在遊戲過程中，遊戲關卡能夠讓玩家詳細了解遊戲進度，並且隨著遊戲關卡的增加，難易度也會提高以及介紹新的遊戲技巧。

Halox' s Dual and Portal 遊戲在 Youtube 上為很好的多層關卡遊戲範例。

重點：這是一項比較進階的編寫範例

1. 黑色分數是用來記錄玩家的闖關進度
2. 設計者所使用的 Bot 不限，但在設計的初期就要先設定好。
3. 其後的關卡可以在複製第二關的開始與結束分數，然後在程式碼裡增加黑色分數
4. 先關閉黑色分數



玩家控制的 bot 是可創造的



第一關開始分數



第一關的結束分數



Stickbots: 第二關之開始分數

7



Stickbots:第二關之結束分數

7



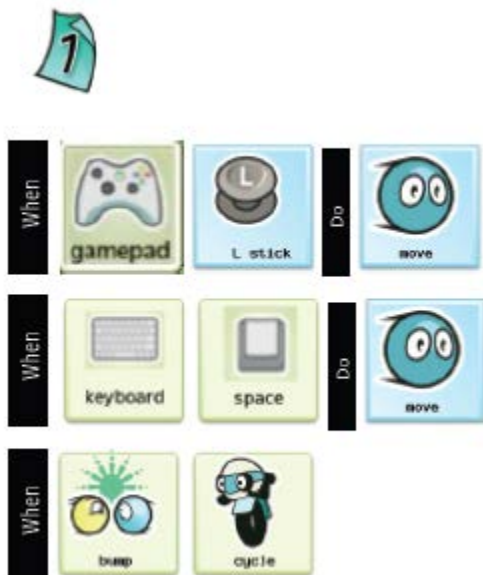
活動 5-2 遊戲操作模式

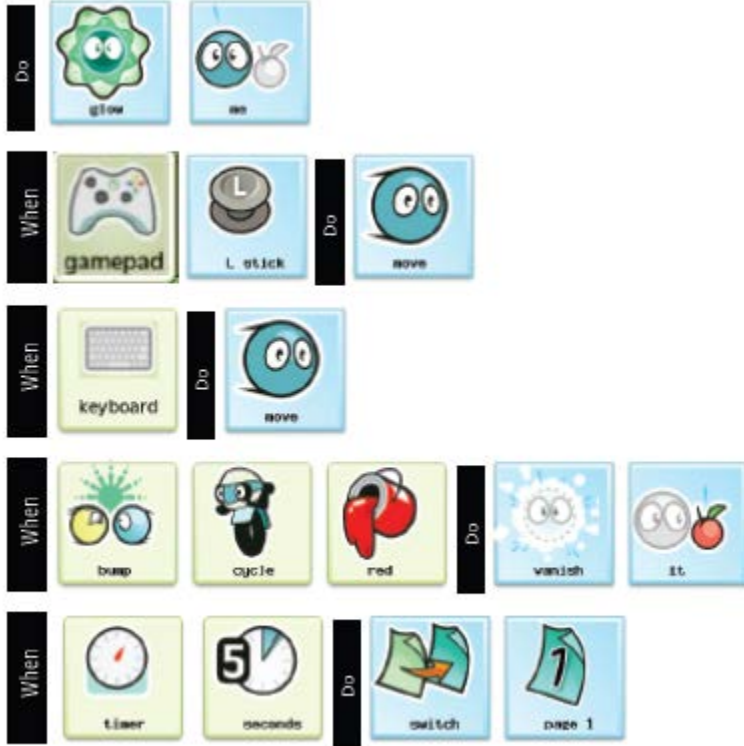
設計模式：力量提升(Power up)

在有時間限制的遊戲中，力量提升將會增加玩家的優勢。協助玩家可以完成許多困難度高的挑戰，bots 力量提升的時機，需要玩家思考決定何時使用，因為力量提升的條件往往牽連到時間以及能力的不同。

重點：這個範例需要藉由編寫頁面的切換以及善用計時器。

設計指引：當玩家所控制的 bot 撞到紅色的 cyclebot 遊戲就結束了，玩家所操縱 bot 吃到蘋果力量就會提升，當玩家所操縱的 bots 力量提升後，紅色的 cyclebot 撞到玩家所操縱的 bot 就會被摧毀，力量提升僅能維持 5 秒鐘，5 秒鐘之後能力便會回到原來的狀態。(也就是如果碰到紅色的 cyclebot，玩家所操縱的 bots，遊戲就結束了)。





設計模式：控制移轉

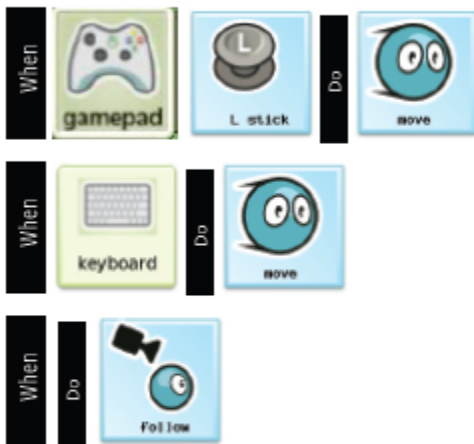
一些遊戲讓玩家在不同遊戲階段中掌控不同的角色。這會包含當一個角色進入車子或者搭乘船。這新的角色通常有不同的特性，導致遊戲通常有不同的結果而增加了玩家的興趣

方法:控制移轉

1. 這範例顯示出一個玩家如何控制 cycle bot，使它進入一艘船後玩家可以控制



Cyclebot:原始玩家控制 bot



Floatbot:當 cycle bot 登船時，玩家掌控的 bot。



When			Do		
------	--	--	----	--	--

{indent}

When		
------	--	--



When			Do	
------	--	--	----	--

When		Do	
------	--	----	--

When	Do	
------	----	--

設計模式：Big Boss 大魔王

遊戲或者關卡中通常結束在與大魔王的一場決鬥。大魔王是更難的對手且通常比起上一場對手有更多變化的能力。與大魔王對戰能夠加深對遊戲的黏著度(也就是喜愛的程度)。

1. 大老闆的難度是可以被調整的，可以被調整到難以擊敗的程度。
2. 大老闆的遭受攻擊時所作的反應以及效果可以被調整，最好調整到讓玩家覺得破壞之後。
3. 以下的範例說明大老闆如何創造出來的。



Rocks are thrown by the Boss, so create a single creatable rock.





活動 5.3 遊戲資料之圖示

跟玩家的互動是很重要的，可以瞭解遊戲型態以及動向。遊戲資訊可以來自於聲音以及分數上面的溝通。清楚的溝通可以幫助避免挫折以及不希望的結果

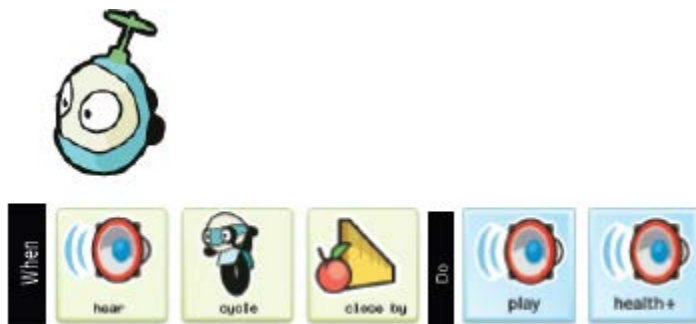
設計模式：鬧鐘

鬧鐘可備用於提醒玩家正處於險境，他們可以用來指示遊戲的開始或結數

不同聲音鬧響可通知玩家不同事件

方法:鬧鐘

此方法顯示出鬧鐘如何指示 cycle bot 的附近位置



設計模式：遊戲資訊

交談視窗可用於對使用者提供資訊，這些資訊可以通知使用再遊戲的進度或是提供怎麼完成遊戲的資訊

方法：對話

Kodu Game 有三種對話視窗的種類:全螢幕跟兩種想法氣球，第一種碼條是連續性的第二種是隨機的，當演講的對話視窗被使用時，全螢幕是用在提供使用者一般的訊息，且特定的角色使用來提供玩家資訊

當黑點數到達五點全螢幕對話會出現



使用  按鈕，當使選擇  圖示來選擇會被使用的對話視窗

六、 進階問題解決

活動 6.1 設計過程

在這個例子我們會研究適合學生的設計過程， Jesse Schell 提過 “想著問題並且確定最佳的解決方式，完成，測試”

一些建議：

1. 在一紙上計畫過程
2. 先由簡單遊戲開始執行，再深入測試，測試，再測試

描述與學生計畫的設計過程



活動 6.2 程式錯誤刪除，錯誤發現，開發工具

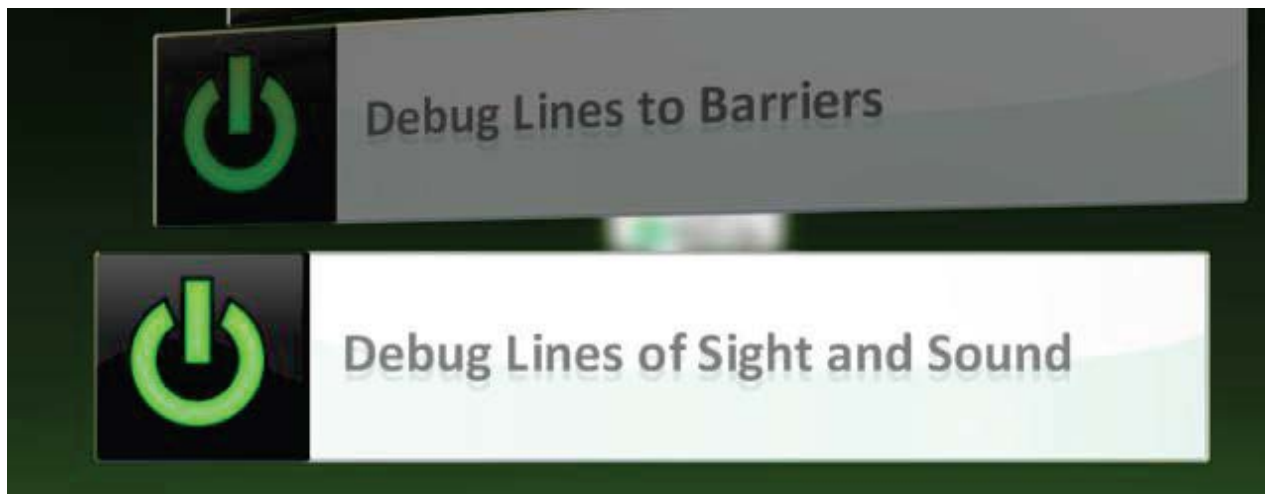
在這活動我們會觀察一些 kodu game lab 的功能，那將會讓遊戲創造者更容易簡單執行

確認風險：我們可以在 KODU 執行嗎？

有很多事可能或不可能在 Kodu Game Lab 上創造出來的。當設計一款新遊戲時，設計者發現遊戲沒辦法被製造時，這是最殘酷的因為這將是在浪費自己的時間。

刪除錯誤程式：為什麼沒辦法作業？


當遊戲沒辦法於自己預期進行時，錯誤程式碼需要被開啓。Sight and Sound 的錯誤程式碼會顯示

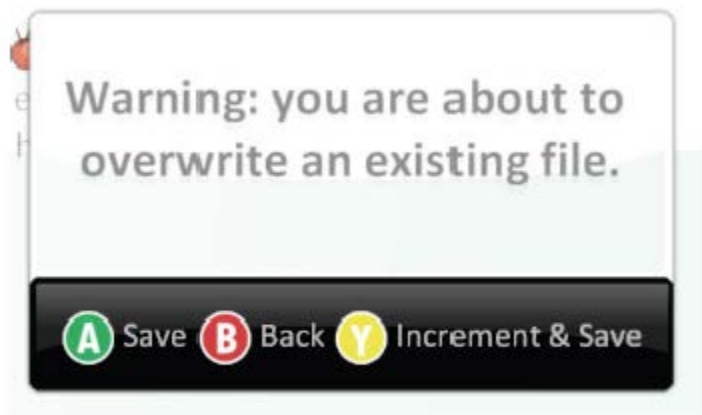


複製，貼上，複製: 還有更快的方法嗎？

當在相同的程式設計下，創造更多的 bots 是相對更快更容易的。Bots 跟 objects 可以在不同的活動中被剪下以及貼上 使用  去剪裁以及  來複製或者使用滑鼠按左右鍵操作。

WOOPS,我的遊戲無法使用了!?

當處存 kodu 遊戲新的重大變更時，最好是存在另外一個新的版本。這有效的建立備份，而當你的遊戲無法作用時，這將會是很有用的。當你儲存新的版本時，遊戲會提醒你儲存並且按 



使用資源偵測計(Resource Meter):為什麼我的遊戲那麼慢?

Kodu Game Lab 是用來做小型遊戲並且當遊戲被製造的很大時，kodu 會開始變慢，通常讓遊戲變得不好玩且無法進行



此圖形可以在打開 World Setting 裡被找到確定此 meter 要持續在紅色的部分，如果太低的話可以降低資源的使用像是減少土地的面積



七、 如何融入學校課程

和小組討論學校能如何使用 Kodu Game Lab 以及他們可以如何在學校推展 Kodu

Kodu 資源

網站:

<http://fuse.microsoft.com/kodu>

<http://planetkodu.com>

<http://boards.kodux.com>

所有的工作訪訊息從以下網址下載：

<http://media.planetkodu.com/workshop/resources.html>

© 2011 Microsoft Corporation. 版權所有.

可至 **XXXXXXXX** 下載此份文件

Microsoft, Windows, Office, Internet Explorer 及其他產品均為已註冊的商標或者微軟公司在美國或其他國家使用的商標。這份文件僅適用於資訊傳播的用途，文件中所記載的資訊僅代表微軟公司對產品的觀點，產品自本文件發布後將會有局部性的變更。

此文件非微軟公司對產品所提出的正式允諾，因此微軟公司不保證此文件的正確性。MICROSOFT 在這篇文件當中沒有做任何的保證、解釋、暗示或法定的公告。

