

要是有一天被關在自己家裡？

簡易密室逃脫遊戲

壹、前言

一、開發動機

作為一名初學者，最重要的莫過於實戰演練。與其進行單一 Function 的練習，不如嘗試製作內容完整的應用程式，能夠更快知曉各種 Function 交互作用的結果。

而選擇製作小遊戲的原因純粹是興趣使然，並將我對神話故事的依戀進行發揮，我從小便夢想著使用各種方法散播歡樂，無論是便捷的工具、抑或是遊戲，程式能力必定是不可或缺的要素。

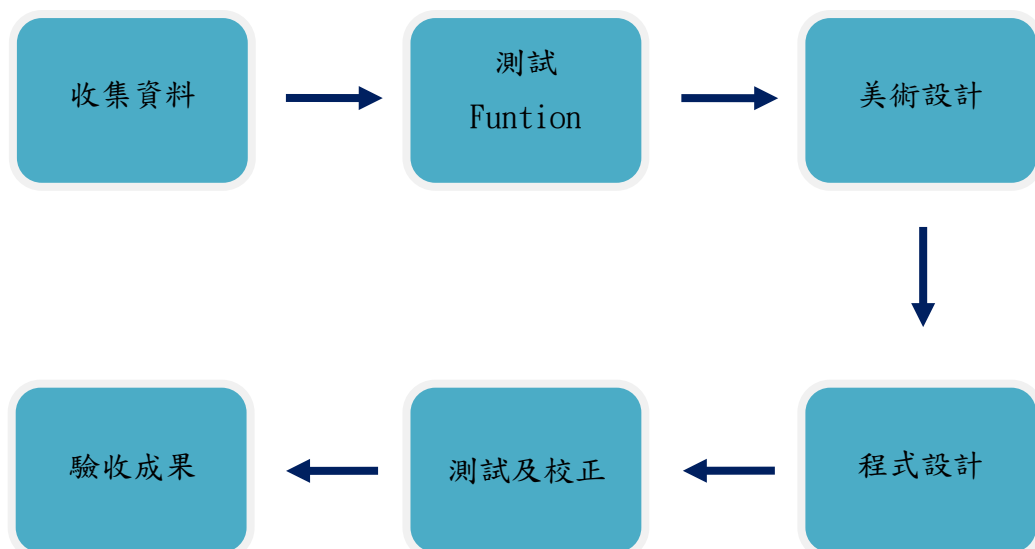
二、開發目的

- 領會程式語言的獨特邏輯，對於往後撰寫其他語言有所幫助。
- 了解各種 Function 的作用，如：微型資料庫、if 述句及圖像精靈等。

三、開發方法

- (一)測試各 Function 的實際作用情形。
- (二)實際操作 App Inventor。
- (三)使用網路及書籍尋找資料或範例。
- (四)將程式打包成 APK 或二維條碼以模擬器及手機驗收成果。

四、研究流程



貳、正文

一、App Inventor 2 介紹

簡稱 AI2，是一款卡通圖形界面的 Android 智慧型手機應用程式開發軟體。起先由 Google 提供的應用軟體，現在由麻省理工學院維護及營運。可以讓任何熟悉或不熟悉程序設計的人來創造基於 Android 作業系統的應用軟體，也適合作為初學者的練習途徑。

二、遊戲介紹

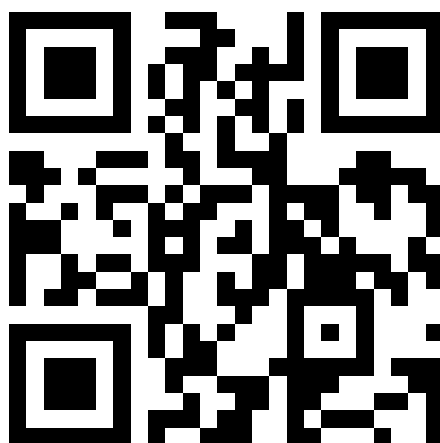
(一)劇情簡述

一個平常的上學日，你一如既往地放學回家，開了門、進了客廳，卻發現大門消失，僅剩牆壁。閣樓依舊一塵不染、地下室仍然空無一物，但浴室和臥室的物品卻都飄浮了起來。為了不喪命於此，你必須找出辦法，從自家大宅逃離出去。

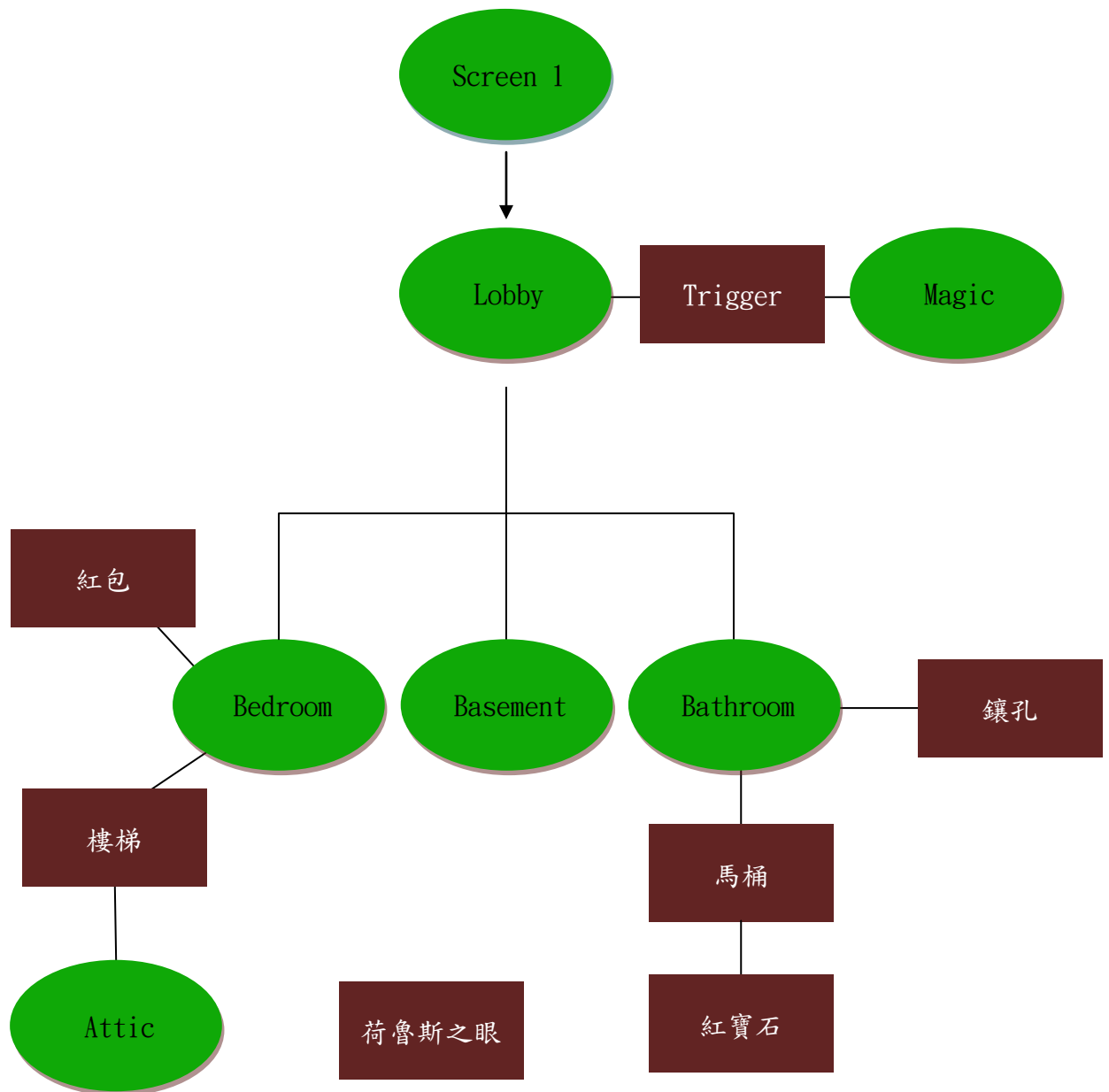
(二)遊戲下載

網址：<https://reurl.cc/96bLn>

QR CODE：



(三)遊戲流程圖



關於 AI2 的各項測試

為了方便測試各 Funtion 的實際運作情形，我特地開了一個專案來進行研究。

● 微型資料庫

The image shows three Scratch code blocks arranged vertically, illustrating a test for TinyDB1 across two screens.

- Screen 1 Initialization:** A "When Screen 1 initializes" block followed by an "Execute" block containing:
 - A "TinyDB1" block with "Store Value" set to "1" and "Store Value" set to "測試".
 - A "Set Button1 Visibility to False" block.
- Screen 1 Click Event:** A "When Button1 is Clicked" block followed by an "Execute" block containing:
 - An "Open Another Screen" block with "Screen Name" set to "Screen2".
- Screen 2 Initialization:** A "When Screen 2 initializes" block followed by an "Execute" block containing:
 - A "Set Label1 Text to" block followed by a "TinyDB1" block with "Retrieve Value" set to "1". The "Label" is "1" and the "Return Value if Label Not Found" is "沒拿到".

A text box on the right states: "於 Screen 1 及 Screen 2 放入微型資料庫，測試是否共用相同的資料庫，結果為是。"

實際上微型資料庫是以"Namespace"來做判別的，點擊畫面編排中的微型資料庫可以看見元件屬性。相同的Namespace即是相同的資料庫。

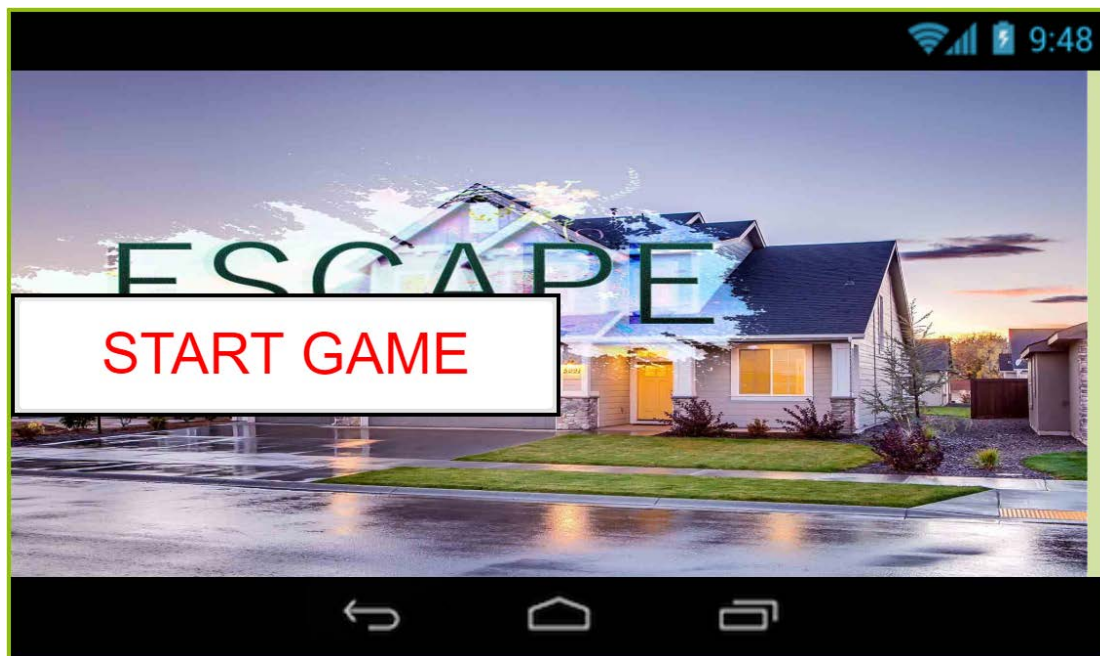


- App Inventor 的解析度是固定值，不同手機會有不同程度的跑板問題。
- App Inventor 檔案大小最多僅能到達 10MB，超過會無法進行打包。
- 對話框沒有 Namespace 的設定，但每一次顯示的警告訊息都不會相互衝突，即使從頭到尾只用一個對話框也是可行的

三、各遊戲場景與內部程式

場景名稱：Screen 1

AI2 畫面(與實機畫面有些微物件位置差距)：



場景說明：遊戲初始畫面。

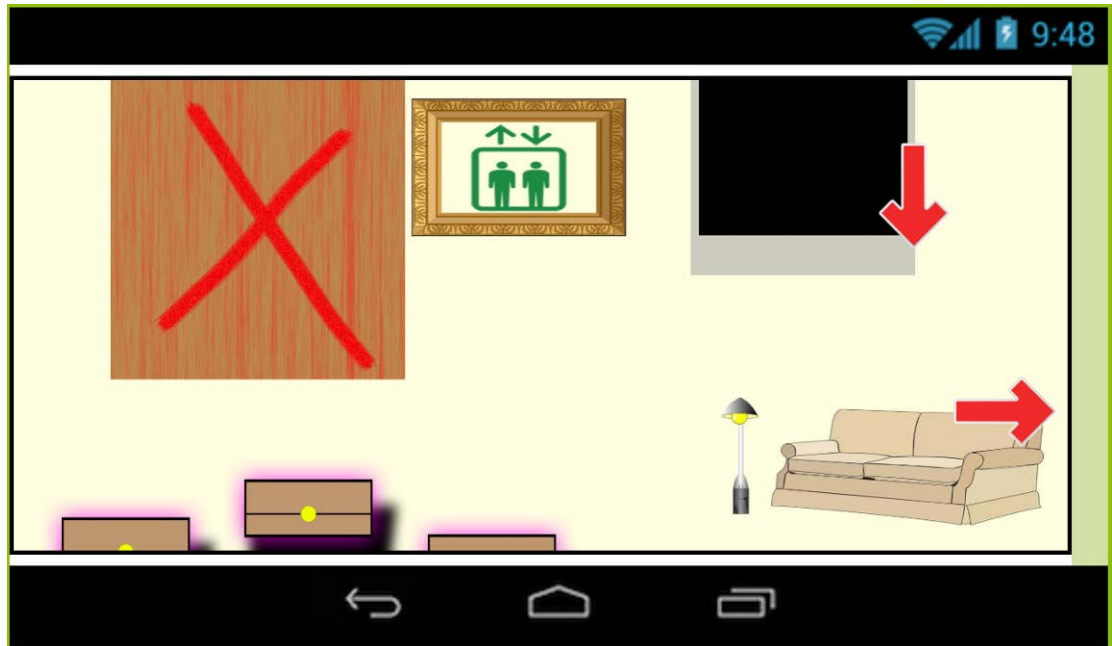
程式：



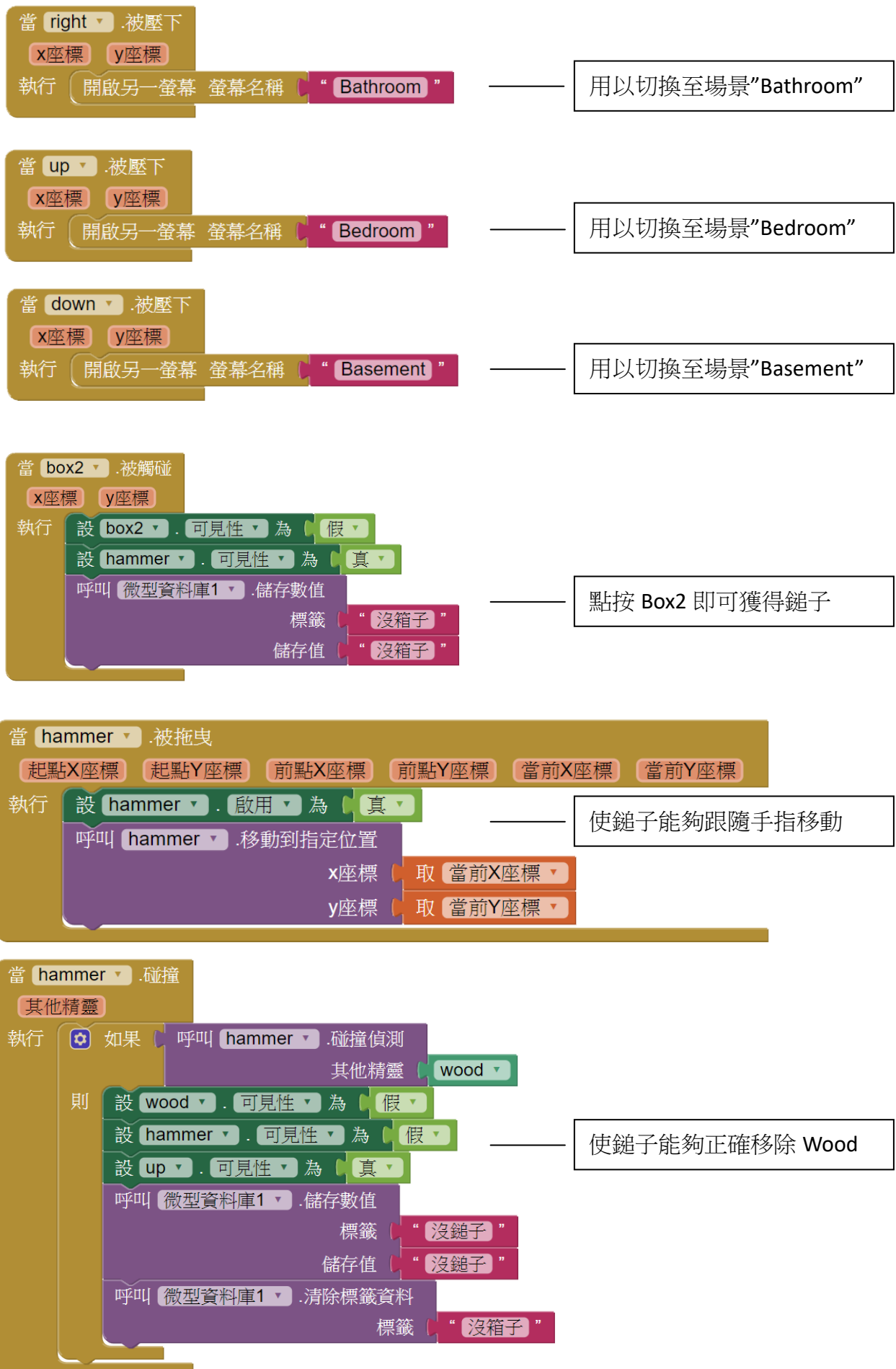
場景名稱：Lobby

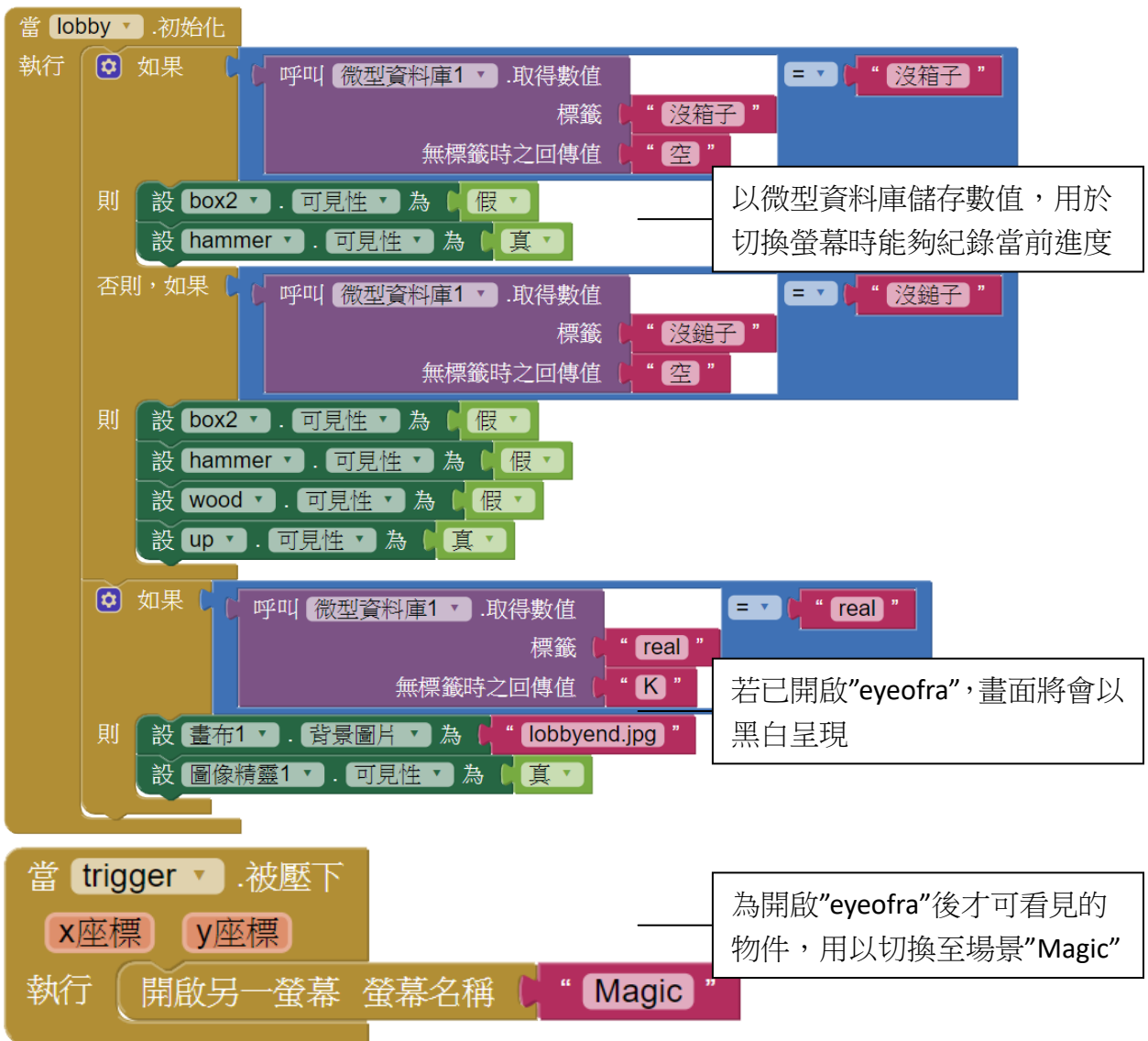
場景說明：遊戲之初始場景，上箭頭連接 Bedroom、下箭頭連接 Basement，而右箭頭則是連接 Bathroom，左上方塗有 X 記號的「Wood」需要使用最右方的箱子「Box2」內的「Hammer」消除，進而啟用通往「Bedroom」的箭頭。

AI2 畫面(與實機畫面有些微物件位置差距)：



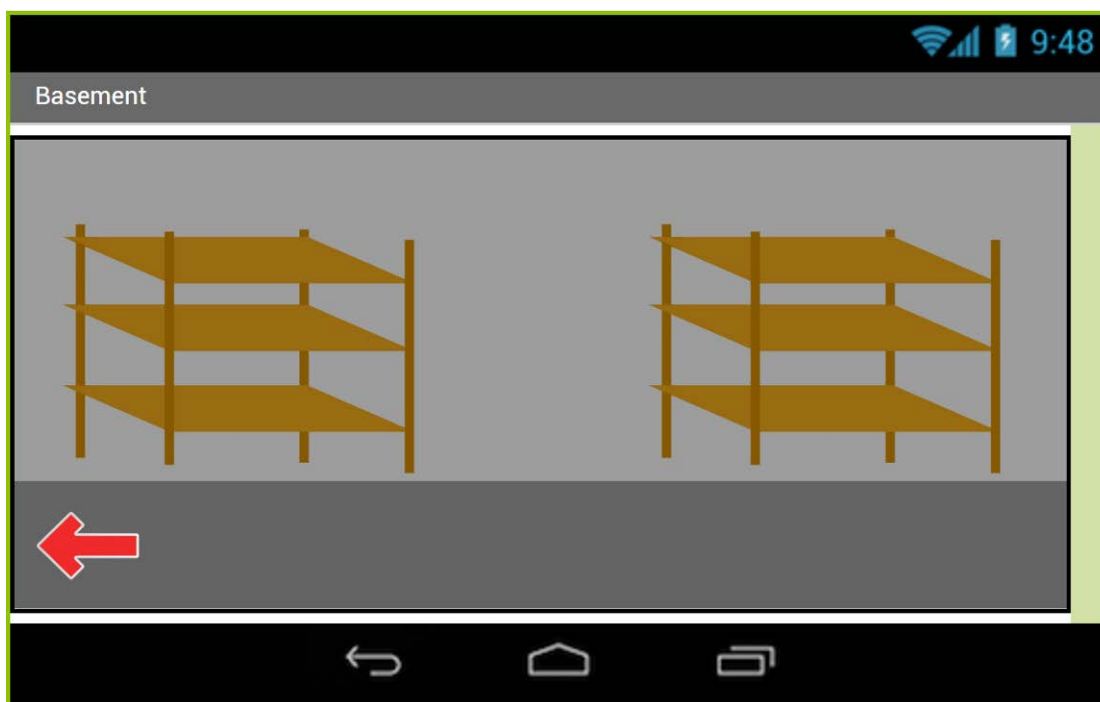
程式：





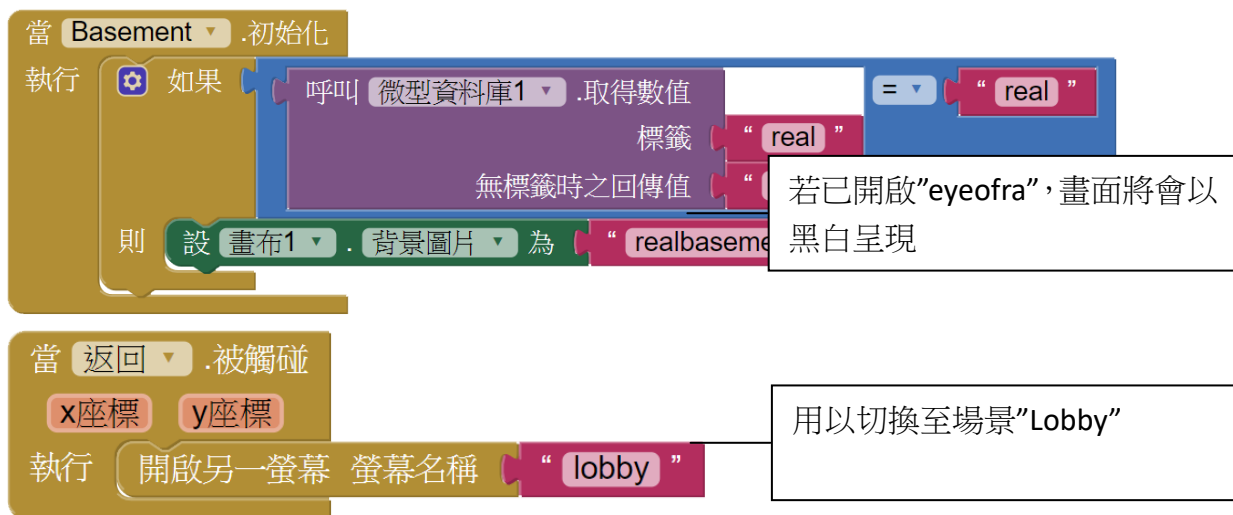
場景名稱：Basement

AI2 畫面(與實機畫面有些微物件位置差距)：



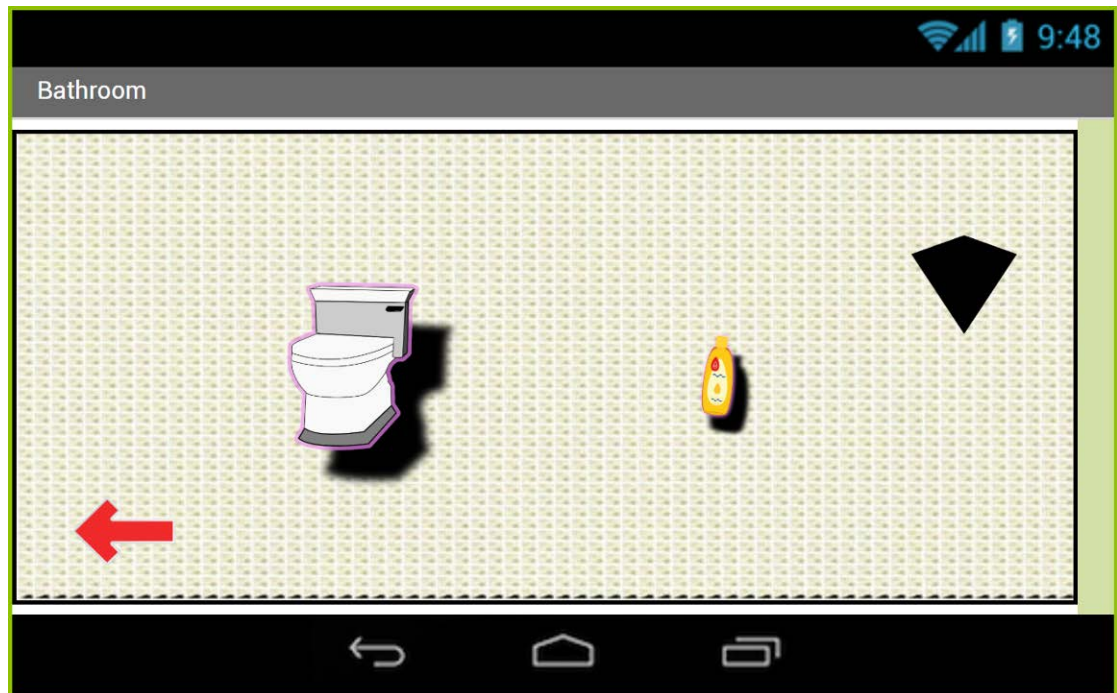
場景說明：僅用於混淆視聽，本身沒有任何其他存在意義。

程式：



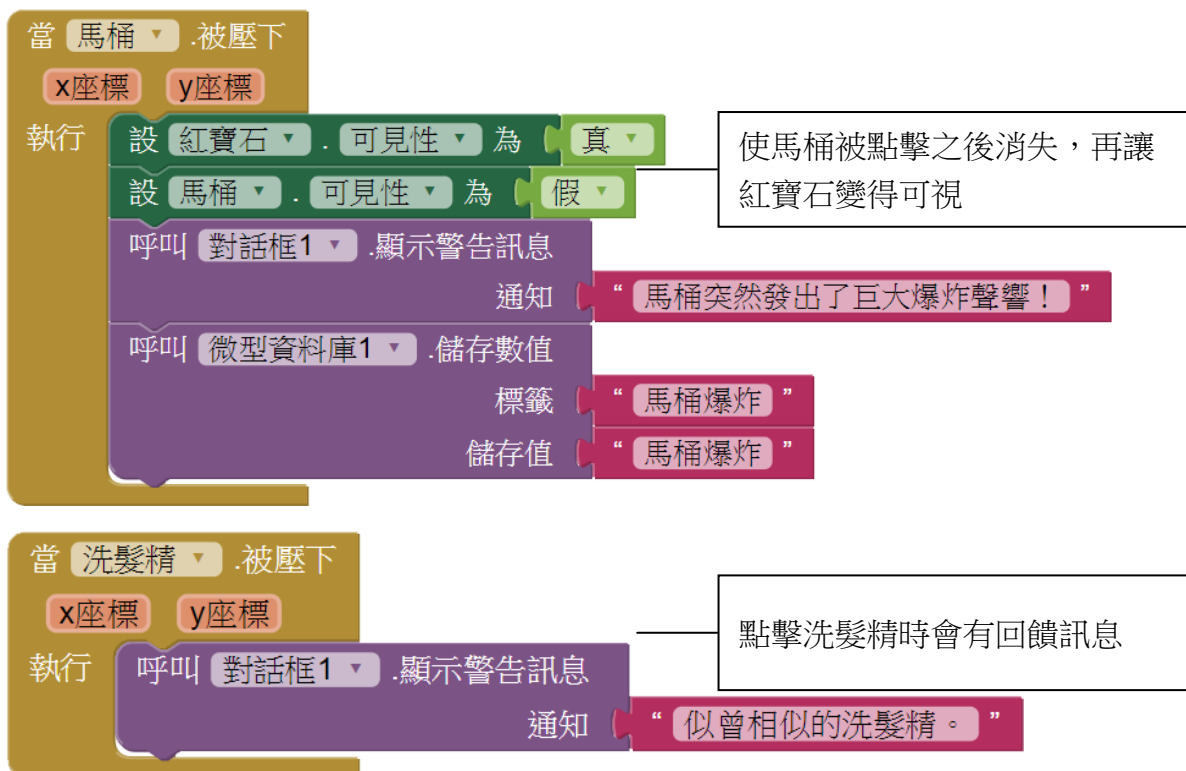
場景名稱：Bathroom

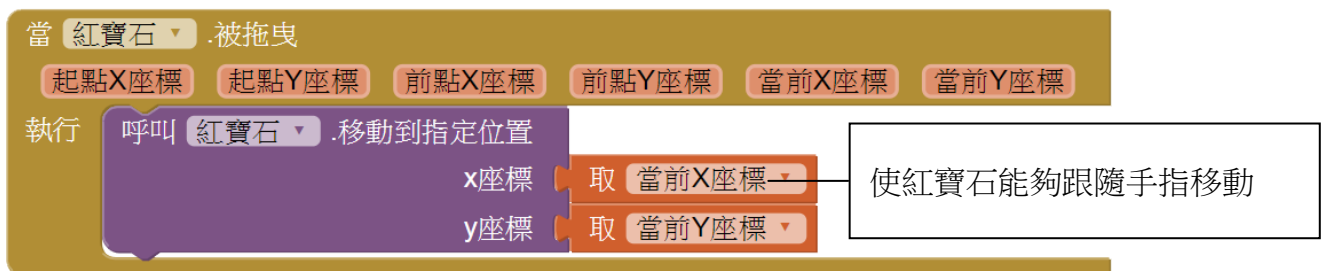
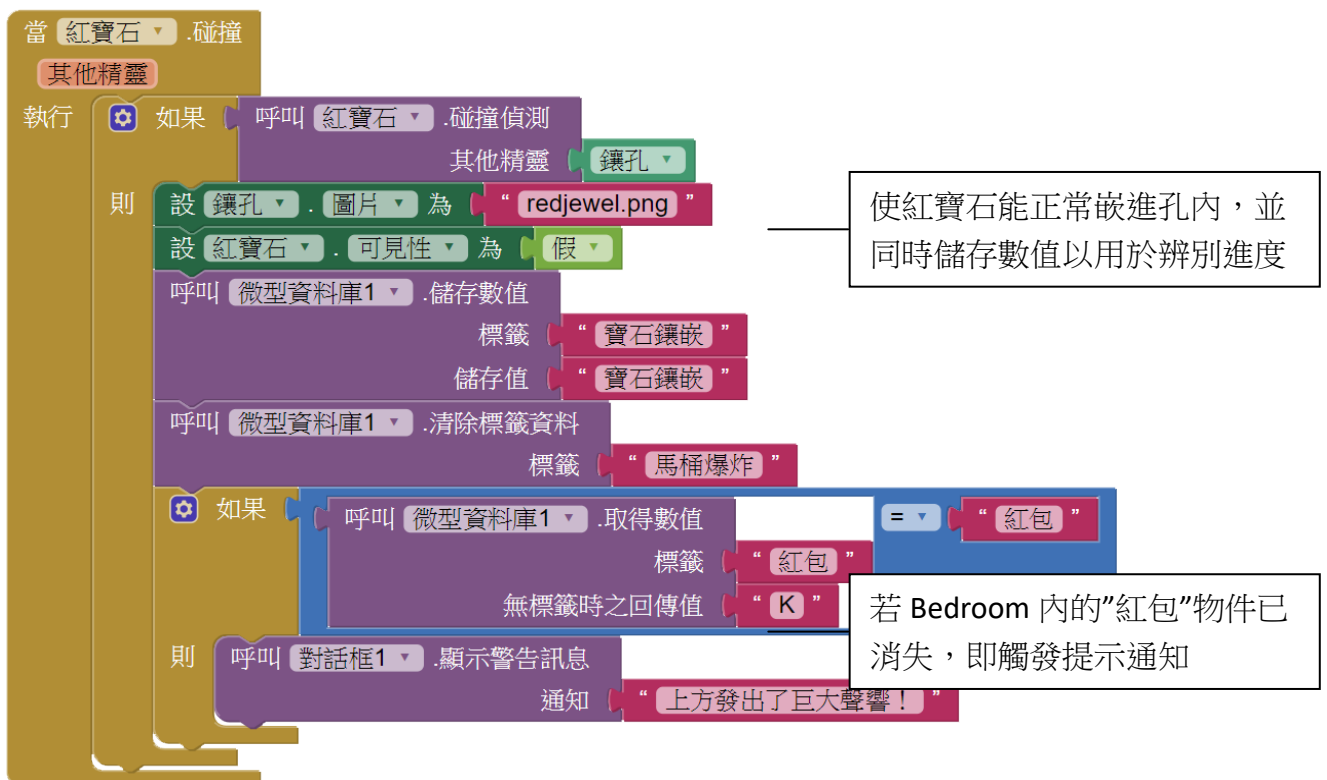
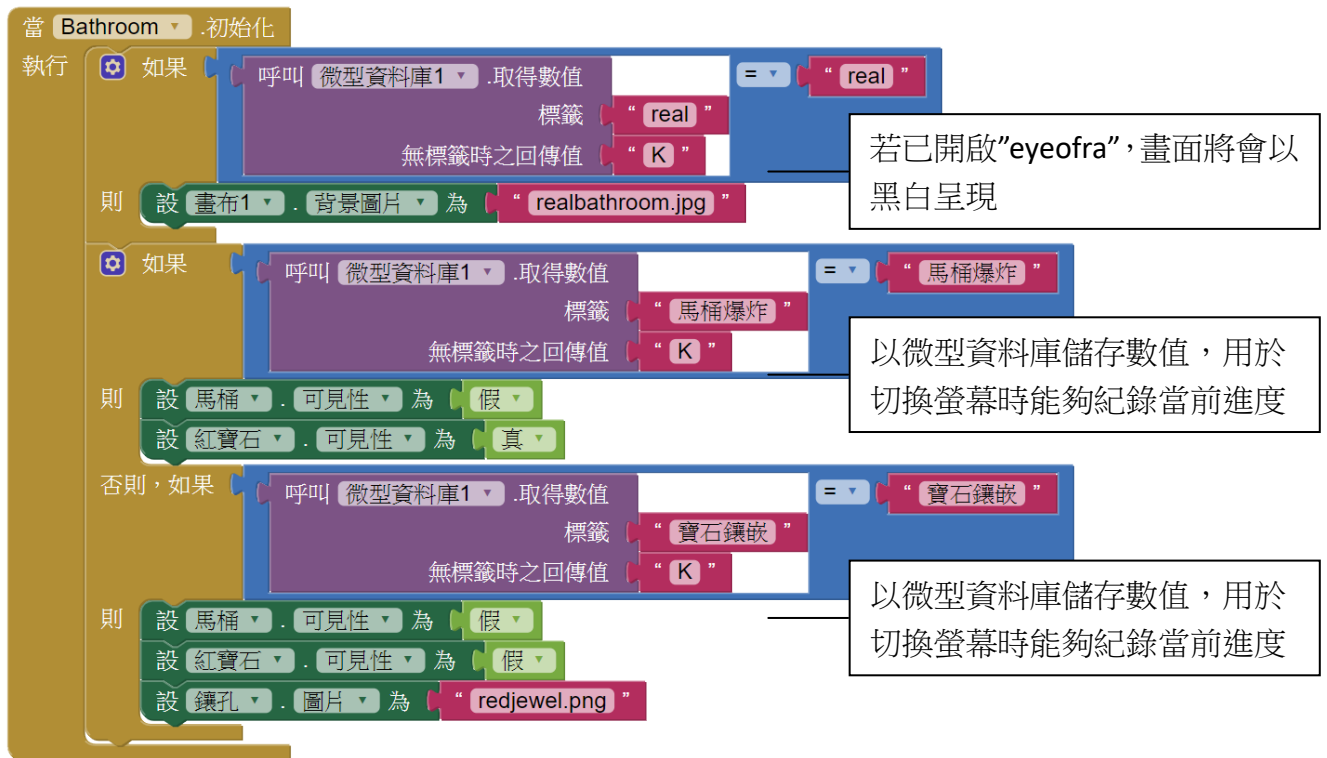
AI2 畫面(與實機畫面有些微物件位置差距)：

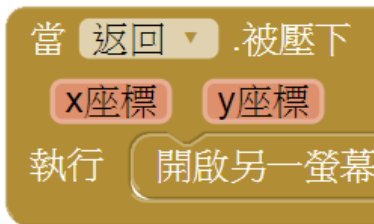


場景說明：點擊馬桶後會獲得紅寶石，可鑲嵌於右上方的鑲孔，為開啟「eyeofra」的條件之一。

程式：



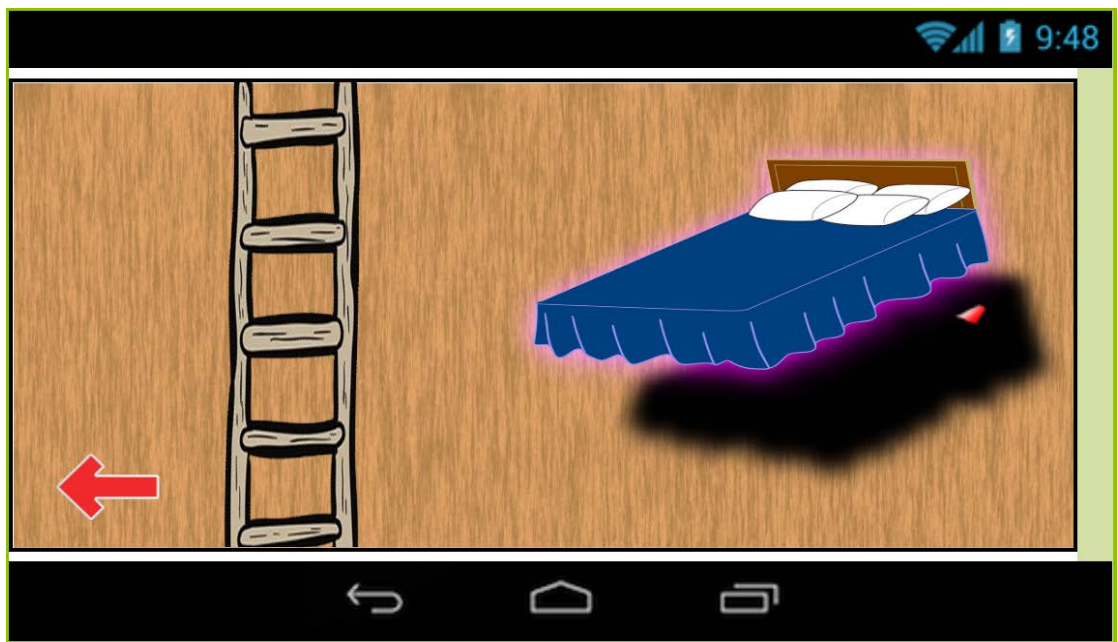




用以切換至場景“Lobby”

場景名稱：Bedroom

AI2 畫面(與實機畫面有些微物件位置差距)：



場景說明：樓梯連接場景 attic，床底下有物件紅包，點擊即會消失，若當下已完成紅寶石事件則會在 attic 啟動「eyeofra」。

程式：

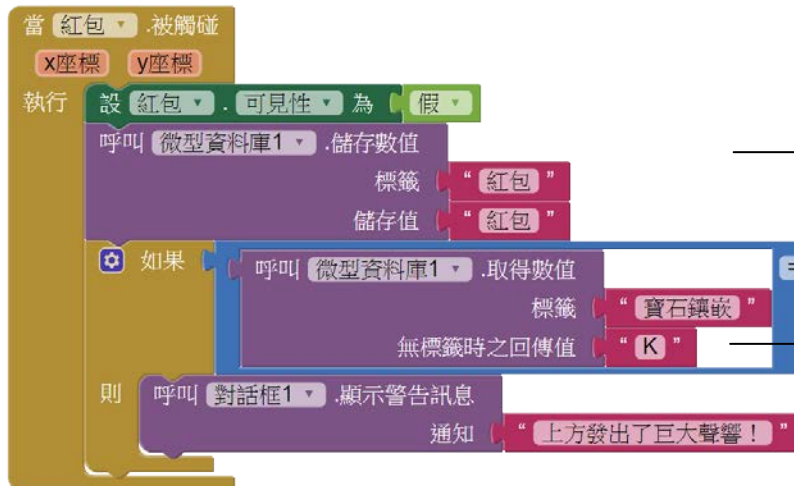


若已開啟“eyeofra”，畫面將會以黑白呈現

以微型資料庫儲存數值，用於切換螢幕時能夠紀錄當前進度



用以切換至場景"attic"



點擊使其消失，並同時儲存數值用於判斷"eyeofra"啟動與否

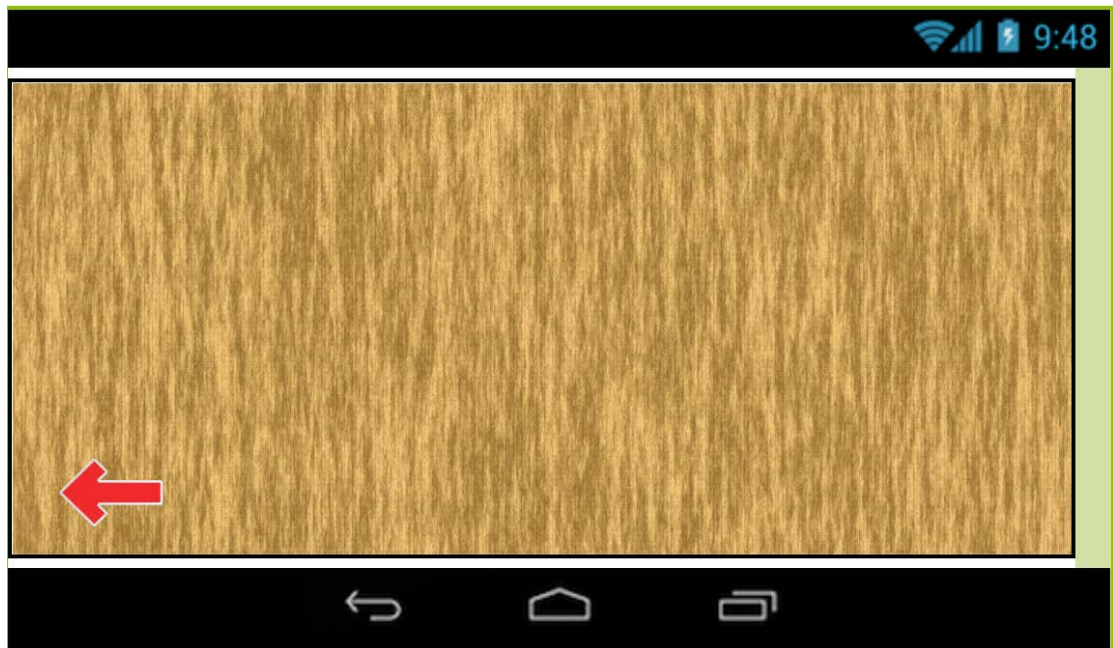
以微型資料庫儲存數值，用於切換螢幕時能夠紀錄當前進度



用以切換至場景"lobby"

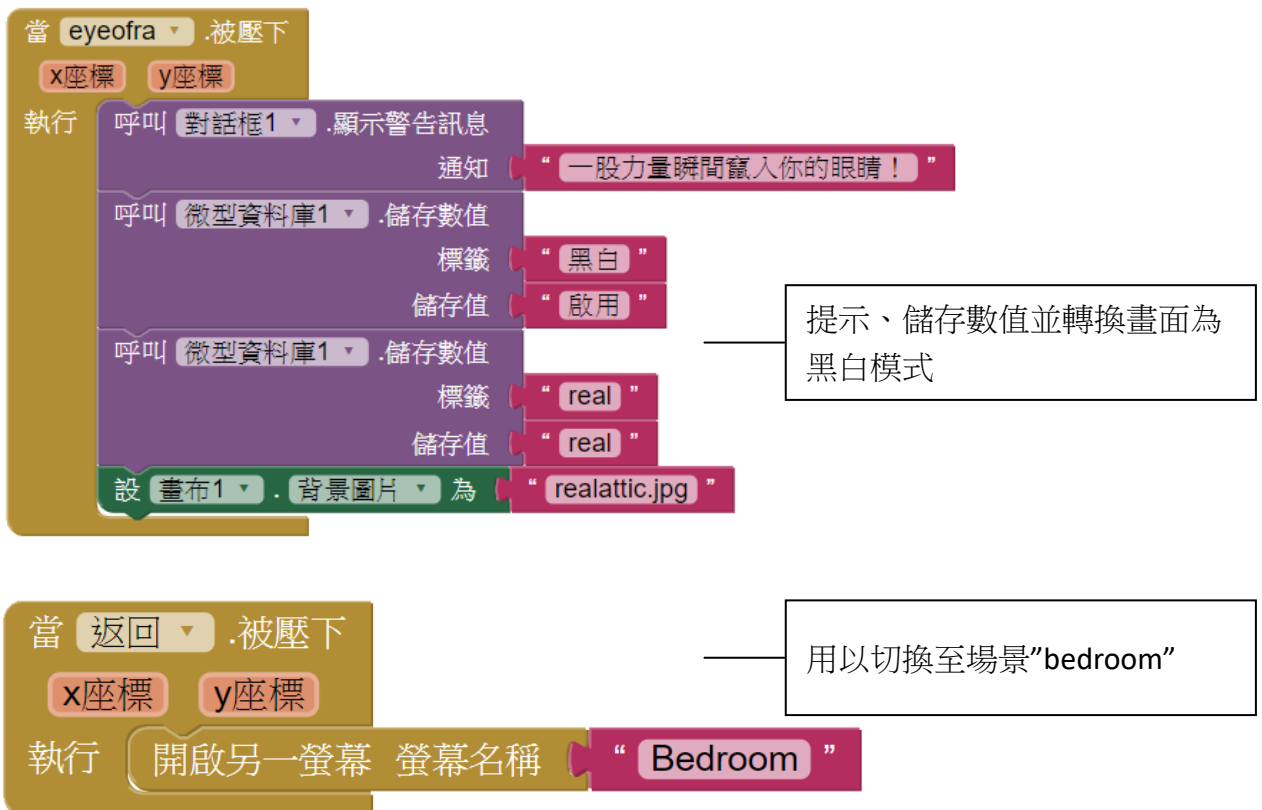
場景名稱：Attic

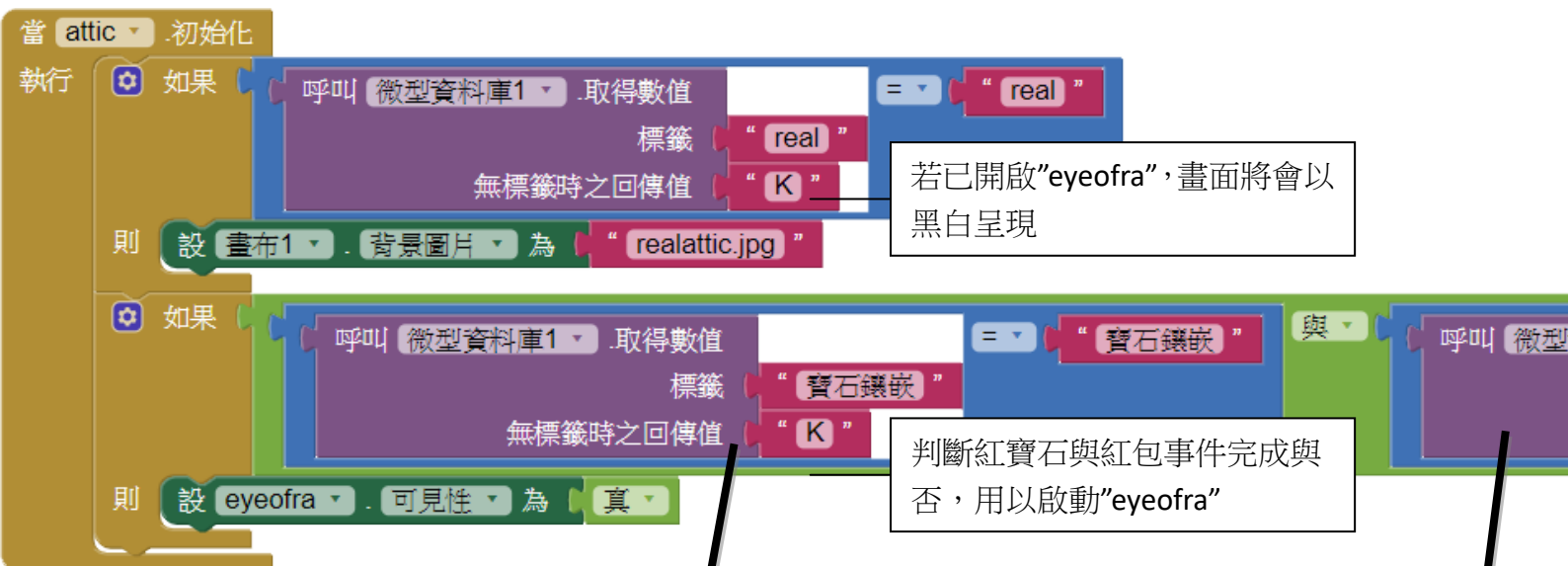
AI2 畫面(與實機畫面有些微物件位置差距)：



場景說明：僅有荷魯斯之眼「eyeofra」，會在完成紅寶石與紅包任務後出現，點擊會使所有畫面變成黑白，並能看見場景「lobby」的「trigger」。

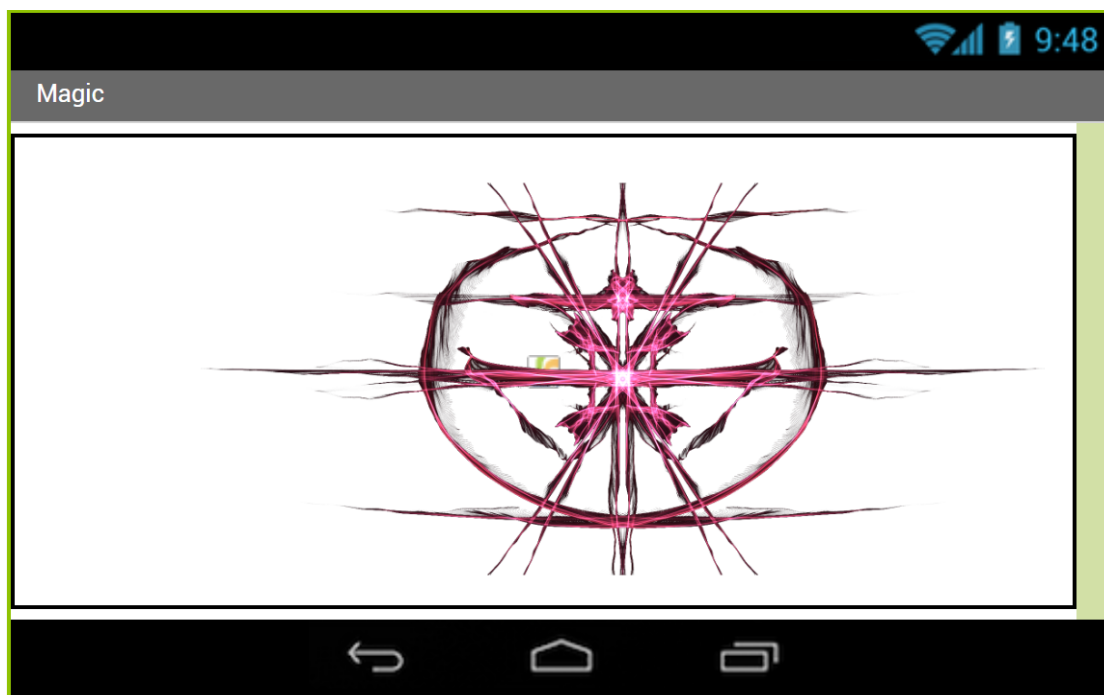
程式：





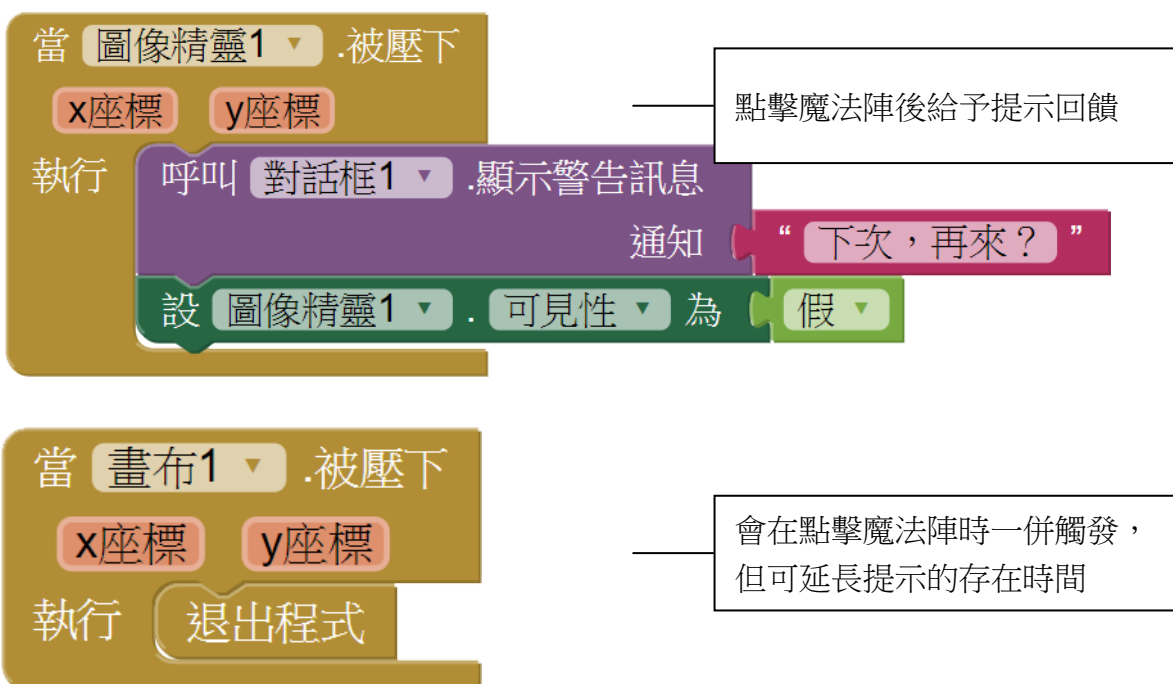
場景名稱：Magic

AI2 畫面(與實機畫面有些微物件位置差距)：



場景說明：遊戲終局，畫面中央有個巨大的魔法陣，點擊即結束遊戲。

程式：



參、心得與結論

AI2 固然較其他程式語言簡易，但在版面美化及各 Function 上確實有其繁瑣且不足之處。以整體功能看來，設計一些功能簡潔俐落的工具類型程式比起遊戲更為合適。

經過此次製作遊戲的過程，我瞭解到「美術」對於一款遊戲的重要性，尤其是密室逃脫等充滿解謎要素的遊戲類型，擁有優秀的美術才能夠將物品巧妙的藏在其他物品之中。當然，「程式」也擁有同等、甚至更高的地位，沒有程式，遊戲就無法啟動。擁有完整的程式能力才能製作完整的遊戲系統。

這款小遊戲僅是藍圖，未來我會以不同的引擎，如：Unity、RPG Maker、Unreal，又甚至是自製的遊戲引擎來進行開發，並融入更豐富的內容、多元的系統、精緻的美術來呈現出我真正想要傳達的信念，以及傳遞更多的歡樂。