

Chapter 2

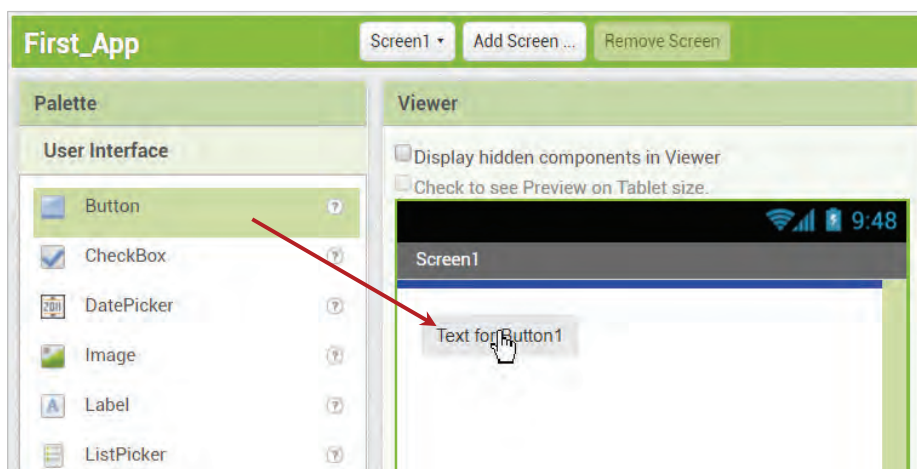
建立第一個 App

- ✧ 2-1 建立專案
- ✧ 2-2 元件配置
- ✧ 2-3 拼塊程式編輯
- ✧ 2-4 執行專案
- ✧ 2-5 專案管理
- ✧ 2-6 離線版 AI2 開發環境

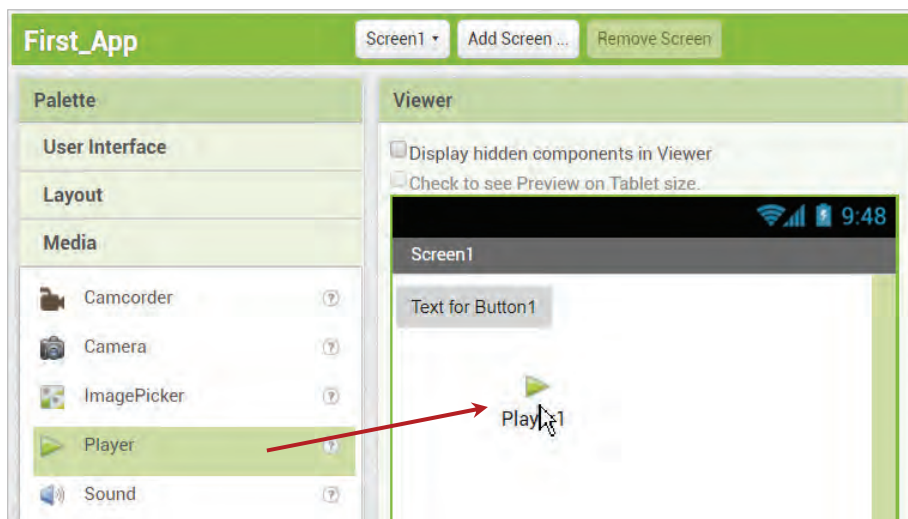
2-2 元件配置

在此專案中，App 應用程式要在版面配置(Designer)頁面中建立一個能與使用者互動的 Button 元件及互動後產生回應聲音的 Player 元件。

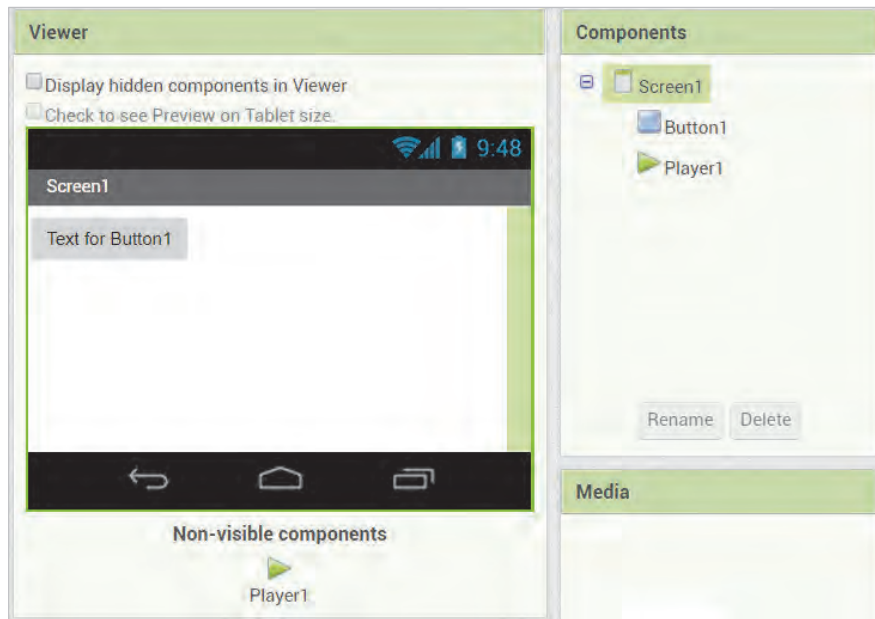
先在 Palette 元件面板區 User Interface 類別中點選 Button 元件，再以拖曳的方式將元件增加至 Viewer 工作面板的螢幕中：



以相同方式，將 Palette 元件面板區 Media 類別的 Player 元件拖曳至工作面板：



其中，Player 元件屬於不可視元件，會置放於 Viewer 工作面板區下方。此時在 Components 元件列表區出現三項物件，畫面如下：



2-2-1 Screen 螢幕元件

App Inventor 2 使用 Screen 元件產生應用程式頁面，一個 Screen 元件就是一個頁面。在建立專案時，系統會自動產生一個名稱為「Screen1」的 Screen 元件，設計者可以在此 Screen 頁面元件中加入其他元件。

在[元件列表區]中點選「Screen1」元件，可由[元件屬性區]設定 Screen 元件屬性。

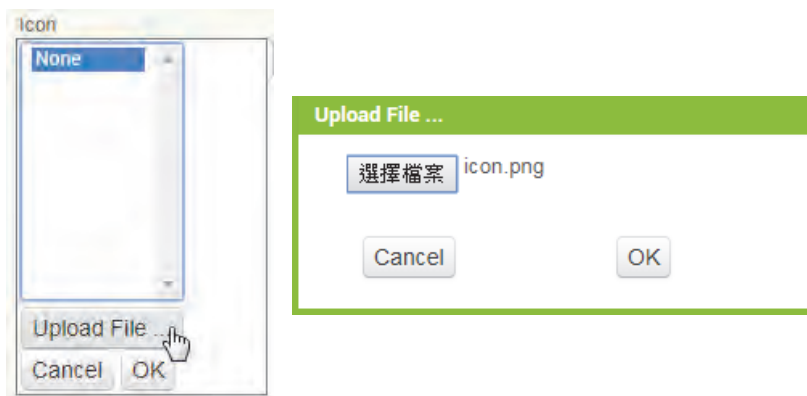
Screen 元件常用的屬性有：

屬性	說明	本例設定值
AboutScreen	設定頁面說明文字	輕輕觸摸一下，可愛的喵星人會撒嬌喔！
AppName	設定應用程式名稱	喵星人
Icon	設定應用程式圖示	icon.png
Title	設定螢幕標題，內容會顯示在螢幕的左上方	我的第一個 App



註

1. 設定 Icon 屬性時，按【Upload File】按鈕選擇 icon.png 圖示檔上傳，圖示檔格式可為 jpg 或 png 檔，大小建議以 48x48 像素為宜。



2. AppName 及 Icon 屬性設定值代表 App 應用程式在裝置桌面的名稱及圖示，必須將 App 應用程式安裝在行動裝置後，才會顯示在裝置的桌面上。

2-2-2 Button 按鈕元件

Button 元件是應用程式與使用者互動的主要元件。最常使用的方式是當使用者點擊 Button 元件時，會執行設計者安排的程序片段，達到互動的目的。

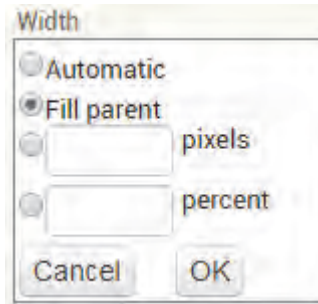
Button 元件常見的屬性有：

屬性	說明	本例設定值
Enable	設定按鈕是否可用	☒ (true)
Height	設定按鈕的高度	Full Parent
Width	設定按鈕的寬度	Full Parent
Image	設定顯示圖片按鈕	cat1.jpg
Text	設定按鈕上的文字	
TextAlignment	設定按鈕文字對齊方式	Center
Visible	設定按鈕顯示狀態	☒



註

元件寬度(Width)屬性設定項目有四種：**Automatic** 會依元件內容自動調整寬度；**Fill Parent** 則會將元件寬度依上層元件填滿，另外也可以設定像素(pixels)值或百分比(percent)值決定元件的寬度。元件高度的屬性設定方式相同。



2-2-3 Player 音頻播放器元件

Player 元件位於 [Media] 類別，屬於不可視元件，主要是用來播放較長的聲音檔，例如遊戲中常用的背景音樂。Player 的功能也可以讓手機產生震動，並且可以依據 Vibrate 屬性設定震動時間。

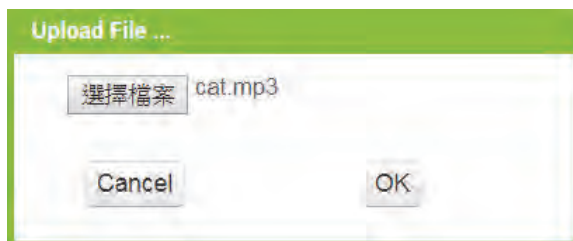
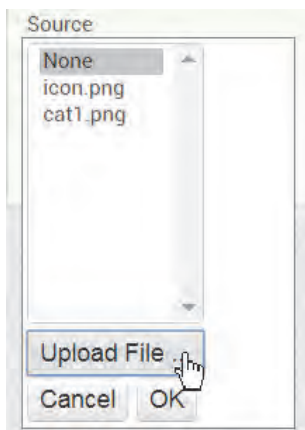
Player 元件常見的屬性有：

屬性	說明	本例設定值
Loop	設定聲音是否循環播放	☐ (false)
PlayOnlyInForeground	目前是否正在播放中	☐ (fasle)
Source	設定播放的聲音檔	cat.mp3
Volumn	設定播放音量大小(0~100)	50



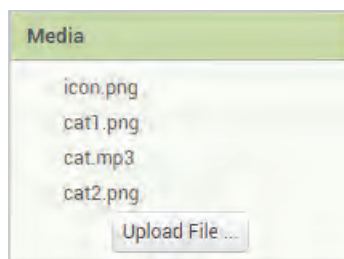
註

Player 元件的聲音檔案格式可以為 wav、mp3、midi...等，本例由 Source 屬性中按【Upload File】按鈕，選擇上傳 cat.mp3。



2-2-4 上傳資源檔

App 程式常要用到圖片、聲音、影片等額外的素材，讓應用程式更加生動。APP Inventor 2 的編譯工作是在伺服器進行，所以所有資源檔案必須上傳至遠端，方能被應用程式使用。所有透過【Upload File】上傳的資源檔都會列表在[上傳素材區]內，在此專案中尚須上傳 cat2.png 檔供 App 應用程式讀取。



2-3 拼塊程式編輯

完成版面配置頁面元件初始屬性設定後，按 App Inventor 2 頁面右上方【Blocks】按鈕，切換到程式設計頁面，進行拼塊程式設計。

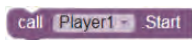
撰寫程式的方式，好比是指揮某一元件在某種事件觸發(如點擊、取得焦點...)的情況下，執行某些程序。

在本專案中的第一個事件是在 Button1 按鈕被按下後，即觸發一事件命令 Player1 元件執行 Start 播放程序，同時將 Button1 的 Image 屬性值以 cat2.png 檔案取代。

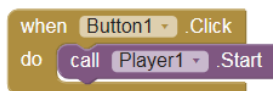
1. 點選 Button1 元件，再按 Button1.Click 事件拼塊。



2. 使用者按下按鈕後，播放 Player1 指定的音樂，點選 Player1 元件，再按 call Player1.Start 拼塊。

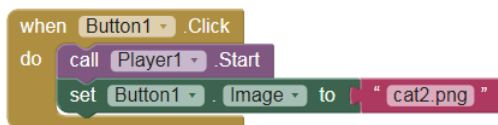


3. 拖曳 call Player1.Start 拼塊至 Button1.Click 事件拼塊內，使得拼塊的缺口接合。



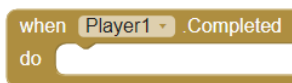
4. 點選 Button1 元件，再按 set Button1.Image to 拼塊，拖曳至 call Player1.Start 拼塊下方，使得拼塊的缺口接合。

5. 在 Built-in 類別，點選 Text 內建方塊，再按 " " 拼塊，拖曳至 set Button1.Image to 拼塊右方，使用拼塊缺口接合。然後在 " " 拼塊中輸入 cat2.png。



第二個事件是在 Player1 指定音樂播放完畢後，觸發一事件命令 Button1 的 Image 自動恢復顯示 cat1.png 圖像。

1. 點選 Player1 元件，再按 Player1.Completed 事件拼塊。



2. 點選 Button1 元件，再按 set Button1.Image to 拼塊，拖曳至 Player1.Completed 事件拼塊內，使得拼塊的缺口接合。

3. 在 Built-in 類別，點選 Text 內建方塊，再按  拼塊，拖曳至  拼塊右方，使用拼塊缺口接合。然後在  拼塊中輸入 cat1.png。

```
when Player1 Completed
do set Button1 Image to "cat1.png"
```



註

可由 Button1.Click 事件拼塊中，點選  拼塊串，按滑鼠右鍵，點選 Duplicate 項目複製拼塊串後，再進行修改。

```
when Button1 Click
do call Player1 Start
  set Button1 Image to "cat2.png"
```

Right-click context menu options:

- Duplicate
- Add Comment
- Collapse Block
- Disable Block
- Add to Backpack (0)
- Delete 2 Blocks
- Help
- Do It

完成後，專案拼塊如下：

```
when Button1 Click
do call Player1 Start
  set Button1 Image to "cat2.png"

when Player1 Completed
do set Button1 Image to "cat1.png"
```

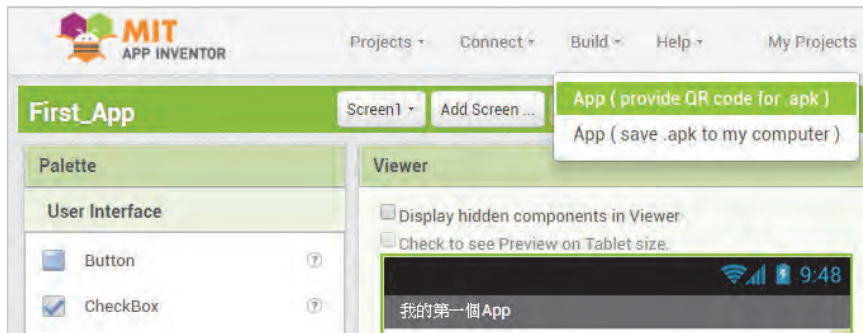
2-4 執行專案

App Inventor 2 是為了開發 Android 系統的應用程式，如果沒有實體裝置裝載，App Inventor 2 也提供線上模擬器提供設計者執行應用程式，甚至網路也提供各種免費的模擬程式，方便設計安裝在個人電腦上執行應用程式。

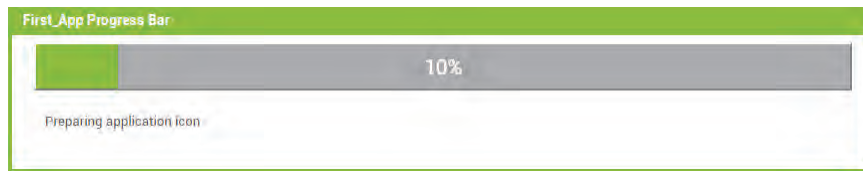
2-4-1 實機安裝安裝檔

執行 App 應用程式最簡單的方法，莫過於直接下載安裝檔(*.apk)檔在 Android 實體裝置安裝，就如同在 Google Player 上下載應用程式一般。

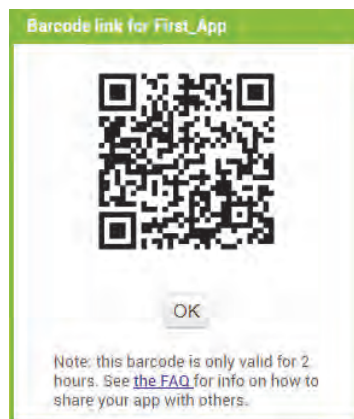
1. 執行[Build/App(provide QR code for .apk)]。



2. 經過一段時間等待應用程式編譯。



3. App Inventor 2 產生一個二維條碼位址，使用 Android 實體裝置中的 Barcode 掃描程式掃描連結下載安裝檔到實體裝置，進而在實體裝置上安裝應用程式。注意：此二維條碼有效時間僅為 2 小時。



如果要在實體裝置上安裝應用程式，基於安全理由，系統預設不允許安裝非 Google Play 下載的應用程式。因此，要能在實體裝置安裝自行開發的 App 應用程式，必先改

變 Android 實體裝置安全性設定，開啟實體裝置的[設定]功能中的[安全與隱私]，核選「不明的來源」項目，允許實體裝置安裝來源不明的應用程式。



安裝 App 應用程式後，裝置桌面出現 App 應用程式的名稱及圖示，開啟 App 應用程式，用手輕輕觸摸桌面，裝置會發出貓咪的叫聲，同時改變圖片內容；在聲音播放結束後，圖片又回復到原始圖案。



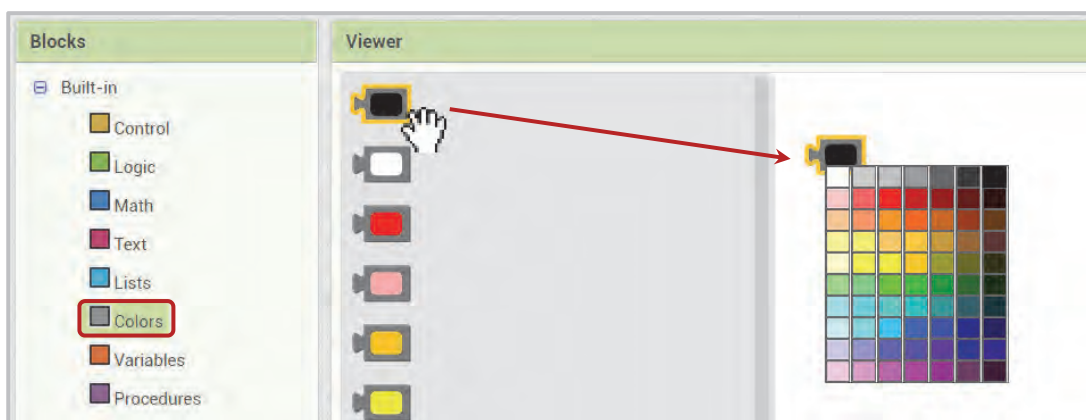
Chapter 6

清單與顏色拼塊

- ✧ 6-1 清單概念
- ✧ 6-2 建立清單
- ✧ 6-3 管理清單
- ✧ 6-4 顏色拼塊
- ✧ 6-5 元件清單

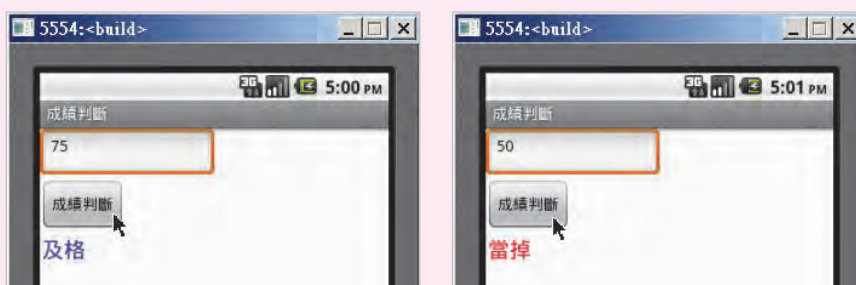
6-4 顏色拼塊

在程式設計過程中，會以不同的顏色來強調輸出或美化裝置的畫面，改變顏色的對象可能是文字的顏色，也可能是畫面或元件的背景顏色。App Inventor 2 在 Blocks 程式設計頁面 Built-in 項目的 Colors 類別，可任意拖曳顏色拼塊作為其他元件的顏色(或背景顏色)屬性，點一下拼塊中央的顏色後，可由色盤中點選其他顏色，變更拼塊設定的顏色。



範例 6-4-01

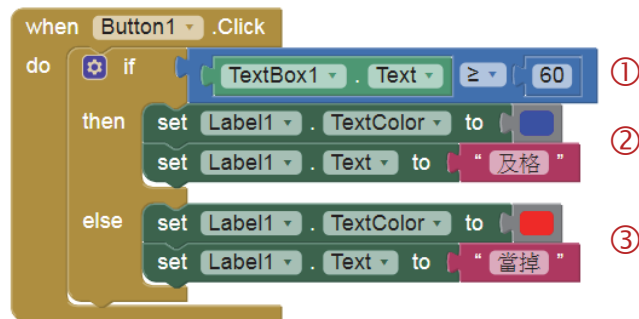
設計成績判斷的 App 程式，讓使用者輸入成績後按下「成績判斷」按鈕，依照輸入成績判別，若大於等於 60，則顯示藍色文字「及格」，否則顯示紅色文字「當掉」。



Designer 界面佈置：



Blocks 程式拼塊：



1：讀取 TextBox1 元件內容進行條件式比較。

2：條件式成立，設定 Label1 元件內文顏色屬性及內容。

3：條件式不成立，設定 Label1 元件內文顏色屬性及內容。

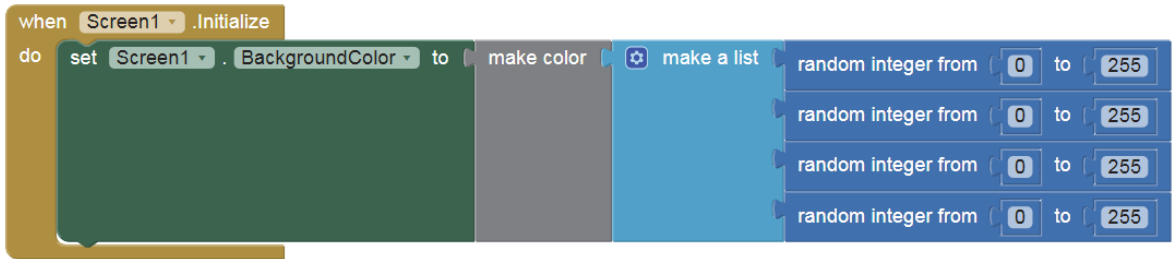
另外一種設定顏色的方式，是使用 Colors 類別中的 [make color] 拼塊，[make color] 拼塊預設接收一個具有 3 個數字項目元素的清單來設定顏色，清單中的數字就是 RGB 色碼，依序代表紅、綠、藍色的強度，每個數值介於 0~255 之間。



按 [make color] 拼塊後方的清單左上方的擴充項目鈕 ，可拖曳增加一個 [item] 項目拼塊作為清單第 4 個項目元素，代表顏色的不透明度(alpha)，alpha 值介於 0~255 之間。如果 alpha 值為 0，就會呈現完全透明的色彩。



例如想在每次執行 App 程式時，螢幕能以隨機方式出現不同背景色彩來轉換心情，程式的拼塊為：

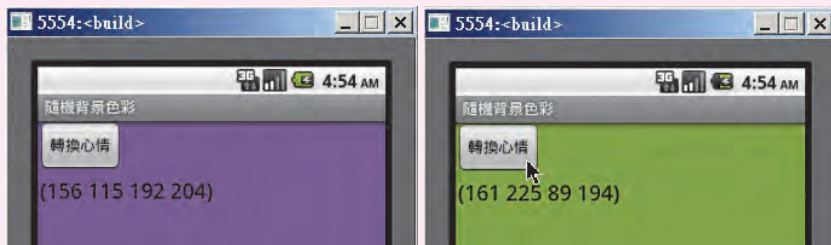


相反的，Colors 類別中的 [split color] 拼塊的功能則是將特定色彩的紅、綠、藍及 alpha 等 4 個色碼值，轉換為 4 個項目元素的清單，每個色碼值介於 0~255 之間。

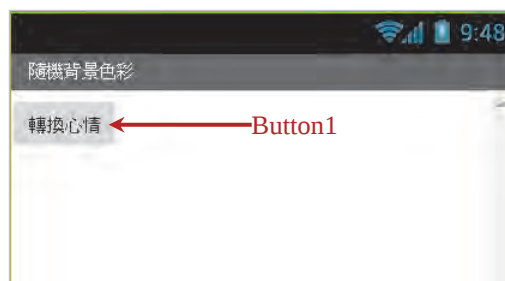


範例 6-4-02

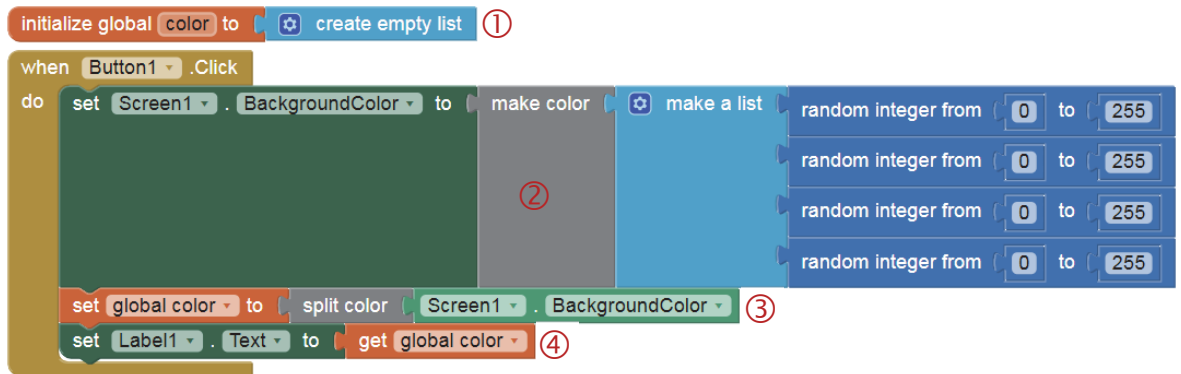
設計隨機背景色彩的 App 程式，讓使用者按下「轉換心情」按鈕，螢幕的背景會隨機變化，並在下方顯示目前背景色彩的 RGB 及 alpha 數值。



Designer 界面佈置：



Blocks 程式拼塊：



1：宣告 color 清單變數。

2：使用隨機函數取得 RGB 及 alpha 值，設定為 Screen1 背景顏色。

3：分解 Screen1 目前背景色彩碼值，並設為 color 清單項目元素。

4：顯示清單內容。