

# Android Tutorial using Kotlin 第一堂

## （4）開發Android應用程式的準備工作

[Android Tutorial using Kotlin 第一堂（3）開始設計Android應用程式 << 前情](#)

對一般應用程式的開發人員來說，開發Android應用程式的規模好像會比較小一些，在瞭解基本的設計方式以後，通常是採用想的方式，過程中遇到問題了，查查文件或搜尋一下，解決問題以後，經過幾次這樣的循環，一個Android應用程式就誕生了

一般Android應用程式的架構，通常不會比企業應用程式專案複雜，如果你已經具備其它應用程式的開發經驗，應該可以把它開發Android應用程式，不過前題是你千萬不要「小」看它，雖然它真的是比較小一些。如果你還沒有任何開發應用程式的經成一個比較好的開發方式與習慣，應該可以節省很多開發與測試的時間。

第一堂的最後一個部份，介紹開發一般Android應用程式的準備工作，在瞭解這些內容以後，就可以在第二堂正式進入開發應專案的階段。

### 4-1 Android應用程式專案資訊

Android提供一個方便與容易的開發環境，可以讓你盡情發揮所有的創意，不論是想要開發一個自己使用的應用程式，或是想到Google Play分享給全世界廣大的使用者，對開發人員來說，Android可以帶來完全不一樣的體驗與樂趣。

在前面介紹的內容中，Android已經是一個有一點歷史的平台，在版本進化的過程，不斷的加入各種最新的功能與支援，跟其程式開發技術一樣，Android也會有版本的問題，例如在Android 4.0（API Level 14）加入一種比較好看的開關畫面元件Switch。如果你在應用程式使用它，這個應用程式就不可以在Android 4.0之前的裝置中執行。

如果你有開發一個Android應用程式的想法，或是接到一個開發應用程式的任務，第一件事應該要先決定好一些關於這個專案資料。

#### 4-1-1 決定應用程式可以運作的版本

如果你想要為自己開發一個Android應用程式，它只會安裝在你的手機或平板電腦中執行，那就容易多了。先進入裝置的設定，在「關於手機」裡面查詢Android的版本，就可以決定應用程式需要設定的版本。例如你的裝置是目前最多的Android 4.4（API Level 19），為開發環境下載與安裝這個版本的SDK Platform，建立專案的時候也把應用程式的版本設定為API Level 19，設計應用程式就可以正確的在你的裝置中執行。

如果你想要把寫好的應用程式上傳到Google Play，讓所有人都可以下載與使用，在這樣的情況下，就應該要到這個網址<http://developer.android.com/about/dashboards/>，看看目前Android裝置的版本統計情況，把應用程式設定為涵蓋最多使用本。例如目前Android 2.X還有7.8%，Android 4.x有90.6%，如果你不想放棄7.8%的使用者，就應該把應用程式的最低版本設定Level 8或10，你的開發環境也要下載與安裝這些版本的SDK Platform。

瞭解這些內容以後，把應用程式的版本資訊寫下來，建立應用程式專案的時候就會用到了。

#### 4-1-2 決定應用程式需要的設備

Android裝置最大的特色就是「太多了」，真的是太多了，這也是你要面對的挑戰與選擇。因為Android是一個開放的作業系統，以廠商可以製造各種硬體規格的行動裝置，光是螢幕的尺寸與解析度就有非常大的差異。如果你想要寫一般的應用程式，例如清單，就不用太在意硬體的規格。不過如果想要設計一個遊戲，就應該要留意螢幕的尺寸與解析度與Open GL的版本。這些資訊也可以在<http://developer.android.com/about/dashboards/>查詢。

根據應用程式的類型，你應該決定它需要哪一些硬體設備才可以正確的運作，例如應用程式需要儲存衛星定位提供的座標資訊，應用程式就一定要在具備GPS設備的裝置才可以運作。如果應用程式需要使用相機拍照與儲存照片，或是讀取裝置的方位資訊，程式就須要加入這些硬體設備需求的設定。

想好應用程式畫面需要的螢幕尺寸與解析度，還有需要哪一些硬體設備，例如相機、GPS與網路，在建立應用程式專案以後，要執行相關的設定。

### 4-1-3 為什麼要做這些工作？

如果是為自己或特定的使用者開發應用程式，行動裝置的版本與規格就很清楚，通常不會有什麼問題。如果想開發上傳到Google Play的應用程式，它面對的就是廣大的使用者，以Android版本的設定來說，一個最低版本設定為Android 4.0（API Level 14）的應用程式，裝置的系統還是Android 2.2的使用者，在Google Play中就無法看到你的應用程式。

如果應用程式需要最低的螢幕尺寸與解析度，才可以正確顯示應用程式的畫面，例如遊戲類的應用程式比較有這類的限制。如果設計好的應用程式，沒有針對螢幕尺寸與解析度執行設定的話，沒有符合最低標準的裝置也可以下載與安裝，可是執行應用程式後，畫面的顯示就會發生問題，例如有部份的畫面無法顯示。如果使用者遇到這樣的情況，那這個應用程式的評分就會很差。

這些重要的設定，在開發環境建立應用程式的時候，可以設定Android平台的版本。在Android應用程式設定檔「AndroidManifest.xml」，可以設定應用程式需要的硬體設備資訊。

## 4-2 規劃你的應用程式

在你的手機或平板電腦裡面，可能安裝了一大堆應用程式，有社群、遊戲、新聞與其它各種生活應用。一般的應用程式通常會有一個畫面，提供資訊的顯示和使用者的操作。根據每一個畫面的用途，會使用一些畫面元件，複雜一點的畫面可能會有許多顯示元件、輸入和選擇元件，在開發應用程式之前，如果可以先想好需要的畫面、元件和操作功能，設計的過程就會比較順利一些。

### 4-2-1 應用程式需要的Activity元件

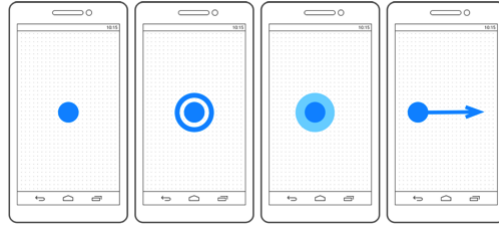
Android提供一個方便與容易的設計架構，一般應用程式需要一個畫面，就需要撰寫一個「Activity」元件，它也是Android應用程式最基本與常用的元件。使用者啟動一個應用程式以後，第一個出現的畫面就是一個Activity元件，通常會稱為應用程式的主元件，隨著使用者的操作，應用程式會啟動其它Activity元件，在畫面上顯示不同的畫面讓使用者查詢資料與執行操作。

一個複雜一點的應用程式，可能包含十幾個Activity元件，在開發應用程式之前，就算只是在紙上用筆畫一些簡單的圖型，也建議規劃一下應用程式需要哪些Activity元件。這時候通常可以依照Activity元件的功能決定它的名稱，例如新增資料的元件可以叫「AddDataActivity」。根據應用程式需要的功能，可以簡單的畫一個類似這樣的圖型：

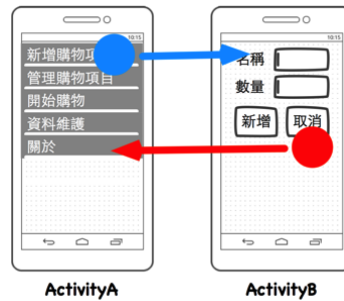


## 4-2-2 設計應用程式與使用者的互動

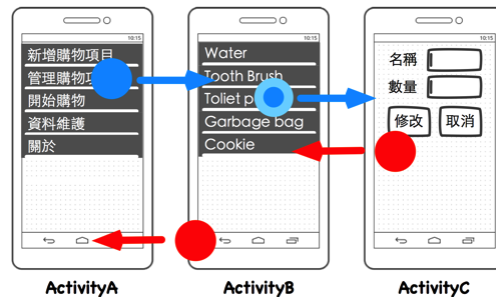
目前的行動裝置大部份都具備觸控螢幕的設備，應用程式在提供使用者互動功能的時候，應該以觸控螢幕的操作為主。應用程式該要提供使用者簡單與流暢的操作方式，而且要儘可能讓使用者使用最少的操作次數，就可以完成應用程式提供的功能。在Android開發人員網站，提供一些基本觸控螢幕的操作說明，由左到右依照順序是「觸控」、「快速觸控兩次」、「長按」和「觸控後拖動」：



在完成應用程式的畫面設計以後，接下來就可以開始規劃應用程式與使用者的互動。一個比較簡單的操作可以使用這樣的圖型表示：



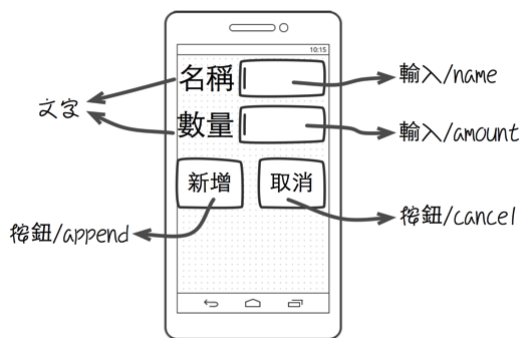
想想應用程式所有Activity元件提供的操作功能，儘可能在圖型中呈現出來，你也可以檢視這些操作的流程是不是簡單又流暢。這個圖型表示三個Activity元件之間的關係與互動：



## 4-2-3 設計應用程式的畫面

Android平台提供非常多各種應用的畫面元件，可以很容易讓你組合成應用程式需要的畫面，這些畫面元件主要在Android API「android.widget」套件。如果你還沒有接觸過Android應用程式開發技術，就還不知道這個套件裡面有哪些東西，不過你還可以先規劃好應用程式需要的畫面，在決定畫面元件的時候，可以先用簡單的文字來表示。如果畫面元件需要在程式碼中使用，比如取文字元件輸入的內容，或是設定列表元件顯示的資料，就可以先為它們取好元件的名稱。

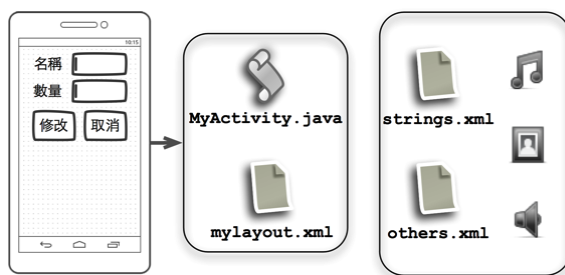
這個圖型表示一個應用程式的畫面，裡面有一些顯示文字、輸入文字與按鈕元件，輸入文字與按鈕元件都依照它們的功能取名稱：



### 4-3 設計應用程式需要的資源

Android應用程式的設計架構，在畫面和資源的部份，有一種比較特別的設計方式。在一般的情況下，應用程式的一個畫面，由一個Activity類別和一個畫面配置檔組合而成的。Activity類別是一個繼承自「android.app.Activity」的子類別，這個類別負責讀取資料、提供互動功能這類工作。搭配使用的畫面配置檔是一個XML格式的文件，負責提供Activity元件使用的畫面。這種和畫面設計分開來的作法，雖然好像比較麻煩一些，不過可以提供應用程式更大的靈活性。

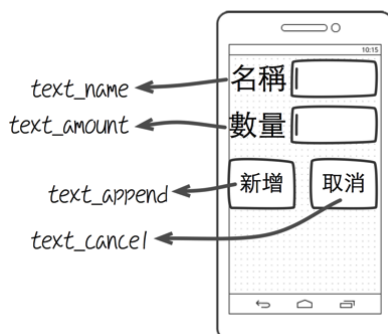
Android應用程式還有一個很重要的設計方式，就是把應用程式使用的資源獨立出來，畫面配置檔就是最明顯的應用。應用程式會使用一些需要的資源，例如在畫面上顯示的文字與圖示，還有播放的音效或影片，都會依照應用程式的需求一起放在專案的目錄。



#### 4-3-1 應用程式需要的文字資源

一般的應用程式都需要在畫面上顯示一些文字的內容，這些文字可以直接寫在程式碼，或是設定在畫面配置檔裡面，這是最直接且簡單的方式。不過這樣的作法通常會在開發的過程發生問題，例如上傳到Google Play上的應用程式，可以讓全世界的使用者下載與安裝，你的應用程式就要考慮到多國語言的問題。如果直接把文字內容寫在程式裡面，需要修改為多國語言應用程式的時候會變得非常麻煩了。

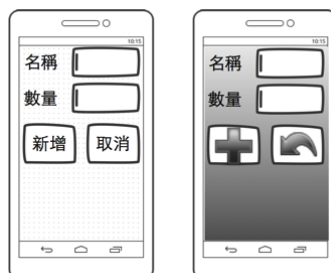
在規劃好應用程式的畫面以後，應該可以很容易整理出需要的文字資源，例如這個規劃好的畫面，針對文字資源的部份，先取出好資源的名稱：



接下來你可以使用這樣的方式，把所有應用程式需要的文字資源整理好，可能是一個列出所有文字資料的表格，或是簡單的純文字檔案。

### 4-3-2 設計應用程式需要的圖形資源

現在的行動裝置應用程式，越來越注重畫面的美觀，應用程式通常會使用一些清楚又好看的圖示，讓應用程式的畫面更美觀。顯的例子就是兩個功能類似的應用程式，其中一個的畫面只有文字與簡單的線條，另一個的畫面使用圖示代替文字，而且也設定一些顏色與效果。雖然提供的功能是一樣的，不過大家應該都會比較喜歡好看的那一個。例如這兩個應用程式的畫面：



雖然希望可以在應用程式盡量使用一些好看的圖示，不過對應用程式開發人員來說，繪圖與設定顏色可能比寫程式更難一些。自己就是最好的例子。網路上有很多可以利用的資源，提供許多套裝的圖示，使用在應用程式的時候也不用擔心風格不統一的。例如這個網站就有很多圖示資源，不過要特別注意使用授權的聲明：

<http://www.smashingmagazine.com/tag/icons/>

對我來說，圖示的問題解決以後，還有一個最大的問題就是「顏色」。設計應用程式的時候，畫面預設的底色都是黑色或白色。就算使用一些圖示加強畫面的美觀，單調的底色還是讓畫面不夠好看。我覺得顏色搭配是一門很高深的學問，為了開發Android再開始學習，好像也來不及了。不過至少可以參考Android開發人員網站提供的一些建議：

<https://developer.android.com/design/style/color.html>

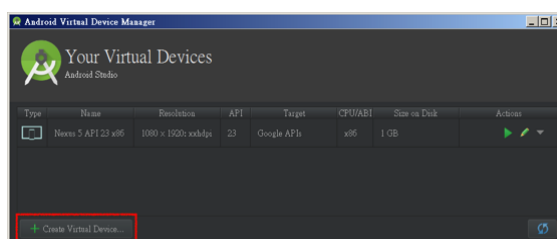
如果覺得Android開發人員網站的資訊還不夠用，可以試試下面這個網站，它可以讓你自由選擇與搭配顏色：

<https://kuler.adobe.com/create/color-wheel/>

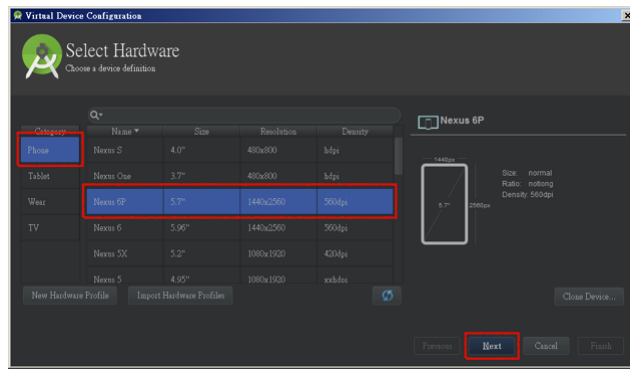
## 4-4 建立Android模擬裝置

開發Android應用程式的作法，通常會希望先在Android模擬裝置中執行測試，因為硬體設備的因素（例如震動效果），才不會在實體裝置中測試。你可以下載與安裝各種平台版本的Android模擬裝置，而且可以選擇行動電話或平板電腦。大部份的Android程式在模擬裝置執行完整的測試以後，最後再安裝到實體裝置測試就可以了。

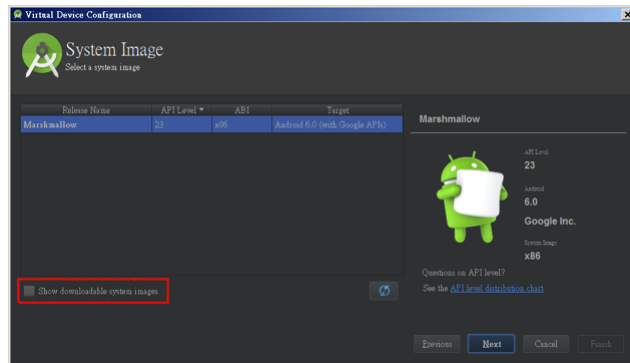
接下來說明在Android Studio建立與管理Android模擬裝置的作法。啟動Android Studio，開啟任何一個應用程式，在應用程式選擇「Tools -> Android -> AVD Manager」啟動AVD Manager，選擇「Create Virtual Device...」準備建立一個新的模擬裝置。



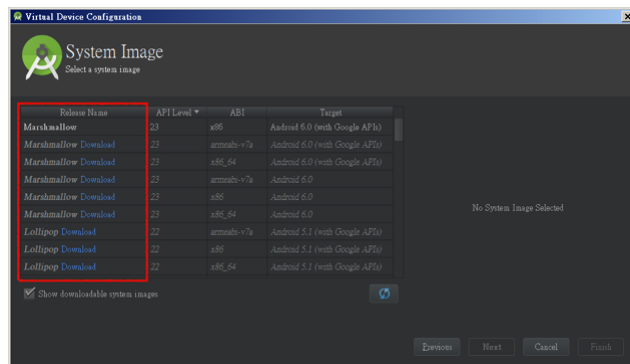
在Category選擇「Phone」，裝置選擇「Nexus 6P」，選擇「Next」：



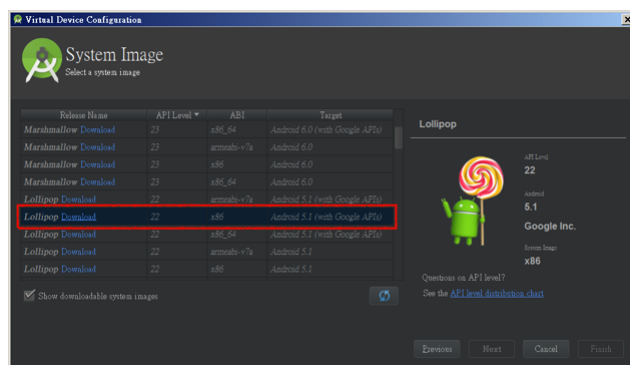
畫面會列出已經下載的Android系統映像檔，現在想要建立Android 5的模擬裝置，所以勾選「Show downloadable system images」：



畫面列出所有可以選擇的Android系統映像檔，在名稱後面如果沒有「Download」，表示已經下載與安裝。名稱後面顯示「Download」，表示還沒有下載與安裝：



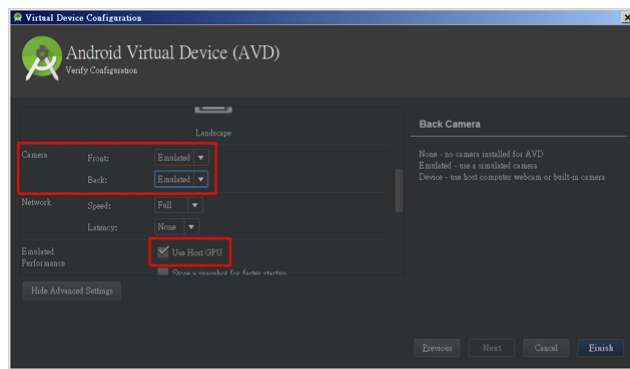
往下捲動，找到「Lollipop – 22 – x86 – Android 5.1 (with Google APIs)」，選擇Lollipop後面的「Download」，準備下載與安裝版本：



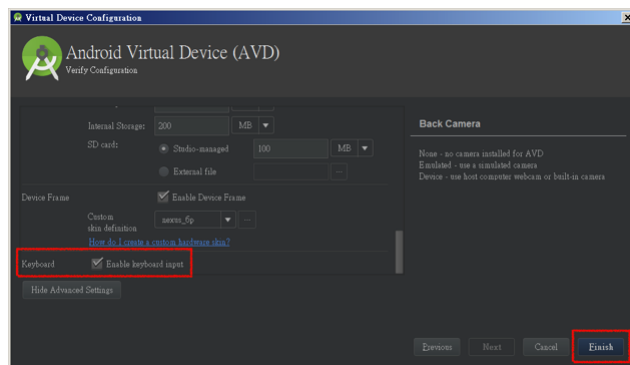
開始下載與安裝選擇的版本：



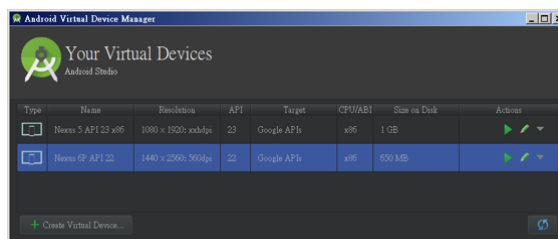




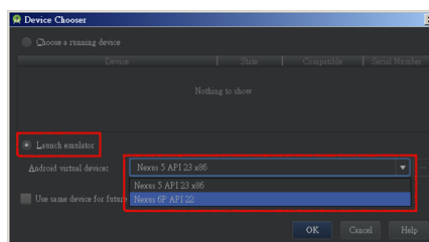
勾選「Enable keyboard input」選項，可以讓你在模擬裝置使用電腦鍵盤輸入。最後選擇「Finish」：



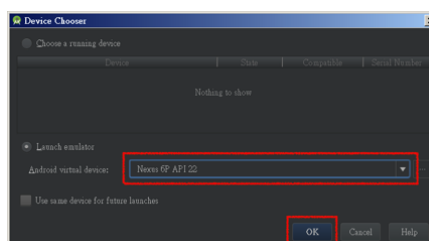
回到AVD Manager視窗以後，就可以看到建立好的模擬裝置。關閉AVD Manager：



你可以使用上面說明的作法，建立其它平版本的模擬裝置。在應用程式畫面選擇「Run -> Run 'app」執行應用程式，在選擇視窗選擇「Launch emulator」後，可以在Android virtual device選擇建立好的模擬裝置：

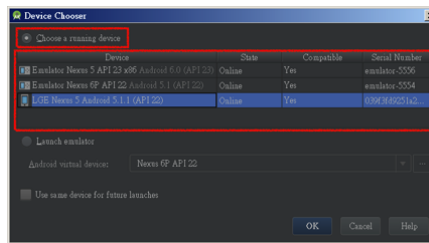


選擇模擬裝置以後，選擇「OK」按鈕，就可以啟動指定的模擬裝置並安裝與執行應用程式：



如果要在實體裝置中執行應用程式，安裝好實體裝置的驅動程式並連接到電腦以後，在選擇裝置的視窗選擇「Choose a runn device」，就可以選擇已經連接的實體裝置，已經啟動的模擬裝置也會在這裡顯示：






完成開發Android應用程式前的準備工作了，後面的內容會使用一個實際的應用程式，開始進入應用程式專案開發的階段。

後續 >> [Android Tutorial using Kotlin 第二章（1）規劃與建立應用程式需要的資源](#)

Does Clearly work fine?



Shortcuts: **SHIFT+CTRL+C** to Toggle, **ESC** to Close.

 Give us feedback

Build upon  with Clearly