**Lab6 清單元件**

1. 本節目的：

* 理解什麼是Adapter
* 理解Adapter與ListView的關係
* 學習如何使用清單元件，Spinner、ListView、Gallery及GridView

1. 觀念說明：

應用程式最主要的目的便是要傳達某個訊息給使用者，而當這訊息量非常多的時候，單純的畫面就無法容納所有的資訊，尤其是在畫面非常小的手機螢幕上。這時我們往往會使用清單或下拉式選單等方式，讓使用者透過滑動查看更多資訊。這類的列表清單元件對於Android應用程式而言就有著舉足輕重的重要性。如以下APP畫面，很常看到清單的呈現方式：

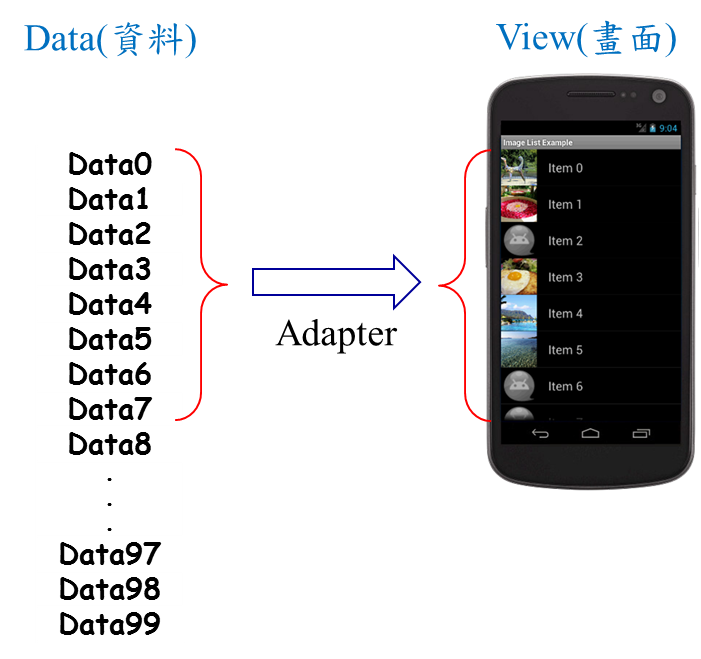


1. **Adapter 介紹**

清單元件在顯示資料內容上有很特別的設計結構。要顯示的資料一般都是來自於外部的程式，例如透過網路或是其他方式取得資料，我們將這些外部資料稱之為**資料來源**。

在Android中，資料來源的處理與顯示畫面的**清單元件**是分開來的操作的，只有在需要顯示的時候，才會把資料來源轉成顯示的清單畫面，而負責做這個轉換動作的介面就是Adapter，而清單元件就像是容器，決定容器內要放入的什麼內容的人就是Adapter。

用簡單的比喻，**清單元件(View)**就像是間飯店，**資料來源(Data)**就像是客人，而Adapter類似於接待人員，客人要進到房前需要先詢問接待人員，房間位置，在由接待人員來安排客人進駐到對應的飯店房間。



1. **Adapter繼承類別與使用**

在使用清單元件時，每一筆項目(Item)的畫面會需要有對應的layout(Xml)來作呈現，而Android SDK本身也有提供的現成的圖檔、layout的資源，我們可以直接使用實現簡單的顯示效果。以下我們用ListView元件說明如何使用ArrayAdapter：



Step2：建立adapter物件，並放入資料來源與要顯示的項目畫面

回傳第幾個項目被按下

Step4：建立項目(Item)點擊事件

Step3：建立清單元件，並連結adapter

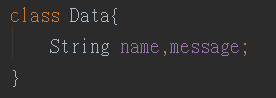
Step1：建立資料來源

1. 我們要先假設擁有一個資料來源，我們要利用ArrayAdapter，ArrayAdapter繼承自BaseAdapter，資料來源需要是陣列格式。
2. 我們要產生出ArrayAdapter的實體，第一個參數要傳入呼叫對象(即this，本身物件)，第二個參數傳入一個layout，這邊我們直接使用Android SDK提供的android.R.layout.simple\_list\_item\_1現成的layout(即最簡單的項目風格)，第三個參數傳進資料(即data)。
3. 我們要將ArrayAdapter指派給畫面元件，這邊我們實做出一個ListView，使用setAdapter()將Adapter做連結。
4. 我們要為ListView設定點擊事件。由於我們不是要對ListView做點擊，而是要對ListView裡的項目(item)作點擊，因此這邊我們使用OnItemClickListener()，OnItemClickListener內的onItemClick方法的第三個參數position會回傳按下的項目編號，可以根據該編號從資料來源中取出對應的資料。
5. **Adapter客製化**

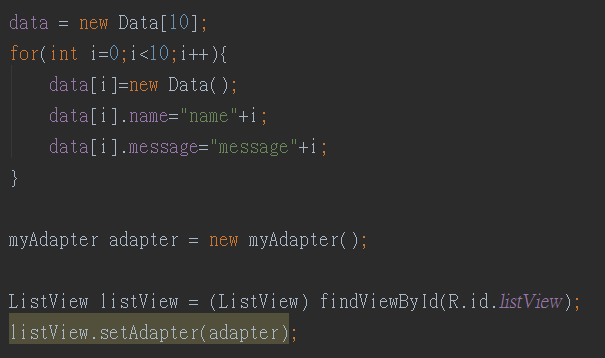
某些情況下，我們需要更複雜的畫面，可能會包含圖片、文字等，可能Android SDK所提供的layout範例就無法滿足我們的使用需求，這時我們就需要實作客製化Adapter。

在實作**Adapter客製化**上我們有三個準備工作：

1. 設計客製化Data：如果我們要顯示的資料項目需要兩種以上的資料內容(如要顯示標題、內容、圖片等多筆資料)，我們就需要設計對應的一個類別去定義這些資料。



與要顯示內容對應的類別宣告



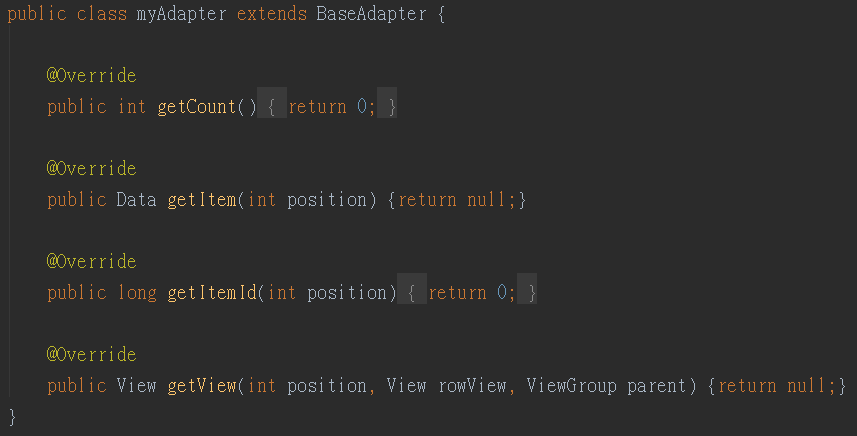
用陣列方式宣告自行設計的類別

用迴圈去產生資料來源，並放入類別陣列之中。

1. 設計客製化layout：我們需要設計要顯示的項目layout。用之前章節所教的layout設計方法去設計客製化的Xml。



1. 建立客製化的Adpater：由於我們的資料內容與格式是自行設計的，因此我們也需要自行設計對應的Adapter。由於我們希望延伸BaseAdapter的功能，我們透過Java的繼承功能建立一個繼承於BaseAdapter的myAdapter，並藉由復寫的方式修改其中幾個方法來建立自己的客製化Adapter。如下所示：



取得指定項目內的資料。

取得指定項目的資料id。

取得資料來源陣列的筆數。

顯示項目(Item)的資料對應的畫面。

創立繼承BaseAdapter的物件後，我們需要覆寫BaseAdapter的幾種方法：

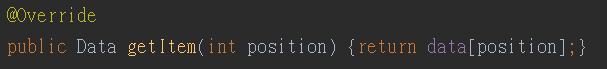
1. public int getCount()

getCount須回傳要顯示的資料筆數，可直接放入陣列的長度，程式碼覆寫如下：



1. public Data getItem(int position)

getItem()可得到position對應的資料，程式碼覆寫如下：

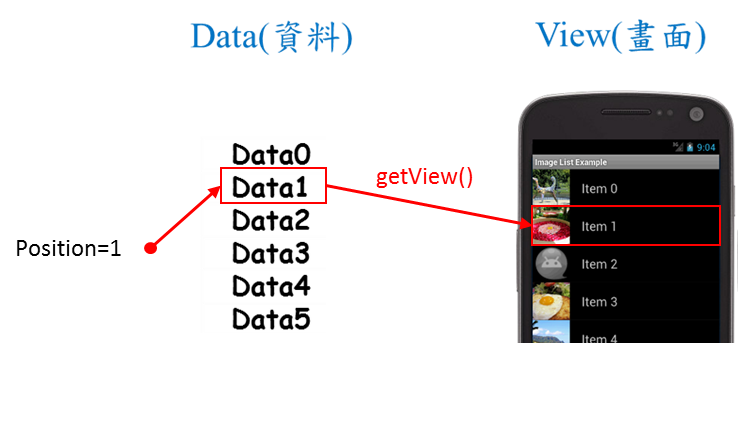


1. public long getItemId(int position)

getItemId ()可得到position對應的id值，這id值應該要為唯一性的編號，一般下非必要的參數，故此處不作修改。

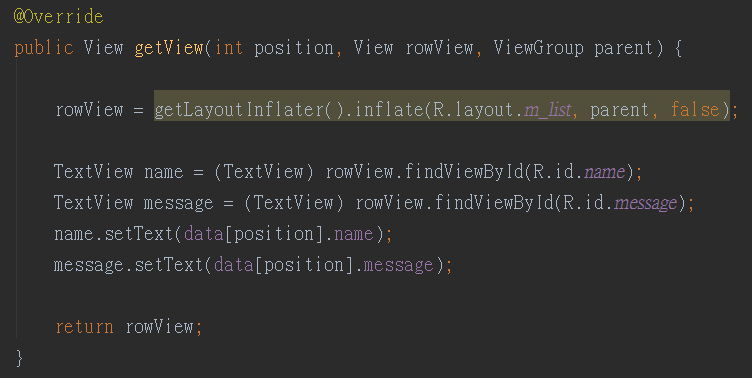
1. public View getView(int position, View rowView, ViewGroup parent)

Adapter的getView ()方法，**主要是將資料來源(Data)顯示在畫面(View)上，**是設計的客製化Adapter的核心。如下圖所示，取得項目編號(position)之後，把資料來源(Data)依照房間編號(position)顯示在畫面中。



根據Position取得Data

放到對應的項目(View)

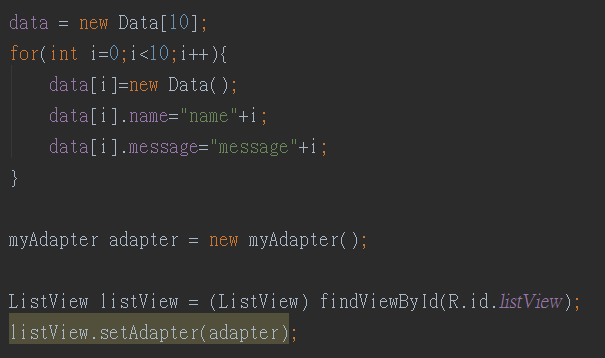


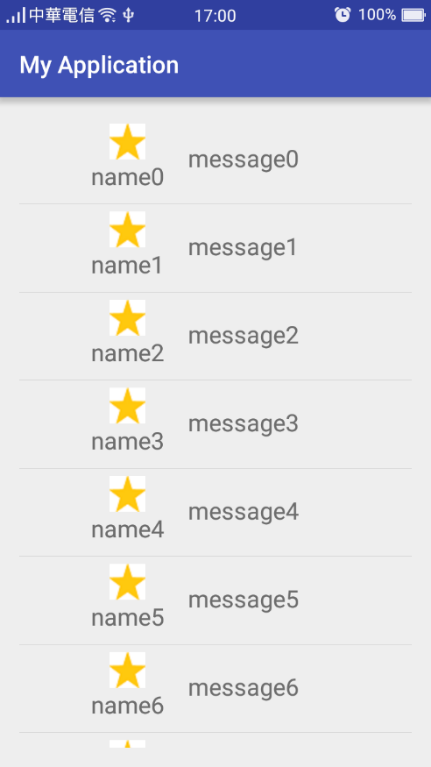
根據項目編號把對應的資料放到畫面元件之中

取得畫面元件

需要顯示的項目編號

設計完myAdapter後，我們將setAdapter換成myAdapter，實作後的結果如下：

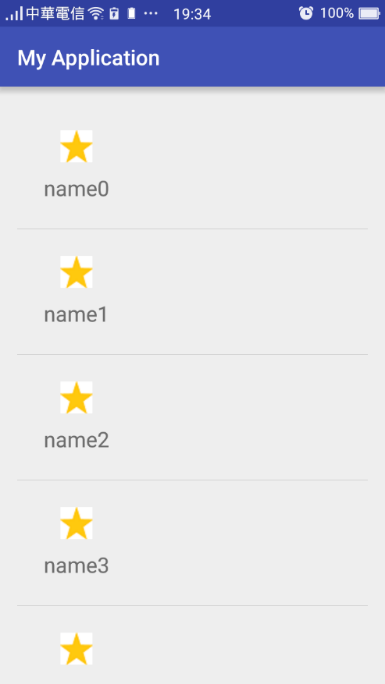




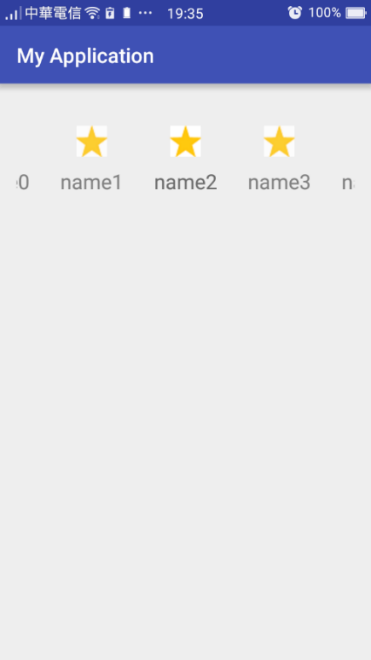
1. **清單元件**

Android有提供幾種清單元件以供顯示，由於資料內容都是由Adapter決定，清單元件扮演著容器的角色，因此只需要替換連結的清單元件就可以實現不同的資料模式，以下分別作介紹：

* ListView(縱向清單)：ListView是最基本的清單元件，他可以將資料垂直排放，由於大部分的行動裝置都是長比寬高，因此ListView能很清楚的顯示資訊。



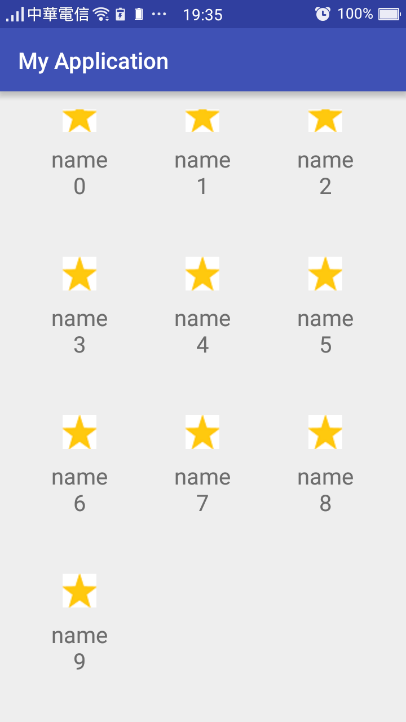
* Gallery (橫向清單/畫廊)：Gallery是可以橫向翻動的清單，在應用程式中常會使用他來展示照片顯示，因此被稱為畫廊。



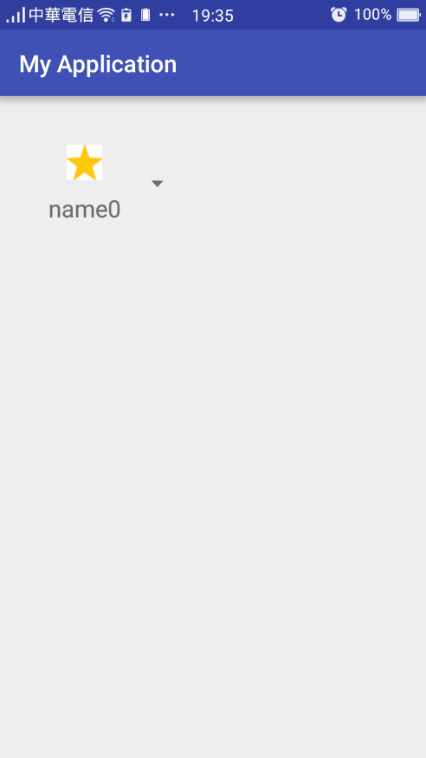
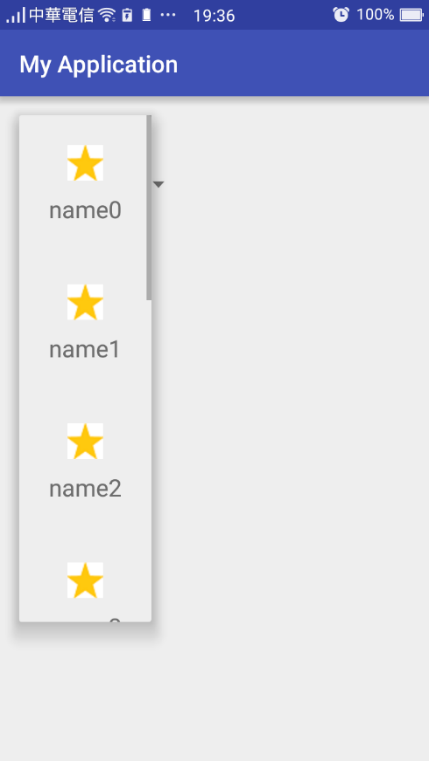
* GridView (格狀清單)：GridView能將將內容縮成方形，透過窗戶般的格狀擺放方式，並依照由左至右、由上到下的排列。GridView比起前述兩個元件，他可以使用setNumColumns()決定橫向要顯示幾列，如果沒有設定則只會顯示一列。



顯示三列



* Spinner (下拉式選單)：Spinner是下拉式選單，一般下只會占用一個TextView元件大小，但被點擊時可以展開清單讓使用者選擇。

1. 設計重點:

* 利用Adapter靈活的配置Spinner、ListView及GridView
* 設計一含有Spinner、ListView及GridView的布局
* ListView使用範例Xml顯示清單
* Spinner與GridView分別使用兩種不同的客製化畫面來實作客製化Adapter顯示清單



Spinner

GridView

ListView

trans\_list.xml

內建

cubee\_list.xml

客製化

1. 設計步驟:

Step1

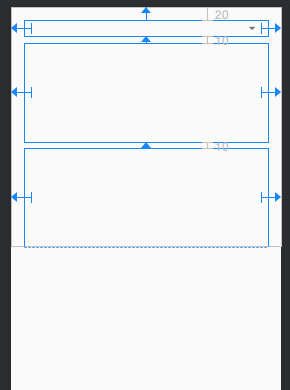
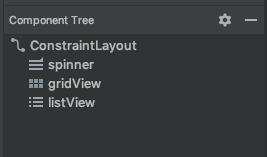
建立專案，並將附件的圖片放於drawable底下

以及建立xml檔

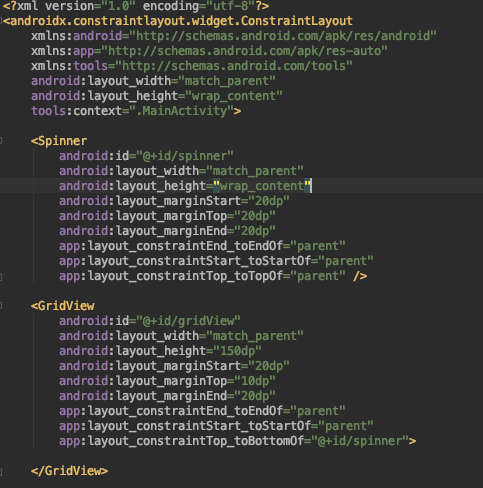
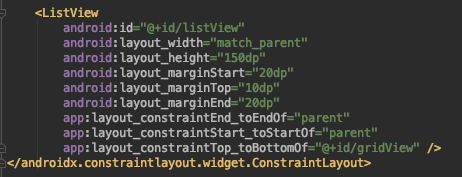
|  |  |
| --- | --- |
|  |  |

Step2

繪製activity\_main. xml

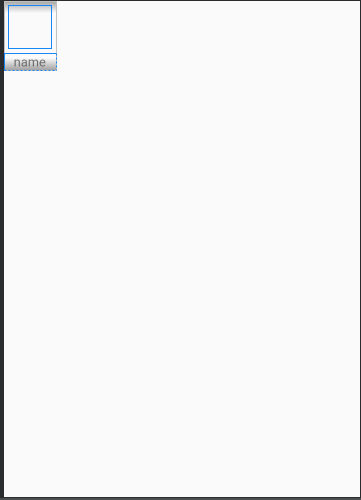
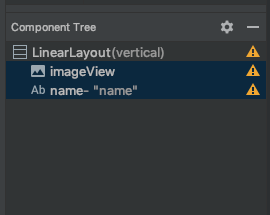
 

對應的xml如下：

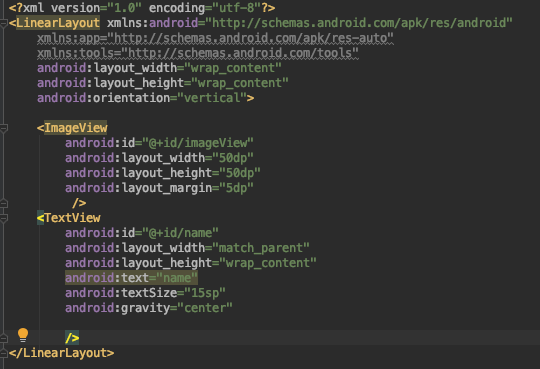


Step3

繪製cubee\_list. xml，顯示客製化的畫面

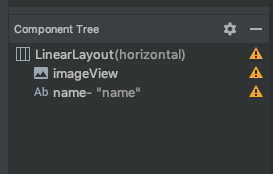
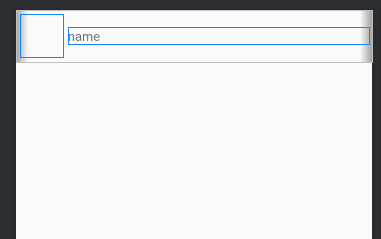
 

對應的xml如下：

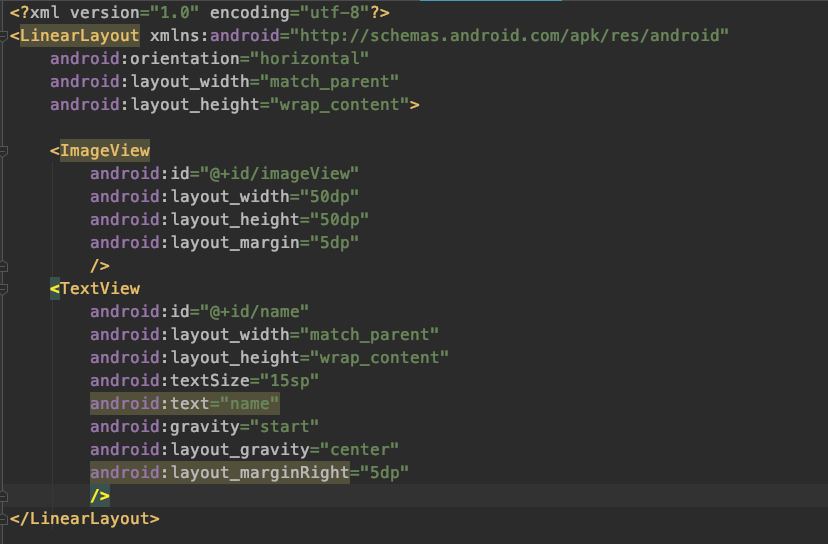


Step4

繪製trans\_list. xml，顯示客製化的畫面

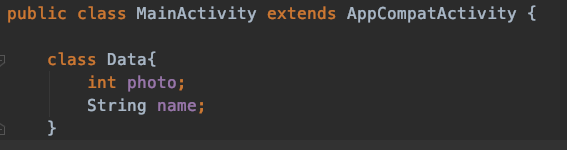


對應的xml如下：



Step5

編寫MainActivity，建立一個客製化的類別Data，包含一張圖片與文字，用於保存之後要顯示於客製化Adapter的資料。

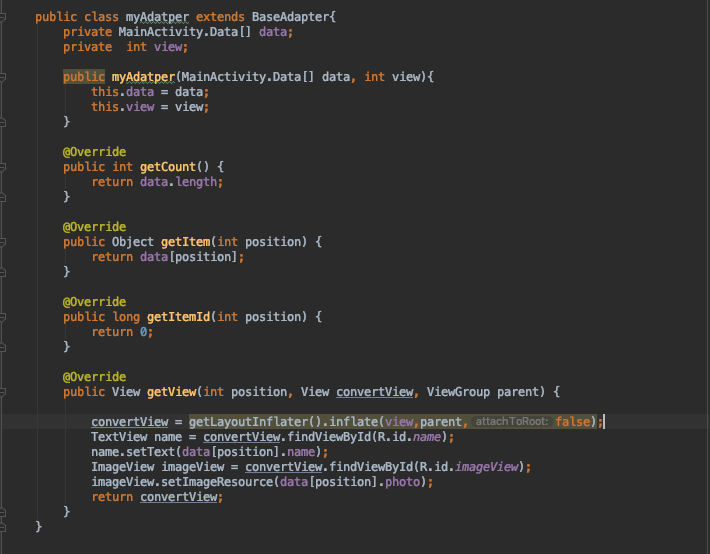


名稱

圖片id

Step6

建立myAdapter來顯示Spinner及GridView的客製化畫面。由於Spinner與GridView有各自要顯示的資料與畫面，因此我們需要先把他們的資料在建構myAdapter就保存在myAdapter中，避免出現取錯資料的問題。



回傳某筆項目id

回傳資料來源筆數

回傳某筆項目

根據position把圖片顯示到ImageView

根據position把字串顯示到TextView

取得畫面元件

需要顯示的項目編號

透過建構子儲存資料來源與畫面到myAdapter之中

連接ImageView元件

連接TextView元件

取得Xml畫面

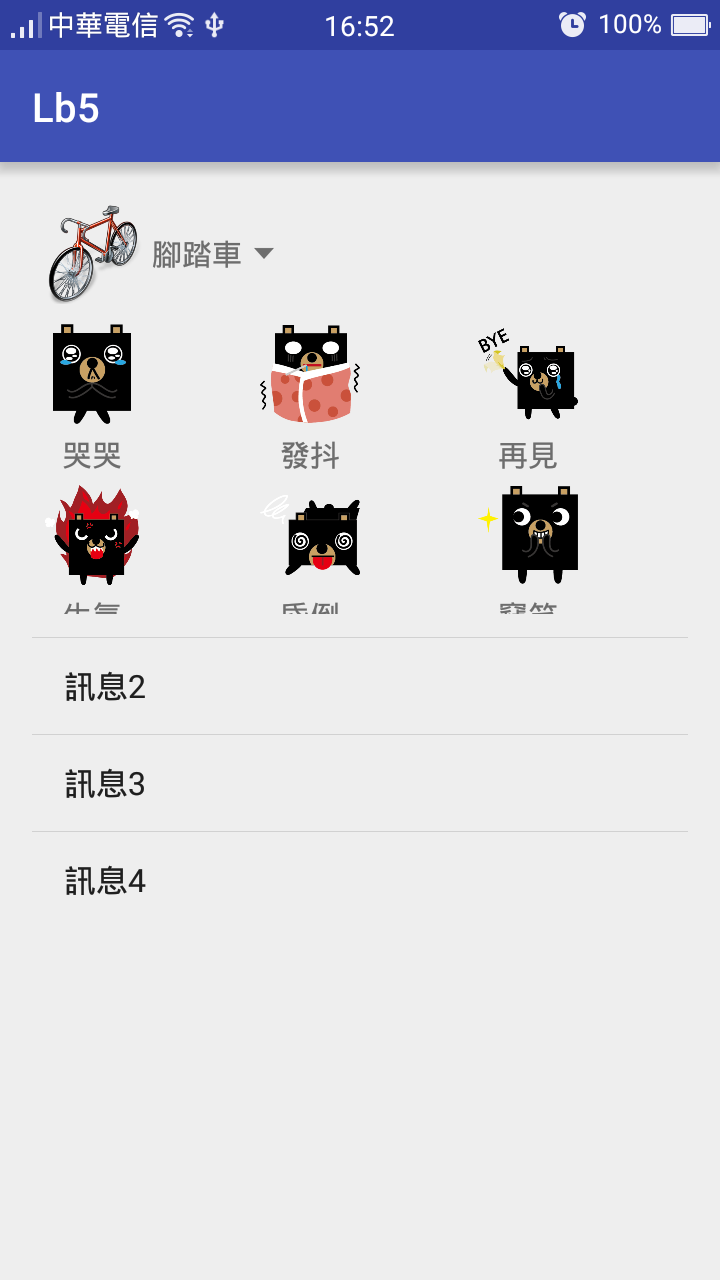
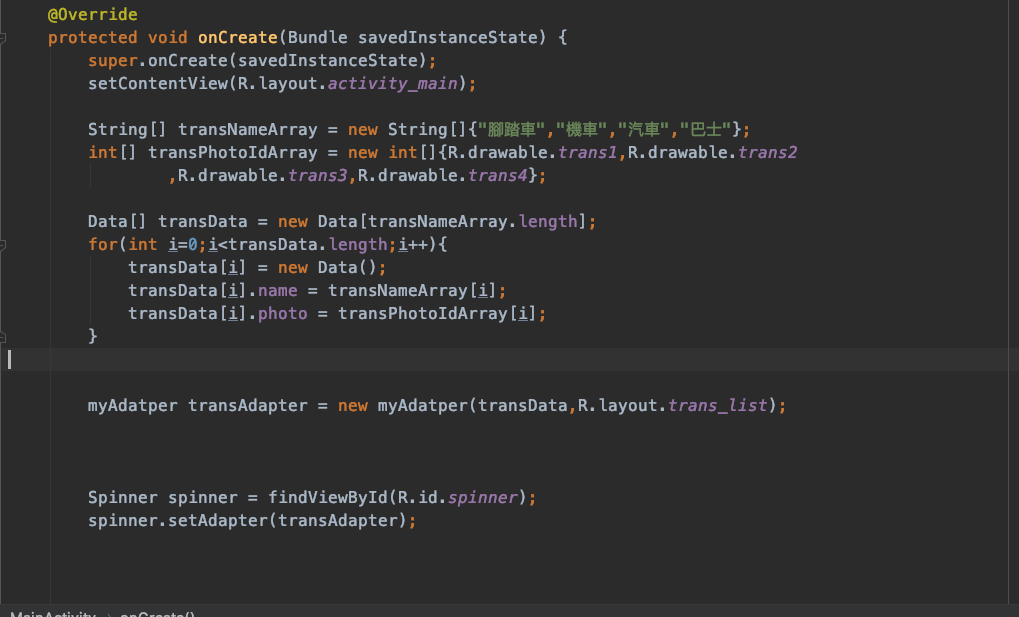
保存在myAdapter之中的畫面

繼承BaseAdapter

保存在myAdapter之中的資料來源

Step7

使用Spinner，透過客製化layout與myAdapter顯示transData(自製的資料)

 Step7

Step1建立資料來源，用陣列方式宣告自行設計的類別，並為陣列的每個內容填入要顯示的資料

Step3：連接Spinner元件，並連結myAdapter

Step2建立myAdapter物件，並放入transData與trans\_list.xml

Java物件導向的特性，每個陣列物件都必須要產生實體

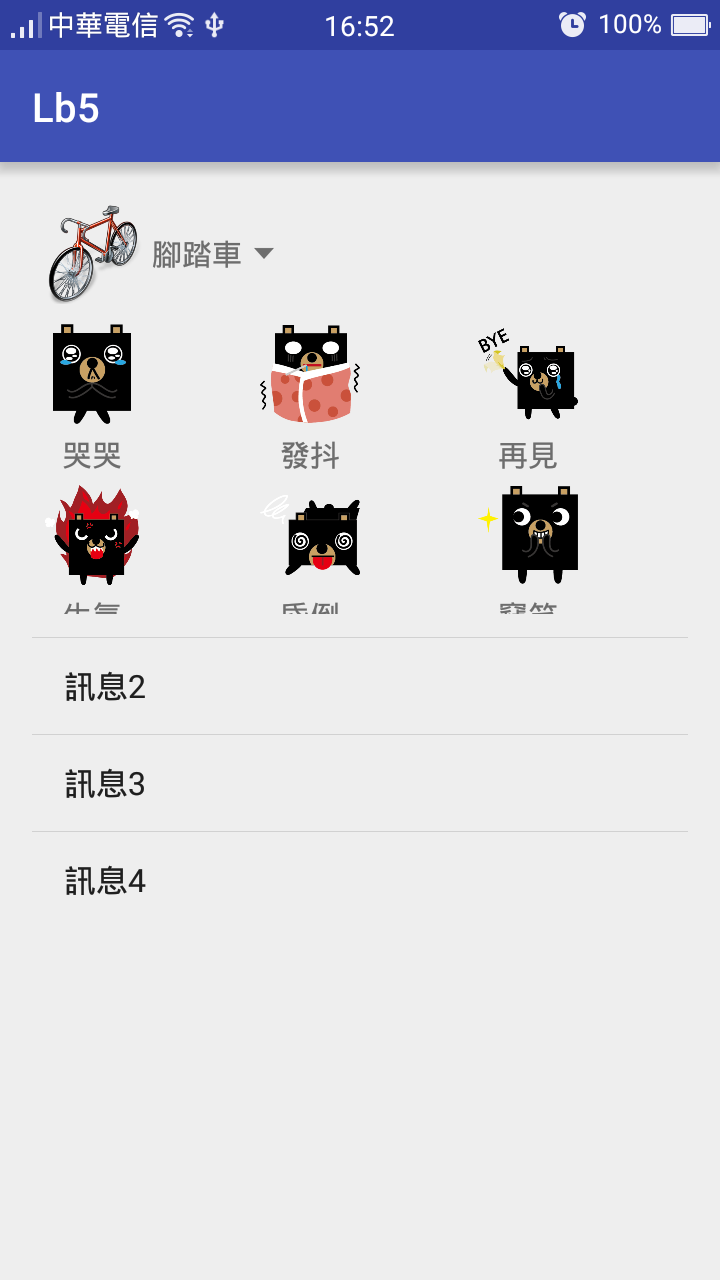
使用ListView，透過Android SDK 提供的現成layout與ArrayAdapter顯示messageArray(字串陣列)



Step3：連接ListView元件，並連結Adapter

Step1：建立資料來源(字串)

Step2：建立Adapter物件，並放入字串與simple\_list\_item\_1.xml

Step8

使用GridView，透過客製化layout與myAdapter顯示cubeeData (自製的資料)



Step1：建立資料來源，用陣列方式宣告自行設計的類別，將陣列的每個內容填入要顯示的資料

Step3：連接GridView元件，並連結myAdapter

Step2：建立myAdapter物件，並放入cubeeData與cubee\_list.xml

物件導向的特性，每個陣列物件都必須要產生實體

