

Something About Android

Dave Lin @ NTUIM
2012 / 11 / 01

在開始之前

- 如果我今天沒心情/想睡/不爽/頭痛，就是沒法聽，那我要怎麼學Android?
- 看這個吧...<http://code.google.com/p/androidbmi/wiki/IntroAndroid>
- 今天的重點在於概念，實做的重點就交給同學自己來了！

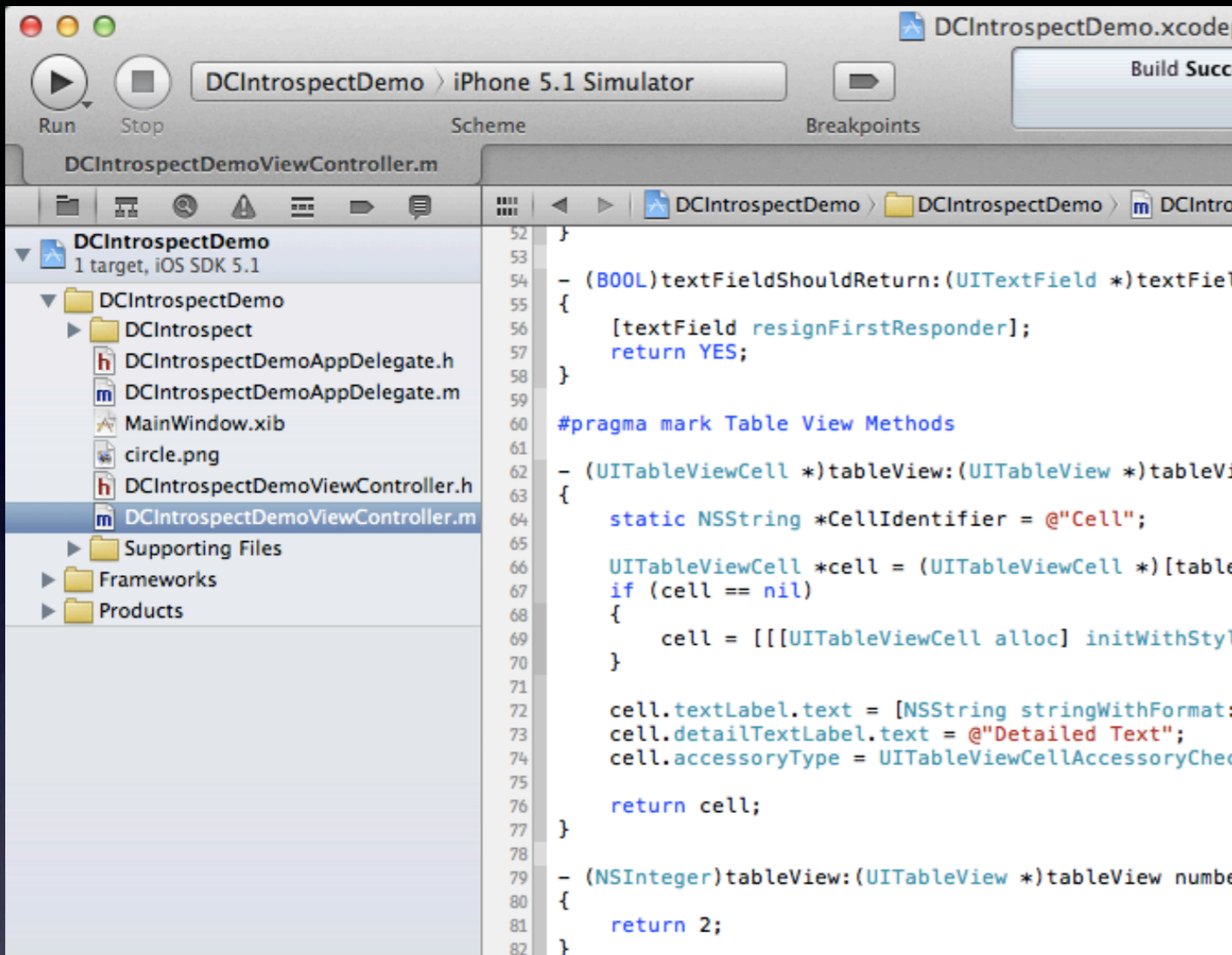
iOS 與 Android 之 主觀對比

Apple v.s. Google



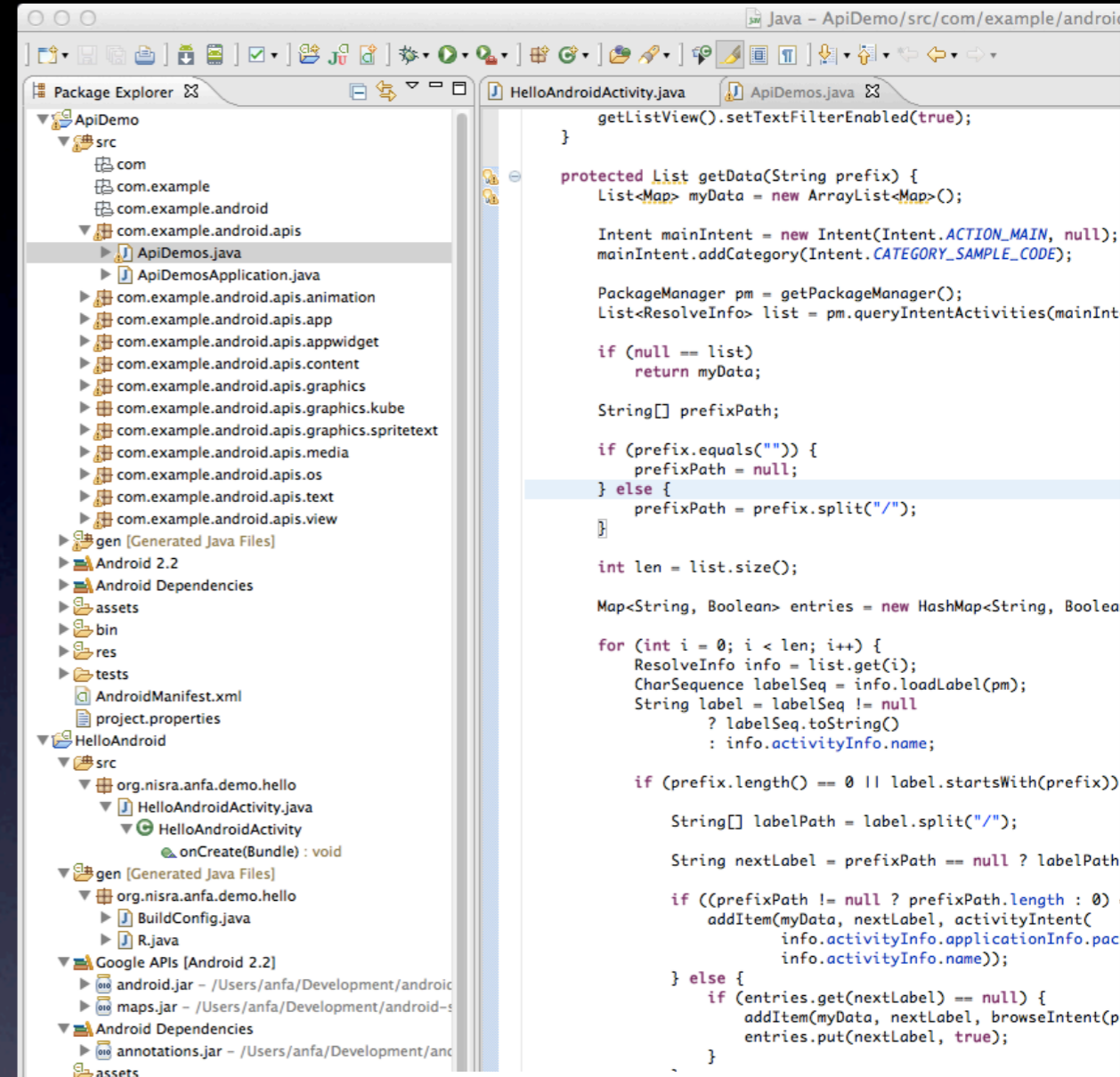
R.I.P

ObjectiveC v.s. Java



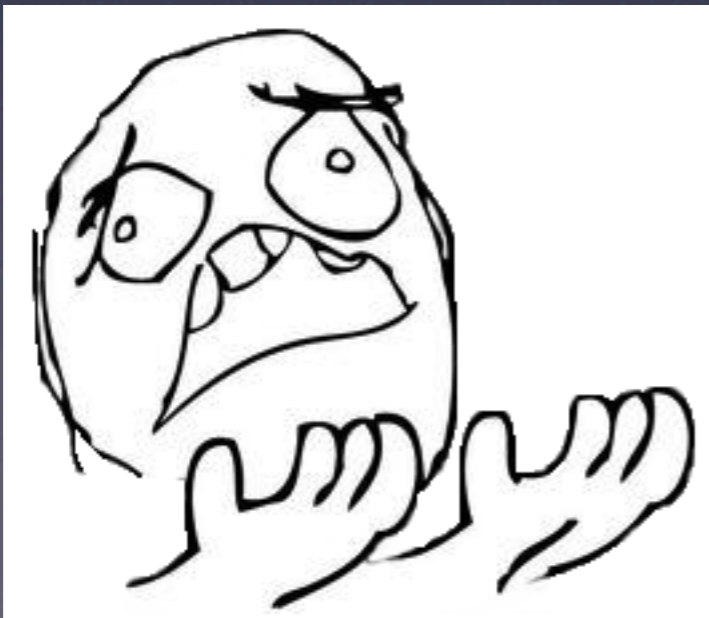
The screenshot shows the Xcode IDE with a project named "DCIntrospectDemo" targeting an iPhone 5.1 Simulator. The code in "DCIntrospectDemoViewController.m" includes a method for handling text field input and a table view method. The table view method uses a static NSString for cell identifiers and allocates UITableViewCell objects.

```
52 }  
53  
54 - (BOOL)textFieldShouldReturn:(UITextField *)textField  
55 {  
56     [textField resignFirstResponder];  
57     return YES;  
58 }  
59  
60 #pragma mark Table View Methods  
61  
62 - (UITableViewCell *)tableView:(UITableView *)tableView  
63 {  
64     static NSString *CellIdentifier = @"Cell";  
65  
66     UITableViewCell *cell = (UITableViewCell *)[tableView  
67     dequeueReusableCellWithIdentifier:CellIdentifier];  
68     if (cell == nil)  
69     {  
70         cell = [[[UITableViewCell alloc] initWithStyle:  
71         UITableViewCellStyleDefault] initWithStyle:  
72         UITableViewCellStyleDefault];  
73         cell.textLabel.text = [NSString stringWithFormat:  
74         @"Detailed Text"];  
75         cell.accessoryType = UITableViewCellAccessoryCheckmark;  
76     }  
77     return cell;  
78 }  
79 - (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSection:  
80 {  
81     return 2;  
82 }
```



The screenshot shows an IDE with a Java project named "ApiDemo". The code in "HelloAndroidActivity.java" includes a method for getting data from a list and a method for displaying the data in a table view. The code uses ArrayList and HashMap to manage data and intents to navigate between activities.

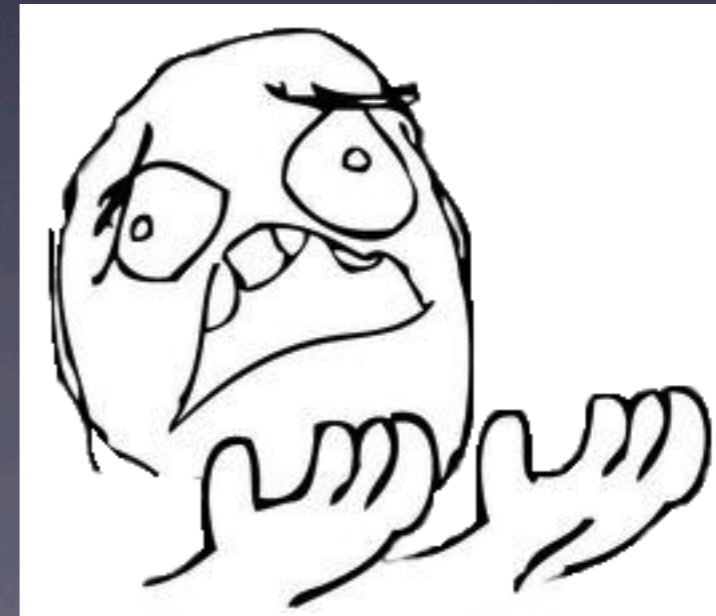
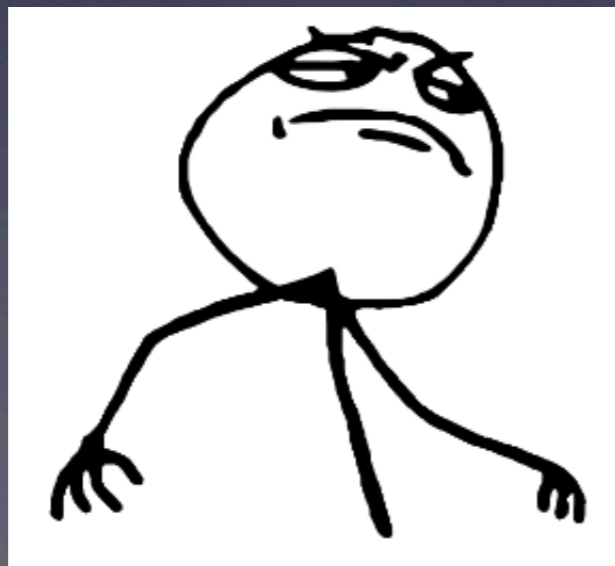
```
getListView().setTextFilterEnabled(true);  
}  
  
protected List getData(String prefix) {  
    List<Map> myData = new ArrayList<Map>();  
  
    Intent mainIntent = new Intent(Intent.ACTION_MAIN, null);  
    mainIntent.addCategory(Intent.CATEGORY_SAMPLE_CODE);  
  
    PackageManager pm = getPackageManager();  
    List<ResolveInfo> list = pm.queryIntentActivities(mainIntent,  
    PackageManager.GET_ACTIVITIES);  
  
    if (null == list)  
        return myData;  
  
    String[] prefixPath;  
  
    if (prefix.equals("")) {  
        prefixPath = null;  
    } else {  
        prefixPath = prefix.split("/");  
    }  
  
    int len = list.size();  
  
    Map<String, Boolean> entries = new HashMap<String, Boolean>();  
  
    for (int i = 0; i < len; i++) {  
        ResolveInfo info = list.get(i);  
        CharSequence labelSeq = info.loadLabel(pm);  
        String label = labelSeq != null  
        ? labelSeq.toString()  
        : info.activityInfo.name;  
  
        if (prefix.length() == 0 || label.startsWith(prefix))  
  
            String[] labelPath = label.split("/");  
  
            String nextLabel = prefixPath == null ? labelPath[0] :  
            prefixPath[prefixPath.length - 1] + labelPath[0];  
  
            if ((prefixPath != null ? prefixPath.length : 0) > 0)  
                addItem(myData, nextLabel, activityIntent(  
                info.activityInfo.applicationInfo.packageName,  
                info.activityInfo.name));  
            } else {  
                if (entries.get(nextLabel) == null) {  
                    addItem(myData, nextLabel, browseIntent(  
                    info.activityInfo.name));  
                    entries.put(nextLabel, true);  
                }  
            }  
        }  
    }  
}
```



ObjectiveC v.s. Java

ObjectiveC	Java
學習門檻較高 有平台限制 應用層面較低	學習門檻較低 跨平台 應用層面較廣

Simulator v.s. Emulator



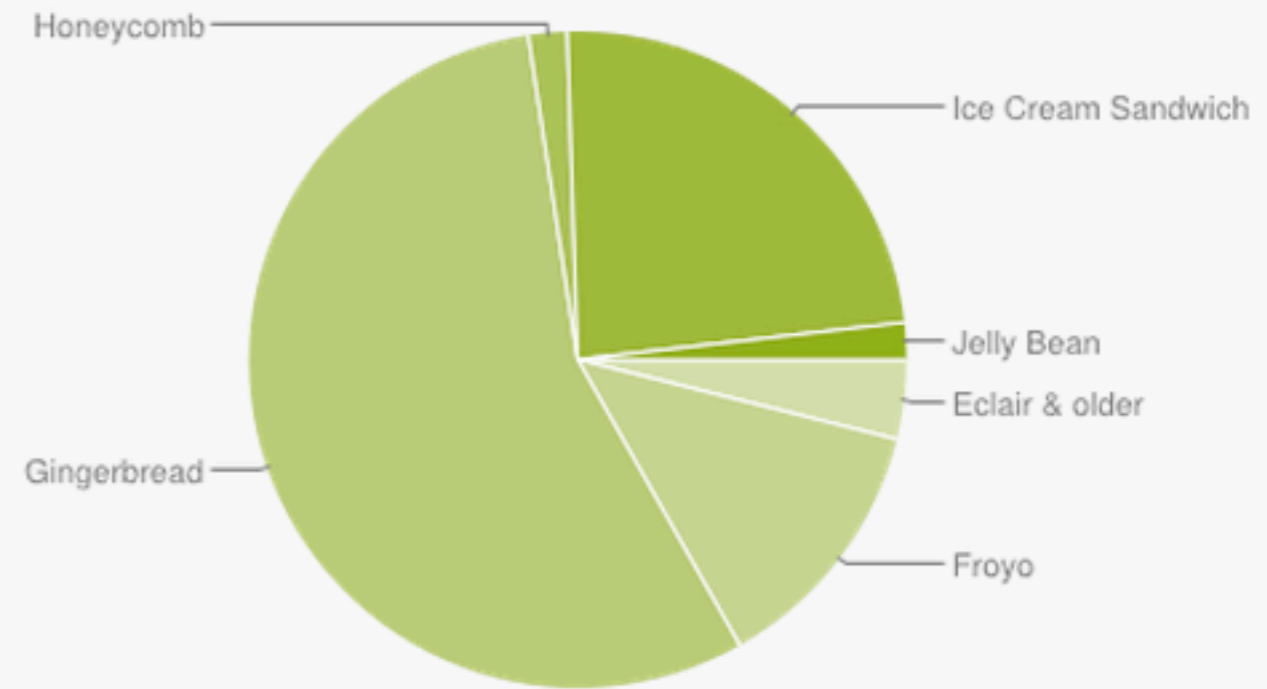
Simulator v.s. Emulator

Simulator	Emulator
<p data-bbox="386 1150 1256 1354">執行速度快 (!!!) 與真實環境差異較大</p>	<p data-bbox="1484 1150 2354 1354">執行速度慢 與真實環境差異較小</p>

Android 生態現況

版本分佈

Version	Codename	API	Distribution
1.5	Cupcake	3	0.1%
1.6	Donut	4	0.4%
2.1	Eclair	7	3.4%
2.2	Froyo	8	12.9%
2.3 - 2.3.2	Gingerbread	9	0.3%
2.3.3 - 2.3.7		10	55.5%
3.1	Honeycomb	12	0.4%
3.2		13	1.5%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	23.7%
4.1	Jelly Bean	16	1.8%



Data collected during a 14-day period ending on October 1, 2012

螢幕大小及解析度分佈

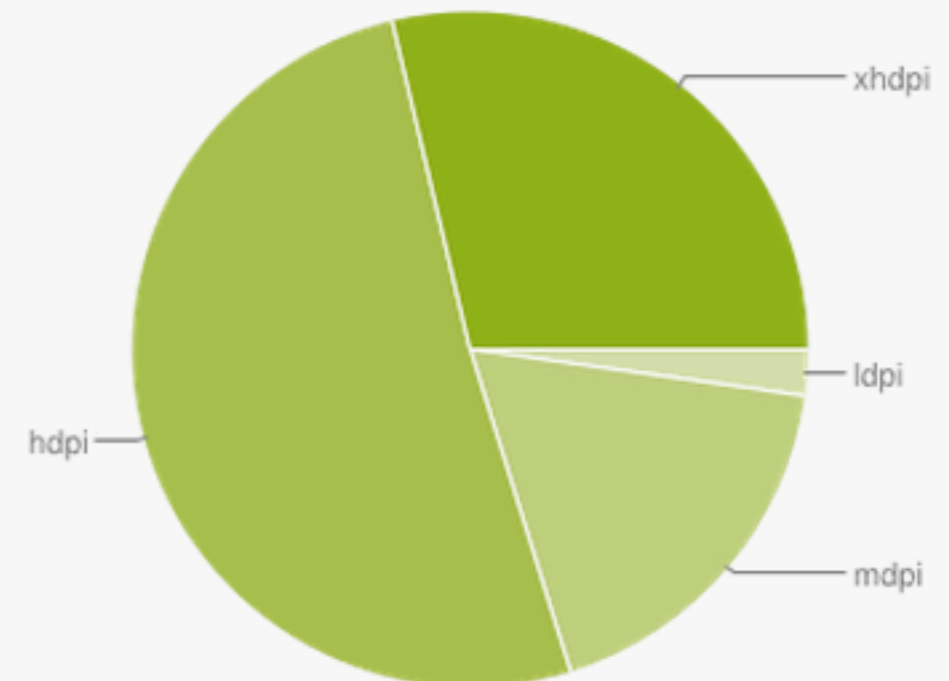
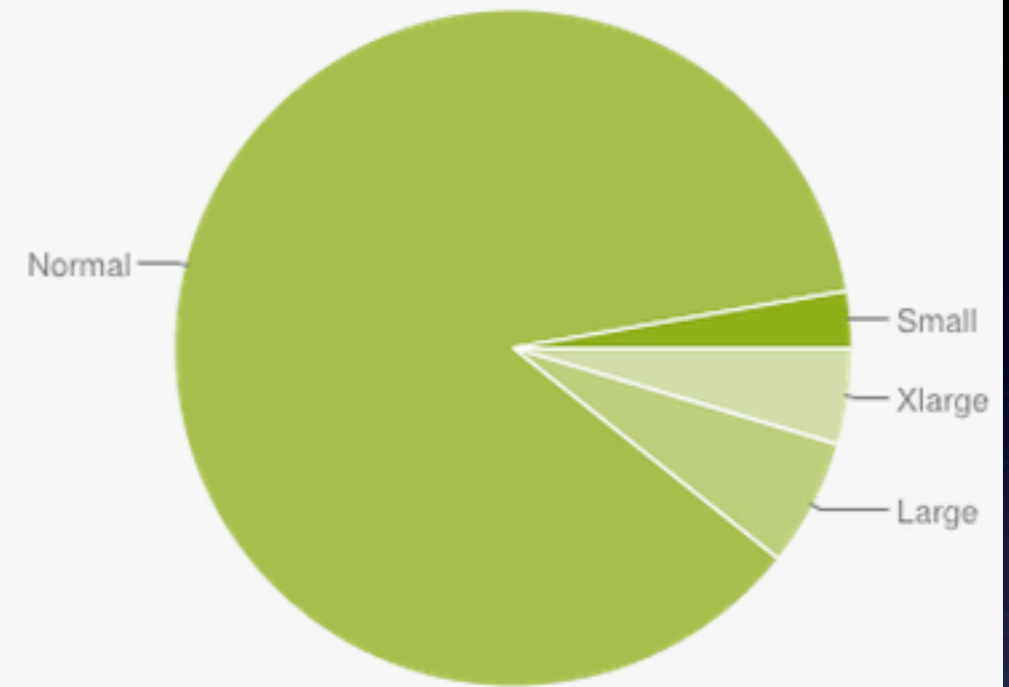
This section provides data about the relative number of active devices that have a particular screen configuration, defined by a combination of screen size and density. To simplify the way that you design your user interfaces for different screen configurations, Android divides the range of actual screen sizes and densities into:

- A set of four generalized **sizes**: *small*, *normal*, *large*, and *xlarge*
- A set of four generalized **densities**: *ldpi* (low), *mdpi* (medium), *hdpi* (high), and *xhdpi* (extra high)

For information about how you can support multiple screen configurations in your application, see [Supporting Multiple Screens](#).

Note: This data is based on the number of Android devices that have accessed Google Play within a 7-day period ending on the data collection date noted below.

	ldpi	mdpi	hdpi	xhdpi
small	1.7%		1.0%	
normal	0.4%	11%	50.1%	25.1%
large	0.1%	2.4%		3.6%
xlarge		4.6%		



Data collected during a 7-day period ending on October 1, 2012

Android 生態系統現況

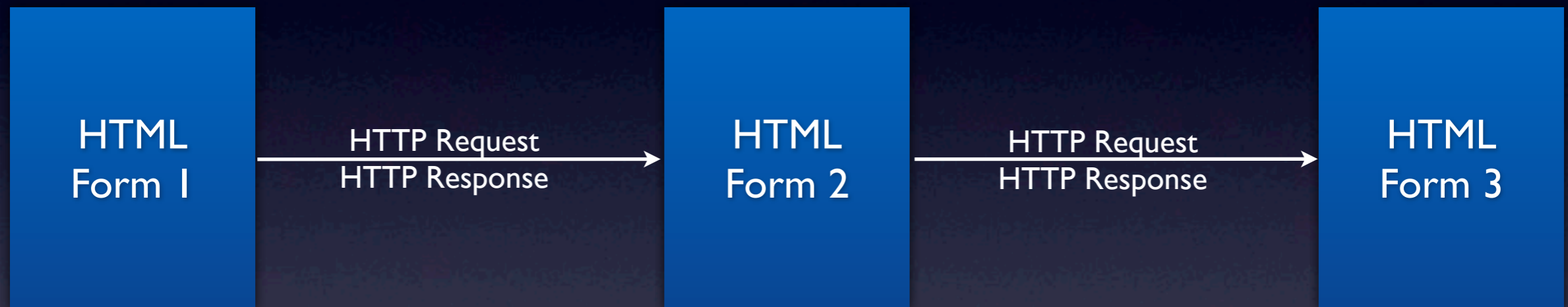
- 2.x 仍佔大多數 (>70%)
- 4.0 以後加入的新功能無法被普及應用
- 螢幕尺寸比例不一造成開發上的困難
- 一直一直一直一直一直一直一直出新版
- 天龍國行政單位萬歲萬歲萬萬歲

進入Android世界之前...

Form-based Application

- 每個畫面都是一個表單
- 程式的主要功能
 - 在表單中與使用者互動
 - 表單到表單之間資料傳遞

Form Based App - Web



Question: what happened if I press “BACK” button?

Form Based App - Android



Question: what happened if I press “BACK” button?

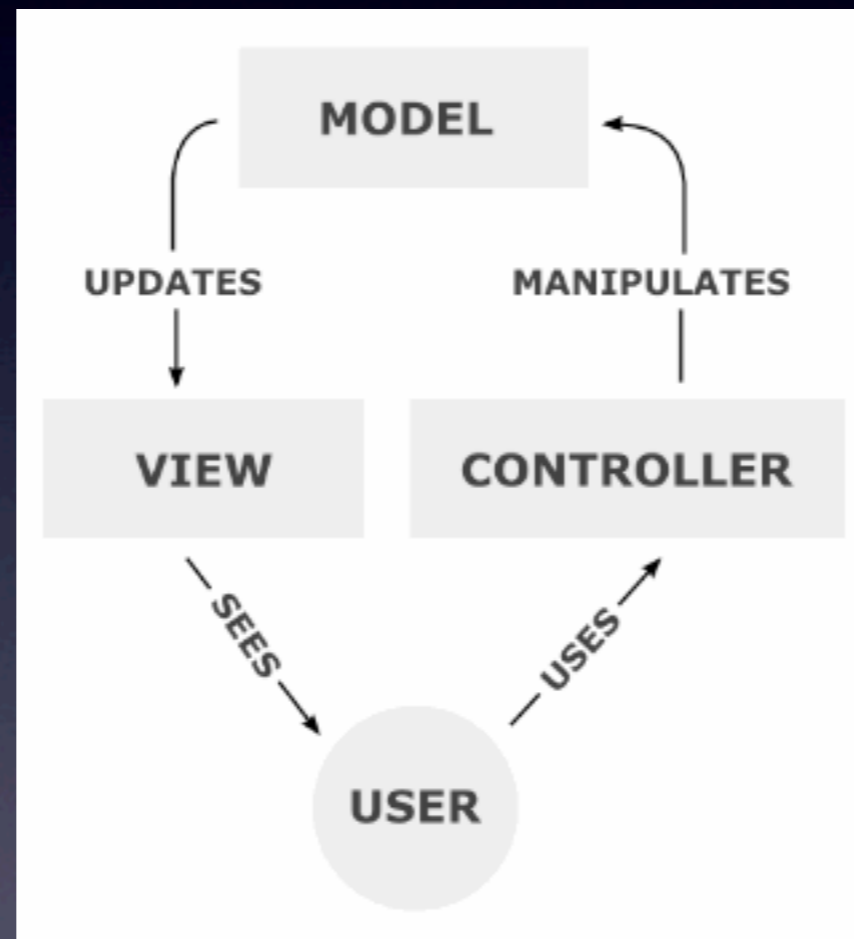
Model-View-Controller (MVC) Pattern

- MVC模式（三層架構模式）（Model-View-Controller）是軟體工程中的一種軟體架構模式，把軟體系統分為三個基本部分：模型（Model）、檢視（View）和控制器（Controller）

MVC

- MVC模式最早由Trygve Reenskaug在1974年提出，是全錄帕羅奧多研究中心（Xerox PARC）在20世紀80年代為程式語言Smalltalk發明的一種軟體設計模式
 - （控制器Controller） - 負責轉發請求，對請求進行處理。
 - （檢視View） - 介面設計人員進行圖形介面設計。
 - （模型Model） - 程式設計師編寫程式應有的功能（實作演算法等等）、資料庫專家進行資料管理和資料庫設計(可以實作具體的功能)。

MVC



MVC的影響

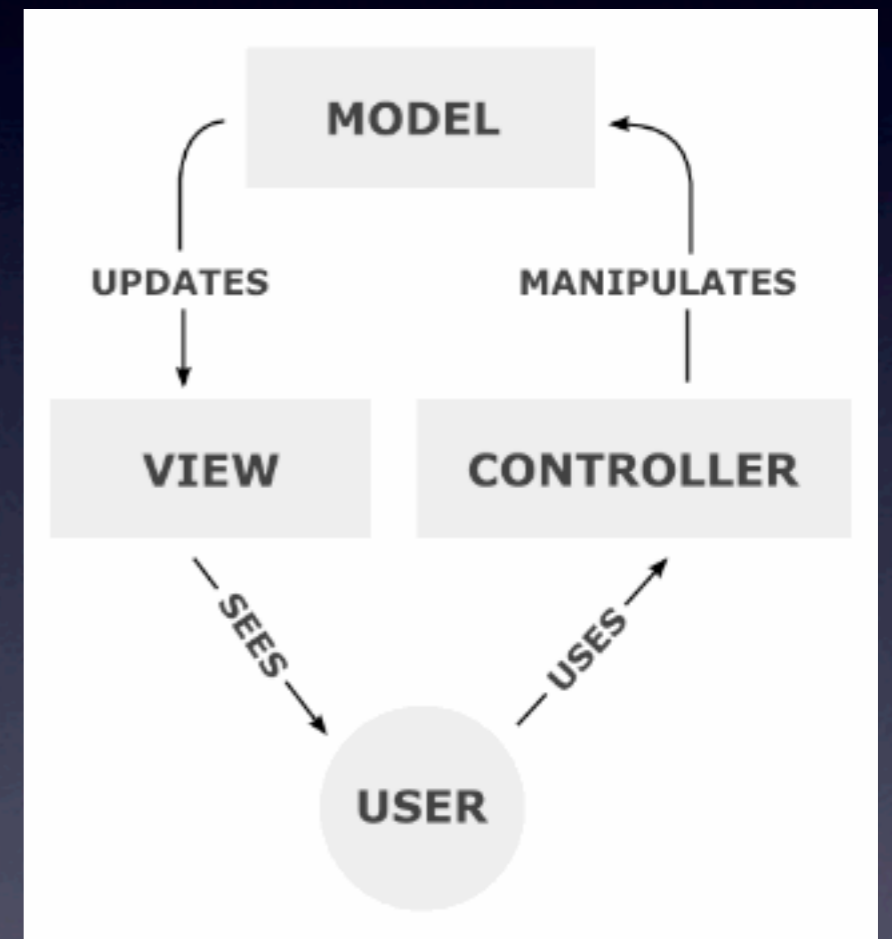
- J2EE (Java)
- .NET (C#)
- RoR (Ruby)
- Backbone.js (Javascript)
- CI, Cake, Yii (PHP)

MVC & Android

- Components
 - Controller == Activity
 - View == Layout
 - Model == 沒有！請自己來！其實也還好...

MVC & Android

- User uses Activity (Y)
- Controller manipulates model (partially Y)
- Model updates view (Nope!!, Activity updates view with model)
- View saw by user (Y)



MVC & Android

- Controller
 - **Activity** in Android. The core of Android application.
 - Activity is a Java class. Which means you can do almost **EVERYTHING** you want in it.
- View
 - **Layout** in Android. Define the Screen saw by user on device.
 - Layout is a XML file. You define it by text or visual editor.

Android & Eclipse

- 非比尋常的(除了Apple出品的...), Android的開發, 幾乎一定要綁定IDE
- 幾乎90%的Android都是以Eclipse開發
- Android SDK本身也提供了command line的SDK, 但幾乎沒人用(除了Geek & auto build之外)

Eclipse

- Eclipse是著名的跨平台的自由整合式開發環境（IDE）。最初主要用來Java語言開發，目前亦有人透過外掛程式使其作為C++、Python、PHP等其他語言的開發工具。
- Eclipse的本身只是一個框架平台，但是眾多外掛程式的支援，使得Eclipse擁有較佳的靈活性。

Eclipse

- Eclipse最初是由IBM公司開發的替代商業軟體Visual Age for Java的下一代IDE開發環境，2001年11月貢獻給開源社群，現在它由非營利軟體供應商聯盟Eclipse基金會（Eclipse Foundation）管理。
- Visual Age Studio當初是以Smalltalk開發的
~~

Application Meta-data

- Application meta-data defines some behaviors of application which is out of the scope of code
- Web meta-data is not explicit defined
 - <http://www.nexdoor.cc>, go to find `index.htm/index.html/index.php` as start page

Application Meta-data

- For android, the application meta-data is defined in `AndroidManifest.xml`
 - What Activity is the start one?
 - What permission to use? E.g. Internet access, camera, GPS, etc.
 - Minimum OS version?
 - etc.

Application Meta-data

- For android, the application meta-data is defined in `AndroidManifest.xml`
 - What Activity is the start one?
 - What permission to use? E.g. Internet access, camera, GPS, etc.
 - Minimum OS version?
 - etc.

Android 應用開發 入門實戰

總要先有開發環境！

環境建置

- Java - JDK
 - JDK 1.6 (Not JRE...)
 - <http://www.oracle.com/technetwork/java/javase/downloads/jdk6u37-downloads-1859587.html>
- Eclipse
 - Eclipse for Mobile Developers

環境建置

- Android Developer Tools (ADT)
 - Android SDK
 - <http://developer.android.com/sdk/index.html>
 - ADT plugins for Eclipse
 - <http://developer.android.com/sdk/installing/installing-adt.html>
- ADT is used to manage Android versions

環境建置

- Android Virtual Device (AVD)
 - Emulator of Android device
 - Create with AVD manager
 - Eclipse > Windows > AVD manager

環境建置 - 總結

- Java - base of android development.
 - JDK
- Eclipse - must have IDE for Android
 - Eclipse for mobile developer
- ADT - must have SDK for Android
- AVD - Android Virtual Device

一切從 Hello World 開始

Hello Android

- 建立新的 Android 專案
 - [File] -> [New] -> [Android Project]
設定專案名稱, 目標版本, 封裝名稱...
 - 若要使用到 Google Maps 請選擇
對應版本的 Google API

New Android Project

Create Android Project

Select project name and type of project

Project Name:

Create new project in workspace
 Create project from existing source
 Create project from existing sample

Use default location

Location:

Working sets

Add project to working sets

Working sets:

New Android Project

Select Build Target

Choose an SDK to target

Build Target

Target Name	Vendor	Platform	API Level
<input type="checkbox"/> Android 1.6	Android Open Source Project	1.6	4
<input type="checkbox"/> Google APIs	Google Inc.	1.6	4
<input type="checkbox"/> Android 2.1	Android Open Source Project	2.1	7
<input type="checkbox"/> Google APIs	Google Inc.	2.1	7
<input type="checkbox"/> Android 2.2	Android Open Source Project	2.2	8
<input checked="" type="checkbox"/> Google APIs	Google Inc.	2.2	8
<input type="checkbox"/> Android 2.3.3	Android Open Source Project	2.3.3	10
<input type="checkbox"/> Google APIs	Google Inc.	2.3.3	10
<input type="checkbox"/> Intel Atom x86 System...	Intel Corporation	2.3.3	10
<input type="checkbox"/> Android 4.0.3	Android Open Source Project	4.0.3	15
<input type="checkbox"/> Google APIs	Google Inc.	4.0.3	15

Android + Google APIs

New Android Project

Application Info

Configure the new Android Project

Application Name:

Package Name:

Create Activity:

Minimum SDK:

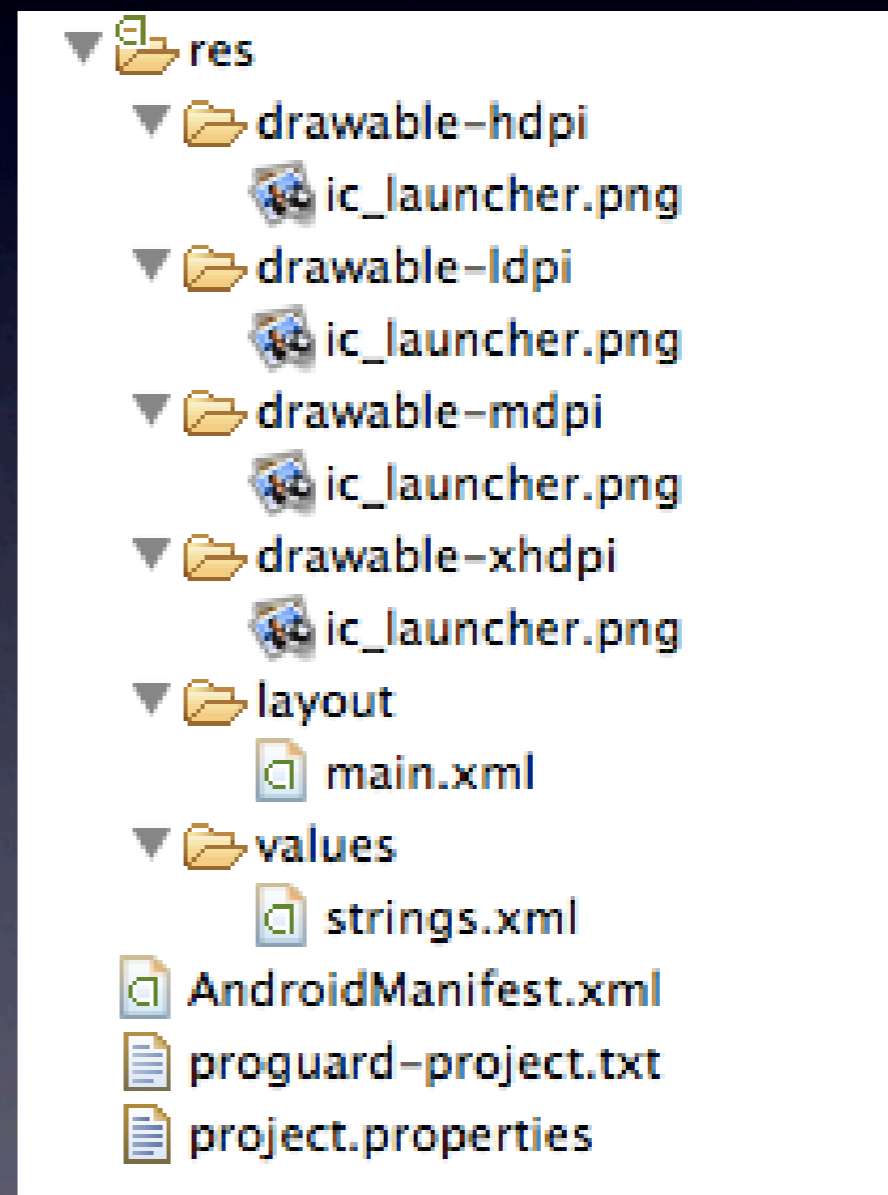
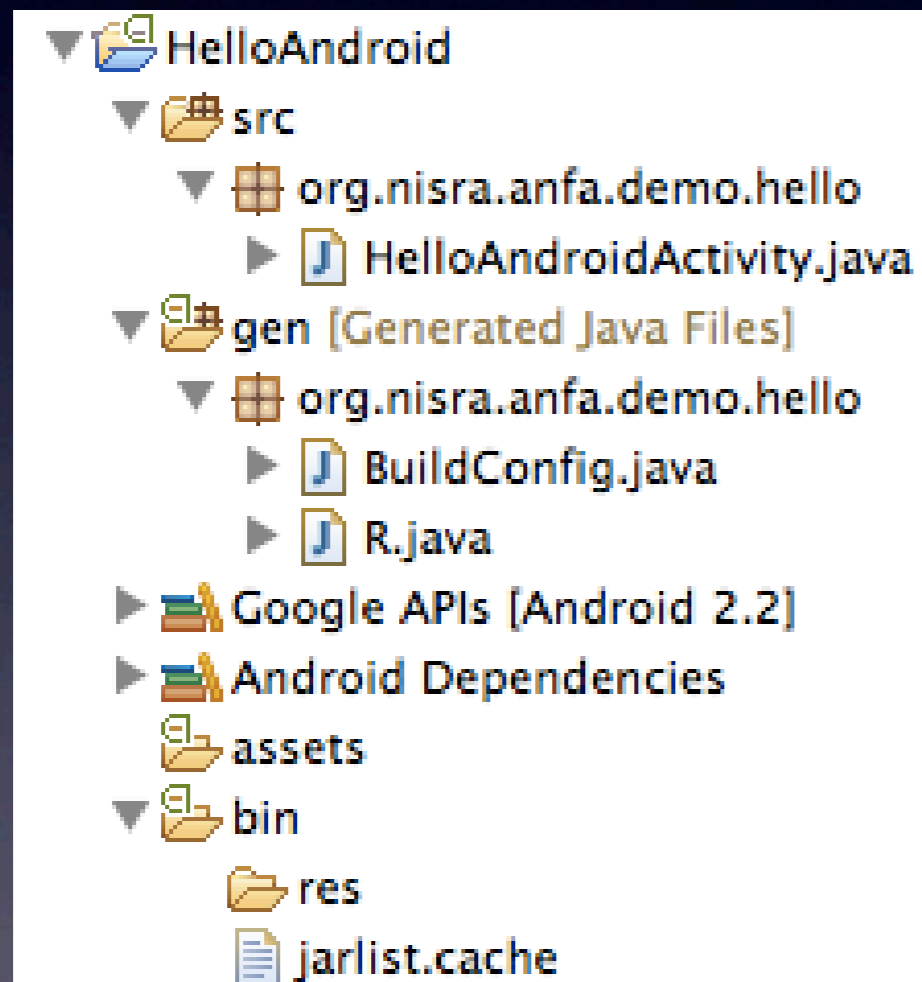
Create a Test Project

Test Project Name:

Test Application:

Test Package:

專案結構

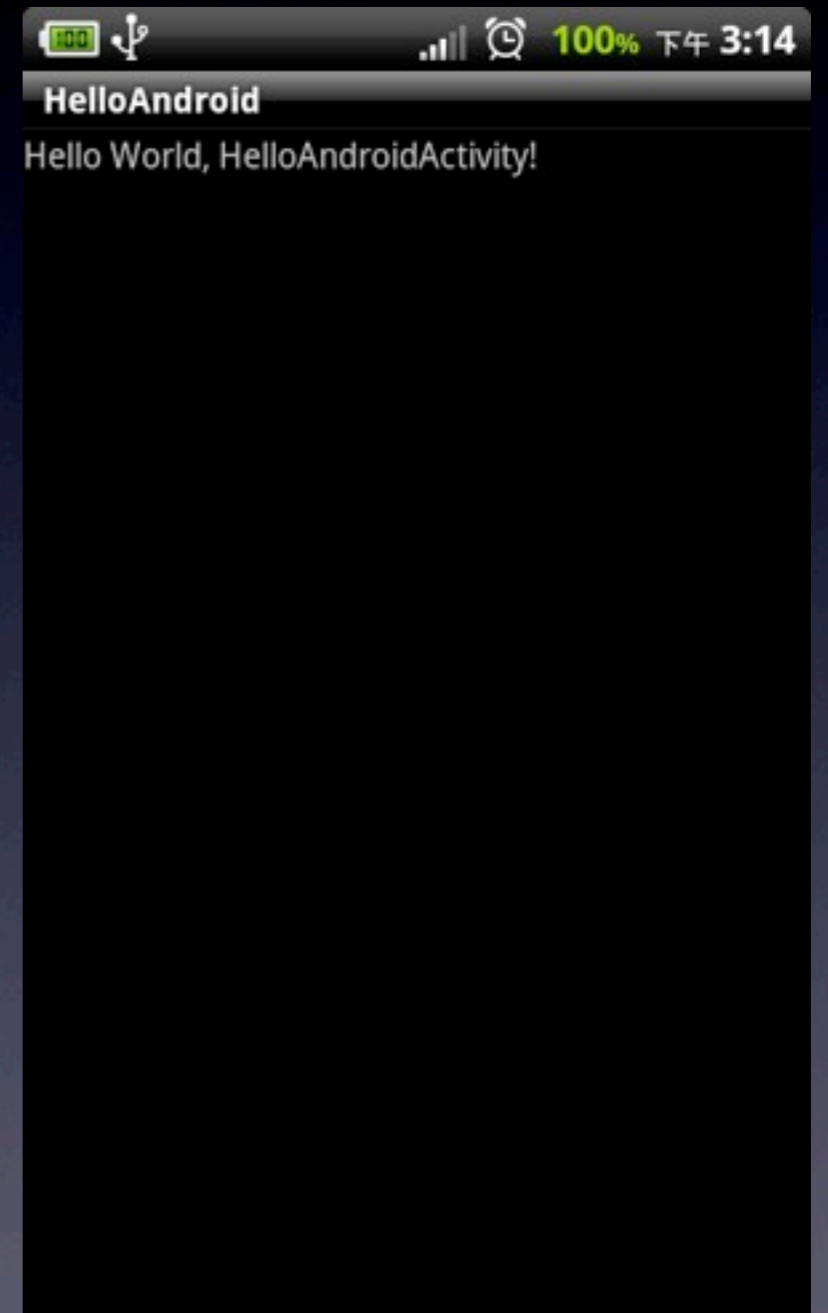


HelloAndroidActivity.java

```
package org.nisra.anfa.demo.hello;

import android.app.Activity;
import android.os.Bundle;

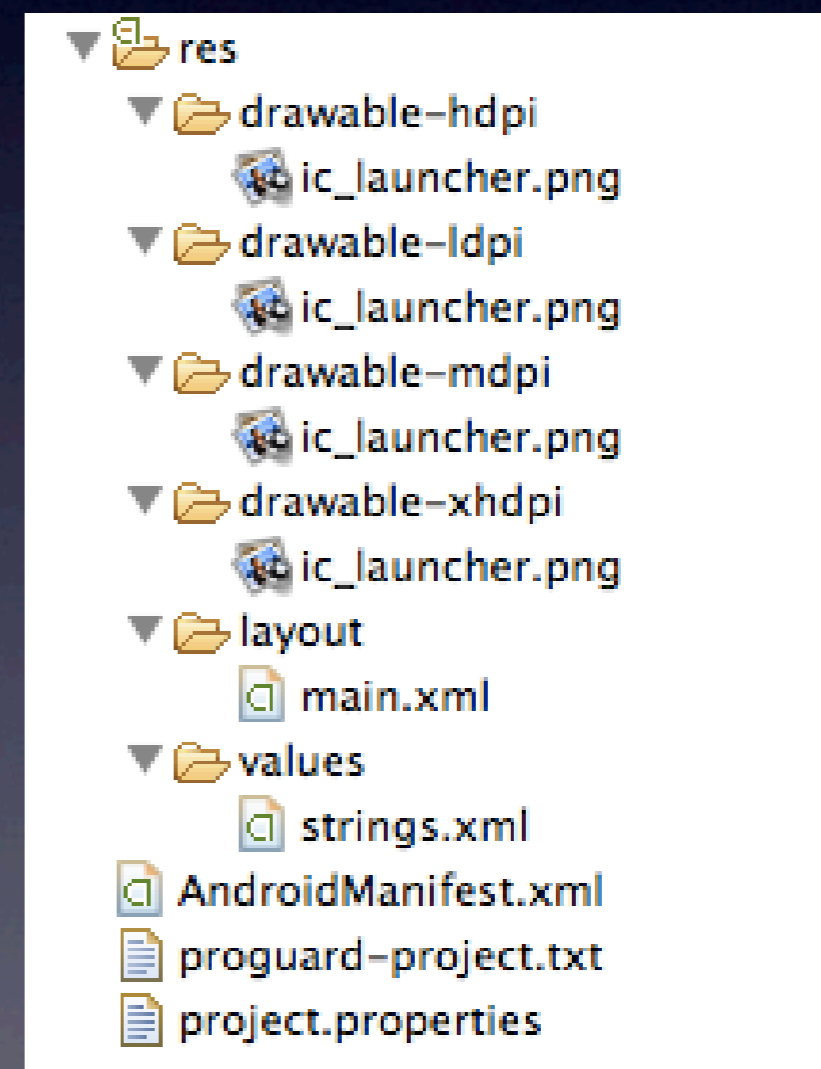
public class HelloAndroidActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```



概念導入 - Resource

- 對資源性質的檔案集中管理

type	path
Animation	res/anim/ res/drawable/
Color State List	res/color/
Drawable	res/drawable/
Layout	res/layout/
Menu	res/menu/
String	res/values/
Style	res/values/
Raw	res/raw/
XML	res/xml/
Other	res/values/



概念導入 - Resource

於程式中引用 Resource:

```
[<package_name>.]R.<resource_type>.<resource.name>
```

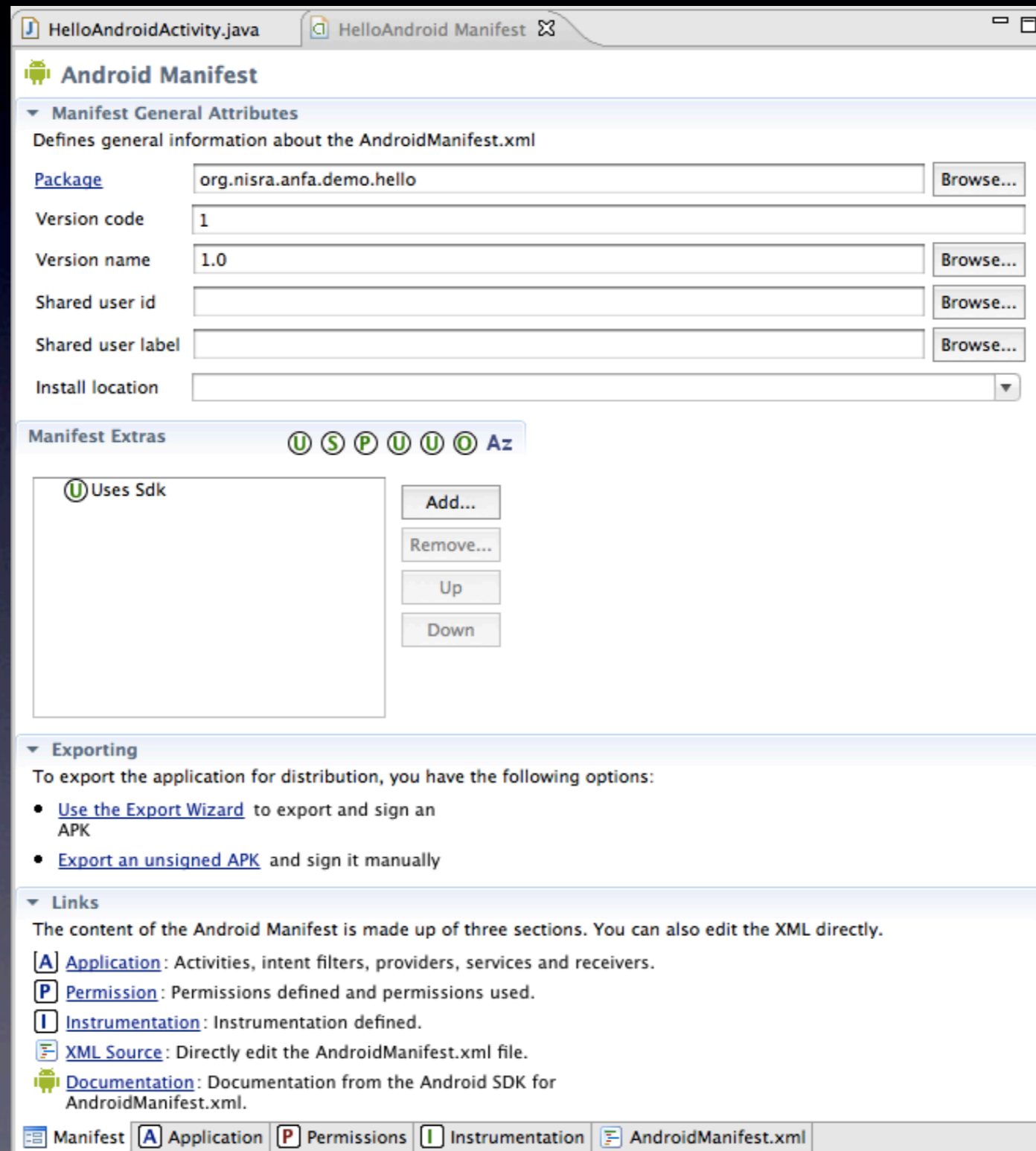
```
R.string.hello
```

於XML中引用 Resource:

```
@ [<package_name>:]<resource_type>/<resource.name>
```

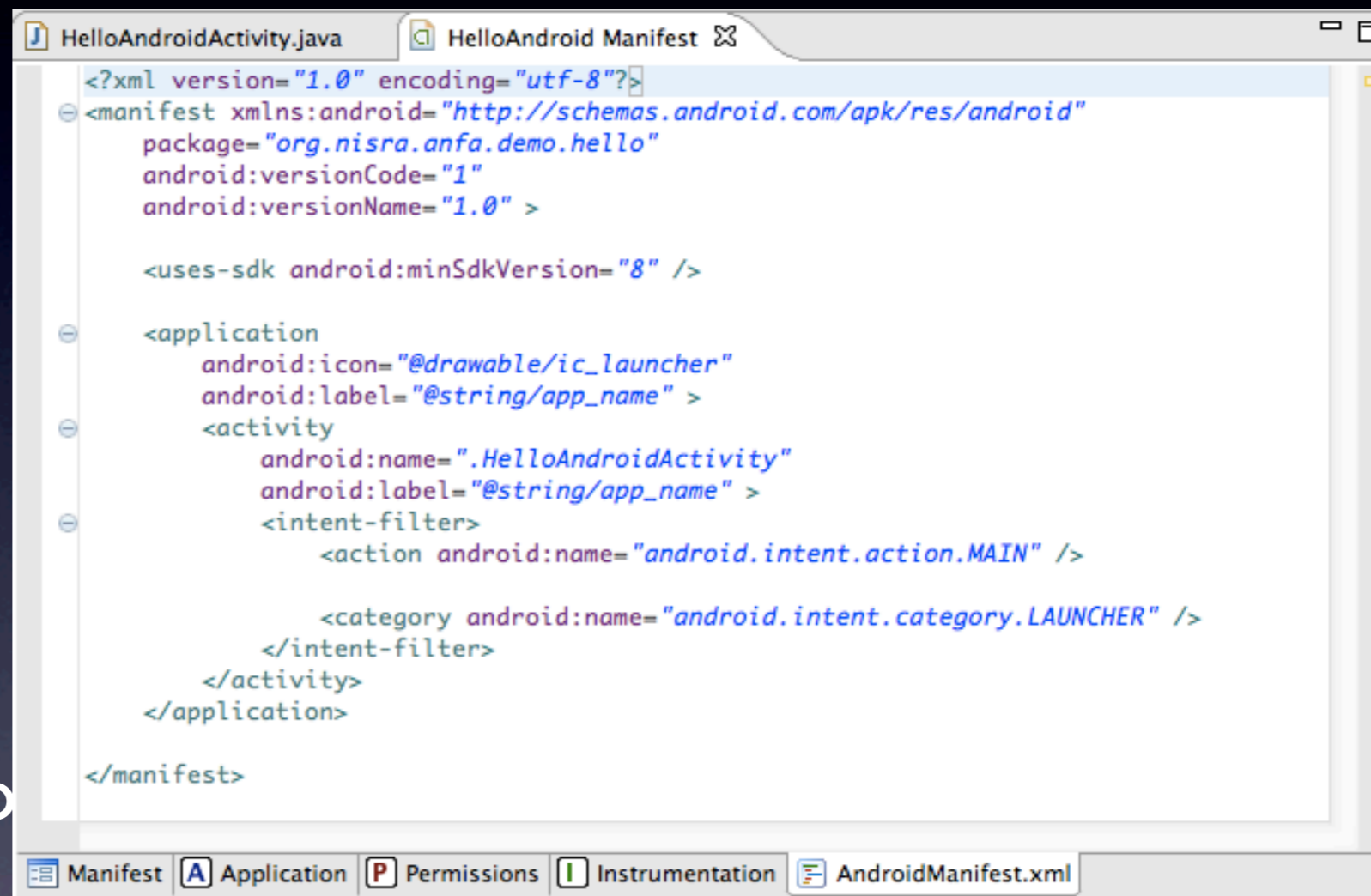
```
@string/hello
```

概念導入 - Manifest



概念導入 - Manifest

- 應用定義檔
 - 封裝名稱 (Unique)
 - 元件 (ex. Activity)
 - 意圖
 - 權限 (ex. Internet)
 - 函式庫連結 (ex. Google)
 - ...



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="org.nisra.anfa.demo.hello"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk android:minSdkVersion="8" />

    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name" >
        <activity
            android:name=".HelloAndroidActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

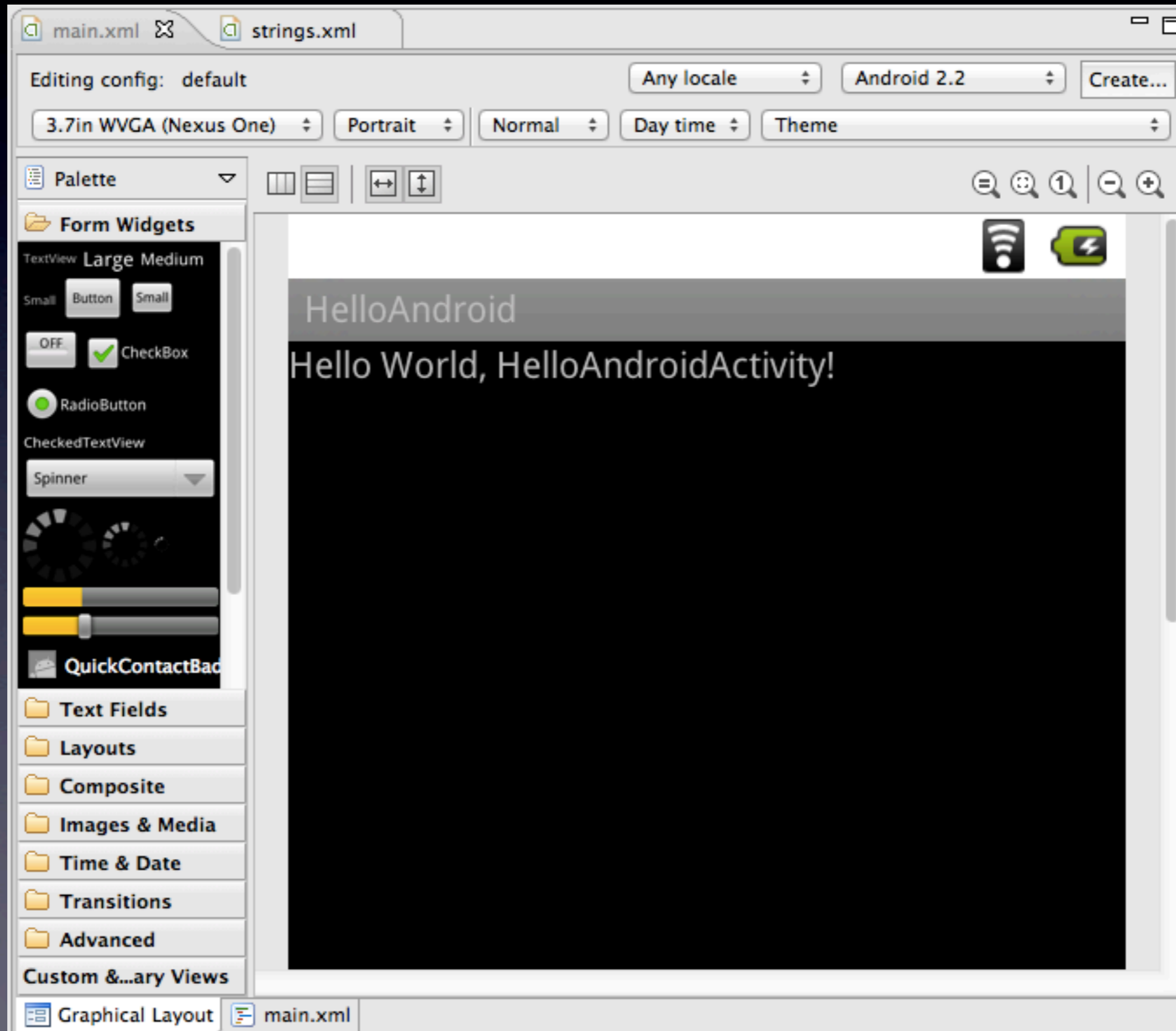
res/layout/main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/
res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/hello" />

</LinearLayout>
```

res/layout/main.xml

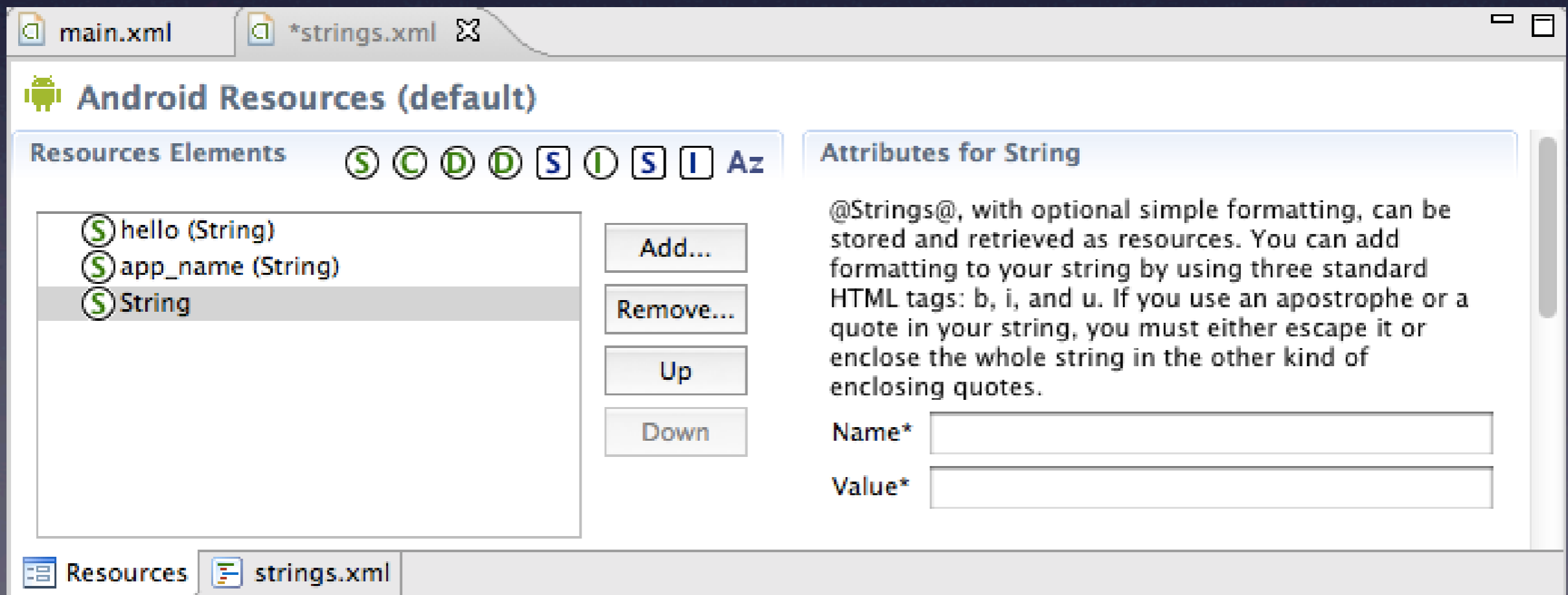


res/values/strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

  <string name="hello">Hello World, HelloAndroidActivity!</string>
  <string name="app_name">HelloAndroid</string>

</resources>
```



The screenshot shows the Android Studio IDE with the 'Android Resources (default)' editor open. The 'Resources Elements' tab is active, displaying a list of string resources: 'hello (String)', 'app_name (String)', and 'String'. The 'String' resource is selected. To the right of the list are buttons for 'Add...', 'Remove...', 'Up', and 'Down'. The 'Attributes for String' tab is also visible, providing documentation for the @Strings@ annotation and fields for 'Name*' and 'Value*'. The bottom of the IDE shows the 'Resources' and 'strings.xml' tabs.

回頭再看 HelloAndroidActivity

```
package org.nisra.anfa.demo.hello;

import android.app.Activity;
import android.os.Bundle;

public class HelloAndroidActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string name="hello">Hello World, HelloAndroidActivity!</string>
    <string name="app_name">HelloAndroid</string>

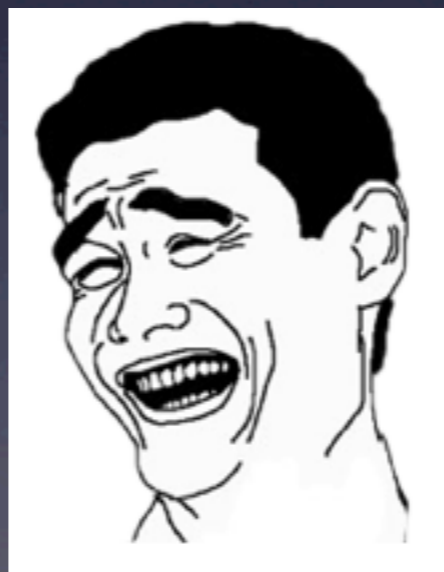
</resources>
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/hello" />

</LinearLayout>
```

恭喜您學會
寫 Hello World 了



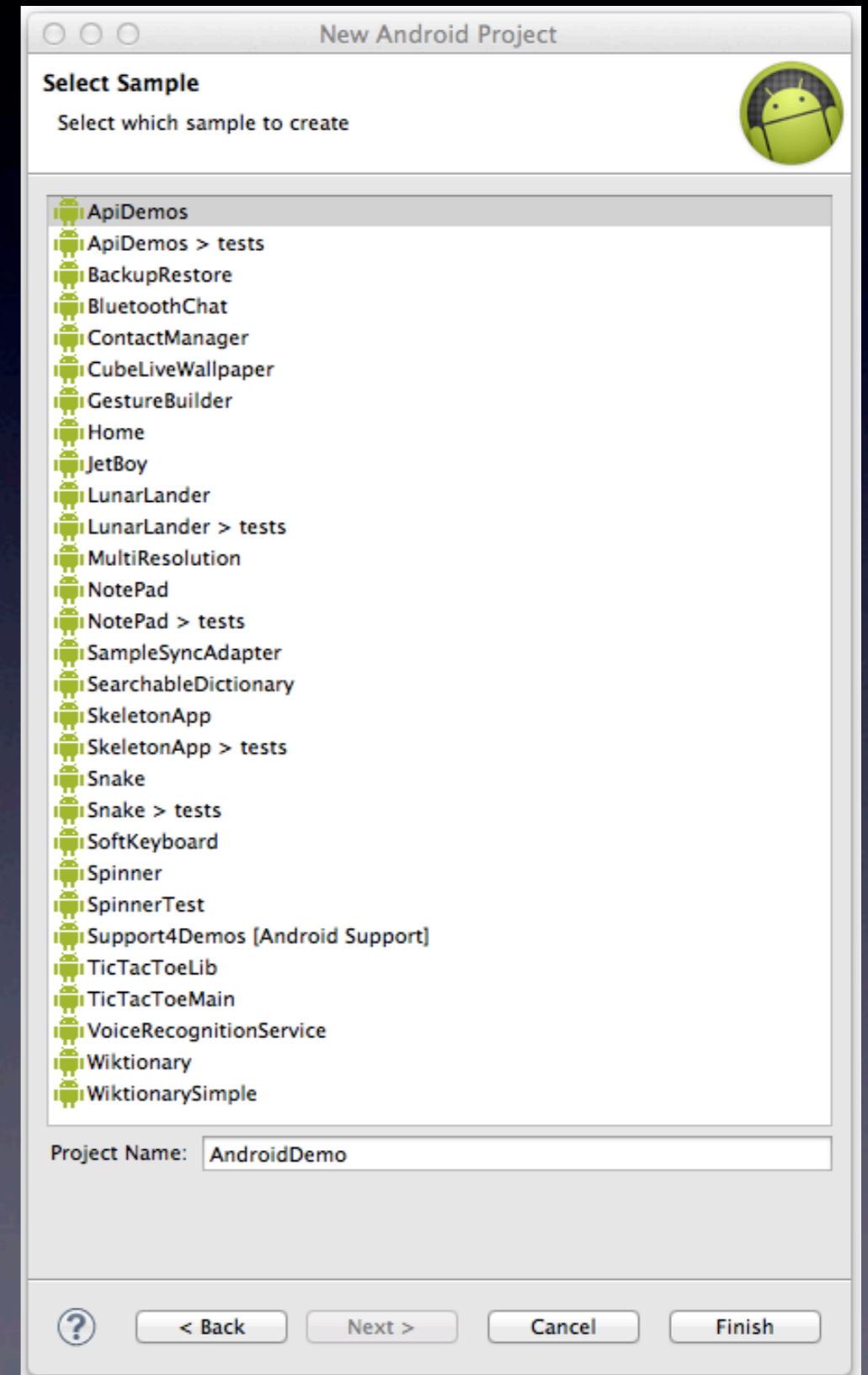
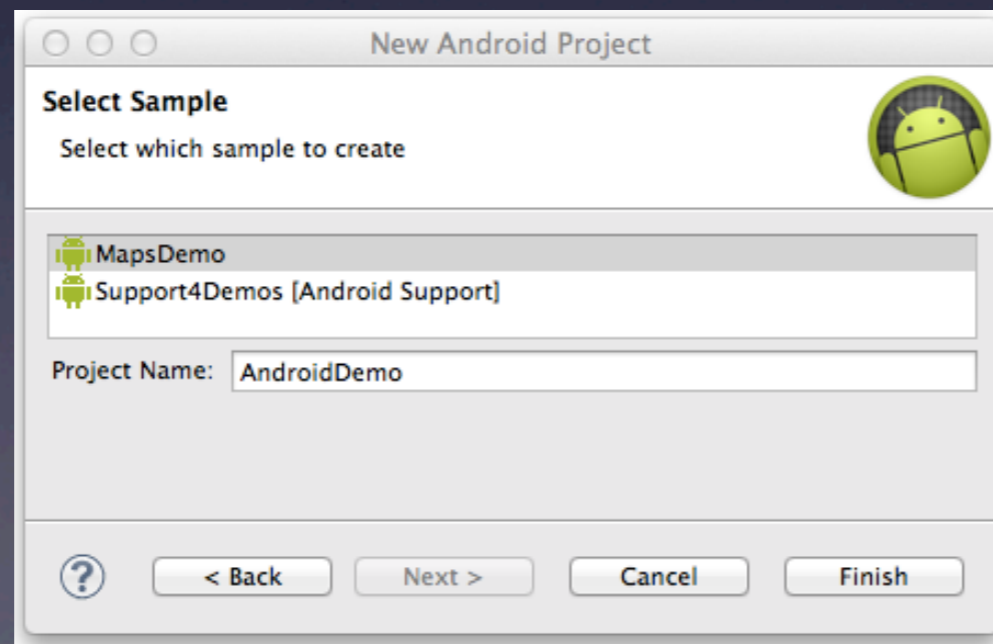
累了，
來看看別人寫了什麼...

一直
土法鍊鋼 閉門造車
也不是辦法

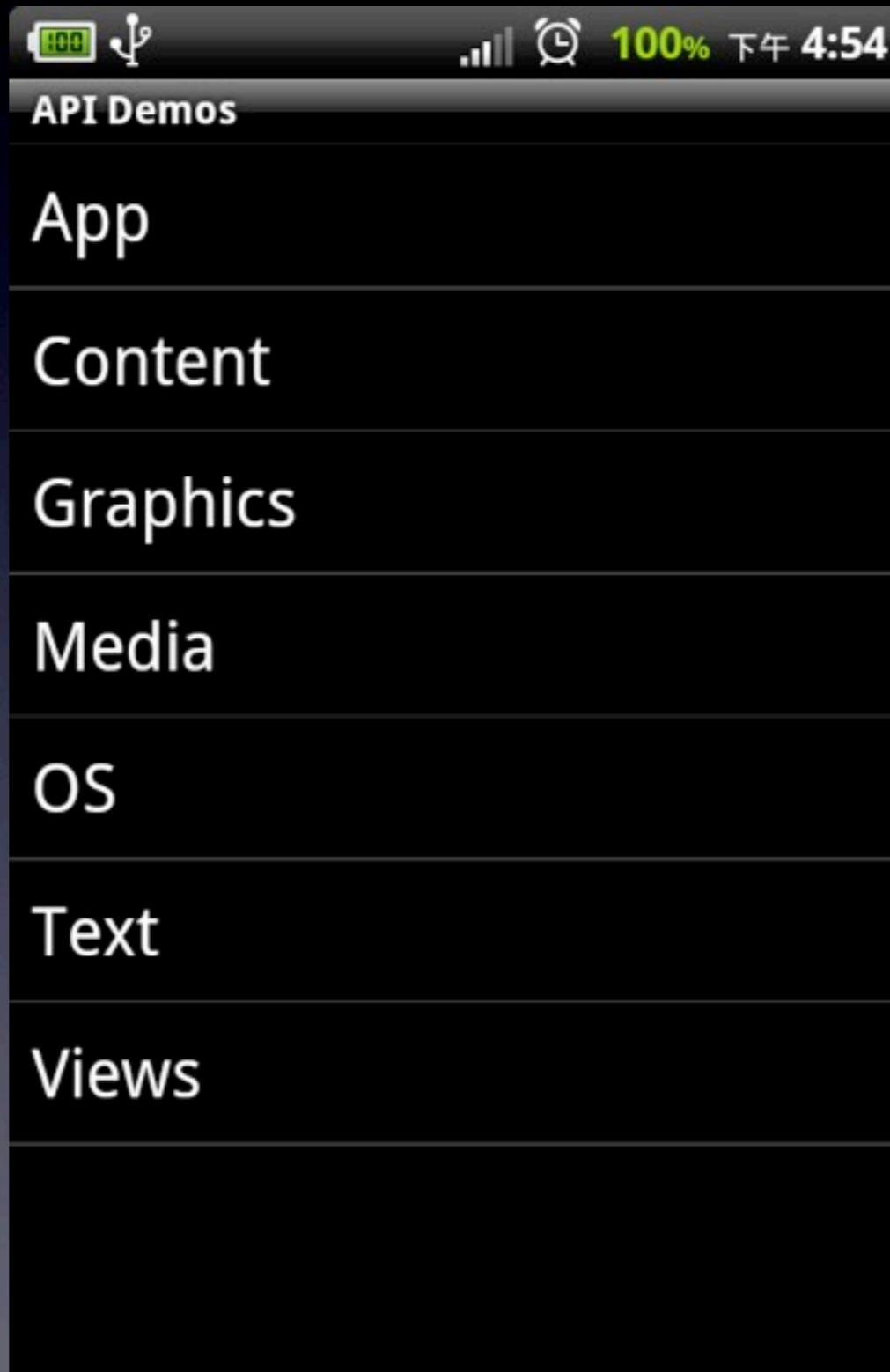
P.S. 不是說你可以拿同學的作業來抄

Official Sample Code

以範例程式作為
專案的基礎



Official Sample Code



CommonsWare Android Components



COMMONSWARE

Books

Services

Training

Warescription

CommonsWare Android Components (CWAC)

The CommonsWare Android Components, or CWAC, are open source libraries to help solve various tactical problems with Android development. Each CWAC component is packaged as a tiny JAR file that you can add to your project (e.g., drop it in libs/), requiring at most other CWAC JARs as dependencies.

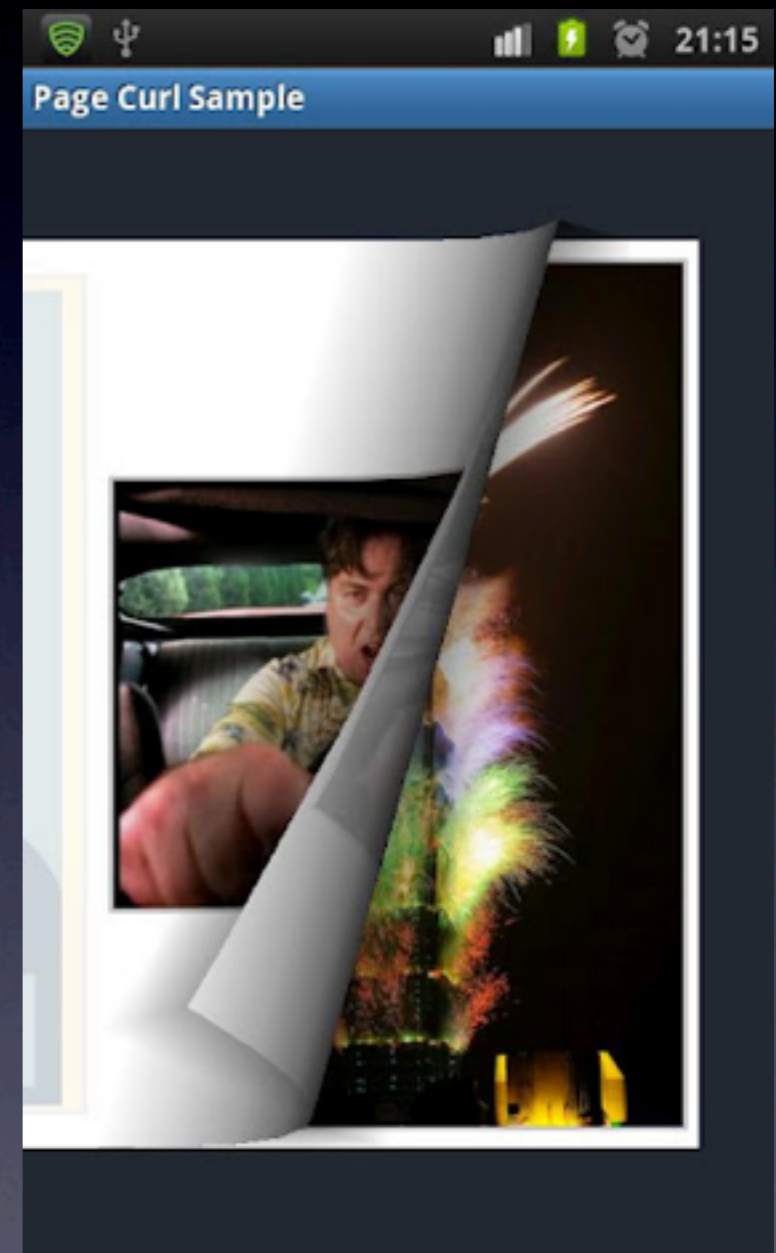
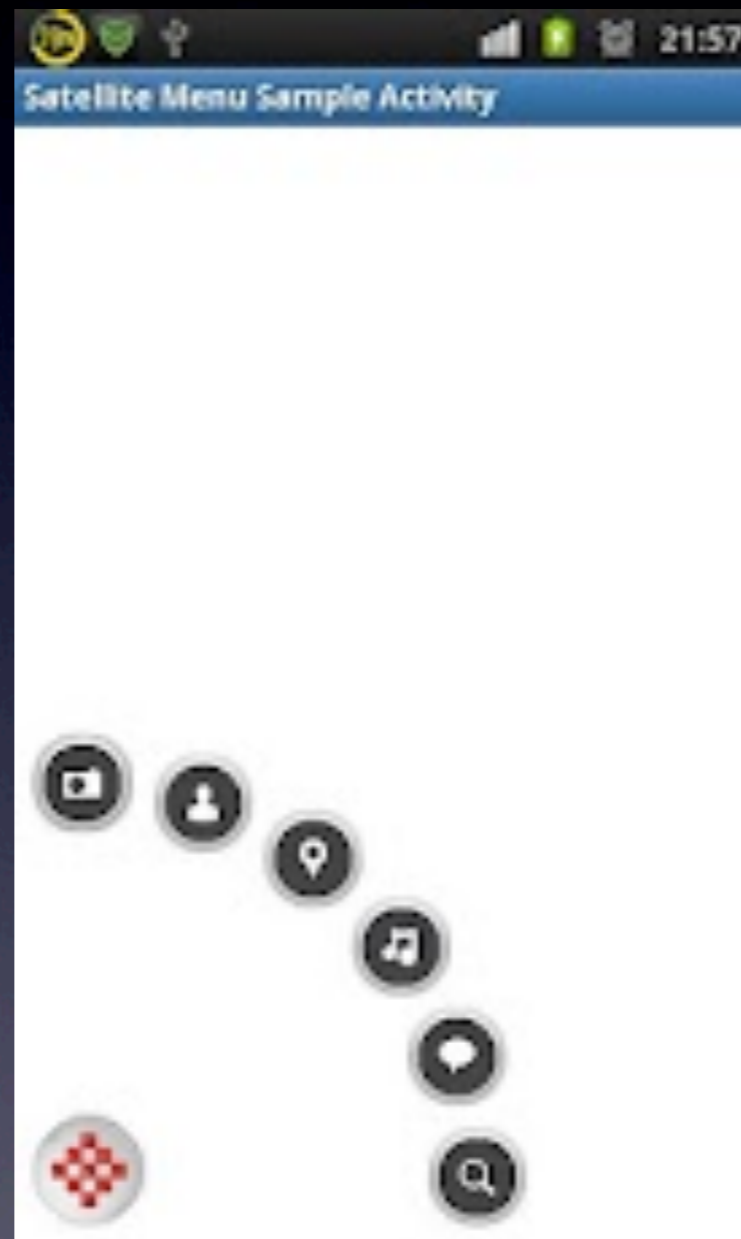
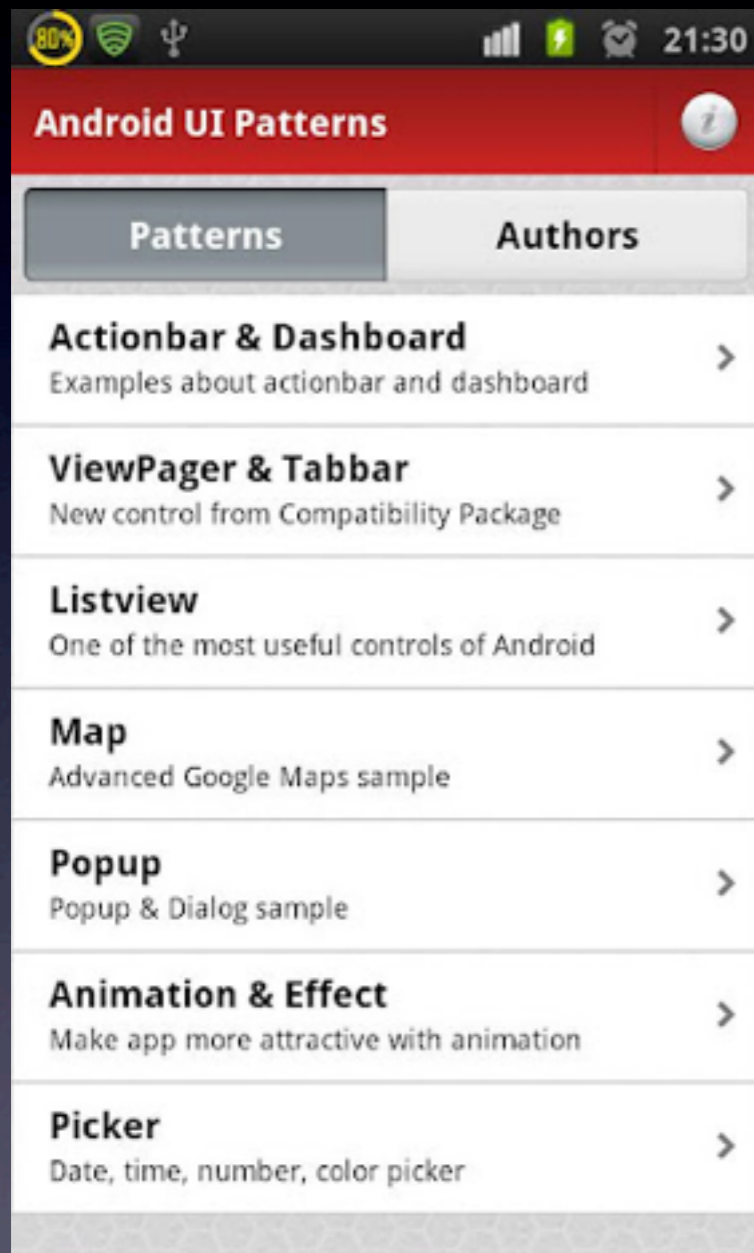
The current supported CWAC components, and their github repositories, are:

- [cwac-adapter](#): Provides an AdapterWrapper, a simple wrapper class that, by default, delegates all ListAdapter methods to a wrapped ListAdapter. The idea is that you can extend AdapterWrapper and only override certain ListAdapter methods, with the rest handled via the wrapped adapter.
- [cwac-anddown](#): Provides support for Markdown syntax in Android, such as a JNI wrapper around the sundown Markdown to HTML converter.
- [cwac-colormixer](#): Provides a simple color mixer widget, dialog, activity, and preference.
- [cwac-endless](#): Provides the EndlessAdapter, a wrapper for an existing ListAdapter that adds

<http://commonsware.com/cwac>

<https://github.com/commonsguy>

Android UI Patterns



<https://play.google.com/store/apps/details?id=com.groidify.uipatterns>

GreenDroid

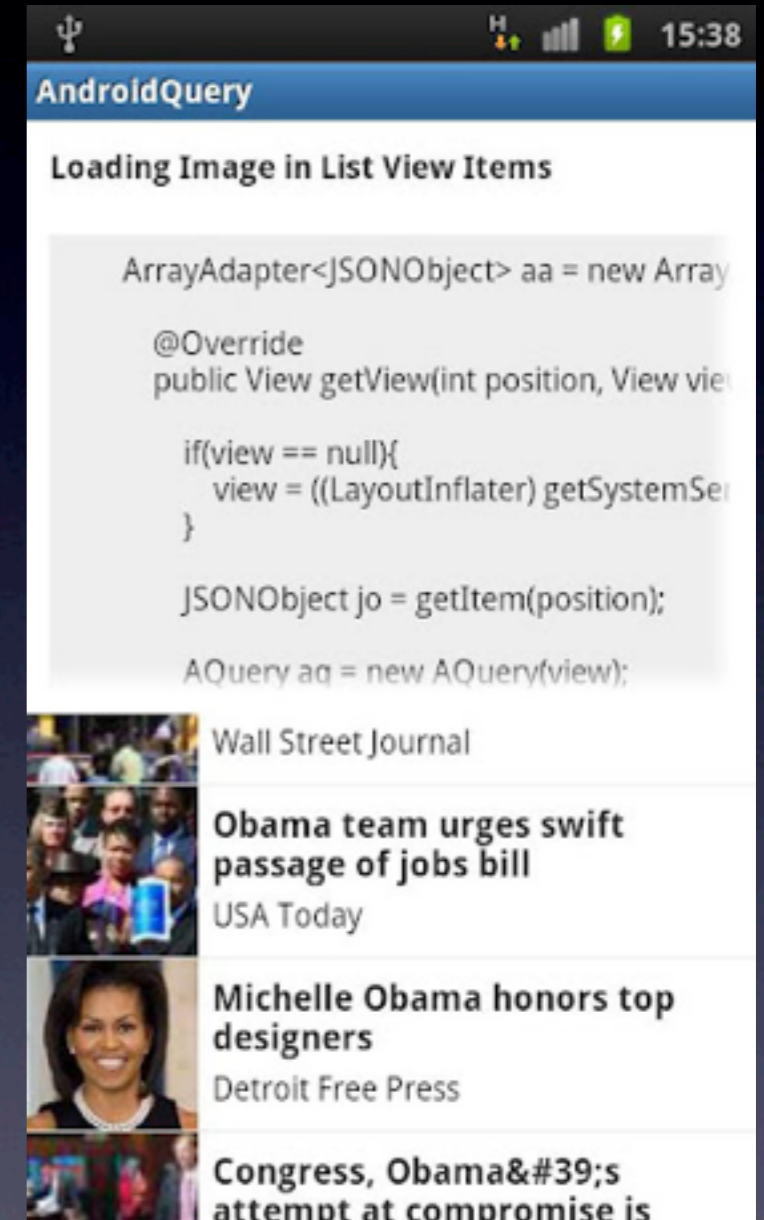
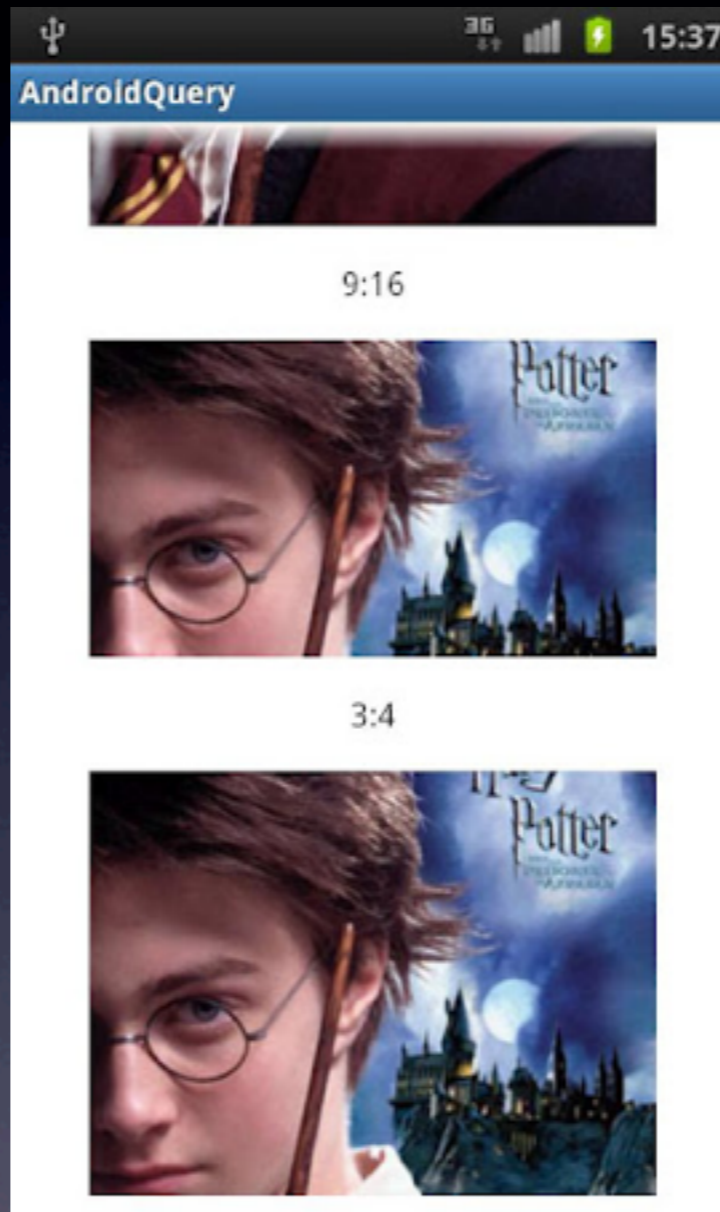


Keep in mind this is still a work in progress. There are thousands of things to do and I hope I'll have enough time to develop all of the features I have in mind :)



<https://github.com/cyrilmottier/GreenDroid>
<https://play.google.com/store/apps/details?id=com.cyrilmottier.android.gdcatalog>

Android Query



<http://code.google.com/p/android-query/>
<https://play.google.com/store/apps/details?id=com.androidquery>

ActionBarSherlock

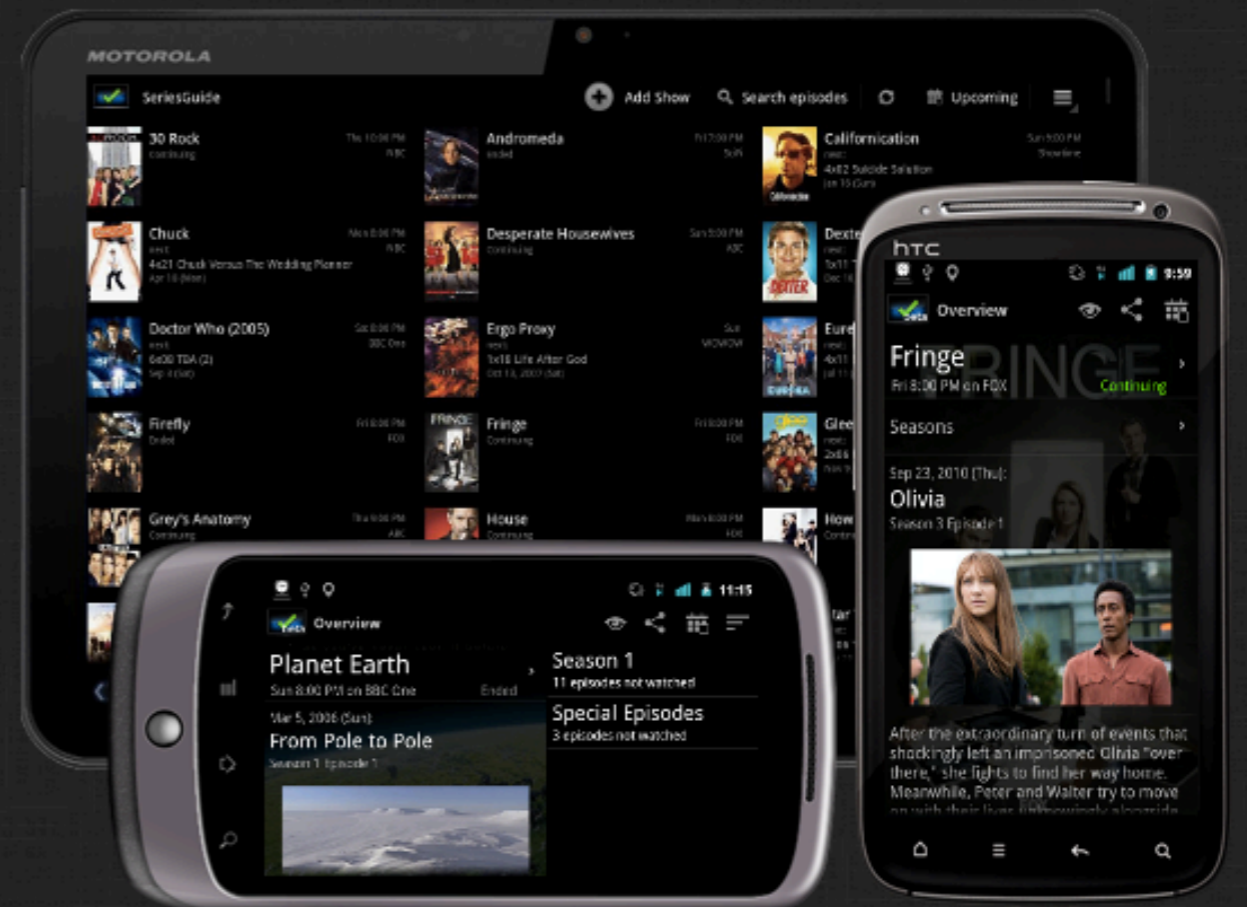
ActionBarSherlock

ActionBarSherlock is an extension of the **compatibility library** designed to facilitate the use of the **action bar design pattern** across all versions of **Android** with a **single API**.



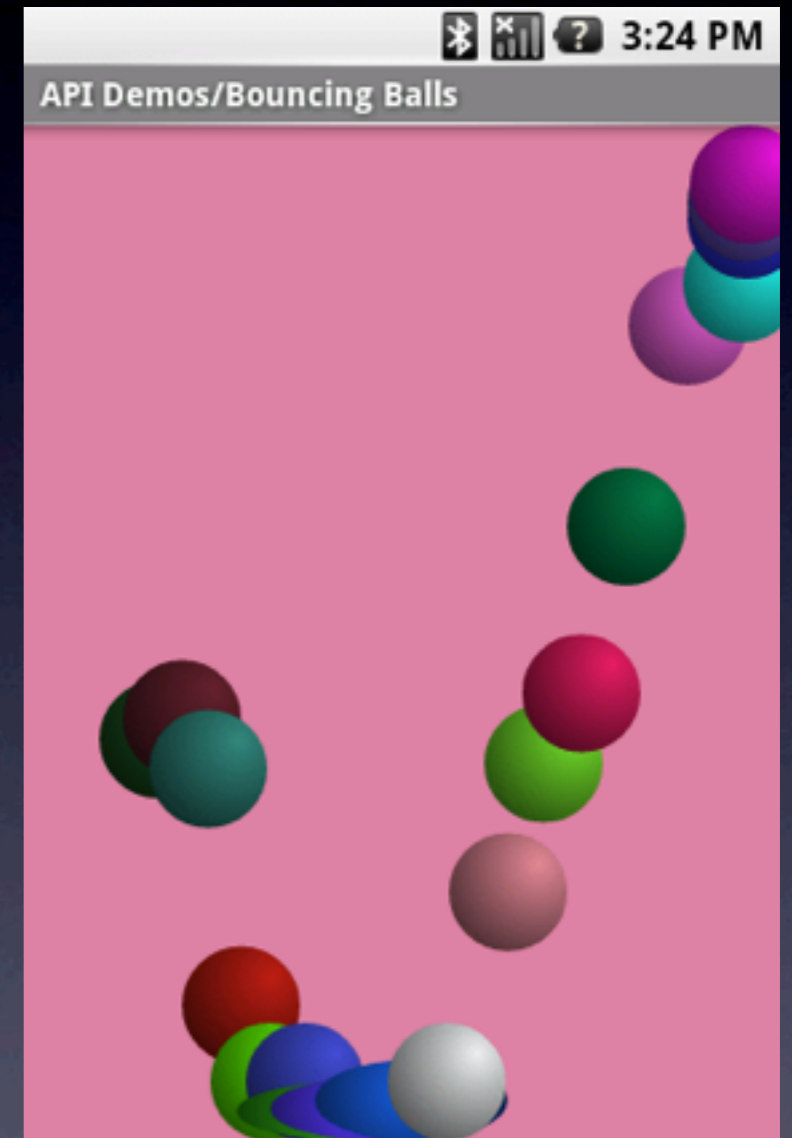
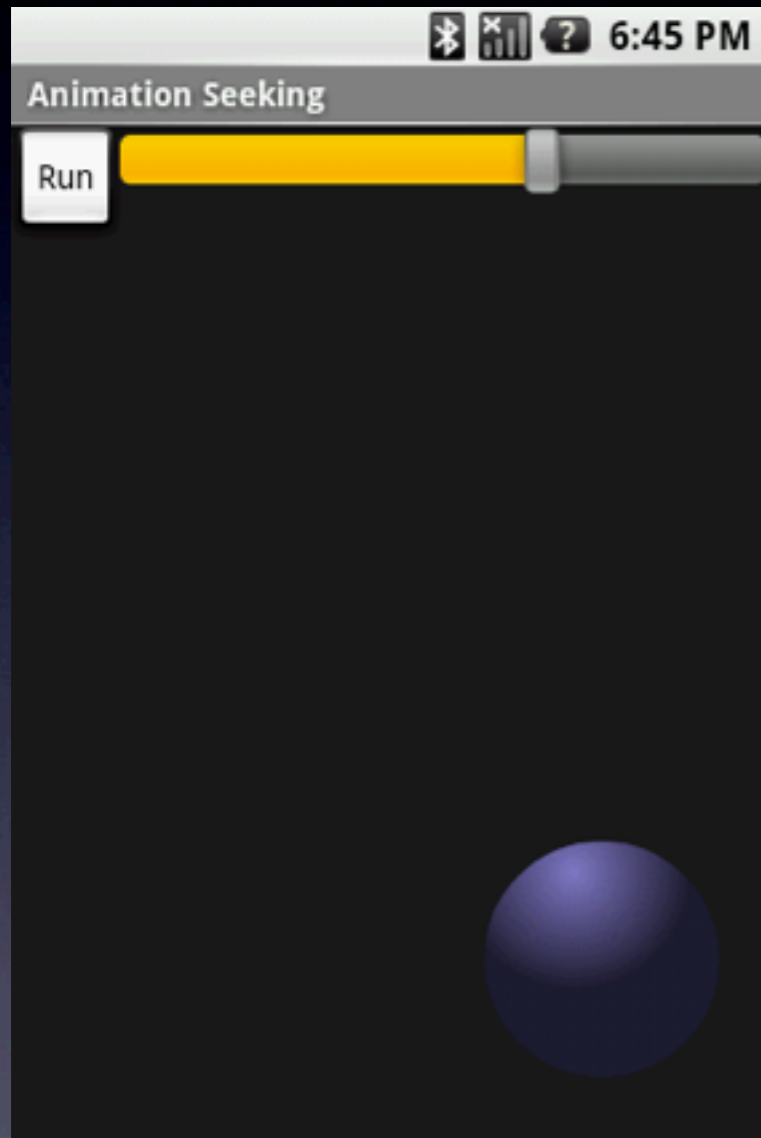
The library will automatically use the native action bar when appropriate or will automatically wrap a custom implementation around your layouts. This allows you to easily develop an application with an action bar for every version of Android from 2.x and up.

Download v4.0.2 : [Zip](#) [Tarball](#) [More...](#)



<http://actionbarsherlock.com/>

Nine Old Androids



<https://play.google.com/store/apps/details?id=com.jakewharton.nineoldandroids.sample>

Life cycle

Android Activity Lifecycle

