

# Android App

程式設計教本  
之無痛起步

引領入門

最簡單、最易懂的初學教材



## 第 13 章

# WebView 與 SharedPreferences

本投影片（下稱教用資源）僅授權給採用教用資源相關之旗標書籍為教科書之授課老師（下稱老師）專用，老師為教學使用之目的，得摘錄、編輯、重製教用資源（但使用量不得超過各該教用資源內容之80%）以製作為輔助教學之教學投影片，並於授課時搭配旗標書籍公開播放，但不得為網際網路公開傳輸之遠距教學、網路教學等之使用；除此之外，老師不得再授權予任何第三人使用，並不得將依此授權所製作之教學投影片之相關著作物移作他用。

# 前言

- 13-1 使用 WebView 顯示網頁
- 13-2 改進 WebView 功能
- 13-3 使用 Preferences (偏好設定) 記錄資訊

# 13-1 使用 WebView 顯示網頁

## • 範例13-1：顯示旗標網站

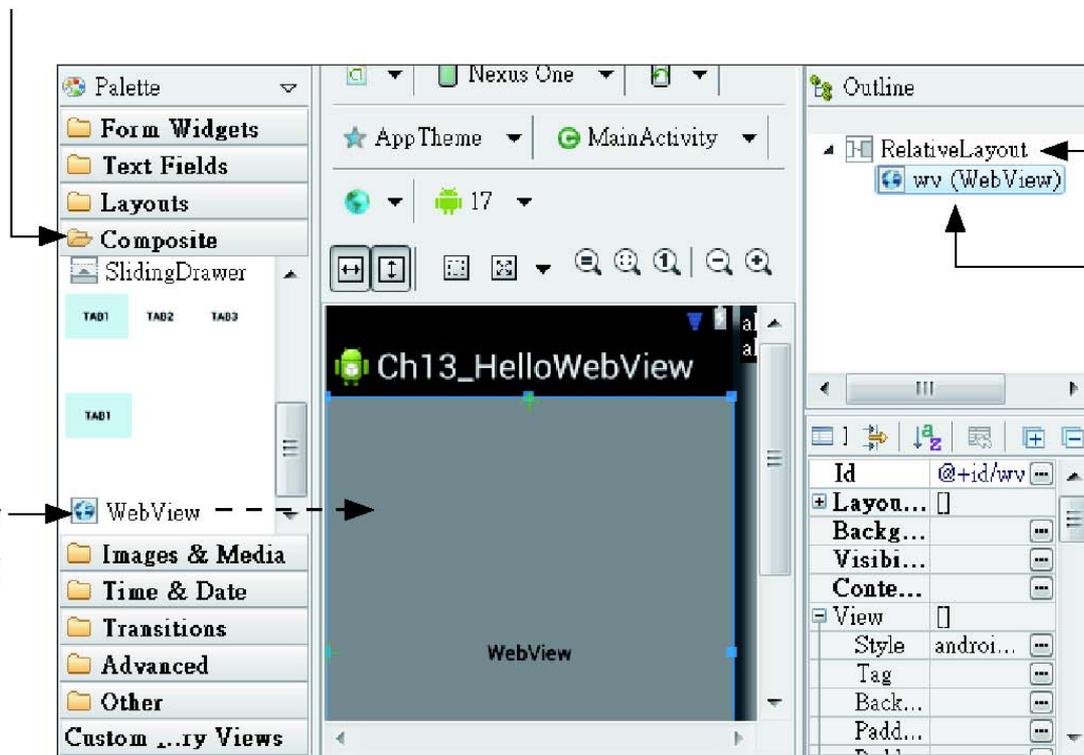
### step 1

2 展開 Composite 項目

1 清除 RelativeLayout 中預建的 TextView

3 將 WebView 元件拉曳到佈局的左上角

4 設定 Id 為 "wv" 並按 **Ctrl + S** 存檔



# 顯示旗標網站

## step 2

```
01  protected void onCreate(Bundle savedInstanceState) {  
02      super.onCreate(savedInstanceState);  
03      setContentView(R.layout.activity_main);  
04  
05      WebView wv = (WebView) findViewById(R.id.wv); ← 取得 WebView 元件  
06      wv.loadUrl("http://www.flag.com.tw"); ← 連到旗標網站  
07  }
```

# 顯示旗標網站

## step 3

3 按 Add 鈕

1 雙按開啟檔案

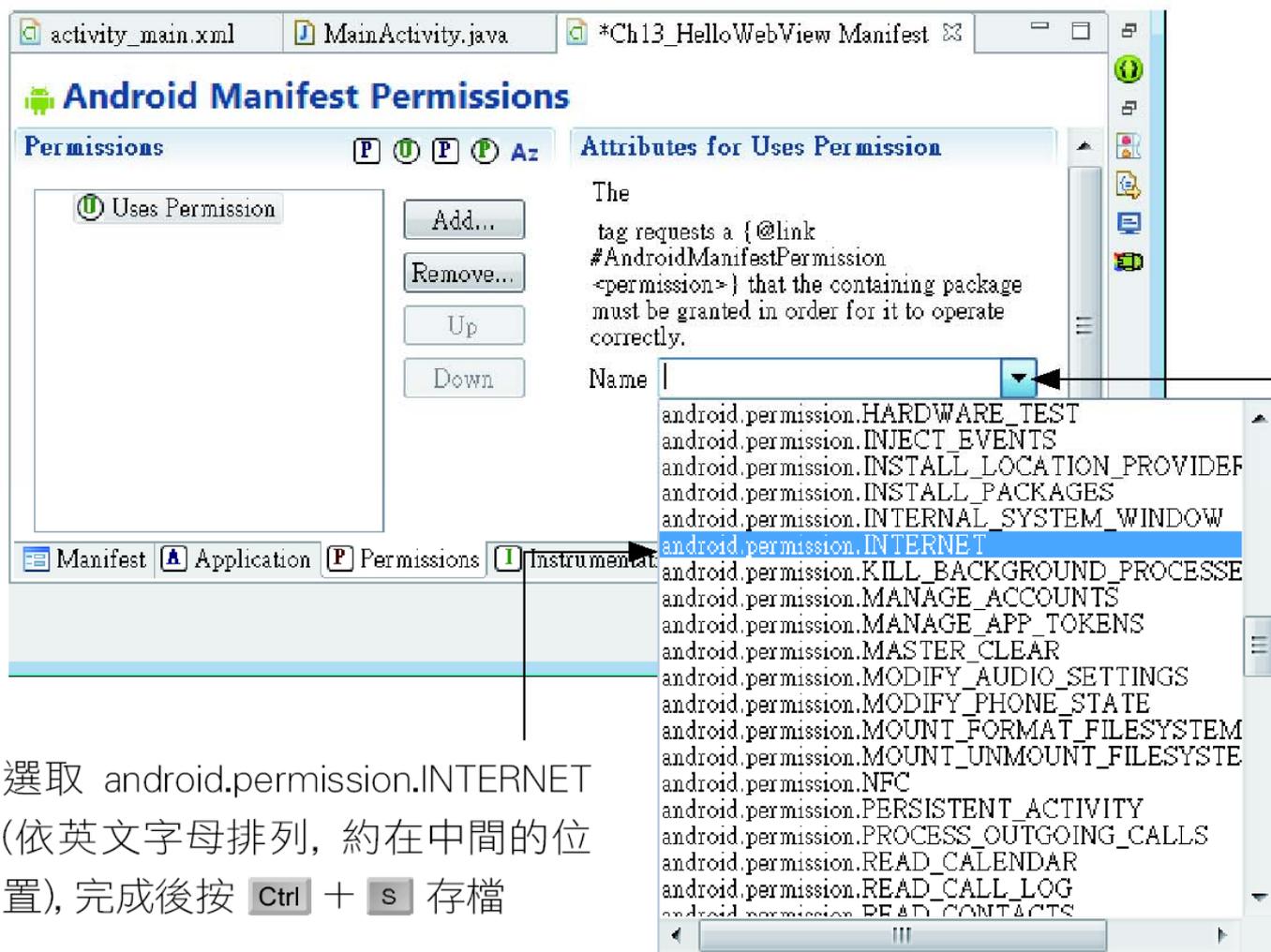
2 切換到 **Permission** 頁次

# 顯示旗標網站



4 雙按 Uses Permission 項目

# 顯示旗標網站



5 按向下鈕展開選單

6 選取 android.permission.INTERNET (依英文字母排列, 約在中間的位置), 完成後按 **Ctrl** + **S** 存檔

# 顯示旗標網站

## step 4



不像瀏覽器會顯示載入網頁的進度，剛開始只有空白，稍等一下（視網路連線的速度）才會出現網頁



# 顯示旗標網站

預設就是以 100% 比例  
(『正常』比例) 顯示網  
頁, 且無法縮放



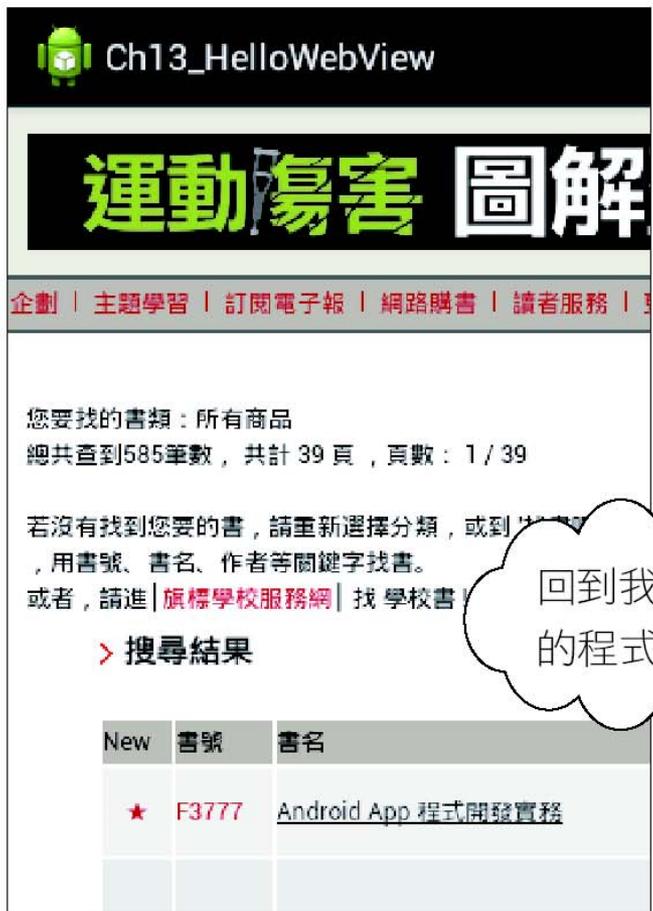
1 先捲動到畫面  
『最右側』

2 按網頁中的 "Go" 鈕



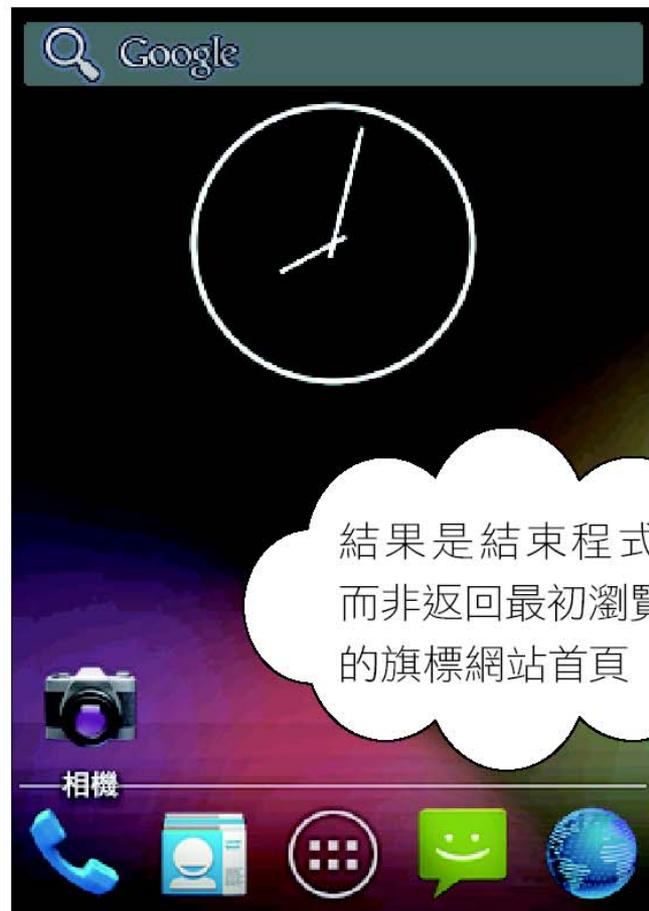


# 顯示旗標網站



回到我們的程式了

5 再按 1 次  
『返回』鍵



## 13-2 改進 WebView 功能

- WebSettings：用於控制 WebView 的基本設定。
- WebViewClient：用於控制WebView 本身的行為，透過此類別物件，可取得網頁相關事件。
- WebChromeClient：用於製作和網頁有關、但屬於WebView 『之外』的效果。

# 改進 WebView 功能

- 使用 WebSettings 啟用網頁縮放及 JavaScript
- 使用 WebViewClient 處理開啟超連結動作
- 使用 WebChromeClient 建立網頁載入進度介面
- 使用 Activity 內建功能顯示網頁載入進度
- 使用 onKeyDown() 實作回上一頁功能

# 使用 WebSettings 啟用網頁縮放及 JavaScript

```
WebView wv = (WebView) findViewById(R.id.wv);  
wv.getSettings().setBuiltInZoomControls(true); ← 啟用縮放功能  
wv.getSettings().setJavaScriptEnabled(true); ← 啟用 JavaScript
```

# 使用 **WebViewClient** 處理開啟超連結 動作

```
// 建立 WebViewClient 物件並設定給 WebView  
wv.setWebViewClient(new WebViewClient() );
```

# 使用 WebChromeClient 建立網頁載入進度介面

```
wv.setWebViewClient(new WebViewClient(){
    ... ← 改寫 WebViewClient 的方法
});

wv.setWebChromeClient(new WebChromeClient(){
    ... ← 改寫 WebChromeClient 的方法
});
```

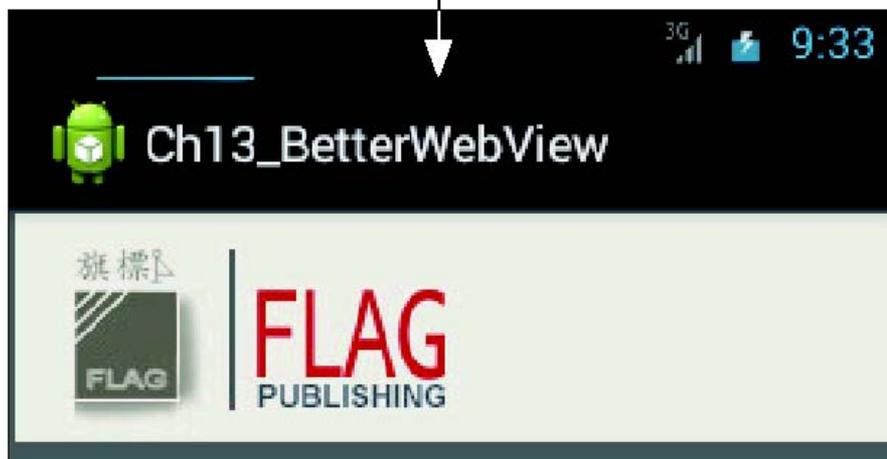
```
wv.setWebChromeClient(new WebChromeClient(){
    public void onProgressChanged(WebView view, int progress) {
        // 加入自己的處理程式
    }
});
```

# 使用 WebChromeClient 建立網頁載入進度介面



# 使用 Activity 內建功能顯示網頁載入進度

視窗的進度顯示功能會顯示在 Action Bar (Android 4.X) 或標題欄 (Android 2.X)



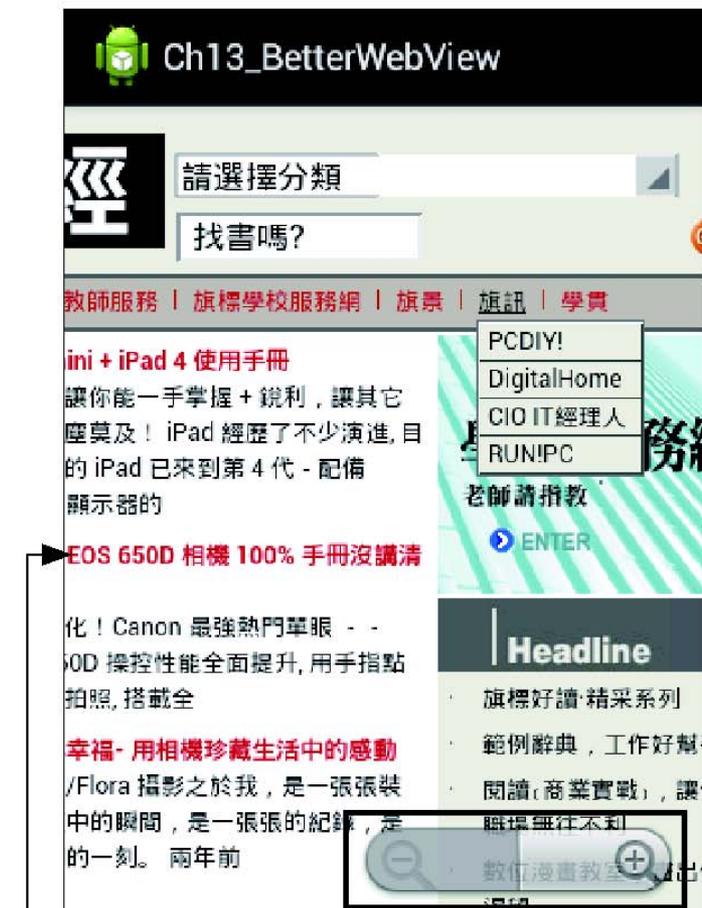
# 使用 Activity 內建功能顯示網頁載入進度

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    getWindow().requestFeature(Window.FEATURE_PROGRESS); ←  
                                                    加上進度顯示功能  
    setContentView(R.layout.activity_main);  
    ...  
}
```

# 使用 onKeyDown() 實作回上一頁功能

```
public boolean onKeyDown(int keyCode, KeyEvent event) {  
    if (keyCode==KeyEvent.KEYCODE_BACK && ← 若按下的是返回鍵, 且  
        wv.canGoBack()) { ← WebView 有上一頁  
        wv.goBack(); ← 回上一頁  
        return true; ← 傳回 true,  
    } ← 表示按鍵事件已處理  
    return super.onKeyDown(keyCode, event);  
}
```

# 範例13-2：改善 WebView 行為



2 按網頁中的超連結

1 可縮放網頁



會在 WebView 中開啟超連結, 不另開瀏覽器

# 改善 WebView 行為

step 1

step 2

```
01 public class MainActivity extends Activity {  
02     WebView wv;  
03     @Override  
04     protected void onCreate(Bundle savedInstanceState) {  
05         super.onCreate(savedInstanceState);  
06         getWindow().requestFeature(Window.FEATURE_PROGRESS); ←  
07         setContentView(R.layout.activity_main);  
08     }  
}
```

在標題欄加上  
進度功能

# 改善 WebView 行為

```
09     wv = (WebView) findViewById(R.id.wv);           啟用 JavaScript
10     wv.getSettings().setJavaScriptEnabled(true); ←┐
11     wv.getSettings().setBuiltInZoomControls(true); ←┐
12     wv.invokeZoomPicker(); ← 顯示縮放小工具       啟用縮放功能
13     wv.setWebViewClient(new WebViewClient()); ←┐
                                           建立及使用 WebViewClient 物件
14     wv.setWebChromeClient(new WebChromeClient() { ←┐
                                           建立及使用 WebChromeClient 物件
15         public void onProgressChanged(WebView view, int progress) {
16             setProgress(progress*100); ← 顯示進度, 標題欄進度
17         }                                     範圍是 0 到 10000
18     });
19
20     wv.loadUrl("http://www.flag.com.tw"); ← 連到旗標網站
21 }
...
```

# 改善 WebView 行為

## step 3

```
01 public boolean onKeyDown(int keyCode, KeyEvent event) {
02     if(keyCode==KeyEvent.KEYCODE_BACK &&    ← 若按下的是返回鍵, 且
03         wv.canGoBack()) {                    ← WebView 有上一頁
04         wv.goBack();                          ← 回上一頁
05         return true;                          ← 傳回 true, 表示按鍵事件已處理
06     }
07     return super.onKeyDown(keyCode, event);
08 }
```

# 改善 WebView 行為

## step 4

程式啟動後會出現載入進度長條 (ProgressBar)



# 改善 WebView 行為



- 1 用多點觸控的方式或按放大鈕放大網頁 (拖拉網頁時就會出現縮放工具)



網頁內容放大了



# 改善 WebView 行為



不會另開  
瀏覽器了

▶  
3 按返回鍵



↑  
回到前一個畫面（再按  
返回鍵則結束畫面）

# 13-3 使用 Preferences (偏好設定) 記錄資訊

- 使用 SharedPreferences 物件儲存資料
- 讀取偏好設定資料
- 儲存/回存資料的時機：onPause()/onResume()

# 使用 SharedPreferences 物件儲存資料

## 1. 取得物件和編輯器

```
SharedPreferences.Editor editor =  
    getPreferences (MODE_PRIVATE) .edit ();
```

取得SharedPreferences 物件

用傳回的物件呼叫 edit()  
取得 Editor 物件

# 使用 SharedPreferences 物件儲存資料

## 2. 儲存資料。

```
putBoolean(String key, boolean value);  
putFloat(String key, float value);  
putInt(String key, int value);  
putLong(String key, long value);  
putString(String key, String value);
```

key 為資料名稱  
value 為資料值  
例如 putInt("金額", 1500);

putInt("金額", 1500) 寫入

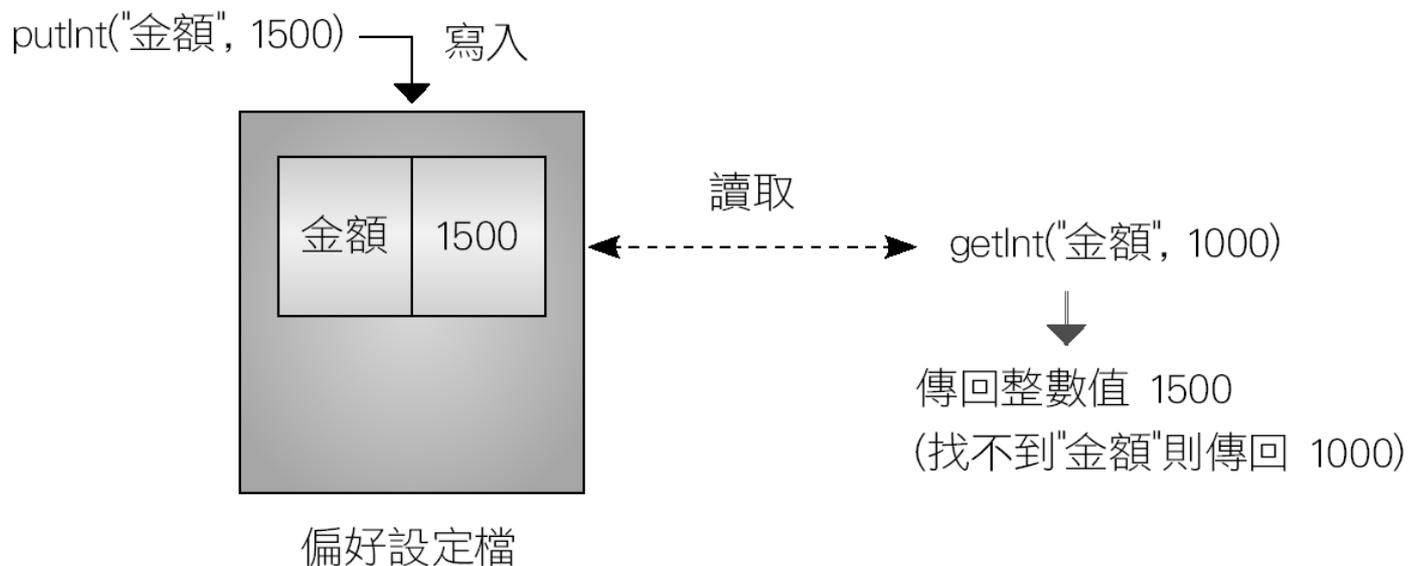


偏好設定檔

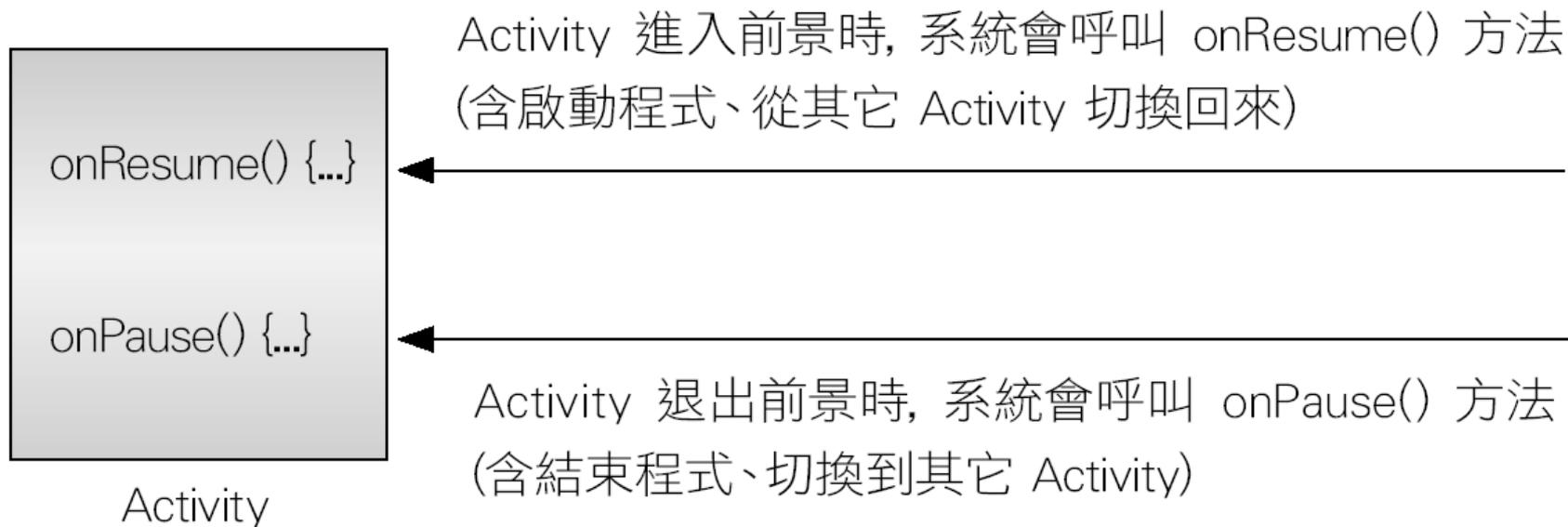
## 3. 存檔寫入。

# 讀取偏好設定資料

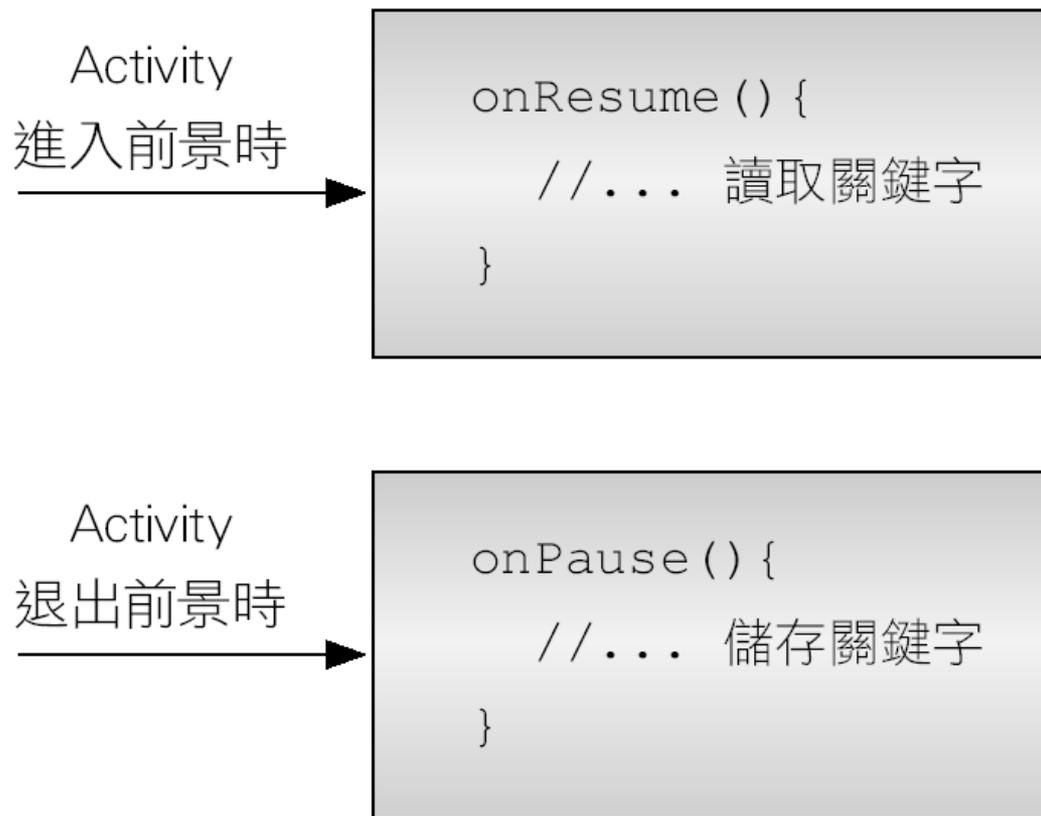
```
getInt("金額", 1000);    ← 若偏好設定資料中沒有 "金額" 就會傳回 1000  
getString("name", "John"); ← 若沒有 "name" 項目, 就傳回 "John"  
// 其它 getBoolean()、getFloat()、getLong() 用法都相似
```



# 儲存/回存資料的時機： onPause()/onResume()



# 儲存/回存資料的時機： onPause()/onResume()



# 範例13-3 : flickr 相片快搜



程式會記錄前次儲存的搜尋關鍵字

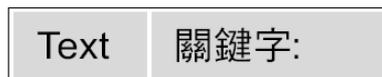
# flickr 相片快搜

## step 1

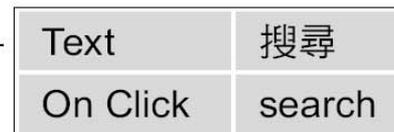
1 加入 LinearLayout (Horizontal)



2 加入 "Large" TextView:



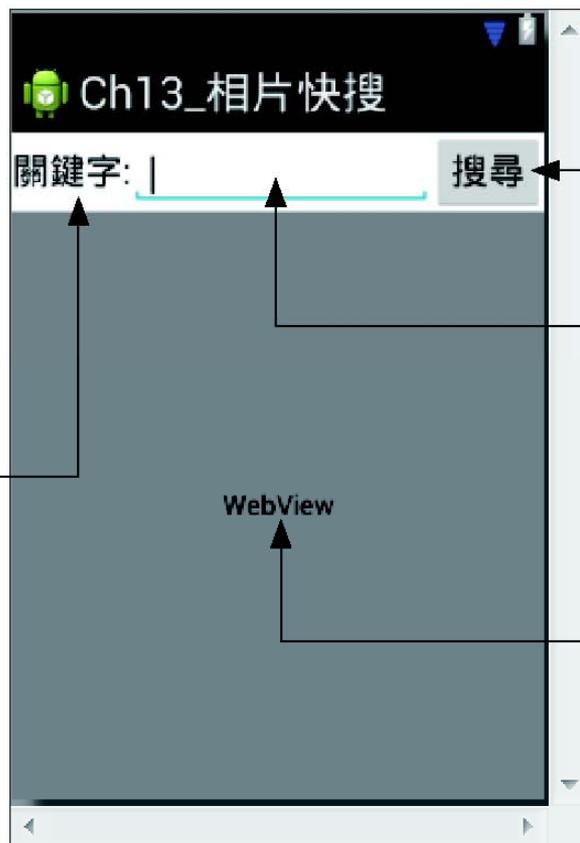
4 加入 Button:



3 加入 EditText:



5 加入 WebView



# flickr 相片快搜

## step 2

```
01 public class MainActivity extends Activity {
02     WebView wv;
03     String query="";           ← 查詢參數字串
04     String baseUrl="http://m.flickr.com/#/search/advanced/_QM_q_IS_";
05
06     @Override
07     protected void onCreate(Bundle savedInstanceState) {
08         super.onCreate(savedInstanceState);
09
10         this.getWindow().requestFeature(           ← 取得視窗功能:
11             Window.FEATURE_PROGRESS);           ← 在Action Bar或
12         setContentView(R.layout.activity_main);   標題欄顯示進度
```

# flickr 相片快搜

```
13
14     wv= (WebView) findViewById(R.id.wv);
15                                     啟用 JavaScript
16     wv.getSettings().setJavaScriptEnabled(true); ←
17     wv.setWebViewClient(new WebViewClient()); ←
                                               建立WebViewClient物件
18     // 自訂 WebChromeClient, 改寫 onProgressChanged() 方法
19     wv.setWebChromeClient(new WebChromeClient() {
20         public void onProgressChanged(WebView view,
                                       int progress)      {
21             setProgress(progress*100); ← 顯示進度, 標題欄進度
22         } ← 範圍是 0~10000
23     });
24 }
```

# flickr 相片快搜

```
25
26  @Override ← 改寫onKeyDown()方法，當按下返回鍵可回上一頁而非結束程式
27  public boolean onKeyDown(int keyCode, KeyEvent event) {
28      if (keyCode==KeyEvent.KEYCODE_BACK && ← 若按下的是返回鍵，
29          wv.canGoBack()) { ← 且 WebView 有上一頁
30          wv.goBack(); ← 回上一頁
31          return true; ← 傳回 true，表示按鍵事件已處理
32      }
33      return super.onKeyDown(keyCode, event);
34  }
```

...

// 與 WebView 相關的程式和前一範例相同，此處不再重複解說。

# flickr 相片快搜

The screenshot shows the mobile interface of Flickr's search results for 'Taipei 101'. At the top, a browser address bar contains the URL: `http://m.flickr.com/#/search/advanced/_QM_q_IS_Taipei+101`. Below the address bar is the Flickr logo with 'from YAHOO!' underneath. A navigation bar features 'Welcome' and 'Explore' links. The main content area is titled 'Search / Results' and displays the message: 'We found 129,070 results matching Taipei and 101.' Below this, there are sorting options: 'Most relevant', 'Most recent', and 'Most interesting'. The search results are listed in a vertical feed, each entry consisting of a small thumbnail image, the title 'Taipei 101', the uploader's name, and the number of views, comments, and favorites.

Thumbnail	Title	Uploader	Views	Comments	Favorites
	Taipei 101	Wang Guowen (gw.wang)	228	0	4
	Taipei 101	Wang Guowen (gw.wang)	161	0	1
	TAIPEI 101	Ansel2007	2905	4	6
	Taipei 101	Colin Lim	1529	44	13

以 "Taipei 101" 在  
行動版 flickr 網  
站搜尋時URL

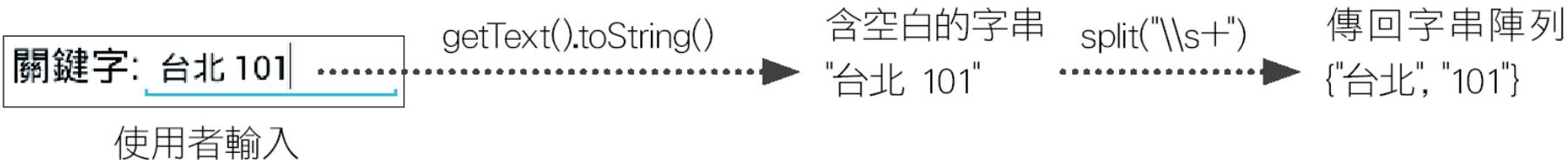
# flickr 相片快搜

## step 3

```
01 public void search(View v) {
02     EditText keyText=(EditText) findViewById(R.id.keywords);
03     //將整個字串依空白分隔成陣列
04     String[] keywords=keyText.getText().toString().split("\\s+");
05
06     query="";
07     for(String s:keywords) ← 用迴圈將陣列中的字串組成查詢參數
08         query+=s+" ";
09     query=query.substring(0, query.length()-1); ← 去掉最後 1 個加號
```

# flickr 相片快搜

```
10
11     if (URLUtil.isHttpUrl(baseUrl+query)) ← 檢查 URL 格式是否正確
12         wv.loadUrl(baseUrl+query);
13 }
```



# flickr 相片快搜

## step 4

```
01 @Override
02 protected void onPause() {
03     super.onPause();
04     SharedPreferences.Editor editor = ← 取得編輯器物件
05         getPreferences(MODE_PRIVATE).edit(); ←
06
07     editor.putString("關鍵字", query); ← 儲存目前的搜尋參數
08     editor.commit();
09 }
10
11 @Override
```

# flickr 相片快搜

```
12 protected void onResume() {  
13     super.onResume();  
14     SharedPreferences myPref = getPreferences(MODE_PRIVATE);  
15     query=myPref.getString("關鍵字", "Taipei+101");  
16  
17     if(wv.getUrl()==null)  
18         wv.loadUrl(baseUrl+query);  
19 }
```

取得偏好設定物件

讀取儲存的字串項目，  
若項目不存在，則傳回  
預設值 "Taipei+101"

# flickr 相片快搜

step 5

step 6

1 輸入想要搜尋  
的關鍵字

程式一開始會搜尋  
"Taipei+101"



# flickr 相片快搜

2 按此鈕



稍候就會出現搜尋結果



3 按返回鍵結束程式，再重新執行



一開始執行就會使用上次的關鍵字做搜尋