|  |
| --- |
| 資訊工業策進會 數位教育研究所 |
| 初級行動裝置程式設計師能力鑑定 考前輔導研習營 |
| **行動裝置程式開發-Android程式設計 Android程式開發** |
|  |
| 林俊昌 |
|  |

目 錄

[1 Android - UI設計 6](#_Toc495591213)

[1.1 UI基本元件應用 6](#_Toc495591214)

[1.1.1 android.view.View類別 6](#_Toc495591215)

[1.1.1.1 隨堂測驗 7](#_Toc495591216)

[1.1.2 TextView 8](#_Toc495591217)

[1.1.2.1 隨堂測驗 10](#_Toc495591218)

[1.1.3 EditText 13](#_Toc495591219)

[1.1.3.1 隨堂測驗 13](#_Toc495591220)

[1.1.4 Button 14](#_Toc495591221)

[1.1.4.1 隨堂測驗 15](#_Toc495591222)

[1.1.5 CheckBox 15](#_Toc495591223)

[1.1.5.1 隨堂測驗 17](#_Toc495591224)

[1.1.6 RadioButton與RadioGroup 17](#_Toc495591225)

[1.1.6.1 隨堂測驗 20](#_Toc495591226)

[1.1.7 Spinner 21](#_Toc495591227)

[1.1.7.1 隨堂測驗 23](#_Toc495591228)

[1.1.8 ListView 24](#_Toc495591229)

[1.1.8.1 隨堂測驗 26](#_Toc495591230)

[1.1.9 ImageView 28](#_Toc495591231)

[1.1.9.1 隨堂測驗 28](#_Toc495591232)

[1.1.10 WebView 29](#_Toc495591233)

[1.1.10.1 隨堂測驗 30](#_Toc495591234)

[1.1.11 MediaPlayer 31](#_Toc495591235)

[1.1.11.1 隨堂測驗 33](#_Toc495591236)

[1.1.12 AlertDialog 35](#_Toc495591237)

[1.1.12.1 隨堂測驗 36](#_Toc495591238)

[1.1.13 DatePickerDialog / TimePickerDialog 36](#_Toc495591239)

[1.1.13.1 隨堂測驗 41](#_Toc495591240)

[1.1.14 Toast即時訊息 42](#_Toc495591241)

[1.1.14.1 隨堂測驗 42](#_Toc495591242)

[1.1.15 Notification 43](#_Toc495591243)

[1.1.15.1 隨堂測驗 46](#_Toc495591244)

[1.1.16 Options Menu 46](#_Toc495591245)

[1.1.16.1 隨堂測驗 49](#_Toc495591246)

[1.2 Layout布局與應用 49](#_Toc495591247)

[1.2.1 Android提供的主要的版面配置元件 49](#_Toc495591248)

[1.2.2 Android版面配置元件共同的父類別 50](#_Toc495591249)

[1.2.3 隨堂測驗 50](#_Toc495591250)

[1.3 Activity運用與生命週期 53](#_Toc495591251)

[1.3.1 重點摘要 53](#_Toc495591252)

[1.3.2 隨堂測驗 55](#_Toc495591253)

[1.4 Touch事件 58](#_Toc495591254)

[1.4.1 重點摘要 58](#_Toc495591255)

[1.4.2 隨堂測驗 59](#_Toc495591256)

[1.5 MVC架構 60](#_Toc495591257)

[1.5.1 重要摘要 60](#_Toc495591258)

[1.5.2 隨堂測驗 60](#_Toc495591259)

[1.6 字串處理 61](#_Toc495591260)

[1.6.1 String類別的字串處理 61](#_Toc495591261)

[1.6.1.1 字串串連運算子 + 61](#_Toc495591262)

[1.6.1.2 隨堂測驗 61](#_Toc495591263)

[1.6.1.3 String類別建構子 62](#_Toc495591264)

[1.6.1.4 String類別實體方法: getBytes() 62](#_Toc495591265)

[1.6.1.5 getBytes() 隨堂測驗 63](#_Toc495591266)

[1.6.1.6 String類別實體方法: matches() 64](#_Toc495591267)

[1.6.1.7 String類別實體方法: split() 65](#_Toc495591268)

[1.6.1.8 split() 隨堂測驗 66](#_Toc495591269)

[1.6.1.9 String類別實體方法: replaceFirst() 66](#_Toc495591270)

[1.6.1.10 String類別實體方法: substring() 67](#_Toc495591271)

[1.6.1.11 String類別實體方法: trim() 67](#_Toc495591272)

[1.6.1.12 substring() / trim() 隨堂測驗 67](#_Toc495591273)

[1.6.1.13 String類別實體方法: indexOf() 68](#_Toc495591274)

[1.6.1.14 String類別實體方法: toUpperCase() 68](#_Toc495591275)

[1.6.1.15 substring() / indexOf() / toUpperCase() 隨堂測驗 68](#_Toc495591276)

[1.6.1.16 String類別實體方法: toCharArray() 69](#_Toc495591277)

[1.6.1.17 String類別靜態方法: format() 69](#_Toc495591278)

[1.6.1.18 format()隨堂測驗 71](#_Toc495591279)

[1.6.2 StringBuffer類別的字串處理 71](#_Toc495591280)

[1.6.2.1 StringBuffer類別實體方法: insert()、append()、toString() 71](#_Toc495591281)

[1.6.3 StringWriter類別的字串處理 72](#_Toc495591282)

[1.6.3.1 StringWriter類別實體方法: write()、append()、toString() 72](#_Toc495591283)

[1.6.3.2 StringWriter隨堂測驗 72](#_Toc495591284)

[1.6.4 Integer類別的字串處理 73](#_Toc495591285)

[1.6.4.1 Integer類別靜態方法: parseInt() 73](#_Toc495591286)

[1.6.4.2 Integer.parseInt()隨堂測驗 73](#_Toc495591287)

[1.7 開發時的圖檔與資源檔案安排 73](#_Toc495591288)

[1.7.1 重點摘要 73](#_Toc495591289)

[1.7.2 隨堂測驗 74](#_Toc495591290)

[2 Android - I/O輸出入串流 76](#_Toc495591291)

[2.1 資料儲存空間與檔案管理 76](#_Toc495591292)

[2.1.1 Android檔案系統 76](#_Toc495591293)

[2.1.1.1 隨堂測驗 77](#_Toc495591294)

[2.1.2 求取Android外部儲存空間相關目錄之API 77](#_Toc495591295)

[2.1.2.1 隨堂測驗 79](#_Toc495591296)

[2.1.3 求取Android內部儲存空間相關目錄之API 82](#_Toc495591297)

[2.1.3.1 隨堂測驗 82](#_Toc495591298)

[2.1.4 使用SharedPreferences存取內部儲存空間的資料 83](#_Toc495591299)

[2.1.4.1 隨堂測驗 85](#_Toc495591300)

[2.1.5 使用SQLite資料庫存取內部儲存空間的資料 87](#_Toc495591301)

[2.1.5.1 隨堂測驗 90](#_Toc495591302)

[2.2 檔案輸出與輸入 92](#_Toc495591303)

[2.2.1 使用openFileInput/Output方法進行內部儲存空間I/O 92](#_Toc495591304)

[2.2.1.1 隨堂測驗 94](#_Toc495591305)

[3 Android - 內建裝置與系統服務 96](#_Toc495591306)

[3.1 內建感應器 96](#_Toc495591307)

[3.1.1 加速度感測器 96](#_Toc495591308)

[3.1.1.1 隨堂測驗 98](#_Toc495591309)

[3.1.2 接近感測器 99](#_Toc495591310)

[3.1.3 亮度感測器 99](#_Toc495591311)

[3.1.4 溫度感測器 100](#_Toc495591312)

[3.1.5 濕度感測器 100](#_Toc495591313)

[3.1.5.1 隨堂測驗 100](#_Toc495591314)

[3.2 內建裝置應用技術 101](#_Toc495591315)

[3.2.1 振動器 101](#_Toc495591316)

[3.2.1.1 隨堂測驗 102](#_Toc495591317)

[3.2.2 相機 102](#_Toc495591318)

[3.2.2.1 隨堂測驗 105](#_Toc495591319)

[3.2.3 電話 107](#_Toc495591320)

[3.2.3.1 隨堂測驗 109](#_Toc495591321)

[3.3 定位功能 109](#_Toc495591322)

[3.3.1 定位技術 109](#_Toc495591323)

[3.3.1.1 隨堂測驗 110](#_Toc495591324)

[3.3.2 使用LocationManager 110](#_Toc495591325)

[3.3.2.1 隨堂測驗 112](#_Toc495591326)

[3.4 Service運用 112](#_Toc495591327)

[3.4.1 隨堂測驗 116](#_Toc495591328)

[3.5 異步執行技術 117](#_Toc495591329)

[3.5.1 執行緒(Thread) 117](#_Toc495591330)

[3.5.2 android.os.AsyncTask<Params, Progress, Result>抽象類別 119](#_Toc495591331)

[3.5.2.1 隨堂測驗 123](#_Toc495591332)

[3.6 Permission設定 123](#_Toc495591333)

[3.6.1 隨堂測驗 124](#_Toc495591334)

[3.7 AndroidManifest設定 125](#_Toc495591335)

[3.7.1 隨堂測驗 126](#_Toc495591336)

[3.8 官方開發工具與程式語言 127](#_Toc495591337)

[3.8.1 Android官方IDE開發工具：Android Studio 127](#_Toc495591338)

[3.8.2 Android官方程式開發語言：Java. 127](#_Toc495591339)

[3.9 Android記憶體管理機制 127](#_Toc495591340)

[3.9.1 隨堂測驗 127](#_Toc495591341)

[3.10 AppWidget 128](#_Toc495591342)

[3.10.1 隨堂測驗 131](#_Toc495591343)

[3.11 Android Platform Architecture 132](#_Toc495591344)

[3.11.1 隨堂測驗 133](#_Toc495591345)

[3.1 Android 2D繪圖 133](#_Toc495591346)

[3.1.1 隨堂測驗 134](#_Toc495591347)

參考資料來源: https://developer.android.com/reference/packages.html

# Android - UI設計

## UI基本元件應用

### android.view.View類別

* View是GUI元件共同的父類別
* View定義了一些**事件監聽器**，可用來監聽GUI元件所觸發的相關事件，例如：
  + **View.onClickListener** (監聽"手指按下然後放開的事件")
  + **View.onLongClickListener**(監聽"手指按住不放約1秒以上"的事件)
  + **View.onTouchListener**(監聽"手指按下或放開"的事件)
  + **View.onKeyListener**(監聽"手機按鍵被按下"的事件)
* View提供的最常用的註冊事件監聽器的方法
  + void setOnClickListener (View.OnClickListener l)：Register a callback to be invoked when this view is clicked.

|  |
| --- |
| View.OnClickListener(介面)  abstract void onClick (View v); |

* 範例：

**package** a.b.c;  
**import** android.app.Activity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.Button;  
**public class** MainActivity **extends** Activity

**implements** View.OnClickListener {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);

Button btn1 = **(Button)findViewById(R.id.*btn1*);** btn1.setOnClickListener(**this**);  
 }  
 **public void** onClick(View view){  
 System.***out***.println( ((Button)view).getText() + **"被點擊了..."**);  
 }  
}



點選上面按鈕可得下列輸出

I/System.out: BUTTON1被點擊了..

#### 隨堂測驗

|  |
| --- |
| (1)  在Android的UI元件中皆支援點擊事件，請問要呼叫哪項功能來註冊監聽器以監聽是否有發覺點擊？  (A) setOnClickListener()  (B) onClick()  (C) onAddClick()  (D) onSubmitClick()  答案: A |

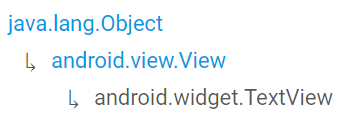
|  |
| --- |
| (2)  在Android UI設計中，可呼叫下列哪項功能並且指定ID來取得Layout檔案內中的UI元件？   1. setContentView(R.layout.main\_activity) 2. findViewById(R.id.xxx) 3. setText(null) 4. 直接使用ID即可（如：R.id.xxx）   答案: B |

|  |
| --- |
| (3)  Android的android.view.View類別定義了多種事件監聽器（Event listeners），可用來監聽GUI元件所觸發的相關事件。下列何者不是android.view.View類別所定義的事件監聽器？  (A) OnClickListener  (B) OnShortClickListener  (C) OnKeyListener  (D) OnTouchListener  答案: B |

|  |
| --- |
| (4)  findViewById的設定，產生Views等靜態設定通常在哪個生命週期執行？  (A) onCreate  (B) onPause  (C) onStop  (D) onDestroy  答案: A |

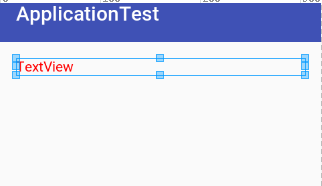
### TextView

* 類別繼承關係



* 利用**XML版面配置檔**之屬性android:layout\_width 、 android:layout\_height 可分別設定UI元件的寬與高的大小。主要設定值如下：
  + **wrap\_content**： UI元件的大小剛好足夠包圍其內容
  + **match\_parent**： UI元件的大小要比照父元件的大小
  + 自訂長度(如50px)
* 利用**XML版面配置檔**之屬性**android:textColor、android:background**可分別設定UI元件內的文字與其背景顏色。主要設定值如下：
  + **color value**：
    - 格式為 #RGB | #RRGGBB | #ARGB | #AARRGGBB
    - 如: #FF0000 (紅色) #7FFF0000(紅色半透明)
  + **color resource**：
    - 格式為@[package:]color/color\_name
* 利用**XML版面配置檔**之屬性**android:textSize**可設定UI元件內的文字大小。可用長度單位如下：
  + **px (pixels)**
  + **dp (density-independent pixels)**
  + **sp (scaled pixels based on preferred font size)**
  + **in (inches)**
  + **mm (millimeters).**
* **在XML版面配置檔宣告<TextView>範例**：

<**TextView  
 android:text="TextView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/textView"  
 android:textColor="#FF0000"** />



* 常用方法
  + CharSequence getText () : Return the text the TextView is displaying.
  + void setText (CharSequence text)：Sets the string value of the TextView.

#### 隨堂測驗

|  |
| --- |
| (1)  下列有關TextView元件的描述何者是錯誤的？  (A) TextView的package名稱為android.widget  (B) TextView的父類別為android.view.View  (C)TextView的getText()方法的回傳資料型別為java.lang.String  (D) TextView的setText()方法可設定TextView的字串值  答案: C |

|  |
| --- |
| (2)  請問下列何種功能無法將資訊輸出到裝置畫面上？  (A) 使用System.out.println顯示  (B) 使用Toast顯示  (C) 使用TextView顯示  (D) 使用Dialog顯示  答案: A |

|  |
| --- |
| (3)  以下何者是Android UI Layout屬性wrap\_content的意義？   1. 讓UI元件的呈現大小隨內容調整 2. 讓UI元件的呈現大小不考慮內容 3. 讓UI元件的顏色隨內容調整 4. 讓UI元件的編排是否置中   答案: A |

|  |
| --- |
| (4)  Android GUI元件的背景顏色可透過相關版面配置檔的background屬性來設定，下列background屬性值之設定，何者正確？   1. #CCFF33 2. color.blue 3. (255,255,255) 4. 以上皆非   答案: A |

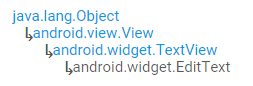
|  |
| --- |
| (5)  Android的UI元件可以透過Layout檔案來完成版面配置，且一般會建議將UI的版面抽離程式碼，單獨放在Layout檔案內，請問下列何項為該Layout檔案的格式？   1. XAML 2. JSON 3. XML 4. HTML   答案: C |

|  |
| --- |
| (6)  以下何者不是Android開發時，可以使用的尺寸單位？  (A) sp  (B) dp  (C) px  (D) ip  答案: D |

|  |
| --- |
| (7)  當我們要清除EditText上的文字時，可用下列何種語法？  (A) editText.setText("");  (B) editText.clear();  (C) editText.setText(‘’);  (D) editText.setText();  答案: A |

### EditText

* 類別繼承關係



* 利用**XML版面配置檔**之屬性android:inputType可設定輸入EditText元件的文字的資料類型(如text、number/正整數、numberSigned/正或負整數、numberDecimal/正浮點數、phone、datetime等)

#### 隨堂測驗

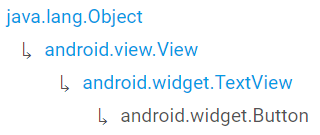
|  |
| --- |
| (1)  下列有關EditText的描述何者是錯誤的？  (A) EditText屬於android.widget套件  (B) EditText 繼承android.widget.TextView  (C)XML版面配置檔的android:inputType屬性可設定為"phone"  (D) 以上皆非  答案: D |

|  |
| --- |
| (2)  當我們要把使用者在EditText中所輸入的一組字串讀出來，存到一個String變數使用時，會使用下列何種方法？  (A) .getText().toString();  (B) .setText(text);  (C) .getString(EditText);  (D) .getView();  答案: A |

|  |
| --- |
| (3)  關於Android的「EditText」GUI元件，下列敘述何者正確？   1. EditText的package為android.text 2. EditText繼承android.widget.TextView，是一種唯讀的文字方塊元件 3. 透過設定版面配置檔的inputType屬性，可限定輸入EditText元件的文字的資料類型 4. EditText的getText方法可直接傳回String型別的文字資料   答案: C |

### Button

* 類別繼承關係



* 利用**XML版面配置檔**在Activity顯示Button的方法如下

<Button

android:id="@+id/button\_id"

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

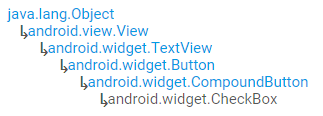
android:text="@string/self\_destruct" />

#### 隨堂測驗

|  |
| --- |
| (1)  在版面配置檔案中的佈局如下，請問會在螢幕上看到什麼？  <Button android:id="@+id/button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="OK" />  (A) 文字  (B) 按鈕  (C) 版面佈局  (D) 圖片  答案: B |

### CheckBox

* 類別繼承關係



* 常用方法
  + boolean isChecked ()
  + void setOnCheckedChangeListener (CompoundButton.OnCheckedChangeListener listener): Register a callback to be invoked when the checked state of this button changes.

|  |
| --- |
| CompoundButton.OnCheckedChangeListener(介面)  abstract void onCheckedChang**ed** (CompoundButton buttonView, boolean isChecked); |

* 範例：

**package** a.b.c;  
**import** android.app.Activity;  
**import** android.os.Bundle;  
**import** android.widget.CheckBox;  
**import** android.widget.CompoundButton;

**public class** MainActivity **extends** Activity

**implements** CompoundButton.OnCheckedChangeListener {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);   
 CheckBox cb = (CheckBox)findViewById(R.id.***checkBox***);  
 cb.setOnCheckedChangeListener(**this**);  
 }  
 @Override  
 **public void** onCheckedChang**ed**(CompoundButton buttonView,

**boolean** isChecked) {  
 System.***out***.printf(**"%s isChecked=%b\n"**,

buttonView.getText(), isChecked);  
 }  
}



勾選CheckBox可得下列輸出

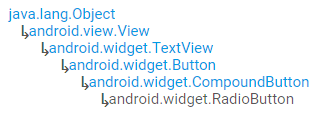
I/System.out: CheckBox isChecked=true

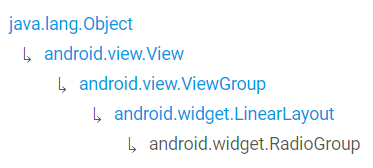
#### 隨堂測驗

|  |
| --- |
| (1)  下列有關Android的「CheckBox」GUI元件的描述何者是正確的？  (A) CheckBox的父類別為android.widget.CompoundButton  (B) CheckBox的isChecked()方法可用來檢查該元件是否被選取  (C) CheckBox的setOnCheckedChangeListener()方法可用來註冊該元件被選取或取消選取的事件監聽器  (D) 以上皆是  答案: D |

### RadioButton與RadioGroup

* 類別繼承關係





* 一個RadioGroup內的多個RadioButton之點選狀態具有互斥效果
* 利用XML版面配置檔將RadioButton元素放置於RadioGroup元素的內容區域(如下)即可達到上述互斥效果

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"** (略) **android:orientation="vertical"**>  
<**RadioGroup  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/radioGroup"**

**android:orientation="horizontal"**>  
  
 <**RadioButton  
 android:text="Male"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/radioButton"  
 android:layout\_weight="1"  
 android:checked="true"** />  
  
 <**RadioButton  
 android:text="Female"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/radioButton2"  
 android:layout\_weight="1"** />  
</**RadioGroup**>

</**LinearLayout**>



* RadioGroup常用方法
  + int getCheckedRadioButtonId () : Returns the identifier of the selected radio button in this group. Upon empty selection, the returned value is -1.
  + void setOnCheckedChangeListener (RadioGroup.OnCheckedChangeListener listener): Register a callback to be invoked when the checked radio button changes in this group.

|  |
| --- |
| RadioGroup.OnCheckedChangeListener(介面)   * abstract void onCheckedChang**ed** (RadioGroup group, int checkedId);   *checkedId: the unique identifier of the newly*  *checked radio button* |

* 範例：

**package** a.b.c;  
**import** android.app.Activity;  
**import** android.os.Bundle;  
**import** android.widget.RadioButton;  
**import** android.widget.RadioGroup;  
  
**public class** MainActivity **extends** Activity

**implements** RadioGroup.OnCheckedChangeListener {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 RadioGroup radioGroup =

(RadioGroup)findViewById(R.id.***radioGroup***);  
 radioGroup.setOnCheckedChangeListener(**this**);  
 }  
@Override  
 **public void** onCheckedChanged(RadioGroup group,

**int** checkedId) {

String s = ((RadioButton)findViewById(checkedId))

.getText().toString();  
 System.***out***.printf(**"%s is checked \n"**, s);  
 }}



點選RadioButton2可得下列輸出

I/System.out: RadioButton2 is checked

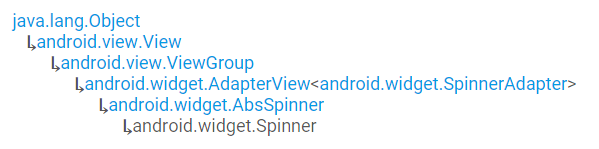
#### 隨堂測驗

|  |
| --- |
| (1)  下列有關Android的事件監聽器(Event Listeners)的描述，何者正確？  (A) RadioGroup.OnCheckedChangeListener物件可被註冊來處理 ComboBox元件之選項點選事件  (B) CompoundButton.OnCheckedChangeListener物件可被註冊來處理 CheckBox元件之點選狀態變更事件  (C) ViewGroup.OnItemSelectedListener物件可被註冊來處理Spinner元件之選項點選事件  (D) 以上皆是  答案: B |

|  |
| --- |
| (2)  常用的UI元件中，下列敘述何者不正確？   1. WebView－讓使用者可以在其所配置的Layout中瀏覽網頁 2. RatingBar－讓使用者可以點選幾顆星的方式來對指定項目評分 3. SeekBar－透過拖曳方式達到選取數值的功能 4. RadioGroup－讓使用者可以透過複選方式選取資   答案: D |

### Spinner

* 類別繼承關係



* Spinner是一種下拉式選單(ComboBox)
* 利用**XML版面配置檔**之屬性android:spinnerMode可設定Spinner之顯示模式: dialog或dropdown(預設)
* 利用**XML版面配置檔**之屬性android:entries可設定Spinner之選項(對應Array Resource)
* 常用方法(繼承自android.widget.AdapterView)
  + int getSelectedItemPosition ()： Return the position of the currently selected item within the adapter's data set
  + Object getSelectedItem ()
  + Object getItemAtPosition (int position)： Gets the data associated with the specified position in the list.
  + void setAdapter (SpinnerAdapter adapter)： Sets the adapter that provides the data and the views to represent the data in this widget.
  + void setOnItemSelectedListener (AdapterView.OnItemSelectedListener listener)：Register a callback to be invoked when an item in this AdapterView has been selected.

|  |
| --- |
| AdapterView.OnItemSelectedListener (介面)   * abstract void onItemSelected (AdapterView<?> parent, View view, int position, long id); *when an item in this view has been selected.* * abstract void onNothingSelected (AdapterView<?> parent); *when the selection disappears from this view.* |

* 範例：

res/values/strings.xml

<**resources**>  
 <**string-array name="hobby"**>  
 <**item**>吟詩</**item**>  
 <**item**>彈琴</**item**>  
 <**item**>作對</**item**>  
 </**string-array**>  
</**resources**>

res/layout/activity\_main.xml

(略)

<**Spinner  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/spinner"  
 android:entries="@array/hobby"** />

MainActivity.java

**package** a.b.c;  
**import** android.app.Activity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.AdapterView;  
**import** android.widget.Spinner;  
**import** android.widget.TextView;  
**public class** MainActivity **extends** Activity

**implements** AdapterView.OnItemSelectedListener {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 Spinner spinner = (Spinner)findViewById(R.id.***spinner***);  
 spinner.setOnItemSelectedListener(**this**);  
 }  
 @Override  
 **public void** onItemSelected(AdapterView<?> parent,

View view, **int** position, **long** id) {  
 System.***out***.printf(**"%s(第%d項) 被點選了...\n"**,

((TextView)view).getText(), position );  
 }  
 @Override  
 **public void** onNothingSelected(AdapterView<?> parent) {  
 }  
}



點選"彈琴"可得下列輸出

I/System.out: 彈琴(第1項) 被點選了...

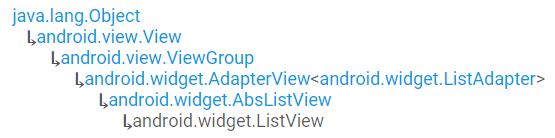
#### 隨堂測驗

|  |
| --- |
| (1)  Spinner優點為節省顯示空間的UI元件，當使用者尚未點選時，僅顯示一筆資料，選擇時會以類似下拉式選單方式呈現清單資料供使用者選擇。開發上需使用何種監聽器 (Listener)監聽哪一個選項被選擇？  (A) OnItemSelectedListener  (B) OnClickListener  (C) OnKeyListener  (D) OnTouchListener  答案: A |

|  |
| --- |
| (2)  下列有關Android的「Spinner」GUI元件的描述，何者正確？  (A) Spinner相當於Java的ProgressBar  (B) 透過設定版面配置檔的spinnerModeType屬性，可設定Spinner以 「toast」 或「dropdown」方式呈現選單  (C) Spinner的getSelectedItemPosition()方法會回傳目前被點選的選項的整數位置索引  (D) 以上皆是  答案: C |

### ListView

* 類別繼承關係



* 利用**XML版面配置檔**之屬性**android:entries**可設定ListView之選項(對應Array Resource)
* 常用方法(繼承自android.widget.AdapterView)
  + int getSelectedItemPosition ()
  + Object getSelectedItem ()
  + Object getItemAtPosition (int position)
  + void setAdapter (ListAdapter adapter)
  + void setOnItemClickListener (AdapterView.OnItemClickListener listener)： Register a callback to be invoked when an item in this AdapterView has been clicked.

|  |
| --- |
| AdapterView. OnItemClickListener (介面)   * abstract void onItemClick (AdapterView<?> parent, View view, int position, long id); *Callback method to be invoked when an item in this AdapterView has been clicked.* |

* 範例：

MainActivity.java

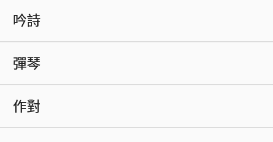
**package** a.b.c;  
**import** android.app.Activity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.AdapterView;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.ListView;  
**import** android.widget.TextView;  
**public class** MainActivity **extends** Activity

**implements** AdapterView.OnItemClickListener {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***); ListView listView = (ListView)findViewById(R.id.***listView***);  
 listView.setOnItemClickListener(**this**);  
 ArrayAdapter<String> arrayAdapter =

**new** ArrayAdapter<String>(**this**,

android.R.layout.***simple\_list\_item\_1***,  
 **new** String[]{**"吟詩"**,**"彈琴"**,**"作對"**});  
 listView.setAdapter(arrayAdapter);  
 }  
 @Override  
 **public void** onItemClick(AdapterView<?> parent, View view,

**int** position, **long** id) {  
 System.***out***.printf(**"%s(第%d項) 被點選了...\n"**,  
 ((TextView)view).getText(), position );  
 }  
}



點選"彈琴"可得下列輸出

I/System.out: 彈琴(第1項) 被點選了...

#### 隨堂測驗

|  |
| --- |
| (1)  請問ListView與ListAdapter之間的關係是？  (A) 兩者沒有任何的關係  (B) ListView 會把使用者的操作結果存在 ListAdapter 中  (C) ListView 的顯示內容仰賴 ListAdapter 來決定  (D) 以上皆非  答案: C |

|  |
| --- |
| (2)  下列有關Android的「ListView」GUI元件的描述，何者正確？   1. ListView具有dropdown與dialog兩種呈現清單的模式 2. ListView清單項目的點擊事件監聽器可使用View.OnItemSelectedListener 3. 透過設定版面配置檔的entries屬性，可設定ListView的清單項目 4. 以上皆是   答案: C |

|  |
| --- |
| (3)  下列有關Android之AdapterView.OnItemClickListener事件監聽器相關描述，何者是錯誤的？  (A) AdapterView.OnItemClickListener適合ListView元件用來監聽選項被點選之事件  (B) AdapterView.OnItemClickListener的型別是介面(interface)  (C) AdapterView.OnItemClickListener定義了onItemClick方法  (D) AdapterView.OnItemClickListener適合Spinner元件用來監聽選項被點選之事件  答案: D |

### ImageView

* 類別繼承關係



* ImageView元件可用來顯示圖片
* 利用**XML版面配置檔**之屬性android:src可設定ImageView元件所欲顯示的圖片，設定格式如下：
  + Android系統內建的圖形資源：@android:drawable/圖檔主檔名
  + 應用程式自訂的圖形資源：@drawable/圖檔主檔名 (基本上對應應用程式res/drawable目錄下的同主檔名之圖檔)
* 利用**XML版面配置檔**之屬性**android:scaleType**可設定ImageView元件所欲顯示的圖片在ImageView元件中的位置與縮放模式，例如
  + **"fitCenter**"： 圖片以符合元件寬或高(取較小者)的方式縮放且置於元件中間
  + **"fitXY"**：圖片以符合元件寬高的方式縮放且置於元件中間(若圖片寬高比與元件不同，則圖片會變形)
  + **"centerInside"**：原圖尺寸比元件小時不縮放且置於元件中間; 否則同"fitCenter"
  + **"center"**：原圖尺寸不縮放，置於元件中間(若圖片比元件大，則圖片會被裁切)

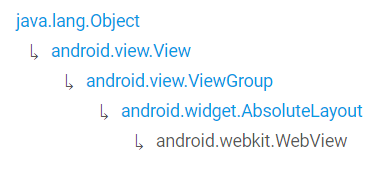
#### 隨堂測驗

|  |
| --- |
| (1)  下列有關ImageView元件的描述何者是錯誤的？  (A) ImageView可用來顯示圖片  (B)利用XML版面配置檔之屬性android:src可設定ImageView元件所欲顯示的圖片  (C) 利用XML版面配置檔之屬性android:scaleType可設定ImageView元件所欲顯示的圖片在元件中的位置與縮放模式  (D) 以上皆非  答案: D |

|  |
| --- |
| (2)  使用ImageView呈現單一圖片，如ImageView長寬設定為 match\_parent，需顯示的圖片尺寸為255×255，須呈現原圖依比例擴大或縮小至ImageView大小，以適應不同螢幕顯示，下列設定何者正確？   1. android:scaleType="fitCenter" 2. android:layout\_width= "match\_parent" 3. android:layout\_height= "match\_parent" 4. android:scaleType="fitXY" 5. 以上皆非   答案: A |

### WebView

* 類別繼承關係



* WebView為Android 用來顯示HTML網頁的元件
* 使用WebView定義的 void loadUrl (String url) 方法即可載入並顯示指定網址之網頁
* 使用WebView元件瀏覽網頁尚需在**AndroidManifest.xml**中宣告使用 android.permission.INTERNET 許可

#### 隨堂測驗

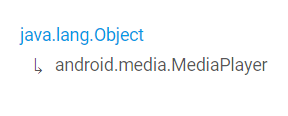
|  |
| --- |
| (1)  下列有關WebView元件的描述何者是錯誤的？  (A)WebView的套件為android.webkit  (B)使用WebView的loadPage()方法可顯示指定網址之網頁  (C) 使用WebView元件瀏覽網頁需在AndroidManifest.xml中宣告使用 android.permission.INTERNET 許可  (D) WebView為Android 用來顯示HTML網頁的元件  答案: B |

|  |
| --- |
| (2)  在應用程式內使用WebView元件瀏覽網頁必須使用到Internet，Android開發上需於何處設定Internet permission？   1. main.class 2. project.properties 3. layout.xml 4. AndroidManifest.xml   答案: D |

|  |
| --- |
| (3)  下列android.webkit.WebView類別的那一個方法，可用來在Android應用程式的Activity相關畫面中載入指定網址的網頁？  (A) loadUrl  (B) loadPage  (C) loadWebPage  (D) loadWeb  答案: A |

### MediaPlayer

* 類別繼承關係



* MediaPlayer 可用來控制audio/video檔案與資料串流的播放
* 透過呼叫MediaPlayer物件的下列方法即可控制上述檔案的播放流程
  + void setDataSource (String path) throws java.io.IOException：設定資料來源(檔案路徑或網址)
    - (Android 4.4 /API level 19起)當資料來源為外部儲存空間的檔案時，則需先取得"android.permission.READ\_EXTERNAL\_STORAGE"許可
  + void prepareAsync () / prepare () ：準備撥放 (prepare方法只適用於files; 不適用於streams)
  + void start ()：開始或恢復撥放
  + void pause ()：暫停撥放 (若想重新撥放，則只需重新呼叫 **start()**方法 )
  + void stop ()：停止撥放 (若想重新撥放，則需重新呼叫 **prepareAsync()/prepare()**方法 )
  + void release ()： 釋放MediaPlayer物件 (It is considered good practice to call this method when you're done using the MediaPlayer.)
  + void setLooping (boolean looping)：設定是否重覆播放
* 範例：

Activity

MediaPlayer **mp** = **new** MediaPlayer();

**protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 **try** {  
 **mp**.setDataSource(**"/sdcard/Music/bs.mp3"**);  
 **mp**.setLooping(**true**);  
 **mp**.prepareAsync();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
}

假設Activity內某按鈕被click時， 將執行下列方法  
@Override  
**public void** onClick(View v) {  
 **try** {  
 **mp**.start();  
 Thread.*sleep*(5000);  
 **mp**.pause();  
 } **catch** (InterruptedException e) {  
 e.printStackTrace();  
 }

}

AndroidManifest.xml

<**manifest xmlns:android=**

**"http://schemas.android.com/apk/res/android"**

**package="xx.yy.zz"**>  
 <**uses-permission android:name=**

**"android.permission.READ\_EXTERNAL\_STORAGE"** />

(略)

</**manifest**>

* Android支援的**媒體檔案類型**(Media File Types)主要有：
  + Audio files：
    - .3gp .mp3 .mp4 .mid
    - .wav .ogg .mkv .aac
  + Vedio files：
    - .3gp .mp4 .mkv .webm
  + Image files：
    - .bmp .jpg .png .webp .gif

#### 隨堂測驗

|  |
| --- |
| (1)  以下何者不是標準Android錄影時可以產出的影片錄製檔案格式？   1. \*.mp4 2. \*.3gp 3. \*.mov 4. \*.webm   答案: C |

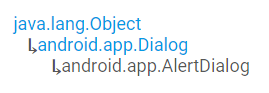
|  |
| --- |
| (2)  Android的MediaPlayer類別可以用來播放音樂。下列敘述何者不正確？   1. MediaPlayer類別定義的start方法可用來開始音樂播放 2. MediaPlayer類別定義的setDataSource方法可用來指定所欲播放的音樂檔的路徑 3. MediaPlayer類別定義的setLooping方法可用來設定播放器是否以重複播放的模式來播放音樂 4. MediaPlayer可用來播放mp3與wma檔   答案: D |

|  |
| --- |
| (3)  Android系統支援許多常見的多媒體格式，讓開發人員可透過Audio、Video等相關API以整合影音撥放功能於應用程式內。MediaPlayer不僅提供撥放Audio功能，也提供了撥放Video功能，如應用程式內使用MediaPlayer進行影音撥放，當系統不再使用MediaPlayer後，應如何處理，以防止產生過多的MediaPlayer物件實體而造成Exception產生？   1. 呼叫stop() 2. 呼叫release() 3. 呼叫reset() 4. 使用try-catch   答案: B |

|  |
| --- |
| (4)  下列何者不是標準Android有支援的音樂播放檔案格式？  (A) \*.mp4  (B) \*.3gp  (C) \*.aac  (D) \*.wma  答案: D |

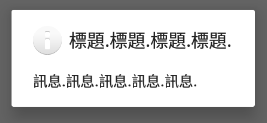
### AlertDialog

* 類別繼承關係



* 在應用程式中顯示AlertDialog的簡便方法為使用AlertDialog的內部類別: Builder
* AlertDialog.Builder常用建構子
  + AlertDialog.Builder (Context context)
* AlertDialog.Builder的常用實體方法
* public AlertDialog.Builder setTitle (CharSequence title)
* public AlertDialog.Builder setMessage (CharSequence message)
* public AlertDialog.Builder setIcon (int iconId)
* public AlertDialog show ()： *Creates an AlertDialog with the arguments supplied to this builder and immediately displays the dialog.*
* 範例：

**new** AlertDialog.Builder(MainActivity.**this**)  
 .setTitle(**"標題.標題.標題.標題."**)  
 .setIcon(android.R.drawable.***ic\_dialog\_info***)  
 .setMessage(**"訊息.訊息.訊息.訊息.訊息."**)  
 .show();

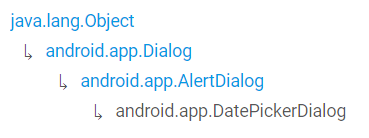


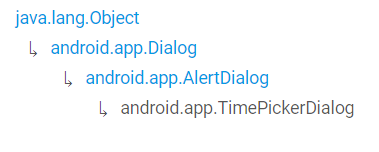
#### 隨堂測驗

|  |
| --- |
| (1)  下列程式的執行結果，何者是正確的？  new AlertDialog.Builder(MainActivity.this)  .setTitle("標題.標題.標題.標題.")  .setIcon(android.R.drawable.ic\_dialog\_info)  .setMessage("訊息.訊息.訊息.訊息.訊息.")  .show();  (A) 程式將顯示一個通知對話框，訊息為"標題.標題.標題.標題."  (B) 程式將顯示一個通知對話框，標題列文字為"訊息.訊息.訊息.訊息.訊息."  (C) 程式將顯示一個不具圖示的通知對話框  (D) 程式將顯示一個通知對話框，訊息為"訊息.訊息.訊息.訊息.訊息."  答案: D |

### DatePickerDialog / TimePickerDialog

* 類別繼承關係





* DatePickerDialog常用建構子
  + DatePickerDialog (Context context)

Creates a new date picker dialog for **the current date** using the parent context's default date picker dialog theme.

* + DatePickerDialog (Context context,

DatePickerDialog.OnDateSetListener listener, int year,

int month(0-11), int dayOfMonth)：\*

Creates a new date picker dialog for **the specified date** using the parent context's default date picker dialog theme.

* 參數 month： 0~11
* 參數dayOfMonth：1~31

**(0 : 表示前一個月最後一天) \***

例如: new DatePickerDialog(this,null,2017, 2, 0) 對話框預設選取日期為2017年2月28日

* DatePickerDialog的常用實體方法
* public void show ()： *Start the dialog and display it on screen.(繼承自android.app.Dialog)*
* DatePickerDialog.OnDateSetListener介面的抽象方法
* void onDateSet (DatePicker view, int year, int month,

int dayOfMonth)：

當使用者完成日期選擇後本方法即被呼叫

* 範例：

**package** com.example.test;  
**import** android.app.DatePickerDialog;  
**import** android.os.Bundle;  
**import** android.app.Activity;  
**import** android.widget.DatePicker;  
**import** android.widget.Toast;  
**import** java.util.Calendar;  
**public class** MainActivity **extends** Activity

**implements** DatePickerDialog.OnDateSetListener {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);

Calendar c = Calendar.*getInstance*();

//假設當日日期為2017/1/24  
 **int** y = c.get(Calendar.***YEAR***);  
 **int** m = c.get(Calendar.***MONTH***);   
 **int** d = c.get(Calendar.***DAY\_OF\_MONTH***);  
 **new** DatePickerDialog(**this**,**this**,y,m,d).show(); //\*}  
 @Override  
 **public void** onDateSet(DatePicker datePicker, **int** y, **int** m, **int** d)

{  
 Toast.*makeText*(**this**,

String.*format*(**"%04d/%02d/%02d"**,y,m+1,d),

Toast.***LENGTH\_SHORT***).show();  
 }  
}



按下「確定」按鈕即顯示下列Toast訊息



* TimePickerDialog常用建構子
  + TimePickerDialog (Context context, TimePickerDialog.OnTimeSetListener listener, int hourOfDay, int minute, boolean is24HourView) \*

Creates a new time picker dialog.

* TimePickerDialog的常用實體方法
* public void show ()： *Start the dialog and display it on screen.(繼承自android.app.Dialog)*
* TimePickerDialog.OnTimeSetListener介面的抽象方法
* void onTimeSet (TimePicker view, int hourOfDay, int minute)：

當使用者完成時間選擇後本方法即被呼叫

* 範例：

**package** com.example.test;

**import** android.os.Bundle;  
**import** android.app.Activity;  
**import** android.app.TimePickerDialog;  
**import** android.widget.TimePicker;  
**import** android.widget.Toast;  
**public class** MainActivity **extends** Activity {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);

onClick(null);  
 }

**public void** onClick(View v){  
 TimePickerDialog.OnTimeSetListener **otsl** =

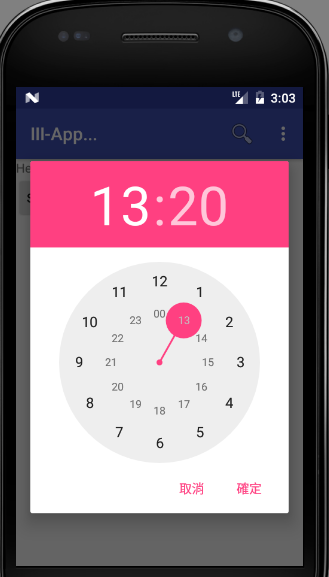
**new** TimePickerDialog.OnTimeSetListener() {  
 @Override  
 **public void** onTimeSet(TimePicker timePicker, **int** h, **int** m)

{  
 Toast.*makeText*(MainActivity.**this**,

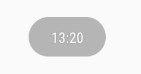
String.*format*(**"%02d:%02d"**, h,m),

Toast.***LENGTH\_SHORT***).show();  
 }  
 };  
 **new** TimePickerDialog(**this**, otsl, 13, 20, **true**).show(); //\*  
}

}



按下「確定」按鈕即顯示下列Toast訊息

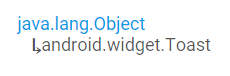


#### 隨堂測驗

|  |
| --- |
| (1)  如果要讓使用者輸入日期格式的資料，下列哪種元件比較適合？   1. DateView 2. EditText 3. DatePickerDialog 4. EditDate   答案: C |

### Toast即時訊息

* 類別繼承關係



* 常用方法
* static Toast makeText (Context context, CharSequence text, int duration) ：*Make a standard toast that just contains a text view.*
* void show ()：*Show the view for the specified duration.*
* 範例：

Toast.*makeText*(MainActivity.**this**, **"Welcome!"**,

Toast.***LENGTH\_LONG*** ).show(); //\*



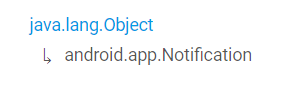
#### 隨堂測驗

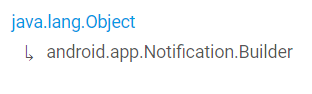
|  |
| --- |
| (1)  如果要出現不中斷使用者操作的訊息通知，下列何種技術比較適合？  (A) AlertDialog  (B) Toast  (C) Builder  (D) MessageAcitvity  答案: B |

|  |
| --- |
| (2)  關於Toast訊息顯示時間設定的方式，下列敘述何者正確？  (A) makeText()中設定duration為Toast.LENGTH\_LONG  (B) makeText()中設定duration為Toast.LONG  (C) makeText()中設定duration為1000  (D) Thread.sleep(1000)  答案: A |

### Notification

* 類別繼承關係





* 應用程式欲發送**通知**需要先建立一個Notification的實例，再利用系統提供的服務(NotificationManager)來發送此Notification實例。
* 建立Notification實例的簡便方法為使用Notification的Builder內部類別
  + Notification.Builder常用建構子
    - Notification.Builder (Context context)
  + Notification.Builder的常用實體方法
  + Notification.Builder setContentTitle (CharSequence title)：

Set the first line of text in the platform notification template.

* + Notification.Builder setContentText (CharSequence text)：

Set the second line of text in the platform notification template.

* + Notification.Builder setSmallIcon (int icon)：

Set the small icon, which will be used to represent the notification in the status bar and content view.

* + Notification build ()：

Combine all of the options that have been set and return a new Notification object.

* 發送通知之基本步驟
  + 在Activity呼叫 android.content.ContextWrapper 的 Object getSystemService (String name (Context.NOTIFICATION\_SERVICE) )方法以取得android.app.**NotificationManager**物件
  + 呼叫 NotificationManager物件的 void notify (int id,Notification notification)方法以發送通知(顯示在狀態列上)
    - 參數id： An **identifier** for this notification unique within your application.
    - 參數notification：A **Notification** object describing what to show the user.
* 範例：

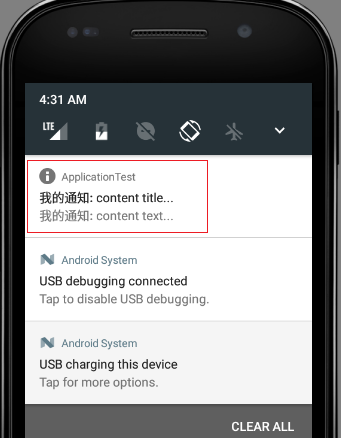
Notification notification = **new** Notification.Builder(**this**) .setSmallIcon(android.R.drawable.***ic\_dialog\_info***)  
 .setContentTitle(**"我的通知: content title..."**)  
 .setContentText(**"我的通知: content text..."**)  .build();NotificationManager notificationManager =

(NotificationManager)**this**.getSystemService(

Context.***NOTIFICATION\_SERVICE***);  
notificationManager.notify(1,notification);



* 點擊「送出通知」按鈕即送出通知



* 開啟「通知匣」即可查看通知之詳細資料

#### 隨堂測驗

|  |
| --- |
| (1)  如果要發送訊息到通知中心，必須使用下列哪種類別？  (A) Toast  (B) AlertDialog  (C) MessageService  (D) NotificationManager  答案: D |

### Options Menu

* **選項選單**為Activity主要的選單項目集合。 在Android 3.0 (API level 11) 以上版本，**選項選單**顯示在動作列(Action bar/App bar)上。
* 在Activity設定**選項選單**需要覆寫**onCreateOptionsMenu()**方法，Android系統會在Activity被啟動後呼叫此方法。
  + boolean onCreateOptionsMenu (Menu menu)
* 參數menu： The options menu in which you place your items. **( You should place your menu items into menu.)**
* 回傳boolean： You must return **true** for the menu to be **displayed**; if you return false it will not be shown.
* 當使用者點選某個選單項目時，Android系統會呼叫Activity的onOptionsItemSelected()方法
  + boolean onOptionsItemSelected (MenuItem item)
* 參數item： The menu item that was selected.
* 回傳boolean： Return false to allow normal menu processing to proceed, true to consume it here.
* 範例：

res/menu/menu.xml

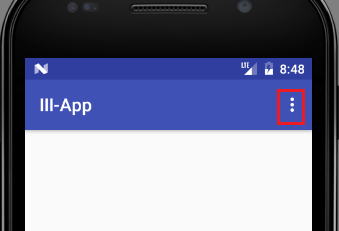
*<?***xml version="1.0" encoding="utf-8"***?>*<**menu xmlns:android="http://schemas.android.com/**

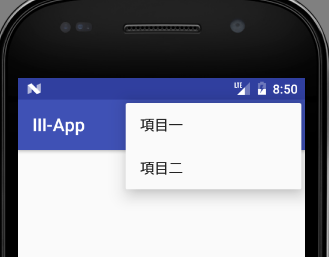
**apk/res/android"**>  
 <**item android:title="項目一" android:id="@+id/mi1"** />  
 <**item android:title="項目二" android:id="@+id/mi2"** />  
</**menu**>

MainActivity.java

**package** com.example.test;  
**import** android.os.Bundle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.view.Menu; **import** android.view.MenuInflater;  
**import** android.view.MenuItem; **import** android.widget.Toast;  
**public class** MainActivity **extends** AppCompatActivity {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 }  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {  
 MenuInflater inflater = getMenuInflater();  
 inflater.inflate(R.menu.***menu***, menu);  
 **return true**;  
 }  
 @Override  
 **public boolean** onOptionsItemSelected(MenuItem item) {  
 Toast.*makeText*(**this**, item.getTitle()+**" 被點擊了 ..."** ,

Toast.***LENGTH\_SHORT***).show();  
 **return false**;  
 }  
}







#### 隨堂測驗

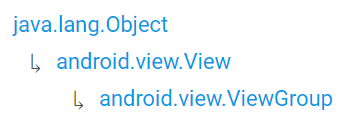
|  |
| --- |
| (1)  建立功能選單，必須透過下列何種事件？   1. onCreate() 2. onCreateOptionsMenu() 3. onCreateMenu() 4. onCreateMenuItem()   答案: B |

## Layout布局與應用

### Android提供的主要的版面配置元件

* LinearLayout (線性方式)
  + 下列XML版面配置檔之屬性可用來設定LinearLayout為水平(row)或垂直(column)的線性版面配置模式
    - android:**orientation**="vertical | horizontal"
* RelativeLayout (相對位置)
  + 下列XML版面配置檔之**屬性**可用來設定GUI元件的相對位置
    - android:layout\_above="@+id/其他UI元件's ID"
    - android:layout\_ below="@+id/其他UI元件's ID"
    - android:layout\_toLeftOf="@+id/其他UI元件's ID"
    - android:layout\_toRightOf="@+id/其他UI元件's ID"
    - android:layout\_alignTop="@+id/其他UI元件's ID"
    - android:layout\_alignBottom="@+id/其他UI元件's ID"
    - android:layout\_alignLeft="@+id/其他UI元件's ID"
    - android:layout\_alignRight="@+id/其他UI元件's ID"
    - android:layout\_align**Parent**Top="true|false"
    - android:layout\_align**Parent**Bottom="true|false"
    - android:layout\_align**Parent**Left="true|false"
    - android:layout\_align**Parent**Right="true|false"
    - android:layout\_**centerHorizontal**="true|false"
    - android:layout\_**centerVertical**="true|false"
    - android:layout\_**centerInParent**="true|false"
* TableLayout (表格(列、欄)方式)
* GridLayout (網格矩陣方式)
* FrameLayout (一個GUI元件佔一個圖層)

### Android版面配置元件共同的父類別



### 隨堂測驗

|  |
| --- |
| (1)  下列何者不是Android支援的版面配置元件？  (A) LinearLayout  (B) RelativeLayout  (C) GridbagLayout  (D) TableLayout  答案: C |

|  |
| --- |
| (2)  下列何者不是Android預設的Layout元件？   1. FrameLayout 2. TabLayout 3. LinearLayout 4. RelativeLayout   答案: B |

|  |
| --- |
| (3)  Android開發上，為了減少程式設計師在UI版面設計所花費的時間，Android提供了不同的layout元件來輔助程式設計師進行開發，下列為Android開發上常用的layout元件特性說明，何者描述有誤？  (A) LinearLayout，以線性方式呈現UI元件  (B) RelativeLayout，以相對位置呈現UI元件  (C) FrameLayout，以列和欄方式呈現UI元件  (D) Gridlayout，以網格矩陣方式呈現UI元件  答案: C |

|  |
| --- |
| (4)  在UI版面設計中，於父元件RelativeLayout中加入一個Button，如需將Button設置於RelativeLayout下方，在<Button>內設定何者正確？  (A) android:layout\_alignParentBottom="true"  (B) android:layout\_alignParentBottom="@+id/parent\_layout"  (C) android:layout\_alignBottom="@+id/parent\_layout"  (D) android:gravity="bottom"  答案: A |

|  |
| --- |
| (5)  Android開發上，為了減少程式設計師在UI版面設計所花費的時間，Android提供了不同的layout元件來輔助程式設計師進行開發。關於Android開發上常用的layout元件特性，下列敘述何者正確？  (A) Gridlayout，以網格矩陣方式呈現UI元件  (B) RelativeLayout，以絕對位置呈現UI元件  (C) FrameLayout，以列和欄方式呈現UI元件  (D) LinearLayout，以非線性方式呈現UI元件  答案: A |

|  |
| --- |
| (6)  關於UI中各Layout元件的介紹，下列敘述何者不正確？  (A) LinearLayout–以線性方式（垂直或水平排列）呈現UI元件  (B) FrameLayout–用來放置一個子元件，如放置多個會採推疊方式疊在一起  (C) RelativeLayout–描述一個UI元件在其他地方附近元件的對應位置  (D) TabLayout–表列式排列UI元件  答案: D |

## Activity運用與生命週期

### 重點摘要

* Activity為Android應用程式元件之一，主要角色為作為使用者與應用程式進行互動的**進入點**(entry point)， 它會建立一個**視窗**讓我們使用**setContentView(View)方法**來設定版面配置與放置UI物件
* Activity的組成可分成視覺介面與程式邏輯兩個部分 \*
  + Activity的視覺介面對應到XML版面配置檔
  + Activity的程式邏輯對應到Java程式檔
* Activity的生命週期(lifetime)



* Activity啟動其他Activity
  + new Intent (Context packageContext, Class<?> cls) 建構子以建立android.content.Intent物件
  + 在Activity中呼叫 android.content.ContextWrapper的void startActivity (Intent intent)方法以啟動一個新的Activity
  + 範例：

android.content.Intent intent =

**new** Intent(MainActivity.**this**, Main2Activity.**class**);  
startActivity(intent);

* Activity可在Intent中夾帶額外資料傳送給其他Activity
  + 呼叫Intent物件的 Intent putExtra (String name, byte value)方法可將額外資料附加到Intent裡
    - value引數亦可為byte、short、int、long、char、float、double、boolean、String資料型別與這些型別的陣列
    - 這些額外資料實際上儲存成android.os.Bundle物件
  + 呼叫Intent物件的 Bundle getExtras ()方法可取得Intent內所儲存的含有額外資料的Bundle物件。呼叫Bundle物件的下列方法可以取得Bundle物件內的額外資料
    - byte getByte (String key)
    - byte getByte (String key, byte defaultValue)
    - byte[] getByteArray (String key)
    - 其他getShort()、getInt()、getLong()、getChar()、getFloat()、getDouble()、getBoolean()、getString()方法可類推
  + 範例：

MainActivity

android.content.Intent intent =

**new** Intent(MainActivity.**this**, Main2Activity.**class**);  
 intent.putExtra(**"message"**,

**"Welcome to android programming..."** );  
 startActivity(intent);

Main2Activity

Bundle bundle = **this**.getIntent().getExtras();  
System.***out***.printf(**"message=%s\n"**,

bundle.getString(**"message"**) );

輸出結果：

I/System.out: message=Welcome to android programming...

### 隨堂測驗

|  |
| --- |
| (1)  關於Android Activity的生命週期，下列敘述何者有誤？  (A) Activity啟動：onCreate() -> onStart() -> onResume()  (B) Activity(1)呼叫Activity(2)啟動：onPause(1) -> onCreate(2) -> onStart(2) - onResume(2) -> onStop(1)  (C) Activity結束或退出：onDestroy() -> onStop()  (D) Activity於執行狀態下，旋轉螢幕：onPause() -> onStop() -> onDestroy() -> onCreate() -> onStart() -> onResume()  答案: C |

|  |
| --- |
| (2)  在Activity之間傳遞資料，可以透過下列何種類別達到？  (A) Extra  (B) Bundle  (C) Builder  (D) Toast  答案: B |

|  |
| --- |
| (3)  若一個Android App從Activity A跳至Activity B時，請問以下哪一個方法一定不會在Activity A內被執行？   1. onStop() 2. onDestroy() 3. onPause() 4. onRestart()   答案: D |

|  |
| --- |
| (4)  開發應用程式時常常會需要將資料從一個Activity傳遞到另一個Activity，例如在瀏覽器中點擊一組電話號碼，則會立即跳至撥號畫面，要達到此一設計，會用到下列哪些物件？   1. Intent、Bundle 2. Notification 3. Shared Preferences 4. ContentProvider   答案: A |

|  |
| --- |
| (5)  在Android開發上，如需鎖定畫面為直式不旋轉，下列設定何者正確？   1. 於Activity的onCreate()方法中setContentView()之後加入 getWindow().setFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN,WindowManager.LayoutParams.FLAG\_FULLSCREEN) 2. 於Activity的onCreate()方法中setContentView()之後加入 setRequestedOrientation(ActivityInfo.SCREEN\_ORIENTATION\_PORTRAIT) 3. 於Layout XML檔內<RelativeLayout>設定android:screenOrientation="portrait" 4. 以上設定方式皆可   答案: B |

|  |
| --- |
| (6)  如要固定螢幕方向為水平，要設置screenOrientation的參數為？  (A) landscape  (B) unspecified  (C) portrait  (D) horizontal  答案: A |

|  |
| --- |
| (7)  Android開發上常需使用轉頁的功能，也就是Activity1啟動另一個Activity2運作，下列程式何者正確？  (A) startActivity(Activity2.this);  (B) startActivityForResult(Activity2.this, 1);  (C) startActivity(new Intent(Activity1.this, Activity2.class));  (D) 以上皆正確  答案: C |

## Touch事件

### 重點摘要

* Touch事件觸發於使用者在裝置螢幕上**按下**手指或**放開**手指的時機
* android.view.View類別的下列方法可用來註冊Touch事件監聽器
  + void setOnTouchListener (View.OnTouchListener l)：Register a callback to be invoked when a touch event is sent to this view.

|  |
| --- |
| View.OnTouchListener(介面)  abstract boolean onTouch (View v, MotionEvent event);   * 參數v：The view the touch event has been dispatched to. * 參數event：The MotionEvent object containing full information about the event. * 回傳boolean：True if the listener has consumed the event, false otherwise. |

* 範例

ImageView iv = (ImageView) findViewById(R.id.***imageView***);  
iv.setOnTouchListener(

**new** View.OnTouchListener(){  
 @Override  
 **public boolean** onTouch(View view, MotionEvent motionEvent) {  
 **if**(motionEvent.getAction()==MotionEvent.***ACTION\_DOWN***)  
 System.***out***.println(**"Touch down..."**);  
 **else if** (motionEvent.getAction()==MotionEvent.***ACTION\_UP***)  
 System.***out***.println(**"Touch up..."**);  
 **return false**; //return true: 後續之onClick()方法不回被呼叫  
 }  
});

iv.setOnClickListener( **new** View.OnClickListener(){  
 @Override  
 **public void** onClick(View view) {  
 System.***out***.println(**"onClick..."**);  
 }  
});

在ImageView點擊後之輸出結果：

I/System.out: Touch down...

I/System.out: Touch up...

I/System.out: onClick...

### 隨堂測驗

|  |
| --- |
| (1)  下列有關Android之觸控事件監聽器的描述何者是錯誤的？  (A) android.view.View.OnTouchListener可用來註冊Touch事件監聽器  (B) android.view.View.OnTouchListener的型別是介面(interface)  (C) android.view.View.OnTouchListener定義了onTouch抽象方法  (D) 以上皆非  答案: D |

## MVC架構

### 重要摘要

* Android讓Activity應用程式元件的開發者可以使用**XML版面配置檔**來定義使用者介面的視覺結構。
* 在XML中宣告UI可以讓程式設計師將應用程式的**顯示畫面**與控制應用程式行為的**程式碼**區隔開來。 如此程式設計師即可修改或調整使用者介面說明，而不必修改並重新編譯原始碼。
* 上述作法大致實踐了MVC模式
* MVC模式(Model-view-controller)是軟體工程中的一種軟體架構模式。它把軟體系統分為三個基本部分
  + Model(模型)：執行企業邏輯與保存應用程式狀態 (Android's Activity以外**其他Java類別**對應Model)
  + View(視圖)：執行使用者介面(畫面)呈現 (Android's **XML版面配置**對應View)
  + Controller(控制器)：接收並處理使用者介面引發的事件與使用者輸入 (Android's **Activity**對應Controller)

### 隨堂測驗

|  |
| --- |
| (1)  在開發Android應用程式時，會建議使用一種軟體架構模式－MVC，關於MVC的敘述，下列何者不正確？   1. M代表Model（模型），可用於儲存資料的地方 2. V代表View（視圖），負責顯示畫面和圖示的元件 3. C代表Controller（控制器），控制程式的流程 4. Activity屬於Model的範圍   答案: D |

## 字串處理

### String類別的字串處理

#### 字串串連運算子 +

* 字串串連算子 (+) 會把兩個字串值串連在一起，返回另一個以兩個運算元字串連結的字串。
  + 範例：

String str = **"Hello "** + **"Android"**;  
System.***out***.println(str);

輸出結果：

I/System.out: Hello Android

#### 隨堂測驗

|  |
| --- |
| (1)  String str = "520";  str += 13;  label.setText(str);  請問label會顯示什麼？  (A) 520  (B) 13  (C) 533  (D) 52013  答案: D |

#### String類別建構子

* String (byte[] bytes, String charsetName) throws java.io.UnsupportedEncodingException

Constructs a **new String** by decoding(解碼) the specified array of bytes using **the specified charset**.

* String (byte[] bytes)

Constructs a **new String** by decoding(解碼) the specified array of bytes using **the platform's default charset(UTF-8)**.

#### String類別實體方法: getBytes()

* byte[] getBytes (String charsetName) throws java.io.UnsupportedEncodingException

Encodes(編碼) this String into a sequence of bytes using **the named charset**, storing the result into a new byte array.

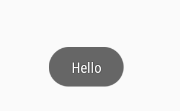
* byte[] getBytes ()

Encodes(編碼) this String into a sequence of bytes using **the platform's default charset(UTF-8)**, storing the result into a new byte array.

* + 範例：

**try** {  
 **byte**[] bytes = **"Hello"**.getBytes(**"utf-8"**);  
 String str = **new** String(bytes,**"utf-8"**);  
 Toast.*makeText*(MainActivity.**this**,str,

Toast.***LENGTH\_LONG***).show();  
} **catch** (UnsupportedEncodingException e) {  
 e.printStackTrace();  
}



#### getBytes() 隨堂測驗

|  |
| --- |
| (1)  下列Android程式片段之空格處，應如何填寫才能讓程式執行後str所參考的字串物件的內容維持為"程式設計"？  String str = "程式設計";  try {  str = new String( str.getBytes("big5"), \_\_\_\_\_\_\_\_\_\_\_\_ );  } catch (java.io.UnsupportedEncodingException e){  e.printStackTrace();  }  (A) "big5"  (B) "shift-jis"  (C) "iso-8859-1"  (D) "ascii"  答案: A |

|  |
| --- |
| (2)  下列程式片段執行以後，str所參考的字串內容何者正確？  try {  String str = new String( "程式設計".getBytes(), "utf-8" );  } catch (UnsupportedEncodingException e) {  e.printStackTrace();  }  (A) "程式設計"  (B) 亂碼  (C) "程 式 設 計"  (D) 以上皆非  答案: A |

#### String類別實體方法: matches()

* boolean matches (String regex)

Tells whether or not this string matches(匹配) the given regular expression.

* + 正規運算式語法摘要

|  |  |
| --- | --- |
| Predefined character classes | |
| . | Any character |
| \d | A digit: [0-9] |
| \s | A whitespace character: [ \t\n\x0B\f\r] |
| \w | A word character: [a-zA-Z\_0-9] |

|  |  |
| --- | --- |
| Character classes | |
| [abc] | a, b, or c (simple class) |
| [^abc] | Any character except a, b, or c (negation) |
| [a-zA-Z] | a  through z  or  A  through  Z, inclusive (range) |

|  |  |
| --- | --- |
| Greedy quantifiers | |
| *X*? | *X*, once or not at all |
| *X*\* | *X*, zero or more times |
| *X*+ | *X*, one or more times |
| *X*{*n*} | *X*, exactly *n* times |
| *X*{*n*,} | *X*, at least *n* times |
| *X*{*n*,*m*} | *X*, at least *n* but not more than *m* times |

|  |  |
| --- | --- |
| Boundary matchers | |
| ^ | The beginning of a line |
| $ | The end of a line |

* + 範例：

String str = **"17#3388"**;  
System.***out***.printf(**"%b\n"**,

str.matches(**"[01][78]#\\d{2}8{2,3}"**)); //\*

輸出結果：

I/System.out: true

#### String類別實體方法: split()

* String[] split (String regex)

Splits this string around matches of the given **regular expression(正規運算式)**.

* + 範例：

String str = **"Welcome,\*to\*\*Android$programming"** ;  
String[] strs = str.split(**"[,$\*]+"**);  
**for**(String s : strs){  
 System.***out***.println(s);  
}

輸出結果：

I/System.out: Welcome

I/System.out: to

I/System.out: Android

I/System.out: programming

#### split() 隨堂測驗

|  |
| --- |
| (1)  下列Android程式片段執行結果的相關描述，何者正確？  String str = "";  for(String s: "AA,BB;CC.DD".split("[,;.]")){  str += s+"-";  }   1. str所參考的字串物件的內容為AABBCCDD 2. str所參考的字串物件的內容為AA-BB-CC-DD- 3. str所參考的字串物件的內容為AABB-CCDD 4. 以上皆非   答案: B |

#### String類別實體方法: replaceFirst()

* String replaceFirst (String regex, String replacement)

Replaces the first substring of this string that matches the given regular expression with the given replacement..

* + 範例：

String str = **"AAA.jpgBBB.jpg"**;  
System.***out***.printf(**"%s\n"**,

str.replaceFirst(**"\\w+.\\w{3,4}$"**,**"\*\*\*"**)); // \*

輸出結果：

I/System.out: AAA.\*\*\*

#### String類別實體方法: substring()

* String substring (int beginIndex/起始索引, int endIndex/結束索引+1)

Returns a new string that is a substring(子字串) of this string. (The length(長度) of the substring is endIndex-beginIndex.)

* + 範例：

String str = **"歡迎Welcome "**.substring(2,5);  
System.***out***.printf(**"%s\n"**, str);

輸出結果：

I/System.out: Wel

#### String類別實體方法: trim()

* String trim ()

Returns a copy of the string, with leading(前) and trailing(後) whitespace(含\t \r \n space) omitted(刪去).

* + 範例：

String str = **"\n 歡迎光臨\t\r "**.trim();   
System.***out***.printf(**"\*\*%s\*\*\n "**, str);

輸出結果：

I/System.out: \*\*歡迎光臨\*\*

#### substring() / trim() 隨堂測驗

|  |
| --- |
| (1)  下列Android程式片段執行結果的相關描述，何者正確？  String str = " android ".trim() + "programming".substring(0,7);  (A) str所參考的字串物件的內容為android programm  (B) str所參考的字串物件的內容為androidprogram  (C) str所參考的字串物件的內容為androidprogramming  (D) str所參考的字串物件的內容為android programming  答案: B |

#### String類別實體方法: indexOf()

* int indexOf (String str)

Returns the index within this string of the first occurrence of the specified substring.

#### String類別實體方法: toUpperCase()

* String toUpperCase ()

Converts all of the characters in this String to upper case using the rules of the default locale.

#### substring() / indexOf() / toUpperCase() 隨堂測驗

|  |
| --- |
| (1)  假設下列程式片段的MainActivity類別為一個Android應用程式的launcher activity，且此應用程式執行後將顯示一個訊息交談窗。那麼依據程式內容可得知，下列何者為交談窗上所顯示的訊息？  String str="android\*program";  str = str.substring(0, str.indexOf('\*')).toUpperCase();  new android.app.AlertDialog.Builder(this)  .setMessage(str).show();   1. program 2. PROGRAM\* 3. ANDROID 4. ANDROID\*   答案: C |

#### String類別實體方法: toCharArray()

* char[] toCharArray ()

Converts this string to a new character array.

* + 範例：

**char**[] chars = **"歡迎光臨"**.toCharArray(); //\*  
System.***out***.printf(**"%c-%c\n "**, chars[0],chars[2]);

輸出結果：

I/System.out: 歡-光

#### String類別靜態方法: format()

* String format (String format, Object... args)

Returns a formatted string using the specified **format string** and arguments.

* + 格式指定器(Format Specifier)：

% [flags][width][.precision]conversion

conversion: 轉換字元

precision: 小數位數

width: 最小寬度/字元數

flags: 控制旗標(**，**表三位數一個逗號 **0**表空白以0取代)

* + - 數目轉換字元

|  |  |
| --- | --- |
| 'd' | 整數 |
| 'f' | 浮點數 |

String str = String.*format*(**"%0,15.1f"**,

11223344.789);



* + - 日期轉換字元

|  |  |
| --- | --- |
| 'Y' | 四碼年. |
| 'y' | 兩碼年. 00 - 99. |
| 'm' | 兩碼月. 01 - 13. |
| 'd' | 兩碼日. 01 - 31 |
| 'e' | 一碼日. 1 - 31. |

* + - 時間轉換字元

|  |  |
| --- | --- |
| 'H' | 24小時制之兩碼時. 00 - 23. |
| 'I' | 12小時制之兩碼時. 01 - 12. |
| 'k' | 24小時制之一碼時 0 - 23. |
| 'l' | 12小時制之一碼時. 1 - 12. |
| 'M' | 分. 00 - 59. |
| 'S' | 秒. 00 - 60 ("60"為特殊狀況:閏秒). |

#### format()隨堂測驗

|  |
| --- |
| (1)  下列Android程式片段執行結果的相關描述，何者正確？  java.util.Calendar c = java.util.Calendar.getInstance();  c.set(2000,1,2);  String str = String.format("%tY-%<tm-%<td", c);  (A) str所參考的字串物件的內容為2000-01-02  (B) str所參考的字串物件的內容為00-1-2  (C) str所參考的字串物件的內容為2000-02-02  (D) 以上皆非  答案: C |

### StringBuffer類別的字串處理

#### StringBuffer類別實體方法: insert()、append()、toString()

* StringBuffer insert (int offset, String str)：

Inserts the string into this character sequence.

* StringBuffer append (String str)：

Appends the specified string to this character sequence.

* String toString ()：

Returns a string representing the data in this sequence

* + 範例：

String str = **new** StringBuffer().insert(0,**"Hello"**).append(**", my friend..."**).toString(); //\*  
System.***out***.println(str);

輸出結果：

I/System.out: Hello, my friend…

### StringWriter類別的字串處理

A character stream that collects its output in a **string buffer**, which can then be used to construct a string.

#### StringWriter類別實體方法: write()、append()、toString()

* void write (String str)： Write a string.
* StringWriter append (CharSequence csq)：

Appends the specified character sequence to this writer

* String toString ()：

Return the buffer's current value as a string.

#### StringWriter隨堂測驗

|  |
| --- |
| (1)  下列Android程式片段執行結果的相關描述，何者正確？  StringWriter sw = new StringWriter();  sw.write("行動裝置");  sw.write("程式設計師");  String str = sw.toString();   1. str所參考的字串物件的內容為"行動裝置" 2. str所參考的字串物件的內容為"程式設計師" 3. str所參考的字串物件的內容為"行動裝置程式設計師" 4. 程式未作例外處理以致無法正常編譯   答案: C |

### Integer類別的字串處理

The Integer class wraps a value of the primitive type int in an object.

#### Integer類別靜態方法: parseInt()

* static int parseInt (String s)：

Parses the **string** argument as a signed decimal **integer**

#### Integer.parseInt()隨堂測驗

|  |
| --- |
| (1)  Android中很常遇到字串的處理，請問String num = "123456"; 如何把num型態轉換為int？   1. int number = (int)num; 2. int number = num.split("1"); 3. int number = num.indexOf("1"); 4. int number = Integer.parseInt(num);   答案: D |

## 開發時的圖檔與資源檔案安排

### 重點摘要

* Android提供一個機制將**應用程式**所使用的**資源**(如圖片、字串等)外部化，以便個別加以維護。
* Android應用程式**資源型別**(Resource Type)主要有

**res/**

**drawable/**

**graphic.png**

**layout/**

**layout\_main.xml**

**mipmap/**

**icon.png**

* 此資料夾用來存放應用程式之launcher icons
* Android會以最佳的解析度來顯示該圖示

**values/**

**strings.xml**

* 內含string、string-array、color、dimen、style等資源

**menu/**

**menu\_main.xml**

**raw/**

**tenyears.mp3**

* 以原始格式儲存的任意檔案
* Android將應用程式的GUI元件歸類於id資源類別
* Android內建許多圖形資源，在XML版面配置檔裡可使用 **@android:drawable/圖形檔主檔名** 之格式指定Android內建之圖形

### 隨堂測驗

|  |
| --- |
| (1)  Android將螢幕版面上的GUI元件都歸屬於一個資源類別，該資源類別為下列那一項？  (A) array  (B) id  (C) string  (D)drawable  答案: B |

|  |
| --- |
| (2)  在UI版面設計常需使用到各種不同的icon圖示或圖片，試問這些圖示或圖片須放置於Android專案的何種資料夾？   1. /res/layout/ 2. /res/values/ 3. /libs/ 4. /res/drawable/   答案: D |

|  |
| --- |
| (3)  Android系統內建許多圖形資源，有關在版面配置檔案裡指定內建圖形的格式，下列何者正確？  (A) @drawable/圖形檔主檔名  (B) @photo/圖形檔主檔名  (C) @image/圖形檔主檔名  (D) @graphics/圖形檔主檔名  答案: A |

|  |
| --- |
| (4)  Android專案下的values資料夾內，存放何種文件？  (A) 多國語字串或顏色等數據  (B) 多媒體文件  (C) UI布局文件的XML檔案  (D) App內建的圖片檔  答案: A |

# Android - I/O輸出入串流

## 資料儲存空間與檔案管理

### Android檔案系統

* 所有 Android 裝置都有兩個檔案儲存區域：內部儲存空間(Internal Storage)與外部儲存空間(External Storage)。
* **內部儲存空間**一般指行動裝置提供的內建靜態記憶體; **外部儲存空間**一般指卸除式儲存媒體(如SD卡)。
* 內部儲存空間具有下列特性
  + 一個行動裝置必定具備內部儲存空間。
  + 預設只有**應用程式**能存取它在內部儲存空間內所儲存的檔案。
  + 使用者**解除安裝**一個應用程式時，系統**會**從內部儲存空間移除該應用程式的所有檔案。
* 外部儲存空間具有下列特性
  + 一個行動裝置不一定具備外部儲存空間。
  + 其他應用程式可以存取外部儲存空間，因此我們無法控制其他應用程式存取儲存在該空間內的檔案。
  + 使用者**解除安裝**一個應用程式時，基本上系統**不會**從外部儲存空間移除該應用程式的檔案。

#### 隨堂測驗

|  |
| --- |
| (1)  關於Android資料存取機制External Storage，下列敘述何者錯誤？  (A) 可將檔案資料寫至外部可移除的儲存媒體上(SD card)  (B) 該檔案資料可以被其他應用程式修改  (C) 安全性與私密性較低  (D) 檔案資料於外部儲存媒體上，如移除應用程式，該檔案資料也會 一併移除  答案: D |

|  |
| --- |
| (2)  Android一般分成內部與外部兩種儲存空間，下列敘述何者不正確？  (A) 應用程式可將大量資料存放於外部儲存空間  (B) 依預設，只有您的應用程式能存取內部空間中儲存的檔案  (C) 使用者解除安裝您的應用程式時，系統會從內部儲存空間自動移除您應用程式的檔案  (D) 您可以讀取其他應用程式的內部儲存空間  答案: D |

### 求取Android外部儲存空間相關目錄之API

* 呼叫 android.os.Environment之 File getExternalStorageDirectory () 靜態方法可以取得主要的外部儲存空間目錄

**範例1：**

System.***out***.printf(**"%s\n"**, Environment.getExternalStorageDirectory() );  
輸出結果如： /storage/emulated/0或 /storage/sdcard0

**範例2：**

**try** {  
 File file =

**new** File( Environment.getExternalStorageDirectory() ,

**"test.txt"**);  
 FileOutputStream fos = **new** FileOutputStream(file);  
 BufferedWriter bw = **new** BufferedWriter(

**new** OutputStreamWriter(fos));

//\* java.io.OutputStreamWriter可依預設或指定的字元編碼進行串流型別轉換(即將輸出資料從character stream轉成byte stream再作輸出)

//註: OutputStreamWriter常用建構子：

//OutputStreamWriter (OutputStream out [, String charsetName]) \*

bw.write(**"歡迎來學Android!"**);  
 bw.close();  
  
 FileInputStream fis = **new** FileInputStream(file);  
 BufferedReader br = **new** BufferedReader(

**new** InputStreamReader(fis));

|  |
| --- |
| 或  BufferedReader br = **new** BufferedReader(  **new** FileReader(file) ); |

String line = **null**;  
 **while**((line=br.readLine()) != **null**){  
 System.***out***.printf(**"%s\n"**, line);  
 }  
 br.close();  
} **catch** (FileNotFoundException e) {  
 e.printStackTrace();  
} **catch** (IOException e) {  
 e.printStackTrace();  
}

|  |
| --- |
| 註:   1. 應用程式欲在外部儲存空間寫入資料，必須先取得"android.permission.WRITE\_EXTERNAL\_STORAGE"許可 2. FileOutputStream/FileInputStream 使用Byte Streams (即以byte為單位) 進行IO 3. FileWriter/FileReader 使用Character Streams (即以character為單位) 進行IO |

輸出結果如： I/System.out: 歡迎來學Android!

* 呼叫 android.os.Environment之 File getExternalStoragePublicDirectory (String type) 靜態方法可以取得系統預設用以存放特定類別(如: Pictures、Movies、Music等)檔案的外部儲存空間目錄

**範例：**

System.***out***.printf(**"%s\n"**, Environment.getExternalStoragePublicDirectory(

Environment.*DIRECTORY\_PICTURES* )

);

輸出結果如：

/storage/emulated/0/Pictures或 /storage/sdcard0/Pictures

#### 隨堂測驗

|  |
| --- |
| (1)  以下何者是Android App取得外部儲存空間目錄的方法？  (A) Activity.getExternalFilesDir();  (B) Environment.getExternalStorageDirectory();  (C) Environment.getExternalDirectory();  (D) Activity.getExternalStorageDirectory();  答案: B |

|  |
| --- |
| (2)  下列程式片段執行結果，何者是正確的？  try {  File file =new File( Environment.getExternalStorageDirectory() ,  "test.txt");  FileOutputStream fos = new **FileOutputStream(file)**;  BufferedWriter bw = new BufferedWriter(  new OutputStreamWriter(fos));  bw.write("歡迎來學Android!");  bw.close();  } catch (FileNotFoundException e) {  e.printStackTrace();  } catch (IOException e) {  e.printStackTrace();  }  (A)程式執行後會在內部儲存空間建立一個檔案  (B) FileOutputStream的write()方法可以直接寫出字串資料  (C)應用程式需要先取得"android.permission.READ\_EXTERNAL\_STORAGE"許可才能正常執行  (D)以上皆非  答案: D |

|  |
| --- |
| (3)  請問當在使用FileOutputStream寫入文字或物件至檔案中時，是以哪個單位來對檔案做存取和儲存？   1. byte 2. bit 3. word 4. dword   答案: A |

|  |
| --- |
| (4)  下列Android應用程式執行後，將讀取指定外部儲存媒體的檔案之第一行內容（假設該檔案已事先建立），其中空格處應該填寫那一個選項？  String str = "";  String fpath = android.os.Environment.  getExternalStorageDirectory().getAbsolutePath()  +"/aaa.txt";  Reader r = new \_\_\_\_\_\_\_\_\_\_\_\_\_\_(fpath);  BufferedReader br = new BufferedReader(r);  str = br.readLine();  (A) Reader  (B) FileReader  (C) FileInputStream  (D) FileInputReader  答案: B |

|  |
| --- |
| (5)  請判斷下列程式碼處理的事情？  File tempFile = new File("sdcard/temp");  if(!tempFile.exists()){  tempFile.mkdirs();  }  (A) 判斷路徑是否為資料夾  (B) 刪除該路徑下的檔案  (C) 檢查路徑的資料夾是否存在，如不存在則建立資料夾  (D) 取得文件大小  答案: C |

### 求取Android內部儲存空間相關目錄之API

* 呼叫 android.content.Context之 File getFilesDir ()方法可以取得以Android API: openFileOutput(String, int)方法建立檔案時預設存檔所在之**內部儲存空間目錄**.

**例如: /data/user/0/<App's package name>/files**

#### 隨堂測驗

|  |
| --- |
| (1)  以下何者是Android App取得內部儲存空間目錄的方法？   1. Activity.getFilesDir(); 2. Environment.getInternalStorageDirectory(); 3. Environment.getHomeDirectory(); 4. Activity.getInternalStorageDirectory();   答案: A |

### 使用SharedPreferences存取內部儲存空間的資料

* SharedPreferences本質上也是一個檔案，只是檔案內的資料為應用程式所**私有**專用的資料(這些資料儲存於Internal Storage)
* SharedPreferences儲存的資料的格式為鍵值對(Key-Value pairs)
* SharedPreferences使用方法
  + 儲存資料
    - 在Activity使用下列方法取得android.content. SharedPreferences(介面)物件

SharedPreferences getSharedPreferences

(String name, int mode) 或

SharedPreferences getPreferences (int mode)

* 參數name：偏好的檔名。
* 上述第二個版本(getPreferences)，預設檔名與路徑如下:

/data/data/package-name### /shared\_prefs/ Activity-name###.xml)

* 參數mode：操作模式。

主要設定值有:

* + Context.MODE\_PRIVATE (0)
* 只有呼叫該方法的App可以存取; 其他App或user不可存取
  + Context. MODE\_WORLD\_READABLE (1)
* 允許其他App讀取
* deprecated in API level 17(Android 4.2 Jelly Bean)
  + Context. MODE\_WORLD\_WRITEABLE (2)
* 允許其他App寫入
* deprecated in API level 17(Android 4.2 Jelly Bean)
  + - 呼叫SharedPreferences物件之edit()方法取得android.content.SharedPreferences.Editor(介面)物件
    - 利用SharedPreferences.Editor物件之下列方法設定偏好資料
      1. SharedPreferences.Editor putBoolean (String key, boolean value)
      2. SharedPreferences.Editor putInt (String key, int value)
      3. SharedPreferences.Editor putLong (String key, long value)
      4. SharedPreferences.Editor putFloat (String key, float value)
      5. SharedPreferences.Editor putString (String key, String value)
    - 利用SharedPreferences.Editor物件之commit() 或 apply()方法將上述異動資料寫回SharedPreferences

boolean commit ()

* 回傳值：true表示資料異動寫檔成功

void apply ()

* 回傳值：無 (非同步確認資料異動)
* apply() commits its changes to the in-memory SharedPreferences immediately but starts an asynchronous commit to disk and you won't be notified of any failures.

**範例：**

SharedPreferences.Editor editor =

getSharedPreferences(**"MainActivity"**,

Context.***MODE\_PRIVATE***(=0))

.edit();  
editor.putInt(**"Id"**,100).putString(**"Name"**,**"Peter"**)

.commit();

* + 讀取資料
    - 在Activity取得android.content. SharedPreferences(介面)物件
    - 利用SharedPreferences物件之下列方法讀取偏好資料

1. boolean getBoolean (String key, boolean defValue)
2. int getInt (String key, int defValue)
3. long getLong (String key, long defValue)
4. float getFloat (String key, float defValue)

(\*註: SharedPreferences不提供getDouble()方法)

1. String getString (String key, String defValue)
   * + 利用SharedPreferences物件之下列方法測試偏好資料(preferences)中是否包含某一項偏好(preference)

boolean contains (String key)

**範例：**

SharedPreferences sp =

getSharedPreferences(**"MainActivity"**,

Context.***MODE\_PRIVATE***);  
System.***out***.printf(**"Id=%d Name=%s\n"**,

sp.getInt(**"Id"**,0),

sp.getString(**"Name"**,**"N/A"**) );

輸出結果：

I/System.out: Id=100 Name=Peter

#### 隨堂測驗

|  |
| --- |
| (1)  使用 SharedPreferences 技術時，請問資料是被儲存於下列何種媒體？  (A) External Storage  (B) Internal Storage  (C) Cloud Storage  (D) Server  答案: B |

|  |
| --- |
| (2)  在程式中 SharedPreferences sp = getSharedPreferences("T", 0);  下列何者可以寫入"Hello"？  (A) sp.edit().putString("K","Hello").commit();  (B) sp.putString("K","Hello").commit();  (C) sp.edit().putString("K","Hello").submit();  (D) sp.putString("K","Hello").apply();  答案: A |

|  |
| --- |
| (3)  在Android開發上使用SharedPreferences來儲存檔案的內容，下列資料型態何者不能透過SharedPreferences儲存？   1. int 2. float 3. String 4. List   答案: D |

|  |
| --- |
| (4)  在SharedPreferences類別中，contains(String key)方法是用來判斷下列何者？   * 1. Preference是否有資料   2. Preference是否有刪除   3. Preference是否有更改   4. Preference是否有關閉   答案: A |

|  |
| --- |
| (5)  要使用SharedPreferences物件，首先要使用getSharedPreferences方法來建立SharedPreferences物件，其語法為SharedPreferences sp = getSharedPreferences(name, mode)，其中mode為儲存檔案的存取權限，下列敘述何者不正確？  (A) MODE\_PRIVATE：只有本應用程式有存取權限  (B) MODE\_WORLD\_READABLE：所有應用程式都具有讀取權限  (C) MODE\_WORLD\_WRITEABLE：所有應用程式都具有寫入權限  (D) MODE\_PUBLIC：所有應用程式都具有存取權限  答案: D |

|  |
| --- |
| (6)  使用SharedPreferences技術時，若要啟動同步寫入機制，請問要呼叫下列哪個方法？  (A) commit();  (B) apply()  (C) submit()  (D) save()  答案: A |

### 使用SQLite資料庫存取內部儲存空間的資料

* SQLite是一套開放原始碼的資料庫引擎， Android內建了SQLite功能讓Android App可以很方便地利用它儲存資料
* SQLite是一個輕量級的關聯式資料庫 (Relational Database)，支援Structured Query Language/SQL語言以存取資料庫資料
* SQLite主要API
  + 開啟/建立資料庫

A. 利用下列方法

android.context.ContextWrapper

SQLiteDatabase openOrCreateDatabase (

String name, int mode,

SQLiteDatabase.CursorFactory factory)

*//參數name: 資料庫名稱 / database file之名稱*

或

B. 利用繼承下列抽象類別的子類別所建立之物件

android.database.sqlite.SQLiteOpenHelper

例如：

**class** DbOpenHelper **extends SQLiteOpenHelper**{  
 **public** DbOpenHelper(Context context, String name,

SQLiteDatabase.CursorFactory factory, **int** version) {  
 **super**(context, name, factory, version);}  
 @Override  
 **public void** onCreate(SQLiteDatabase sqLiteDatabase) {

//本方法為SQLiteOpenHelper抽象方法  
 }  
 @Override  
 **public void** onUpgrade(SQLiteDatabase sqLiteDatabase,

**int** i, **int** i1) { //本方法為SQLiteOpenHelper抽象方法}  
}

之SQLiteDatabase getWritableDatabase () 方法

* + 執行SQL指令

android.database.sqlite.SQLiteDatabase

* void execSQL (String sql): 用於執行非Select SQL指令
* Cursor rawQuery (String sql, String[] selectionArgs): 用於執行Select SQL指令
  + 擷取查詢結果內容

android.database.Cursor(介面)

* short getShort (int columnIndex)
* int getInt (int columnIndex)
* long getLong (int columnIndex)
* float getFloat (int columnIndex)
* double getDouble (int columnIndex)
* String getString(int columnIndex)
* 範例1：

SQLiteDatabase db = openOrCreateDatabase(**"MyDB.db"**,   
 Context.***MODE\_PRIVATE***,**null**); //\* 資料庫名稱: MyDB.db  
db.execSQL(**"create table if not exists emp"** + //\* 資料表名稱: emp  
 **"(empid int, name varchar(15) )"**); //\* 欄位名稱:

empid、name  
db.execSQL(**"insert into emp "** +  
 **"values(1,'Michael')"**);  
Cursor cursor = db.rawQuery(

**"select \* from emp"**, **null**);  
**while**(cursor.moveToNext()){ //\*  
 System.***out***.printf(**"id=%d name=%s\n"**,

cursor.getInt(0), cursor.getString(1));  
}  
db.close();

輸出結果：

I/System.out: id=1 name=Michael

* 範例2：

**public class** MainActivity **extends** AppCompatActivity {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
  
 DbOpenHelper dbOpenHelper =

**new** DbOpenHelper(**this**,**"MyDB.db"**,**null**,1);

SQLiteDatabase db = dbOpenHelper.getWritableDatabase(); *//1.當資料庫尚不存在時,此時資料庫會被建立,*

*然後onCreate() 與 onOpen()方法會被呼叫*

*//2.當資料庫已存在時,則onOpen()方法會被呼叫  
 //3.資料庫建立後,若再度執行上面二敘述且*

*參數version增大,則onUpgrade()方法會被呼叫* db.execSQL(**"insert into emp values(1,'Mary')"**);

(略)  
 db.close();  
 }  
}  
  
**class** DbOpenHelper **extends** SQLiteOpenHelper{  
 **public** DbOpenHelper(Context context, String name,

SQLiteDatabase.CursorFactory factory, **int** version) {  
 **super**(context, name, factory, version); }  
 @Override  
 **public void** onCreate(SQLiteDatabase sqLiteDatabase) {String sql = **"create table if not exists emp(\_id int,**

**name varchar(20))"**;  
 sqLiteDatabase.execSQL(sql);  
 }  
 @Override  
 **public void** onUpgrade(SQLiteDatabase sqLiteDatabase, **int** i,

**int** i1) { }  
 @Override  
 **public void** onOpen(SQLiteDatabase db) {  
 }  
}

#### 隨堂測驗

|  |
| --- |
| (1)  假設有一段SQL指令：String sql="Select \* From tCustomers";  請問下列方法何者可以執行取得上述指令的查詢結果？  (A) rawQuery(sql)  (B) execSQL (sql)  (C) queryBySQL(sql)  (D) executeSQL(sql)  答案: A |

|  |
| --- |
| (2)  下列何者是Android平台內建支援的資料庫？  (A) MySQL  (B) SQL Compact Edition  (C) Firebird  (D) SQLite  答案: D |

|  |
| --- |
| (3)  假設一個Android應用程式的Activity正常地開啟了一個SQLite資料庫，之後欲使用SQL的查詢指令查詢該資料庫內特定資料表的資料，可以使用下列哪一項方法？   1. execSQL 2. query 3. rawQuery 4. select   答案: C |

|  |
| --- |
| (4)  請問當Android應用程式要存取SQLite資料庫時，必須繼承下列何項抽象類別？   * 1. SQLiteOpenHelper   2. AndroidSQLiteClass   3. SQLiteDatabase   4. SQLiteDataRead   答案: A |

|  |
| --- |
| (5)  假設下列程式片段為一個可以正常啟動執行的Android應用程式，那麼關於執行結果，下列敘述何者不正確？  SQLiteDatabase db = openOrCreateDatabase("db1",  Context.MODE\_PRIVATE, null);  db.execSQL("create table if not exists" +  " T1( name nvarchar(10))");  db.close();  (A) 程式會開啟或建立一個名稱為db1的SQLite資料庫  (B) db1資料庫內有一個名稱為T1的資料表  (C) T1資料表只具有一個名稱為name的欄位  (D) T1資料表的primary key欄位是 \_id  答案: D |

## 檔案輸出與輸入

### 使用openFileInput/Output方法進行內部儲存空間I/O

* 在Activity呼叫 android.content.ContextWrapper 的 FileInputStream openFileInput (String name) throws FileNotFoundException方法以讀取私有檔案

Open a private file associated with this Context's application package for **reading**.

* 在Activity呼叫 android.content.ContextWrapper 的 FileOutputStream openFileOutput (String name, int mode) throws FileNotFoundException方法以寫入私有檔案
* 參數name: 檔案名稱
* 參數mode: 操作模式，可設定之值主要有：
  + MODE\_PRIVATE(0) for the default operation. (代表原則上只有呼叫該方法的App可以存取; 其他App或user不可存取)
  + MODE\_APPEND to append to an existing file.

Open a private file associated with this Context's application package for **writing**. Creates the file if it doesn't already exist.

* 範例：

**try** {  
 *//寫檔* FileOutputStream fos = openFileOutput(**"file1.txt"**,0);  
 fos.write(**"歡迎光臨"**.getBytes());  
 fos.close();  
   
 *//讀檔* FileInputStream fis = openFileInput(**"file1.txt"**);  
 BufferedReader br = **new** BufferedReader(

**new** InputStreamReader( fis ) );  
 String str;  
 **while**( (str = br.readLine()) != **null** ){  
 System.***out***.println(str); //輸出 I/System.out: 歡迎光臨  
 }  
 br.close();  
} **catch** (FileNotFoundException e) {  
 e.printStackTrace();  
} **catch** (IOException e) {  
 e.printStackTrace();  
}

#### 隨堂測驗

|  |
| --- |
| (1)  下列Android應用程式執行後將讀取指定檔案內容(假設該檔案已事先建立)，其中空格處應該填寫那一個選項才能讓程式正確編譯與執行？  (假設下列MainActivity類別為launcher activity)  package com.example.xxx.test;  import java.io.\*; import android.app.\*; public class MainActivity extends Activity{  protected void onCreate(android.os.Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  String str = "";  try {  FileInputStream fis = this.**openFileInput("temp.txt")**;  BufferedReader br =  new BufferedReader( );  String line;  while((line=br.readLine()) != null){  str += line+"\n";  }  br.close();  } catch (IOException e) {  e.printStackTrace();  }  } }  (A) fis  (B) new InputStream(fis)  (C) new InputStreamReader(fis)  (D)new ReaderInputStream(fis)  答案: C |

|  |
| --- |
| (2)  使用openFileOutput(String p1,int p2)創建檔案時，若第二個參數p2傳入 Activity.MODE\_PRIVATE 請問代表意義為何？   1. 創建的檔案只限儲存於主記憶體 2. 創建的檔案只有呼叫該方法的程式能使用 3. 創建的檔案只有在該裝置上才能使用 4. 創建的檔案只有在該事件中才能使用   答案: B |

|  |
| --- |
| (3)  這是一段寫入檔案的程式碼：  OutputStream os = openFileOutput("t.txt",Activity.MODE\_PRIVATE);  請問下列何者可以在「t.txt」檔案中寫入”Hello” ？   1. os.persist("Hello".getBytes("utf-8")); 2. os.persist("Hello"); 3. os.write("Hello".getBytes("utf-8")); 4. os.write("Hello");   答案: C |

|  |
| --- |
| (4)  關於Android檔案輸出入API，下列敘述何者不正確？  (A) android.content.ContextWrapper.openFileInput方法可用來開啟為了讀檔用的應用程式自己私有的檔案  (B) android.content.ContextWrapper.openFileOutput方法的回傳資料型別為java.io.DataOutputStream  (C) android.content.ContextWrapper.openFileInput方法的回傳資料型別為java.io.FileInputStream  (D) android.content.ContextWrapper.openFileOutput方法可用來開啟為了寫檔用的應用程式自己私有的檔案  答案: B |

|  |
| --- |
| (5)  下列何者不是Android平台內建的資料存取技術？  (A) File I/O  (B) SQLite  (C) JDBC  (D) SharedPreferences  答案: C |

# Android - 內建裝置與系統服務

## 內建感應器

### 加速度感測器

* Android手機內建許多感測器(如加速度感測器、接近感測器等)，可用以偵測手機的移動、方向以及周遭環境的現況。
* Android 提供了一套API(Android sensor framework)，幫助我們在程式中執行與感測器相關的工作
  + 在Activity呼叫 android.content.ContextWrapper 的 Object getSystemService (String name (Context.SENSOR\_SERVICE) )方法以取得android.hardware.**SensorManager**(抽象類別) 物件
  + 呼叫 SensorManager物件的 Sensor getDefaultSensor (int type (Sensor.TYPE\_ACCELEROMETER))方法以取得android.hardware.**Sensor** 物件
  + 呼叫 SensorManager 物件的 boolean registerListener (SensorEventListener listener, Sensor sensor, int samplingPeriodUs) 方法為**加速度感測器**註冊感測器事件監聽器

***samplingPeriodUs****(取樣頻率): 一般設定SensorManager.SENSOR\_DELAY\_NORMAL(3) 約 0.2秒*

* + 透過android.hardware.SensorEventListener物件的onSensorChanged()方法的event參數可取得手機X軸、Y軸、Z軸的加速度值
    - void onSensorChanged (SensorEvent event)

event.values[0]：X軸的加速度值(單位: m/s^2)

event.values[1]：Y軸的加速度值(單位: m/s^2)

event.values[2]：Z軸的加速度值(單位: m/s^2)

( 註: *event.values為 float[] )*

* + - void onAccuracyChanged (Sensor sensor, int accuracy)
* 範例：

SensorEventListener sensorEventListener = **new** SensorEventListener() {  
 @Override  
 **public void** onSensorChanged(SensorEvent event) {  
 System.***out***.printf(**"x軸加速度=%.2f y軸加速度=%.2f**

**z軸加速度=%.2f\n"**,  
 event.**values**[0],event.**values**[1],event.**values**[2]);  
 }  
 @Override  
 **public void** onAccuracyChanged(Sensor sensor, **int** accuracy) {  
 }  
};  
SensorManager sm = (SensorManager)getSystemService(Context.***SENSOR\_SERVICE***);  
Sensor sr = sm.getDefaultSensor(Sensor.***TYPE\_ACCELEROMETER***);  
sm.registerListener(sensorEventListener,sr,SensorManager.***SENSOR\_DELAY\_NORMAL***);

輸出結果：

I/System.out: x軸加速度=0.00 y軸加速度=9.80 z軸加速度=0.00

I/System.out: x軸加速度=0.00 y軸加速度=9.80 z軸加速度=0.00

I/System.out: x軸加速度=0.00 y軸加速度=9.80 z軸加速度=0.00

. . .

#### 隨堂測驗

|  |
| --- |
| (1)  Android應用程式的Activity若欲讀取「加速度感測器」的值，一般需先取得對應「加速度感測器」的Sensor物件。  下列程式片段內的空格應該如何填寫才能使Activity正確取得「加速度感測器」對應之Sensor物件？  import android.content.Context;  import android.hardware.SensorManager;  import android.hardware.Sensor;  SensorManager sm = (SensorManager)getSystemService( (1) ); Sensor sr = sm.getDefaultSensor( (2) );  (A) (1)為Context.SENSOR\_SERVICE 且 (2)為SensorManager.ACCELEROMETER  (B) (1)為Context.SENSOR\_SERVICE 且 (2)為Sensor.TYPE\_ACCELEROMETER  (C) (1)為Sensor.TYPE\_ACCELEROMETER 且 (2)為Context.SENSOR\_SERVICE  (D) (1)為Context.ACCELEROMETER 且 (2)為Sensor.TYPE\_ACCELEROMETER  答案: B |

### 接近感測器

* API使用方法與加速度感測器類似

SensorManager sm = (SensorManager)getSystemService(Context.***SENSOR\_SERVICE***);  
Sensor sr = sm.getDefaultSensor(Sensor.***TYPE\_PROXIMITY***);  
sm.registerListener(sensorEventListener,sr,SensorManager.***SENSOR\_DELAY\_NORMAL***);

* 透過android.hardware.SensorEventListener物件的onSensorChanged(SensorEvent event)方法的event參數可取得手機與物件之距離：

event.values[0]：距離 (單位:cm)

* Android接近感測器屬於Binary Sensor; 實際上只能測得0.0(表示靠近)與5.0(表示遠離)

### 亮度感測器

* API使用方法與加速度感測器類似

SensorManager sm = (SensorManager)getSystemService(Context.***SENSOR\_SERVICE***);  
Sensor sr = sm.getDefaultSensor(Sensor.***TYPE\_LIGHT***);  
sm.registerListener(sensorEventListener,sr,SensorManager.***SENSOR\_DELAY\_NORMAL***);

* 透過android.hardware.SensorEventListener物件的onSensorChanged(SensorEvent event)方法的event參數可取得周遭亮度：

event.values[0]：亮度 (單位:lux/流明)

### 溫度感測器

* API使用方法與加速度感測器類似

SensorManager sm = (SensorManager)getSystemService(Context.***SENSOR\_SERVICE***);  
Sensor sr = sm.getDefaultSensor(Sensor. ***TYPE\_AMBIENT\_TEMPERATURE***);  
sm.registerListener(sensorEventListener,sr,SensorManager.***SENSOR\_DELAY\_NORMAL***);

* 透過android.hardware.SensorEventListener物件的onSensorChanged(SensorEvent event)方法的event參數可取得周遭溫度：

event.values[0]：溫度 (單位:攝氏°C)

### 濕度感測器

* API使用方法與加速度感測器類似

SensorManager sm = (SensorManager)getSystemService(Context.***SENSOR\_SERVICE***);  
Sensor sr = sm.getDefaultSensor(Sensor. ***TYPE\_RELATIVE\_HUMIDITY***);  
sm.registerListener(sensorEventListener,sr,SensorManager.***SENSOR\_DELAY\_NORMAL***);

* 透過android.hardware.SensorEventListener物件的onSensorChanged(SensorEvent event)方法的event參數可取得周遭相對溼度：

event.values[0]：濕度 (單位:%)

#### 隨堂測驗

|  |
| --- |
| (1)  Android的 android.hardware.SensorEventListener介面定義了兩個抽象方法，可被用來接收當感測器的值有所變動時從SensorManager所送出的通知。下列何者正確地描述了這兩個抽象方法的名稱？   1. onSensorChanged與onAccuracyChanged 2. onSensorChanged與onValueChanged 3. onValueChanged與onAccuracyChanged 4. 以上皆非   答案: A |

## 內建裝置應用技術

### 振動器

* 使用手機震動功能
  + 在Activity呼叫 android.content.ContextWrapper 的 Object getSystemService (String name (Context.VIBRATOR\_SERVICE) )方法以取得android.os.Vibrator(抽象類別) 物件
  + 呼叫 Vibrator物件的 void vibrate (long milliseconds)方法可讓手機**持續震動**指定長度時間 (執行本方法前需先取得"android.permission.VIBRATE"許可)
* 範例：

Activity

Vibrator vibrator = (Vibrator) getSystemService ( Context.***VIBRATOR\_SERVICE*** );  
vibrator.vibrate(1000);

AndroidManifest.xml

<**manifest xmlns:android=**

**"http://schemas.android.com/apk/res/android"**

**package="xx.yy.zz"**>  
 <**uses-permission android:name=**

**"android.permission.VIBRATE"** />

(略)

</**manifest**>

#### 隨堂測驗

|  |
| --- |
| (1)  下列那一項Activity的方法呼叫可以取得振動器物件？  (A) getSystemService(android.content.Context.VIBRATOR\_SERVICE);  (B) getSystemService(android.content.Context.SHAKER\_SERVICE);  (C) getSystemService(android.os.Vibrator.VIBRATOR\_SERVICE);  (D) getSystemService(android.os.Vibrator.SHAKER\_SERVICE);  答案: A |

|  |
| --- |
| (2)  如果要讓手機持續震動2分鐘，要呼叫下列何種方法？  (A) startVibrate(120000)  (B) startVibrate(2)  (C) vibrate(120000)  (D) vibrate(2)  答案: C |

### 相機

* 使用手機內建相機程式
  + 建立android.content.Intent物件(建構子引數設為android.provider.MediaStore.ACTION\_IMAGE\_CAPTURE | "android.media.action.IMAGE\_CAPTURE" )
  + 亦可建立android.content.Intent物件後呼叫Intent物件之 Intent setAction (String action)方法以設定待執行的動作
    - action引數設為: android.provider.MediaStore.ACTION\_IMAGE\_CAPTURE
  + 在Activity呼叫 android.content.ContextWrapper 的 void startActivity(Intent intent)方法以啟動系統內建相機程式
  + 範例：

android.content.Intent intent = **new** Intent(android.provider.MediaStore.***ACTION\_IMAGE\_CAPTURE***);   
startActivity(intent);

* 在應用程式啟動手機內建相機程式，拍照後顯示相片縮圖
  + 範例：

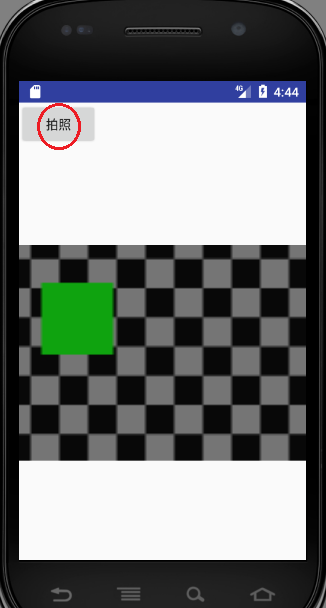
**package** com.example.test.myapplication;  
**import** android.app.Activity; **import** android.content.Intent;  
**import** android.graphics.Bitmap; **import** android.os.Bundle;  
**import** android.provider.MediaStore; **import** android.view.View;  
**import** android.widget.Button; **import** android.widget.ImageView;  
**public class** MainActivity **extends** Activity

**implements** View.OnClickListener {  
 **private** Button **btn\_photo**;  
 **private** ImageView **iv\_photo**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main5***);  
 **btn\_photo** = (Button)findViewById(R.id.***btn\_photo***);  
 **btn\_photo**.setOnClickListener(**this**);  
 **iv\_photo** = (ImageView)findViewById(R.id.***iv\_photo***);  
 }  
 **private int requestCode** = 100;  
 @Override  
 **public void** onClick(View v) {  
 Intent intent =

**new** Intent(MediaStore.***ACTION\_IMAGE\_CAPTURE***);  
 startActivityForResult(intent,**requestCode**);  
 }  
 @Override  
 **protected void** onActivityResult(**int** requestCode, **int** resultCode,

Intent data) {  
 **super**.onActivityResult(requestCode, resultCode, data);  
  
 **if**(requestCode==**this**.**requestCode** &&

resultCode== Activity.***RESULT\_OK***){  
 Bundle bundle = data.getExtras();  
 Bitmap bitmap = (Bitmap)bundle.get(**"data"**);  
 **iv\_photo**.setImageBitmap(bitmap);  
 }  
 }  
}



* 查詢**應用程式**安裝所在裝置所支援與相機有關的系統功能
  + 在Activity呼叫 android.content.ContextWrapper 的 PackageManager getPackageManager ()方法以取得android.content.pm.PackageManager(抽象類別)物件
  + 呼叫PackageManager物件的 boolean hasSystemFeature (String name)方法以測試應用程式安裝所在裝置是否支援指定的系統功能。
  + 上述方法的name引數可設定之值主要有
    - PackageManager.FEATURE\_CAMERA(支援後鏡頭)
    - PackageManager.FEATURE\_CAMERA\_FRONT
    - PackageManager. FEATURE\_CAMERA\_ANY(支援至少一種鏡頭)
    - PackageManager. FEATURE\_CAMERA\_EXTERNAL
    - PackageManager. FEATURE\_CAMERA\_FLASH
    - PackageManager. FEATURE\_CAMERA\_AUTOFOCUS
  + 範例：

System.***out***.printf(

**"FEATURE\_CAMERA\_AUTOFOCUS = %b\n"**,

getPackageManager().hasSystemFeature(

PackageManager.***FEATURE\_CAMERA\_AUTOFOCUS***)

);

輸出結果：

I/System.out: FEATURE\_CAMERA\_AUTOFOCUS = true

#### 隨堂測驗

|  |
| --- |
| (1)  假設下列程式片段為一個可以正常啟動執行的Android應用程式，那麼下列空格應填寫那一項才可以在應用程式被執行後即啟動系統的相機程式？  package com.example.xxx.test;  import android.content.Intent;  import android.provider.MediaStore;  public class MainActivity extends android.app.Activity{  protected void onCreate(android.os.Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  Intent intent = new Intent(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_); startActivity(intent);  }  }  (A) MediaStore.ACTION\_PHOTO\_CAPTURE  (B) MediaStore.ACTION\_PICTURE\_CAPTURE  (C) MediaStore.ACTION\_IMAGE\_CAPTURE  (D) 以上皆非  答案: C |

|  |
| --- |
| (2)  請問以下何者不是Android有支援的相機System Feature？  (A) PackageManager.FEATURE\_CAMERA  (B) PackageManager.FEATURE\_CAMERA\_FRONT  (C) PackageManager.FEATURE\_CAMERA\_ANY  (D) PackageManager.FEATURE\_CAMERA\_ALL  答案: D |

|  |
| --- |
| (3)  一般Android應用程式可以啟動系統的相機程式來拍照，並可將拍照後的相片縮圖透過onActivityResult方法傳回給Android應用程式。假設下列onActivityResult方法如上所述，在拍照動作完成後會被呼叫執行，那麼下列空格應填寫那一項來取得相機程式所傳回的相片縮圖？  protected void onActivityResult(int requestCode, int resultCode,  android.content.Intent data) {  super.onActivityResult(requestCode, resultCode, data);  android.os.Bundle bundle = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;  android.graphics.Bitmap bm =  (android.graphics.Bitmap)bundle.get("data");  }  (A) data.getData()  (B) data.getExtras()  (C) data.getAction()  (D) data.getBundle()  答案: B |

### 電話

* 啟動手機內建電話撥號程式
  + 建立android.content.Intent物件
  + 呼叫Intent物件之 Intent setAction (String action)方法以設定待執行的動作
    - action引數設為: android.content.Intent.ACTION\_VIEW 或android.content.Intent.ACTION\_DIAL
  + 呼叫Intent物件之 Intent setData (Uri data)方法以設定待操作的資料
    - data引數設為: android.net.Uri.parse("tel:12345678")
  + 在Activity呼叫 android.content.ContextWrapper 的 void startActivity(Intent intent)方法以啟動系統內建電話撥號程式
  + 範例：

android.content.Intent intent = **new** Intent();  
intent.setAction(android.content.Intent.***ACTION\_VIEW***) ;

intent.setData( android.net.Uri.*parse*(**"tel:12345678"**) );  
startActivity(intent);

* 啟動手機內建電話撥號程式**直接撥出電話**
  + 步驟類似上述「啟動手機內建電話撥號程式」
  + 除了(步驟二) 呼叫Intent物件之 Intent setAction (String action)方法以設定待執行的動作
    - action引數需改設為: android.content.Intent.ACTION\_CALL
  + 另外應用程式必須先取得"android.permission.CALL\_PHONE" 許可
  + 範例：

Activity

android.content.Intent intent = **new** Intent();  
intent.setAction(Intent.***ACTION\_CALL***);  
intent.setData(Uri.*parse*(**"tel:12345678"**));  
startActivity(intent);

AndroidManifest.xml

<**manifest xmlns:android=**

**"http://schemas.android.com/apk/res/android"**

**package="xx.yy.zz"**>  
 <**uses-permission android:name=**

**"android.permission.CALL\_PHONE"** />

(略)

</**manifest**>

* 啟動手機內建的簡訊傳送程式
  + 步驟類似上述「啟動手機內建電話撥號程式」(實際用法參考下列範例)
  + 另外應用程式必須先取得"android.permission.SEND\_SMS" 許可
  + 範例：

Activity

android.content.Intent intent = **new** Intent();  
intent.setAction(Intent.***ACTION\_VIEW***);

intent.setData(Uri.*parse*(**"sms:5556?body=welcome..."**)); \*

startActivity(intent);

AndroidManifest.xml

<**manifest xmlns:android=**

**"http://schemas.android.com/apk/res/android"**

**package="xx.yy.zz"**>  
 <**uses-permission android:name=**

**"android.permission.SEND\_SMS"** />

(略)

</**manifest**>

#### 隨堂測驗

|  |
| --- |
| (1)  如果要透過Intent直接撥打電話，請問下列程式碼中的問號，要使用何  種參數？  Intent x = new Intent();  x.setAction(？);  x.setData(Uti.parse(“tel:0932123456”));  startActivity(x);  (A) Intent.ACTION\_PHONE  (B) Intent.ACTION\_CALL  (C) Intent.ACTION\_DIAL  (D) Intent.ACTION\_NAVIGATE  答案: B |

## 定位功能

### 定位技術

* Android提供兩種主要的定位方式
  + gps：利用GPS (全球定位系統/ Global Position System)衛星所發出的訊號來定位
  + network：利用行動電話與Wi-Fi基地台所發出的訊號來定位
* Android支援Location Provider讓我們在程式中使用上述兩種定位方式以取得使用者手機的位置
* 應用程式若欲使用Location Provider以取得使用者的位置，必須先取得相關許可
  + 使用**gps提供者**需要下列許可"android.permission.ACCESS\_FINE\_LOCATION"
  + 使用**network提供者**需要下列許可"android.permission.ACCESS\_COARSE\_LOCATION"

#### 隨堂測驗

|  |
| --- |
| (1)  以下何者不是Android定位時需要的Permission？  (A) android.permission.ACCESS\_COARSE\_LOCATION  (B) android.permission.ACCESS\_FINE\_LOCATION  (C) android.permission.ACCESS\_GPS\_LOCATION  (D) android.permission.ACCESS\_INTERNET  答案: C |

### 使用LocationManager

* 使用手機定位服務
  + 在Activity呼叫 android.content.ContextWrapper 的 Object getSystemService (String name (Context.LOCATION\_SERVICE) )方法以取得android.location.LocationManager 物件 \*
  + 呼叫 LocationManager物件的 String getBestProvider (Criteria criteria, boolean enabledOnly)方法以取得符合criteria引數所設定的標準的最佳供應者之名稱
  + 呼叫 LocationManager物件的 void requestLocationUpdates (String provider, long minTime(毫秒), float minDistance(公尺), LocationListener listener)方法為**定位供應者**註冊**位置監聽器**
  + 透過android.location.LocationListener物件的onLocationChanged()方法的location參數可取得手機所在位置之緯度、經度等資料
    - void onLocationChanged (Location location)

location.getLatitude( ): 傳回緯度(單位:degree)

location.getLongitude( ): 傳回經度(單位:degree)

* + - void onProviderDisabled (String provider)
    - void onProviderEnabled (String provider)
    - void onStatusChanged (String provider, int status, Bundle extras)
* 範例：

Activity

android.location.LocationListener locationListener = **new** android.location.LocationListener(){  
 @Override  
 **public void** onLocationChanged(Location location) {  
 System.***out***.printf(**"緯度=%f 經度=%f\n"**,

location.getLatitude(), location.getLongitude());

}  
 @Override  
 **public void** onStatusChanged(String provider, **int** status,

Bundle extras) {

*This method is called when a provider is unable to fetch a location or if the provider has recently become available after a period of unavailability.*

}  
 @Override  
 **public void** onProviderEnabled(String provider) {  
 }  
 @Override  
 **public void** onProviderDisabled(String provider) {  
 }  
};  
LocationManager locationMgr = (LocationManager)getSystemService(Context.***LOCATION\_SERVICE***);  
String locationProvider = locationMgr.getBestProvider(

**new** android.location.Criteria(), **true**);  
locationMgr.requestLocationUpdates(locationProvider,5000,5,

locationListener);

AndroidManifest.xml

<**manifest xmlns:android=**

**"http://schemas.android.com/apk/res/android"**

**package="xx.yy.zz"**>  
 <**uses-permission android:name=**

**"android.permission.ACCESS\_FINE\_LOCATION"** />

(略)

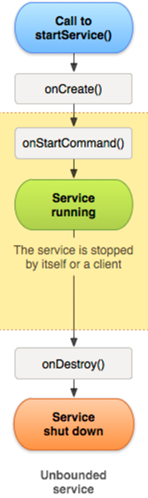
</**manifest**>

#### 隨堂測驗

|  |
| --- |
| (1)  Android API的android.location.LocationListener介面可以用來監聽行動裝置的位置事件，下列那一項不是該介面所定義的抽象方法？  (A) onLocationChanged  (B) onDataChanged  (C) onProviderEnabled  (D) onProviderDisabled  答案: B |

## Service運用

* Service是Android的**應用程式元件**之一， 適合用來執行需用較長時間且不與使用者互動的作業。
* Service如同Activity一樣都是在**UI/Main執行緒**執行的程式，因此當我們欲進行需耗用較長CPU時間的作業時，應該要另建**背景執行緒**來執行該作業。
* 啟動一般Service的主要方式是利用Context.startService(Intent)方法
* 應用程式元件(如Activity)啟動Service以後，就算使用者切換至其他應用程式，Service也會繼續在背景中執行。
* 停止一般Service的方式
  + Service本身呼叫stopSelf()方法
  + 其他應用程式元件(如Activity)呼叫stopService(Intent) 方法
* Service的生命週期



*註：*

* *void onStart (Intent intent, int startId) deprecated in API level 5(Android 2.0).*
* *Implement onStartCommand(Intent, int, int) instead.*
* 範例：

MainActivity

(假設點擊**按鈕A**後即執行下列方法)  
**public void onClick**(View v) {  
 startService(**new** Intent(MainActivity.**this**, MyService.**class**));

|  |
| --- |
| * startService()第一次被呼叫後，   Service's onCreate() 🡪 onStartCommand() 依序會被呼叫   * startService()再次被呼叫後，只有Service's onStartCommand()會被呼叫 |

}

(假設點擊**按鈕B**後即執行下列方法)  
**public void** onClick2(View view){  
 stopService(**new** Intent(MainActivity.**this**, MyService.**class**) );

|  |
| --- |
| * Service's onDestroy()會被呼叫 |

}

Service

**package** a.b.c;  
**import** android.app.Service;   
**import** android.content.Intent;  
**import** android.os.IBinder;  
  
**public class** MyService **extends** Service { //Service是一個抽象類別  
 @Override //實作Service之抽象方法:onBind()  
 **public** IBinder onBind(Intent intent) {   
 **return null**;  
 }  
 @Override  
 **public void** onCreate() {

* 當使用者呼叫startService()方法之後，
* 若此時相關Service 物件尚未被建立時，系統會先建立此Service物件，然後呼叫此方法
* 若此時相關Service 物件已被建立時，系統就不會呼叫此方法

}  
 @Override  
 **public int** onStartCommand(Intent intent, **int** flags, **int** startId) {

* 每當使用者呼叫startService()方法之後，系統都會呼叫此方法
* **參數flags**: Additional data about this start request
  + 0：一般狀況
  + 1：if the Intent is a re-delivery
  + 2：if the Intent is a retry
* **參數startId**:
  + 當Service被系統建立後，後續的onStartCommand()方法被系統呼叫時，系統都會傳一個唯一的整數值給startId參數(從一開始，每次遞增一)
  + 與boolean **stopSelfResult** (int startId)方法搭配使用
* **回傳值**(例如):
* Service.**START\_STICKY**：

此Service's process如果被killed，**稍後系統會重新建立此Service**，並保證呼叫onStartCommand()方法**(但不保留this delivered intent)** (效果相當於 We want this service to continue running until it is explicitly stopped)

* Service.**START\_NOT\_STICKY**：

參考上述：系統不會重新建立此Service

* Service.**START\_REDELIVER\_INTENT**：

參考第一項：(但the last delivered Intent 會被re-delivered給此Service)

}

@Override  
 **public void** onDestroy() {

* 當使用者呼叫stopService()方法之後，系統會呼叫此方法然後移除此Service物件

}  
}

AndroidManifest.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="a.b.c"**>  
 <**application** (略)>  
<**activity  
 android:name=".MainActivity"**>  
 <**intent-filter**>  
 <**action android:name=**

**"android.intent.action.MAIN"** />  
 <**category android:name=**

**"android.intent.category.LAUNCHER"** />  
 </**intent-filter**>  
 </**activity**>  
 <**service  
 android:name=".MyService"  
 android:enabled="true"  
 android:exported="true"**></**service**>  
 </**application**>  
</**manifest**>

### 隨堂測驗

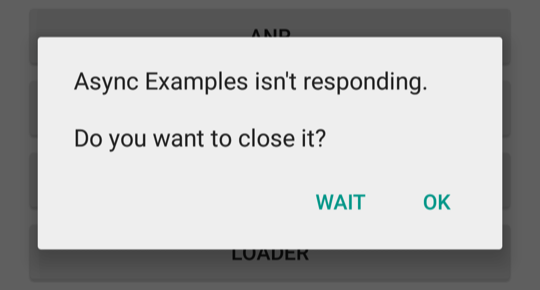
|  |
| --- |
| (1)  有關service的生命週期，以下敘述何者不正確？   1. 第一次啟動的時候會先呼叫onCreate() 2. 第一次啟動時呼叫onCreate()後接著呼叫onStart() 3. 如果service已經啟動，會先呼叫onCreate()接著呼叫onStart() 4. 如果service已經啟動，只會呼叫onStart()，不會呼叫onCreate()   答案: C |

|  |
| --- |
| (2)  在開發應用程式時，通常會需要提供一個畫面來提供基本的操作，但在開發哪一類應用程式時可不需要畫面？  (A) Service  (B) Map  (C) Canvas  (D) Keyboard  答案: A |

## 異步執行技術

### 執行緒(Thread)

* Android的**UI物件**是在一個特別的執行緒執行的，此執行緒稱為main執行緒，又稱為UI執行緒
* 當應用程式有費時的工作需要執行時，應該另外開啟其他執行緒(即所謂的background threads或worker threads)來執行，以免阻礙應用程式UI的回應能力
* 否則，Android偵測到下列狀況之一時，將顯示一個ANR/Application Not Responding對話框(如下圖) 提示使用者 :
  + 應用程式對輸入事件沒有回應超過5秒
  + 廣播接受器(BroadcastReceiver)執行程式超過10秒



* 又Android 3.0 (API level 11)起預設要求應用程式必須開啟一個背景執行緒來執行網路作業，否則Android將產生 NetworkOnMainThreadException例外
* 另外Android限制應用程式「只有在**UI執行緒**執行的物件才能存取在**UI執行緒**執行的其他物件」，意即在**背景執行緒**執行的物件無法存取在**UI執行緒**執行的物件。(解決方案可使用下列範例或AsyncTask等)
* 範例：
  1. 定義java.lang.Thread子類別並在子類別內覆寫父類別之run() 方法
  2. 建立上述Thread子類別物件後呼叫其start()方法以啟動**背景執行緒**來執行費時的工作( run()方法內的程式即為待執行的工作)
  3. 在run()方法內呼叫Activity的 **void runOnUiThread (Runnable action)**方法以在**UI執行緒**執行存取UI物件的工作(action參數所參考的**Runnable物件**的run()方法內的程式即為待在**UI執行緒**執行的工作)

**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapFactory;

**import** java.net.URL;

…

**public void** onClick(View v){  
 **new** Thread(){  
 **public void** run(){  
 **try** {  
 *//在背景執行緒 執行費時之工作*

*//如從Internet下載圖片*

(註:需在**AndroidManifest.xml**中宣告使用

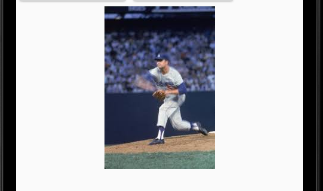
android.permission.INTERNET 許可)**final** Bitmap bm;  
 InputStream is = **new** URL(

**"https://encrypted-tbn3.gstatic.com/images?q=tbn:ANd9GcQ-u4bJbn\_osyqJIdxHawxadQQKH1tWEUgu8wb0CixfF0hUc46agA"**).openStream();  
 bm = BitmapFactory.*decodeStream*(is);

*//在UI執行緒 執行設定UI物件之工作*

*//設定ImageView之圖片內容* runOnUiThread( **new** Runnable(){  
 **public void** run(){  
 ((ImageView)findViewById(

R.id.***imageView***)).setImageBitmap(bm);  
 }  
 });  
  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }.start();  
}



### android.os.AsyncTask<Params, Progress, Result>抽象類別

* AsyncTask可讓我們透過**使用者介面**執行**非同步工作**
* AsyncTask會以**背景執行緒**執行費時工作，再將結果發行到**UI執行緒** (我們完全不需要自行處理執行緒問題)
* Asynchronous Task被執行之後，將依序進行**四個**動作(分別對應呼叫下列四個方法)
  + void onPreExecute ()：
    - Runs on the **UI thread** before doInBackground(Params...).
  + **abstract** Result doInBackground (Params... params)：
    - Override this method to perform a computation on a **background thread**.
  + void onProgressUpdate (Progress... values)
    - Runs on the **UI thread** after **publishProgress(Progress...)** is invoked.
    - The specified values are the values passed to publishProgress(Progress...).
  + void onPostExecute (Result result)：
    - Runs on the **UI thread** after doInBackground(Params...).
    - The specified result is the value returned by doInBackground(Params...).
* 其他常用方法
  + AsyncTask execute (Params... params)：
    - Executes the task with the specified parameters.
  + void publishProgress (Progress... values)
    - This method can be invoked from **doInBackground(Params...)** to publish updates on the **UI thread** while the background computation is still running.
    - Each call to this method will trigger the execution of **onProgressUpdate(Progress...)** on the **UI thread**.
* 範例：

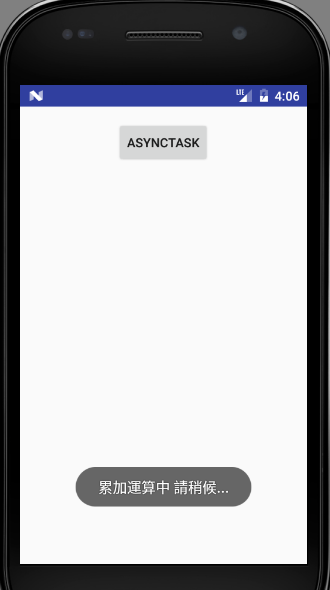
**public class** MainActivity **extends** Activity

**implements** View.OnClickListener {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 Button btn1 = (Button)findViewById(R.id.***btn1***);  
 btn1.setOnClickListener(**this**);  
 }  
 @Override  
 **public void** onClick(View v) {  
 **new MyAsyncTask().execute(1,2,3,4,5);** }

**class** MyAsyncTask **extends** AsyncTask<Integer,Integer,Double>{  
 Toast **toast** = Toast.*makeText*(MainActivity.**this**, **""**,

Toast.***LENGTH\_SHORT***);  
 @Override  
 **protected void** onPreExecute() {  
 **super**.onPreExecute();  
 **toast**.setText(**"累加運算中 請稍候..."**);  
 **toast**.show();  
 }  
 @Override  
 **protected** Double doInBackground(Integer... integers) {  
 **int** total = 0;  
 **for**(**int** i=0; i<integers.**length**; i++){  
 **try** {  
 Thread.*sleep*(1000);  
 } **catch** (InterruptedException e) {  
 e.printStackTrace();  
 }  
 total += integers[i];  
 publishProgress(100\*(i+1)/integers.**length**);  
 }  
 **return** (**double**)total;  
 }  
 @Override  
 **protected void** onProgressUpdate(Integer... values) {  
 **super**.onProgressUpdate(values);  
 **if**(values != **null**) {  
 **toast**.setText(**"進度: "** + values[0] + **"%"**);  
 **toast**.show();  
 }  
 }  
 @Override  
 **protected void** onPostExecute(Double aDouble) {  
 **super**.onPostExecute(aDouble);  
 **toast**.setText(**"運算結果 total="**+aDouble);  
 **toast**.show();  
 }  
}

}







#### 隨堂測驗

|  |
| --- |
| (1)  AsyncTask是官方建議的多執行緒處理方式，在執行耗時的背景處理時，需要更新進度條時，會呼叫publishProgress通知進度狀態，此時要在何處更新進度條畫面？   1. onPreExecute 2. onPostExecute 3. doInBackground 4. onProgressUpdate   答案: D |

|  |
| --- |
| (2)  AsyncTask是官方建議的多執行緒處理方式，執行耗時的背景處理後，需要把處理結果更新到UI畫面的部分要寫在何處？  (A) onPreExecute  (B) onPostExecute  (C) doInBackground  (D) onBackgroundUpdate  答案: B |

|  |
| --- |
| (3)  關於ANR（Application Not Responding），下列敘述何者正確？  (A) 表示手機空間不足的警告  (B) 表示手機App耗電量過大導致App crash  (C) 表示main thread對輸入事件在5秒內沒有處理完畢  (D) 表示網路server超時沒有回應  答案: C |

## Permission設定

* Android應用程式預設不具有任何許可去影響手機內的資料或使用者操作經驗。
* Android要求應用程式必須在AndroidManifest.xml裡宣告其欲使用的許可。
  + 如果此許可不會影響使用者的隱私或其他App的操作 (即Normal Permission)，例如 INTERNET、VIBRATE許可，系統會在應用程式**安裝**時自動授與此許可。
  + 如果此許可會影響使用者的隱私或其他App的操作 (即Dangerous Permission)，例如ACCESS\_FINE\_LOCATION、ACCESS\_COARSE\_LOCATION、READ\_EXTERNAL\_STORAGE (註:Android 4.4/API level 19起始為必要)、WRITE\_EXTERNAL\_STORAGE、 CALL\_PHONE、SEND\_SMS等，系統會要求使用者在**安裝或執行時期**進行許可授與的同意作業。

### 隨堂測驗

|  |
| --- |
| (1)  下列哪一個許可是Android應用程式順利顯示網頁內容所必須宣告使用的許可？  (A) android.permission.LAN  (B) android.permission.INTERNET  (C) android.permission.WWW  (D) android.permission.WAN  答案: B |

|  |
| --- |
| (2)  如果要發送簡訊，必須開放下列何種權限？   1. android.permission.SEND\_SMS 2. android.permission.NOTIFY\_SMS 3. android.permission.ALLOW\_SMS 4. android.permission.SUBMIT\_SMS   答案: A |

## AndroidManifest設定

* 每個Android應用程式的根目錄都必須包含 AndroidManifest.xml (宣示說明)檔案。
* **宣示說明檔案**可以提供 Android 系統在執行該應用程式之前必須具備的基本資訊。
* **宣示說明檔案**基本結構範例

*<?***xml version="1.0" encoding="utf-8"***?>*<**manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="aa.bb.cc"**>

<**uses-permission**

**android:name="android.permission.VIBRATE"** />  
 <**uses-permission**

**android:name="android.permission.CALL\_PHONE"** />  
   
 <**application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:theme="@style/AppTheme"**>  
<**activity android:name=".MainActivity"**>  
 <**intent-filter**>  
 <**action android:name=**

**"android.intent.action.MAIN"** />  
 <**category android:name=**

**"android.intent.category.LAUNCHER"** />  
 </**intent-filter**>  
 </**activity**>  
 <**activity android:name=".Main2Activity"**></**activity**>  
 </**application**>  
</**manifest**>

|  |
| --- |
| * 上述aa.bb.cc.MainActivity是應用程式的主要進入點(即**Launcher/Main Activity**)，當使用者啟動應用程式時，這個Activity就會被開啟。   + <intent-filter>內之<action android:name="android.intent.action.MAIN">表示這個Activity是應用程式的主要進入點   + <intent-filter>內之<category android:name="android.intent.category.LAUNCHER">表示這個Activity的圖示應該顯示在系統的應用程式啟動器(App Launcher)中。   (如果這個Activity未設定圖示，則系統將會顯示 <application> 元素中所設定的圖示)   * + 上述兩項必須成對，Activity才會顯示在應用程式啟動器中。 |

### 隨堂測驗

|  |
| --- |
| (1)  一個Activity若想當作應用程式的Launcher Activity，則必須在AndroidManifest.xml中設定這個Activity的意圖篩選器，下列哪一項是正確的設定？  (A)  <intent-filter>  <category android:name="android.intent.category.LAUNCHER" />  </intent-filter>  (B)  <intent-filter>  <action android:name="android.intent.action.MAIN" />  </intent-filter>  (C)  <intent-filter>  <action android:name="android.intent.action.MAIN" />  <category android:name="android.intent.category.LAUNCHER" />  </intent-filter>  (D) 以上皆非  答案: C |

## 官方開發工具與程式語言

### Android官方IDE開發工具：Android Studio

### Android官方程式開發語言：Java.

## Android記憶體管理機制

* Android支援記憶體自動垃圾回收(Garage Collection)機制
* 當一個物件在程式裡不再被使用時，也就是程式中不再保留該物件的參考時，即該物件會被Android標記為"待垃圾回收"
* Android會在適當時機呼叫垃圾回收者(garbage collector)將該物件佔用的記憶體回收再用

### 隨堂測驗

|  |
| --- |
| (1)  關於Android的記憶體回收機制，何者正確？   1. Garbage collection機制會自動清除無用的記憶體空間 2. 程式人員必須撰寫清除記憶體的程式 3. Garbage collection機制可以指定特定時間執行清理記憶體 4. 程式人員可以要求Garbage collection機制立刻執行清理記憶體空間的動作   答案: A |

## AppWidget

* App Widgets 是一種小型的應用程式，可嵌入於其他應用程式 (例如Home screen) 並可定時執行更新作業。
* 範例：

AndroidManifest.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.test.myapplicationa"**>  
 <**application** (略)>  
 <**activity android:name=".MainActivity"**>  
 <**intent-filter**>  
 <**action android:name="android.intent.action.MAIN"** />  
 <**category android:name=**

**"android.intent.category.LAUNCHER"** />  
 </**intent-filter**>  
 </**activity**>  
 <**receiver android:name=".MyAppWidget"**>  
 <**intent-filter**>  
 <**action android:name=**

**"android.appwidget.action.APPWIDGET\_UPDATE"** />  
 </**intent-filter**>  
 <**meta-data  
 android:name="android.appwidget.provider"  
 android:resource="@xml/my\_app\_widget\_info"** />  
 </**receiver**>  
 </**application**>  
  
</**manifest**>

res/xml/my\_app\_widget\_info.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**appwidget-provider xmlns:android=**

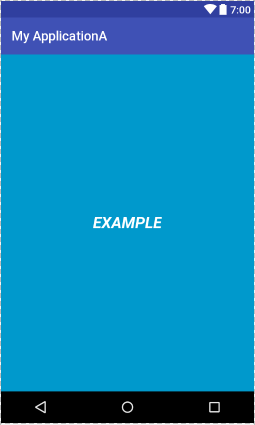
**"http://schemas.android.com/apk/res/android"  
 android:initialKeyguardLayout="@layout/my\_app\_widget"  
 android:initialLayout="@layout/my\_app\_widget"  
 android:minHeight="40dp"  
 android:minWidth="110dp"  
 android:previewImage="@drawable/example\_appwidget\_preview"  
 android:resizeMode="horizontal|vertical"  
 android:updatePeriodMillis="86400000"** (1天) **android:widgetCategory="home\_screen|keyguard"**>

</**appwidget-provider**>

res/layout/my\_app\_widget.xml

<**RelativeLayout xmlns:android=**

**"http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#09C"  
 android:padding="@dimen/widget\_margin"**>  
 <**TextView  
 android:id="@+id/appwidget\_text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:background="#09C"  
 android:contentDescription="@string/appwidget\_text"  
 android:text="@string/appwidget\_text"  
 android:textColor="#ffffff"  
 android:textSize="24sp"  
 android:textStyle="bold|italic"  
 android:layout\_centerVertical="true"  
 android:layout\_centerHorizontal="true"** />  
</**RelativeLayout**>

****

MyAppWidget.java

**package** com.example.test.myapplicationa;  
**import** android.appwidget.AppWidgetManager;  
**import** android.appwidget.AppWidgetProvider;  
**import** android.content.Context;  
**import** android.widget.RemoteViews;  
**import** java.util.Date;  
**public class** MyAppWidget **extends** AppWidgetProvider {  
 **static void** updateAppWidget(Context context,

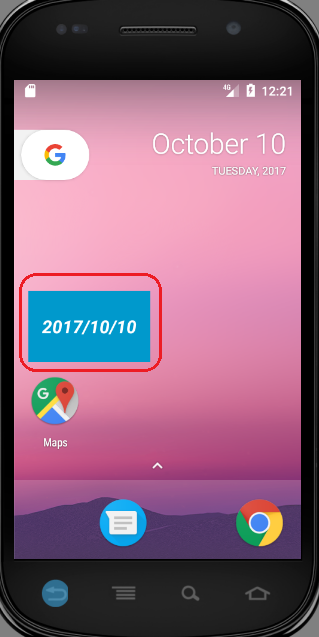
AppWidgetManager appWidgetManager, **int** appWidgetId) {  
 *// Construct the RemoteViews object* RemoteViews views = **new** RemoteViews(context.getPackageName(),

R.layout.***my\_app\_widget***);  
 CharSequence widgetText = String.*format*(**"%tY/%<tm/%<td"**,

**new** Date());  
 views.setTextViewText(R.id.***appwidget\_text***, widgetText);  
  
 *// Instruct the widget manager to update the widget* appWidgetManager.updateAppWidget(appWidgetId, views);  
 }

@Override  
 **public void** onEnabled(Context context) {  
 *//when the first widget is created* }  
 @Override  
 **public void** onUpdate(Context context,

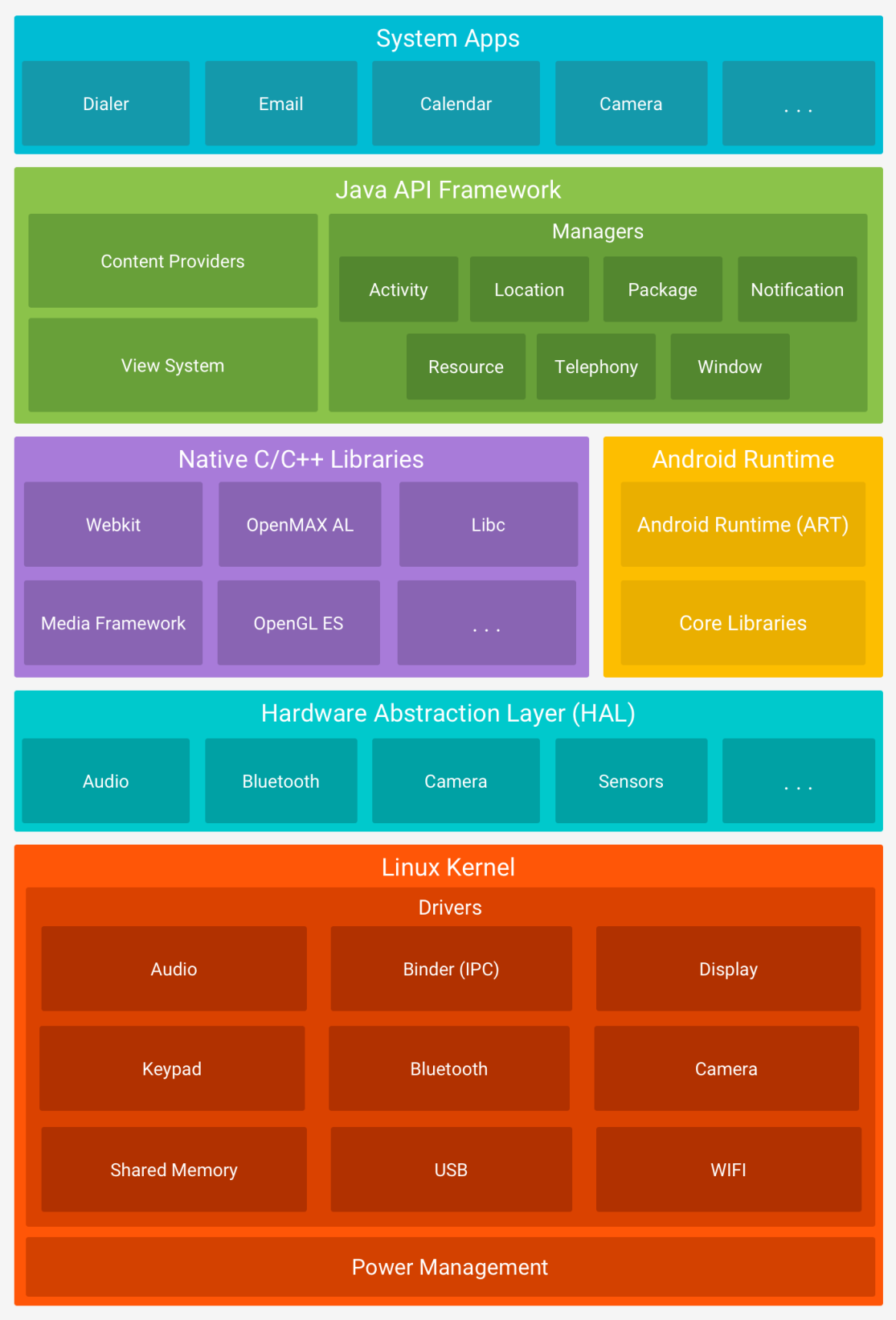
AppWidgetManager appWidgetManager, **int**[] appWidgetIds) {  
 *// There may be multiple widgets active, so update all of them* **for** (**int** appWidgetId : appWidgetIds) {  
 *updateAppWidget*(context, appWidgetManager, appWidgetId);  
 }  
 }  
 @Override  
 **public void** onDeleted(Context context, **int**[] appWidgetIds) {  
 *// When the user deletes the widget* }  
 @Override  
 **public void** onDisabled(Context context) {  
 *// when the last widget is disabled* }  
}

****

### 隨堂測驗

|  |
| --- |
| (1)  設計Wiget小工具，必須繼承下列何種類別？  (A) WidgetActivity  (B) WidgetProvider  (C) AppWidgetProvider  (D) Context  答案: C |

## Android Platform Architecture





### 隨堂測驗

|  |
| --- |
| (1)  用Android手機內的撥號App打電話，是屬於Android框架的哪一層？  (A) Application Framework層  (B) Libraries層  (C) Kernel層  (D) Application層  答案: D |

## Android 2D繪圖

* 在Android繪圖需要四個基本元件:
* A Bitmap to hold the pixels,
* a Canvas to host the draw calls (writing into the bitmap),
* a drawing primitive (e.g. Rect, Path, text, Bitmap),
* and a paint (to describe the colors and styles for the drawing).
* 範例：

Bitmap bitmap = Bitmap.*createBitmap*(300,100,Bitmap.Config.***ARGB\_8888***);

*//Bitmap* ***createBitmap*** *(int width, int height, Bitmap.Config config)*

*//第三個參數設為****Bitmap.Config.ARGB\_8888: 表示每個pixel被儲存成*** *4 bytes.*Canvas canvas = **new** Canvas(bitmap);  
Paint paint = **new** Paint();  
paint.setColor(Color.***RED***);  
paint.setAntiAlias(**true**); *//設定反鋸齒*paint.setTextSize(28); *//單位:pixels*canvas.drawText(**"\*\*\*Welcome\*\*\*"**,50,50, paint);  
**imageView**.setImageBitmap(bitmap);



### 隨堂測驗

|  |
| --- |
| (1)  在Android平台上使用2D繪圖技術時，下列何者可以改善螢幕鋸齒效果？  (A) setSawtooth(true)  (B) setSawtooth(false)  (C) setAntiAlias(true)  (D) setAntiAlias(false)  答案: C |