

# Intent

- 瀏覽網站：ACTION\_VIEW
  - **Intent 物件名稱 = new Intent(Intent.ACTION\_VIEW,uri) ;**
  - **Uri uri = Uri.parse("網址");**
- 呼叫撥號按鈕：ACTION\_DIAL
  - **Intent 物件名稱 = new Intent(Intent.ACTION\_DIAL,uri) ;**
  - **Uri uri = Uri.parse("tel:0912345678");**
- 撥打電話：ACTION\_CALL
  - **Intent 物件名稱 = new Intent(Intent.ACTION\_CALL,uri) ;**
  - **Uri uri = Uri.parse("tel:0912345678");**

# Intent

## ■ 權限設定

- Android 6.0 以後的版本，需要在『AndroidManifest.xml』檔案中開啟權限。第一次執行時必須再次允許行動裝置取得權限，方可正常執行。
- 設定撥打電話的權限：
  - 於『AndroidManifest.xml』檔案中加入：(放在<application>標籤外)
    - **<uses-permission android:name="android.permission.CALL\_PHONE"/>**

# Intent

## ► 使用 Intent 傳遞資料

- 除了可以切換不同的 Activity 到畫面中顯示之外，也可以將資料從目前的 Activity 傳遞到指定的 Activity 中使用。

- 原理：

- 在目前的 Activity 中將要傳遞的資料放在【Bundle】物件中。

- 當對方收到【Intent】附帶的【Bundle】物件後，再從【Bundle】物件中取出附帶的資料。



# Intent

- 使用 `setClass` 方法指定執行類別
  - 設定要執行的自訂類別，即確定 `Intent` 中的寄件者與收件者。
  - 如 `MainActivity` 寄給 `Second`。
    - **`Intent intent=new Intent();`  
`intent.setClass(MainActivity.this , Second.class);`**

# Intent

- 依照資料的型別以 Bundle 物件進行打包
  - Bundle 物件類似郵件包裹的概念，包裹裡包含許多郵包，各個郵包就是單筆 Intent 要傳遞的資料。
  - 使用 Bundle 的『putXXX()』方法，將郵包依照不同的資料型別打包並寫上郵包名稱。
  - XXX 表示資料的型別。
    - **String name="Simon";**  
**int age=50;**  
**Double tall=177.5;**  
**Bundle bundle=new Bundle();**  
**bundle.putString("NAME", name);**  
**bundle.putInt("AGE", age);**  
**bundle.putDouble("TALL", tall);**

# Intent

put() 方法	功能
putString()	將 String 型別資料放入 Bundle 中。
putBoolean()	將 Boolean 型別資料放入 Bundle 中。
putByte()	將 Byte 型別資料放入 Bundle 中。
putChar()	將 Char 型別資料放入 Bundle 中。
putDouble()	將 Double 型別資料放入 Bundle 中。
putFloat()	將 Float 型別資料放入 Bundle 中。
putInt()	將 Int 型別資料放入 Bundle 中。
putShort()	將 Short 型別資料放入 Bundle 中。
putStringArray()	將 String 型別資料陣列放入 Bundle 中。
putIntArray()	將 Int 型別資料陣列放入 Bundle 中。

# Intent

- 使用 putExtras 方法加入 Bundle 物件
  - 使用 『putExtras()』 方法將資料傳遞給其他的 Activity。
  - Intent 如同郵差，『intent.putExtras(bundle)』 則是帶有郵寄包裹的郵差。
- 使用 startActivity 方法執行 Intent
  - 使用 『startActivity()』 方法將包裹寄出。
    - **startActivity(intent);**

# Intent

- Intent 傳遞資料
  - 附加資料到 Intent 中：
    - **putExtra (資料名稱, 資料)**
  - 取得傳入的 Intent 物件：
    - **getIntent()**
  - 讀取 Intent 物件的資料：
    - **getXXXExtra(資料名稱, 預設值)**
    - XXX 為資料型別名稱



# Intent

- Bundle 物件打包資料的方式
  - **putXXX( 資料名稱 , 資料內容)**
- 取出 Intent 資料
  - 以【getIntent()】方法取得傳送的 Intent。
    - **Intent intent = this.getIntent();**
  - 利用 Intent 的【getExtras()】方法，從 Intent 中取得 Bundle 物件。
    - **Bundle bundle = intent.getExtras();**
  - 依據郵包名稱取得 Bundle 物件中的資料。

# Intent

- 取出 Bundle 物件資料
  - 以【getXXX()】方法取得 Bundle 物件中不同型別的資料。
    - **getXXX(名稱);**
    - XXX 為資料型別。

# Intent

get() 方法	功能
getString()	取得 Bundle 中 String 型別資料。
getBoolean()	取得 Bundle 中 Boolean 型別資料。
getByte()	取得 Bundle 中 Byte 型別資料。
getChar()	取得 Bundle 中 Char 型別資料。
getDouble()	取得 Bundle 中 Double 型別資料。
getFloat()	取得 Bundle 中 Float 型別資料。
getInt()	取得 Bundle 中 Int 型別資料。
getShort()	取得 Bundle 中 Short 型別資料。
getStringArray()	取得 Bundle 中 String 陣列型別資料。
getIntArray()	取得 Bundle 中 Int 陣列型別資料。

# 訊息顯示元件

## ■ AlertDialog (對話方塊)

- 顯示訊息，並以按鈕與使用者互動。
- 使用『AlertDialog.Builder』類別。
- 語法：**AlertDialog.Builder 變數名稱 = new AlertDialog.Builder(主程式類別) ;**  
**變數名稱.show() ;**
- 可使用匿名物件的方式：
  - **類別名稱.方法1()**  
**.方法2()**  
**.....**  
**.方法n() ;**

# 訊息顯示元件

## ➤ AlertDialog (對話方塊)

➤ 常用方法：

方法	功能
setTitle()	設定對話方塊的標題
setIcon()	設定對話方塊的圖示
setMessage()	設定對話方塊的內容
setItems()	設定對話方塊的項目內容
setPositiveButton()	在對話方塊中加入 確定按鈕
setNegativeButton()	在對話方塊中加入 取消按鈕
setNeutralButton()	在對話方塊中加入 忽略按鈕