

第十一章

畫面轉向與資源選擇

OrientationChange/ActivityOrientationChangeNonKeepInstance

OrientationChange/ActivityOrientationChange

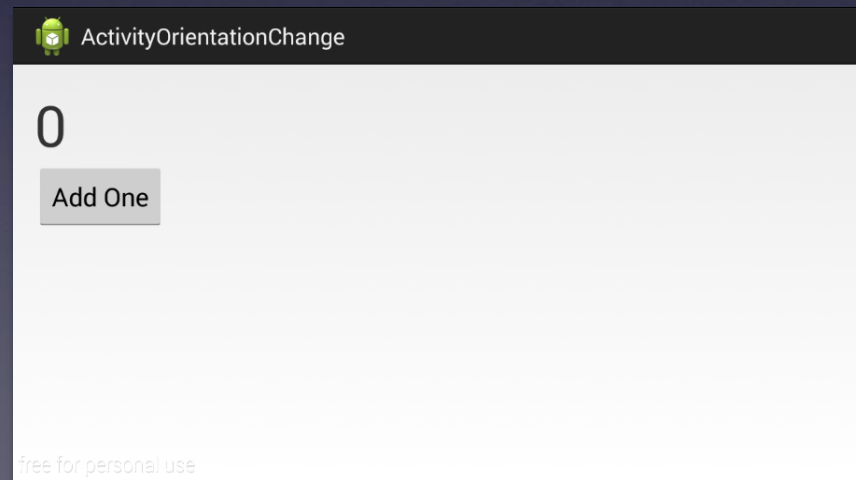
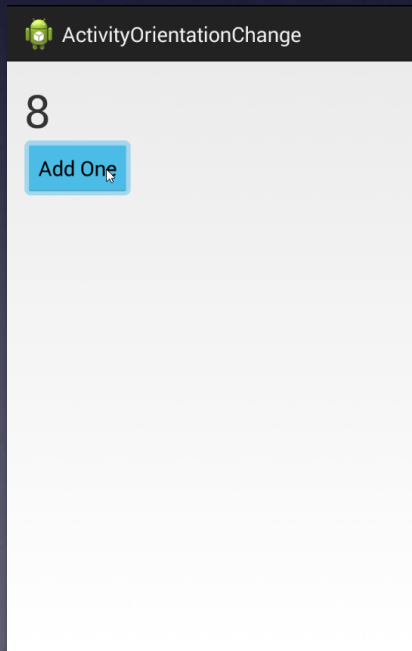
畫面轉向的生命週期

畫面轉向

- 看專案

ActivityOrientationChangeNonKeepInstance中
按下按鈕就可以讓畫面上的數字加1

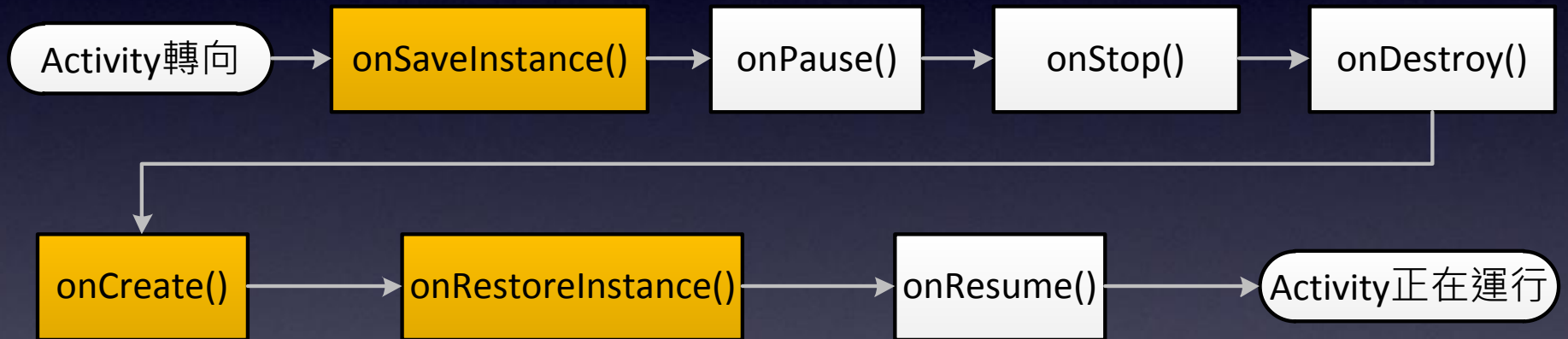
- 但手機轉方向後，又回到0了，怎麼回事？



畫面轉向

- 手機畫面轉換方向，會觸發App自我摧毀再次重建
- 所以如何保留畫面上的資料就是一件重要的事情
- 藉由Activity的生命週期來儲存及恢復資料
 - onSaveInstanceState() 儲存
 - onCreate() 恢復
 - onRestoreInstanceState() 恢復

轉向時的生命週期



儲存狀態

- `onSaveInstanceState(Bundle outState)`
 - 使用outState儲存
 - e.g. `outState.putString("text", ...);`

恢復狀態

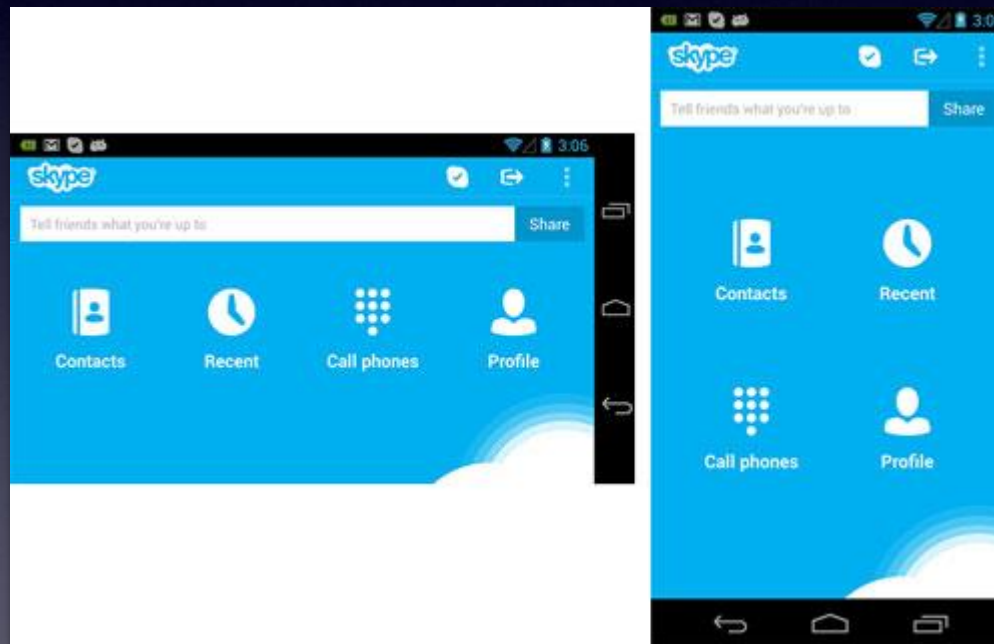
- Android提供兩個地方回復狀態
- `onCreate(Bundle savedInstanceState)`
 - 參數內含`onSaveInstanceState()`時儲存的資料
 - e.g. `savedInstanceState.getString("text")`
- 為了配合生命週期，另一個啟動點是`onRestoreInstanceState(Bundle savedInstanceState)`
 - 使用方式與`onCreate()`中相同

OrientationChange/ActivityResourceChooser

轉方向時的資源選擇

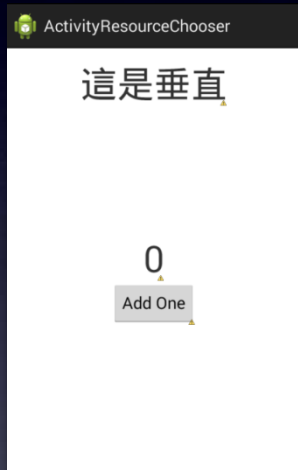
資源選擇

- 現在裝置螢幕越來越大，為了提供更好的使用者經驗，常有直向和橫向畫面長的不同的App出現

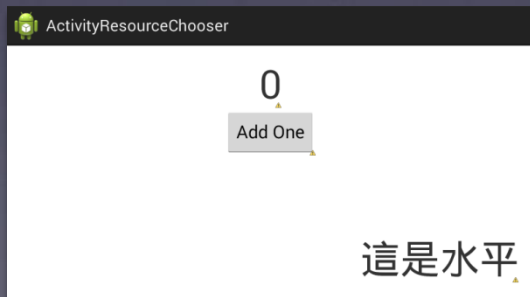


資源選擇

- 藉由Activity動態資源選擇機制 (見第三章)來做
- 直向時的layout擺在layout-port



- 橫向時的layout擺在layout-land



資源選擇

- 接下來只要在程式碼中照正常使用setContentView()的方式即可
- 轉向時要儲存的資訊使用上一段介紹的onSaveInstanceState()來儲存
- 並使用onCreate()的參數或onRestoreInstanceState()的參數來恢復

執行結果

