**Lab8 Service**

1. 本節目的：

* 了解什麼是Service。
* 使用Service執行背景工作。

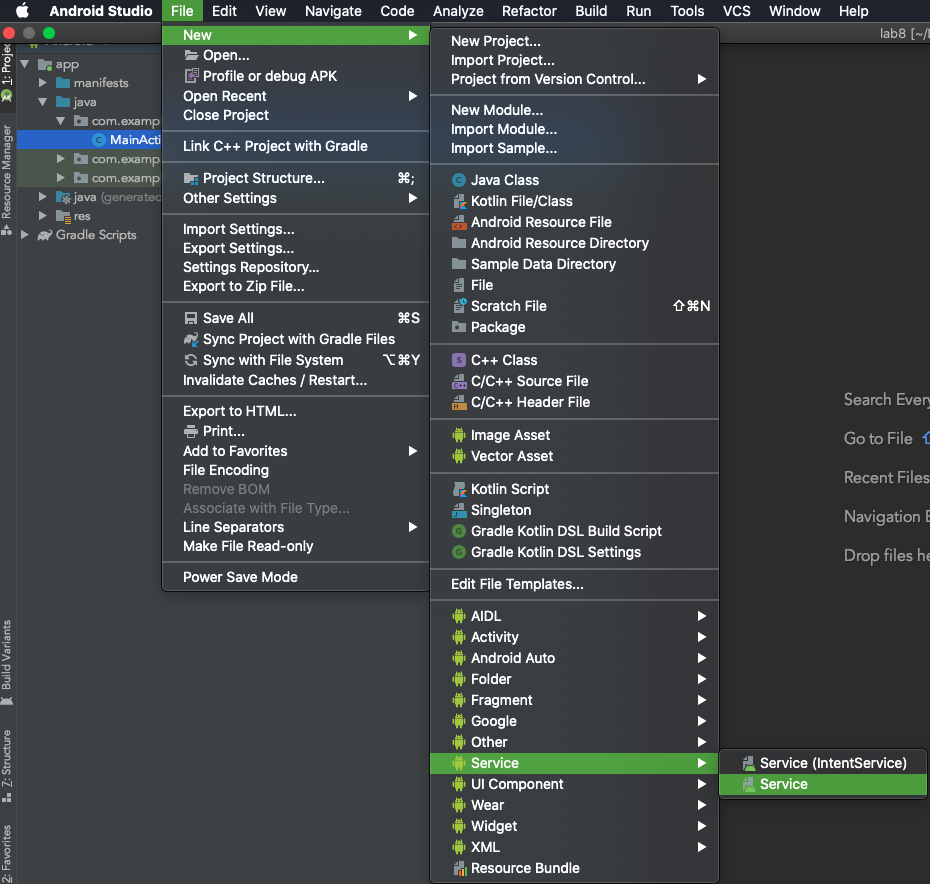
1. 觀念說明：

Activity在離開畫面後會進入停止狀態，在這狀態下我們是無法控制Activity，除非透過調用Thread的方法，但是當APP完全結束關閉後，這些工作也會停止，如果希望能在背景中繼續執行工作，我們就會需要使用到Service來執行背景作業，像是在背景等待網路連線、背景作業等工作。

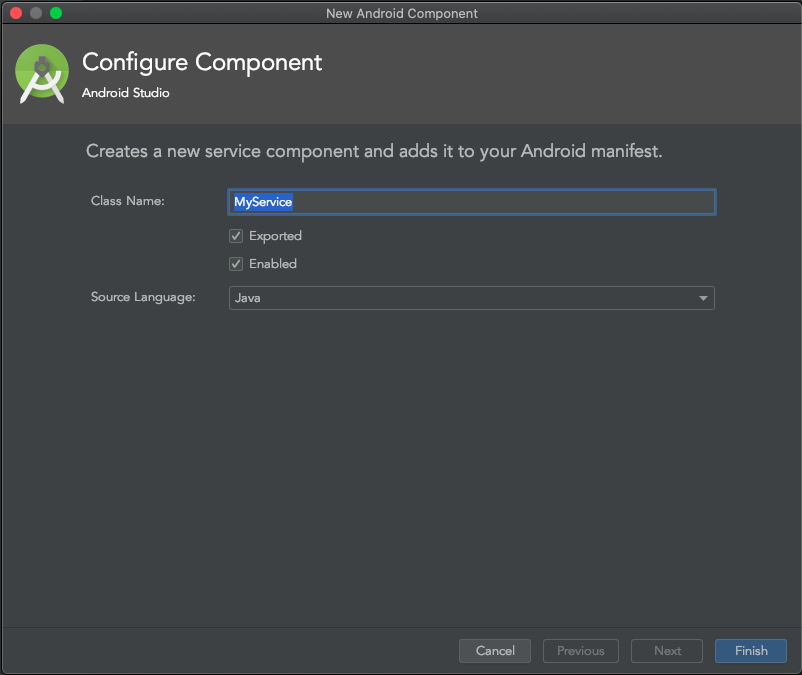
Service最大的特色是他的執行任務與使用者的操作無關，Service會獨立運行於背景，因此能夠APP完全結束關閉後能保持啟動，甚至能在使用者操作別的APP時繼續執行，如等待網路訊息通知、下載資料等作業，一旦任務完成Service就可以被結束，也可能就此常駐於裝置上。

1. 創建Service

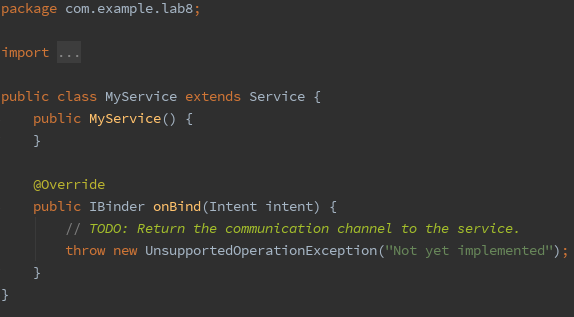
要產生出一個新的Service，首先先選擇File/New/Service/Service來產生出空白的Service。



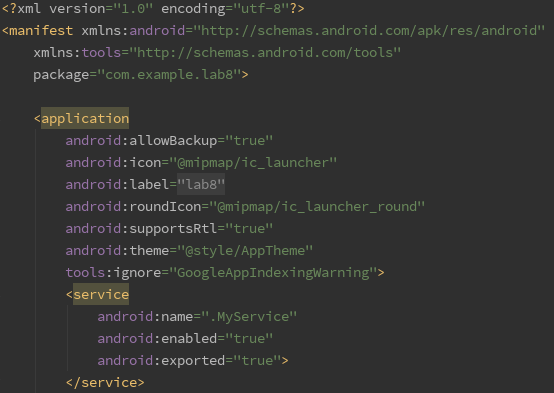
選擇後可於下面的視窗中修改Service的名稱，完成後按下Finish。



完成後，系統會幫你產生出Service的java檔，之後就可以在此編寫Service。

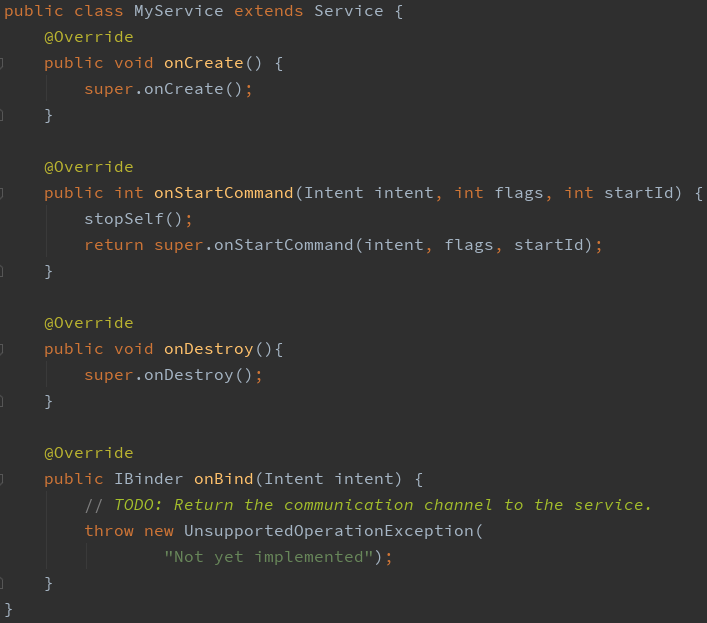


在AndroidManifest.xml中也會自動增加Service的資訊。



1. **啟動Service**

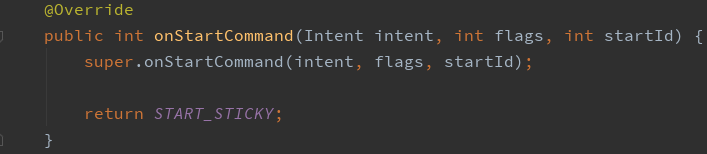
Service擁有獨立運行的能力，第一次產生的Service會執行onCreate()方法，之後會自動調用onStartCommand()方法，我們可以將要執行的工作寫於onStartCommand()之中，而當需要結束時可以調用stopSelf()來讓Service自行進入結束程序，觸發結束程序後皆會調用onDestroy()來結束服務。



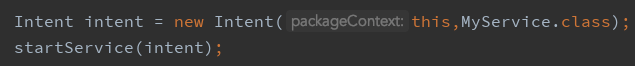
Service最重要的執行階段便是在調用onStartCommand()，onStartCommand()會告知系統如何重啟Service，如異常終止後是否要重新啟動。而運行中的Service如果又接收到Activity發出startService()的請求，Service會執行onStartCommand()而不會再次執行onCreate()。換言之，onStartCommand()扮演著接收外來請求並操作Service的角色。

onStartCommand()的第一個參數為Intent，如果從Activity啟用，可以藉由Intent夾帶資訊傳遞到Service，而返回值主要有三種定義：

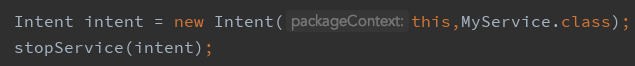
* START\_NOT\_STICKY：如果Service被結束時，便結束服務。
* START\_STICKY：如果Service被結束時，系統會嘗試重啟並再次調用onStartCommand()，不過Intent會被清空。
* START\_REDELIVER\_INTENT：如果Service被結束時，系統會嘗試重啟並再次調用onStartCommand()，不過Intent會保留前次的並重新傳入。



當要從Activity中啟用一個Service，需要在Activity中透過Intent來做切換，並調用startService()方法發出訊息。



如果要關閉Service可以從Activity調用，stopService()方法來做停止。



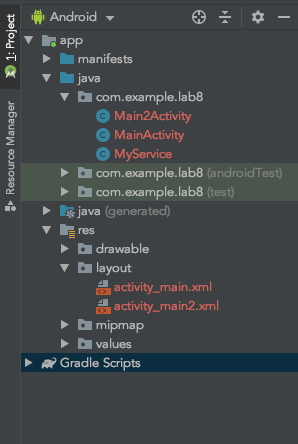
1. 設計重點:

* 本次範例要透過Service來啟動另一個Activity。
* 在MainActivity中，按下”啟動SERVICE”按鈕後啟動Service，並結束MainActivity。
* 在Service中，透過Thread延遲5秒後啟動Main2Activity。

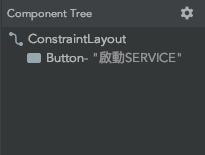
1. 設計步驟:

Step1

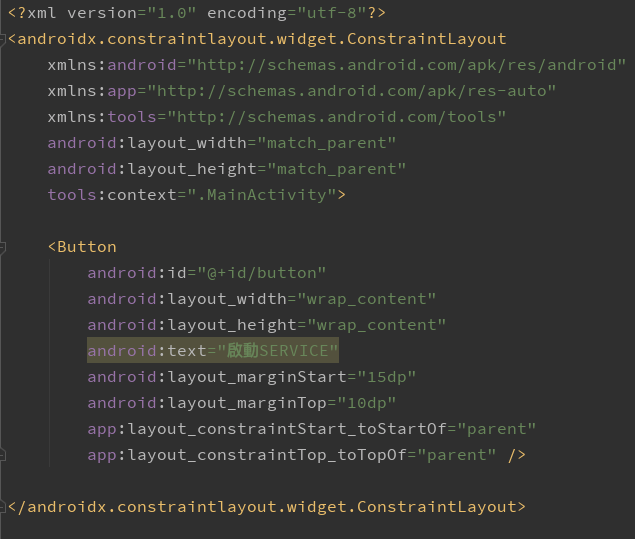
新建專案，並建立兩個Activity與一個Service

Step2

繪製activity\_main.xml檔



對應的xml如下：

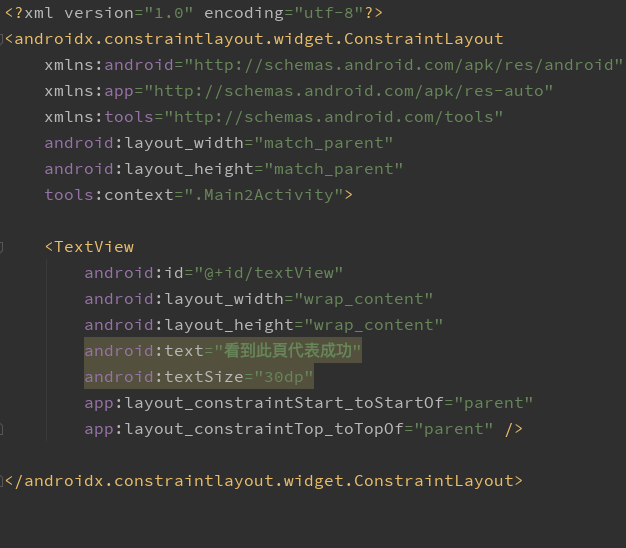


Step3

繪製activity\_main2.xml檔

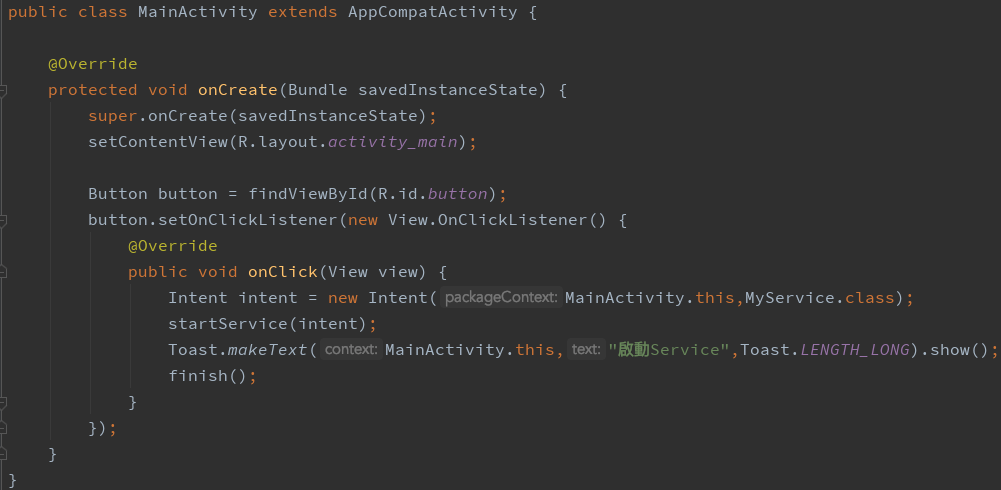


對應的xml如下：



Step4

編寫MainActivity程式，按下按鈕後啟動Service，Main2Activity不做設計，僅用於顯示結果



Step5

編寫MyService程式，在其中加入執行緒延遲5秒後，啟動Main2Activity

