**Lab12**

**API**

本節目的：

* 了解Http Get與Post觀念
* 了解JSON觀念
* 使用GSON轉換JSON與String。
* 了解透過OkHttp來進行網路請求。
  1. **觀念說明：**

在Android中想要透過網路取得資料，或是與他人交換訊息，就需要使用Http通訊機制，Http通訊協定被廣泛應用於網路環境，除了常見的瀏覽器外，Android應用程式也能使用Http協定存取伺服器的資料。

* + 1. **Http觀念**

Http(Hypertext Transfer Protocol)是一種網路通訊協定，用於客戶端發出請求(Request)給伺服器，伺服器將要給予的資料回傳(Response)給客戶端使用，一般我們稱之為API(Application Programming Interface)，如下圖 1手機送出Request與伺服器建立連線，伺服器回傳Response資料，資料通常為Xml或JSON…等格式的字串，需要再透過翻譯器轉換成Android看得懂的資料型態。

****

圖 1、API示意圖

Http通訊協定中常用的方法(HTTP methods)有Http Get與Http Post兩種，用信件來比喻的話，信封的格式就是HTTP，信封外的內容為http-header，信封內的書信為message-body，那麼HTTP Method就是你要告訴郵差的寄信規則，Get就像廣告信，參數直接附加於網址後面，人人都看得到，而Post就是私人信件，裡面的參數只有寄信者與收信者能看到，相對的安全性也較高。

* Http Get

當我們要從網路取得資料時，就需要從客戶端發出Http Get Request跟伺服器建立連線，Get Request請求的參數會附在URL之後(就是直接加在網址後面)， 網址以「?」分割URL和傳輸參數，參數之間以「&」相連，如下網址所示name1和name2是參數的名稱，value1和value2是參數的值，傳送的連結如下，黑字的部分為網址，「?」前是URL，「?」後是要傳輸的參數.多個在and之後便可加上我們的參數，並以「&」做區別：

http://網址?name1=value1&name2=value2

Get可以直接使用瀏覽器顯示。圖 2我們以瀏覽器測試以下網址，從伺服器撈出postId為1的所有文章資料：

https://jsonplaceholder.typicode.com/comments?postId=1

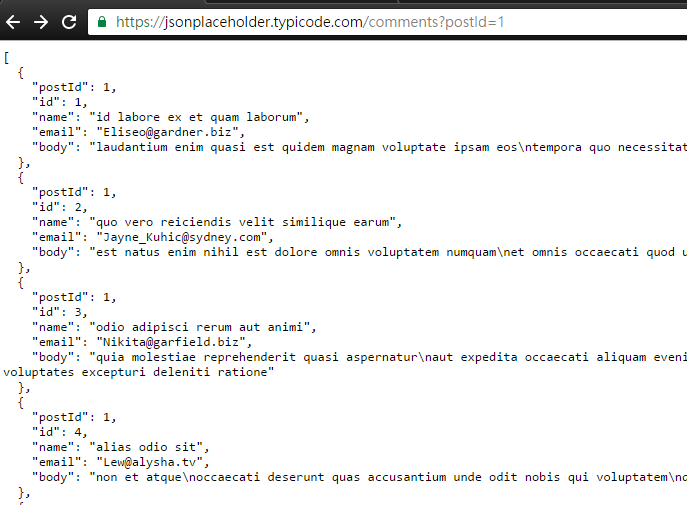


圖 2、Http Get示範

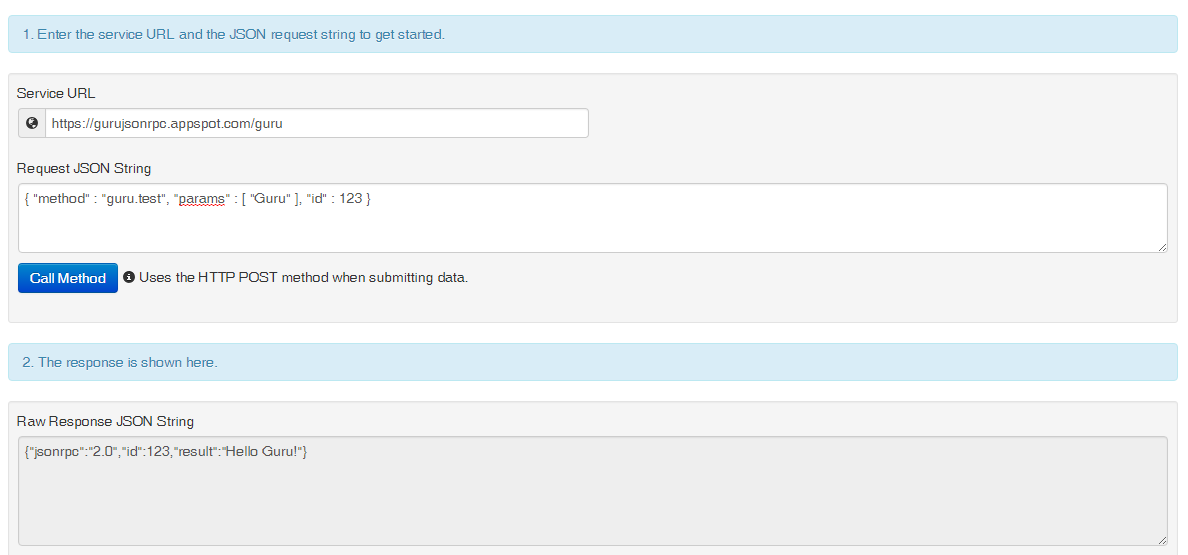
執行之後，瀏覽器的顯示區會出現一組字串，此字串即為伺服器回傳的資料，並採用JSON字串格式表示，有關JSON字串格式會在下一節做說明。

Get使用非常簡單方便，不過也有其缺點：

* 由於Get是直接在網址後加上參數，任何人都能看到。因此不該使用Get Request傳送重要的資料。
* Get Request有長度限制。
* Http Post

不同於Get直接在網址上寫上要傳送的資料，Post將要傳送給伺服器的資料用body另外包起來，隨著request一起送給伺服器，如下圖 3採用[https://gurujsonrpc.appspot.com](https://gurujsonrpc.appspot.com/)網頁來測試Post。

圖 3例子中，採用Post對<https://gurujsonrpc.appspot.com/guru>做請求，搜尋資料的方法為"guru.test"，params為"Guru"字串陣列且id為123的資料。

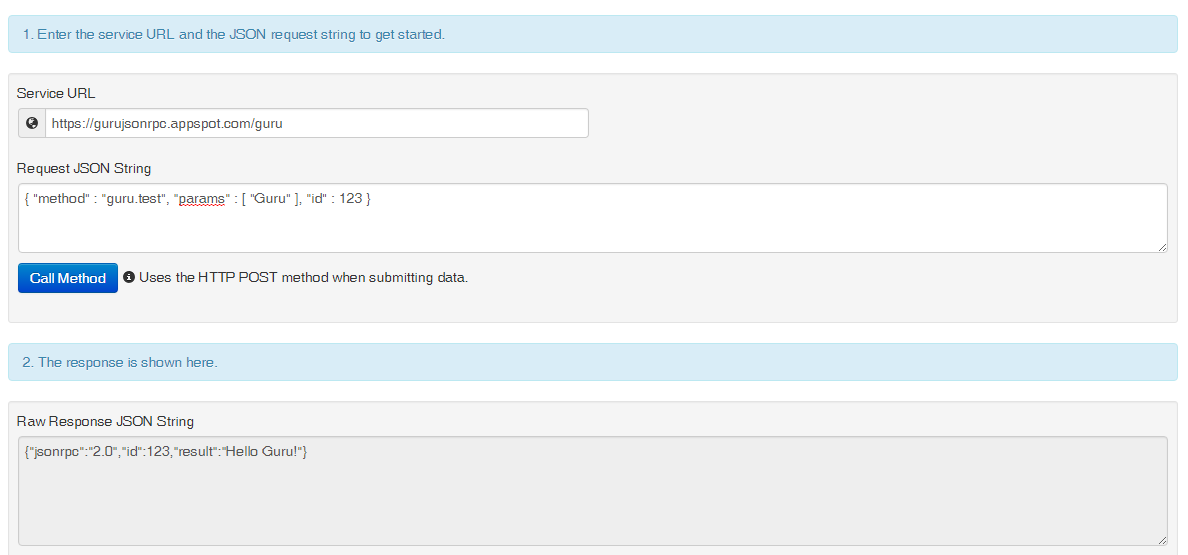


Request的資料

要求的URL

圖 3、Http Post Request

執行之後，伺服器可以接收到客戶端發送過去的資料，經由伺服器執行相關處理後會Response一組JSON字串資料，如下圖 4所示id為123且result帶有Guru的資料。



伺服器Response的資料

圖 4、Response

理解了Get與Post的觀念之後，本教學中會教導在Android上採用JSON 字串格式，透過OkHttp來與後台溝通。

說明

* + 1. **JSON觀念**

HTTP的操作上，傳遞的資料必須是字串形式，因此為了讓資料能透過單一字串送出多組的資料，就必須要設計一套標準化的格式，此教學中會教導使用的是JSON字串格式。

JSON 是個以純文字為基底去儲存和傳送簡單結構資料，你可以透過特定的格式去儲存任何資料(字串、數字、陣列、物件)，也可以透過物件或陣列來傳送較複雜的資料。

JSON 字串採用name : value的表示方式，name表示這個變數的名稱，而value表示其內容`.物件或陣列的 value 值的表示方式如下：

|  |  |
| --- | --- |
| 整數或浮點數 | 直接寫入數值 |
| 字串 | 加上"" |
| 布林函數 (boolean) | true 或 false |
| 陣列 | 加上[ ] |
| 物件 | 加上{ } |

如下方所示，左邊的類別結構以JSON來表示後，會呈現圖 5的結果，變數名稱以字串的方式表示，並以冒號連接變數內容，陣列物件會以一個大括弧包含所有成員，而布林、整數與字串則是直接顯示數值。

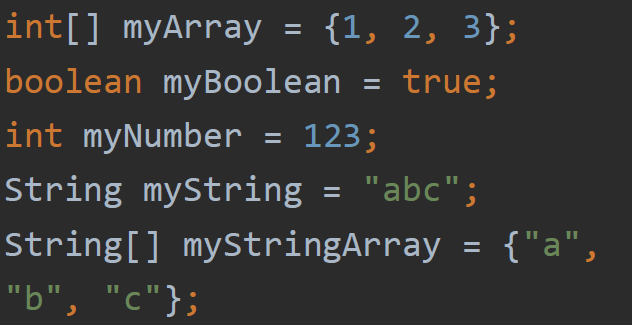
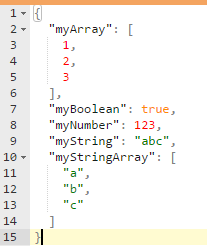


圖 5、JSON資料結構

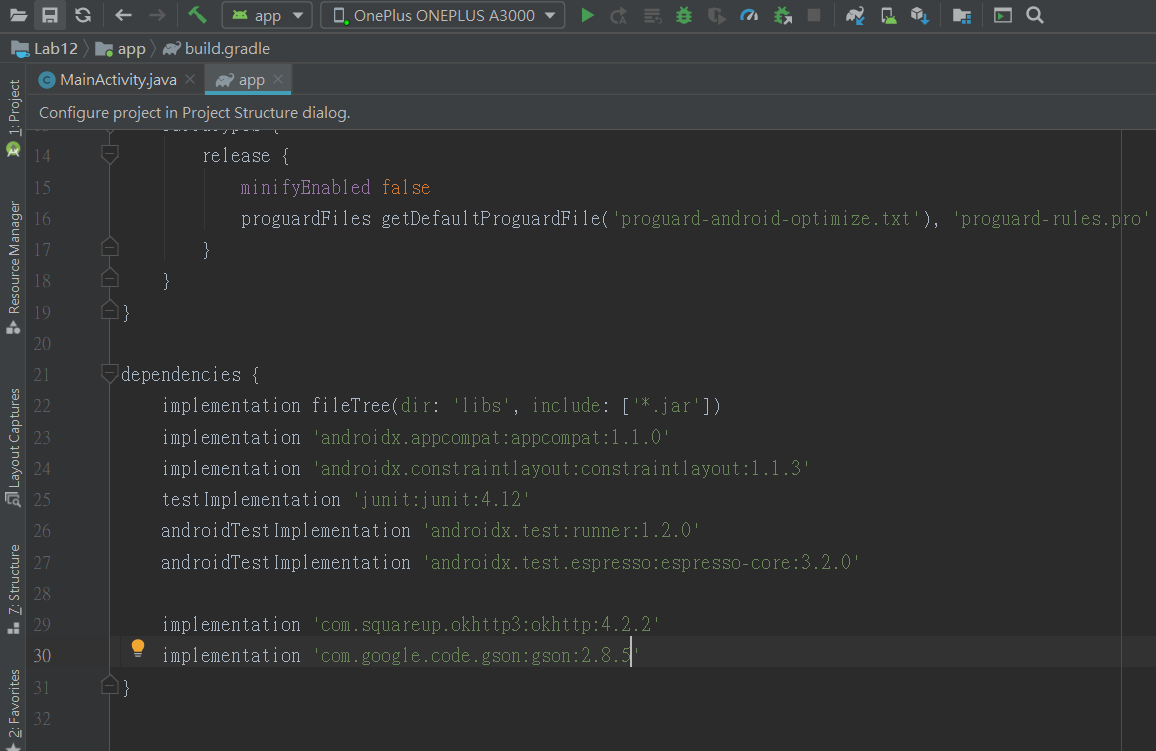
* + 1. **GSON**

圖 1手機透過API與伺服器溝通,由於Http連線存在資料格式的問題，客戶端看不懂伺服器的資料型態，而伺服器看不懂客戶端的類別，因此在送出Http Request前，我們需要將程式的物件轉成JSON字串，這動作叫做序列化(Serialization)，而接收Http Response後，需要將JSON字串轉成程式的物件，這動作叫做反序列化(Deserialization)。Google提供一個開源library **GSON**可以快速的處理物件與JSON格式轉換。



圖 1、API示意圖

要使用GSON，首先我們要引入GSON的library，下圖 6在gradle的dependencies中，我們加入'com.google.code.gson:gson:2.8.5'後讓系統將其匯入。



加入 'com.google.code.gson:gson:2.8.5'

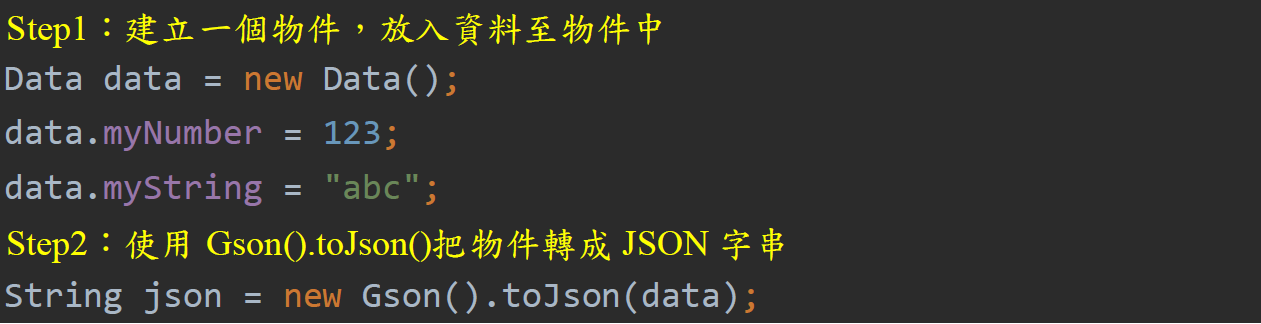
圖 6、加入GSON library

* 序列化(把物件轉成JSON字串)

1. 在轉換之前，我們要先設計好一個要被轉換的Data類別，來接收來自伺服器的資料：



1. 要透過GSON把該物件轉成JSON字串有兩個步驟：



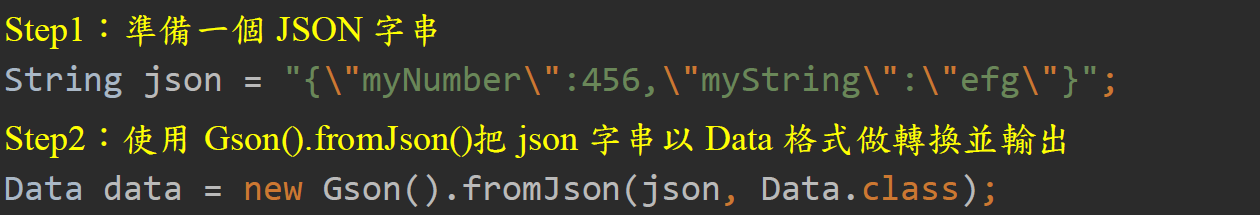
1. 輸出字串後可以看到圖 7的字串結果：



圖 7、JSON字串

* 反序列化(把JSON字串轉成物件)

1. 要透過GSON把JSON字串轉成物件有兩個步驟：



1. 把Data物件中myNumber與myString輸出後的結果如下圖 8：

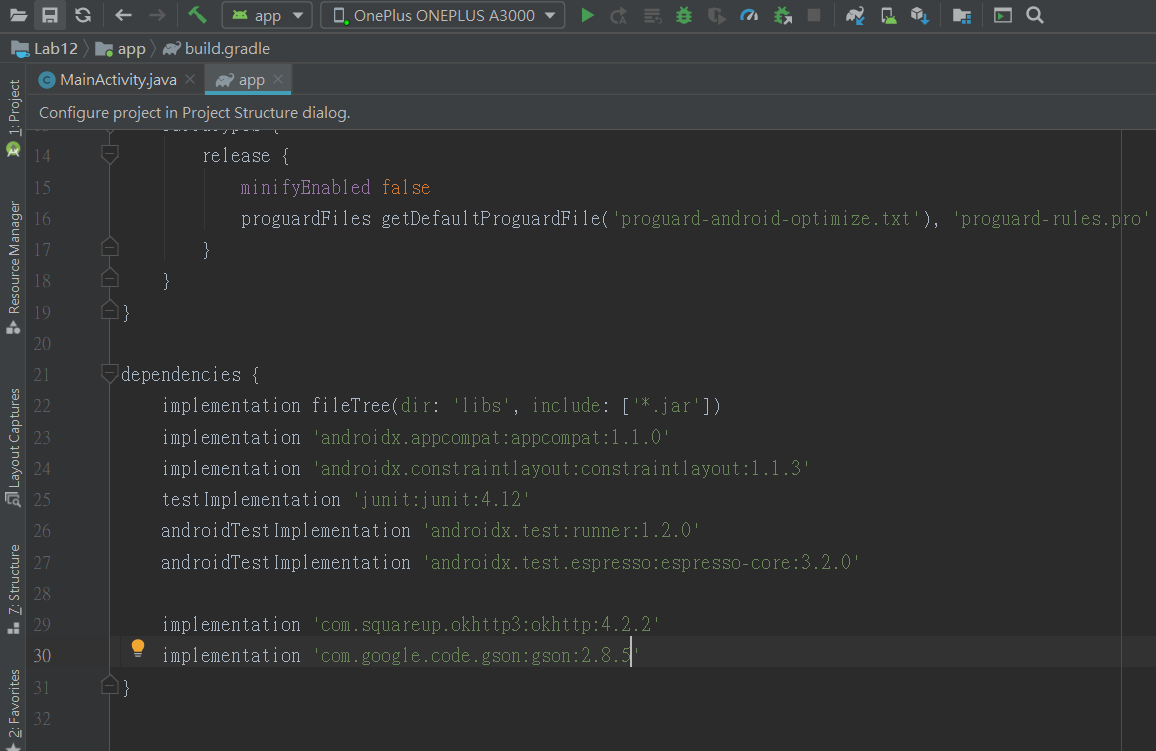


圖 8、JSON資料

* + 1. **OkHttp**

OkHttp 是Square出產的一個Open source project，是一個Http連線的第三方library，使用它快速實作Http Get/Post資料交換的工作，讓HTTP連線的過程更加有效率，能避免人為的錯誤設計，加快程式執行的速度。

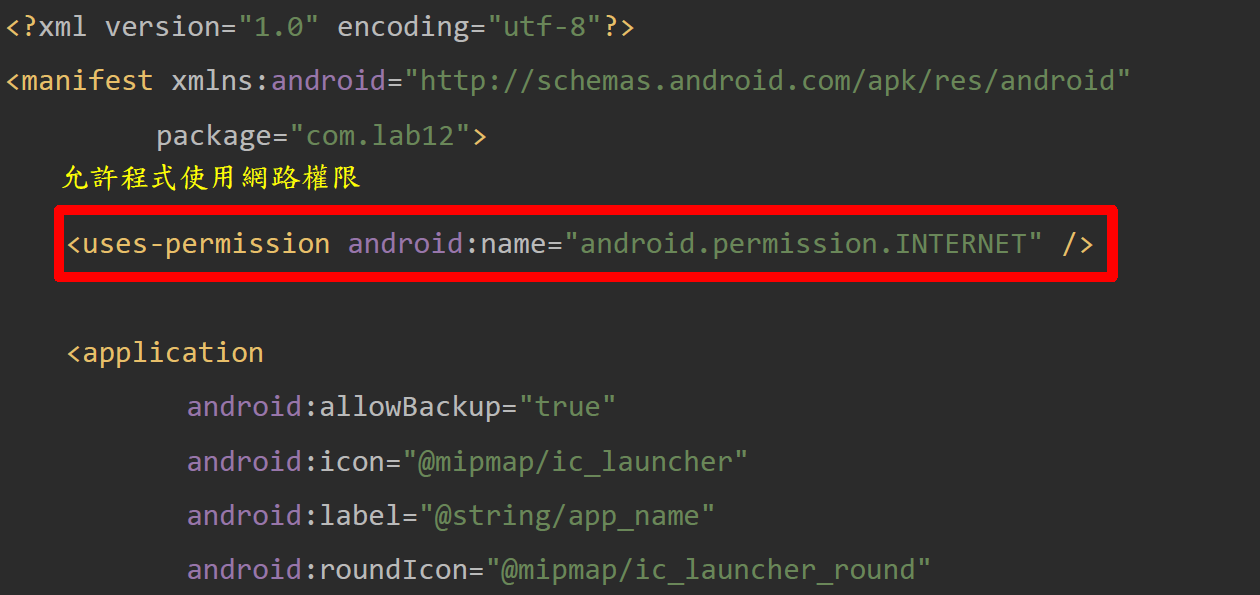
圖 7要使用OkHttp，首先我們要引入OkHttp的library，在gradle的dependencies中，我們加入implementation 'com.squareup.okhttp3:okhttp:4.2.2'，然後讓系統將其匯入。



加入 'com.squareup.okhttp3:okhttp:4.2.2'

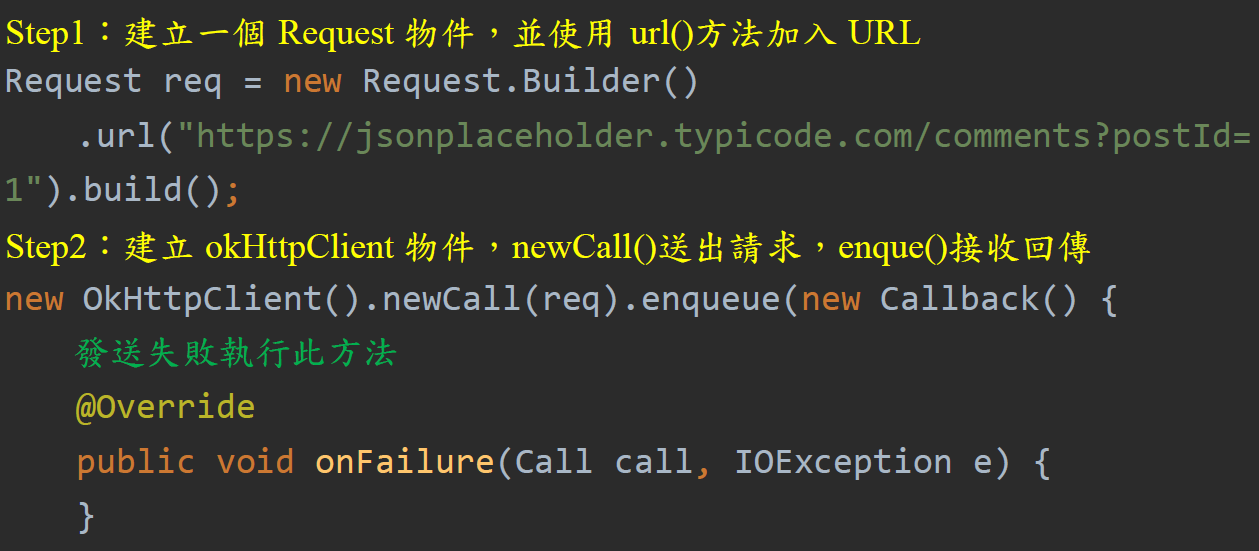
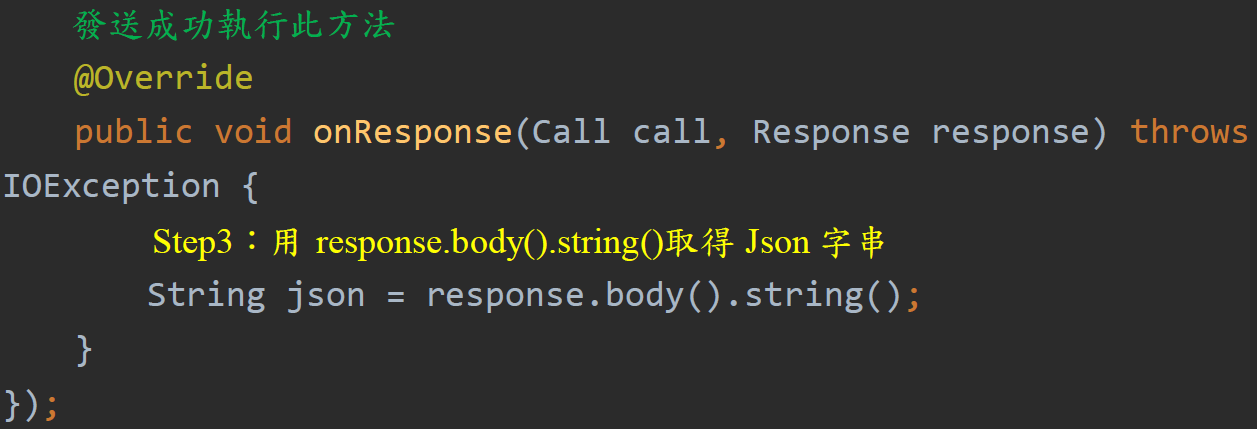
圖 9、加入OKHttp library

接著，由於我們要讓Android進行網路服務，因此需要在AndroidManifest.xml加入網路連線的權限。



* Http Get

OkHttp使用Get Request有以下步驟：

enqueue()接收回傳

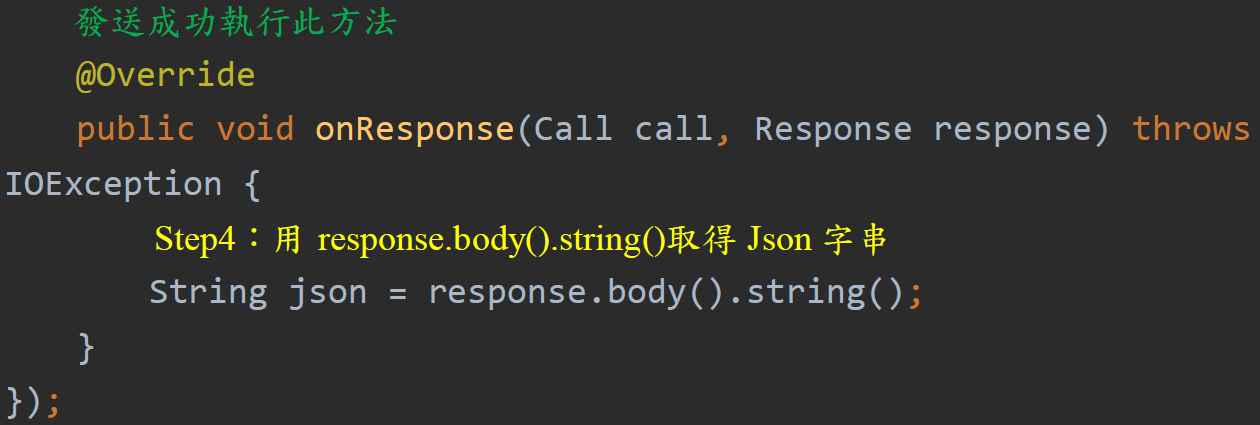
以上就是發送一個get請求的步驟，首先建立一個Request物件，參數要有url來設定網址或是Get Request。

然後通過request的對象去產生得到一個Call物件，類似於將Request封裝成了任務，既然是任務，就會有execute()和cancel()等方法。

最後採用非同步執行的方式去執行，這裡使用newCall().enqueue然後等待任務執行完成，我們便可在Callback中即可得到結果。

* Http Post

OkHttp使用Post Request有以下步驟：

enqueue()接收結果

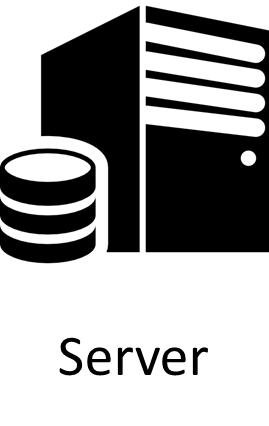
Http Post與Get唯一的差別是Request物件需要多一個post()參數，post()中我們要加入RequestBody物件，RequestBody用來封裝資料(第一個參數)，以及Body的格式(第二個參數)。由於我們要採用JSON的格式傳送，因此格式部分要加入"application/json; charset=utf-8"，資料部分則加入要傳送過去的JSON字串。

說明

* 1. **設計重點：**
* 練習使用Http Get取得“臺北捷運列車到站站名”API資料，流程如下：

API連結：<https://data.taipei/opendata/datalist/apiAccess?scope=resourceAquire&rid=55ec6d6e-dc5c-4268-a725-d04cc262172b>

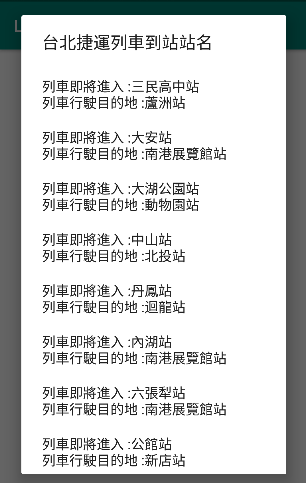
1. 按下按鈕後，送出Http Get取得API response(JSON字串)
2. 使用GSON將Response的JSON字串轉換成物件
3. 從物件中取得Station(列車進入月台車站站名)與Destination(行駛目的地)資訊，並使用AlertDialog顯示



1. Get Request[

<https://data.taipei/opendata/datalist/apiAccess?scope=resourceAquire&rid=55ec6d6e-dc5c-4268-a725-d04cc262172b>]

Station



1. 透過Dialog顯示資訊
2. Get Response(JSON)，使用GSON轉換JSON成物件

Destination

GSON

* 1. **設計步驟：**

Step1 建立新專案，以及下圖 10對應的class和xml檔：

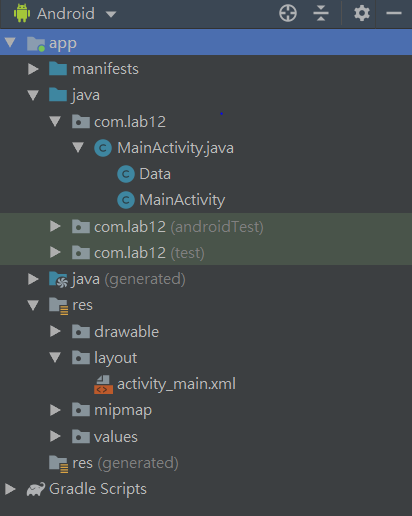
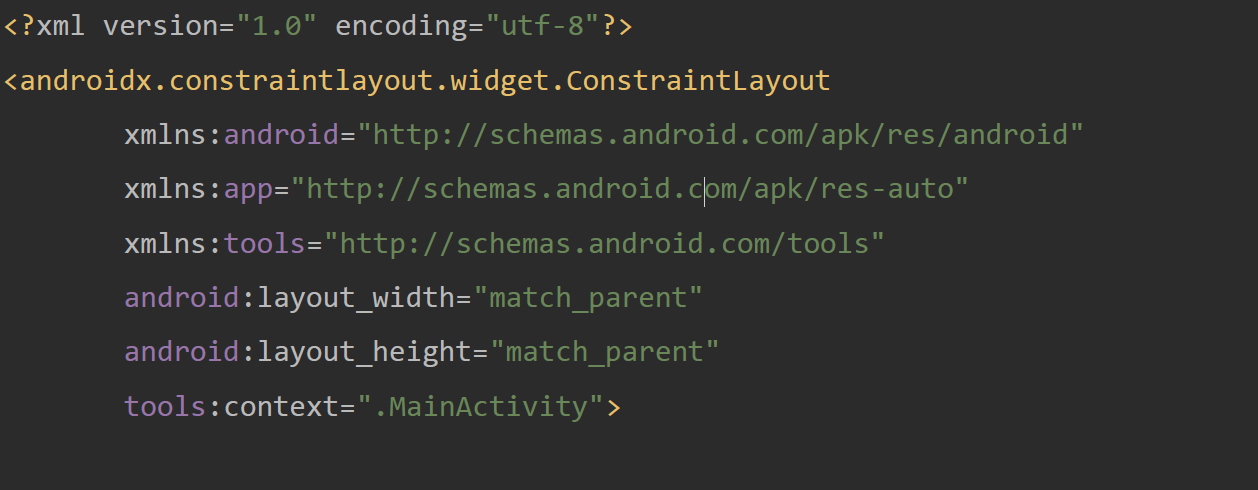


圖 10、專案架構

Step2 繪製activity\_main.xml如圖 11所示:

|  |  |
| --- | --- |
| 圖 11-1、預覽畫面 | 圖 11-2、元件樹 |

對應的xml如下：


Step3 圖 12在gradle的dependencies中引入Gson與OkHttp的library，加入'com.google.code.gson:gson:2.8.5' 與 'com.squareup.okhttp3:okhttp:4.2.2'，然後讓系統將其匯入。

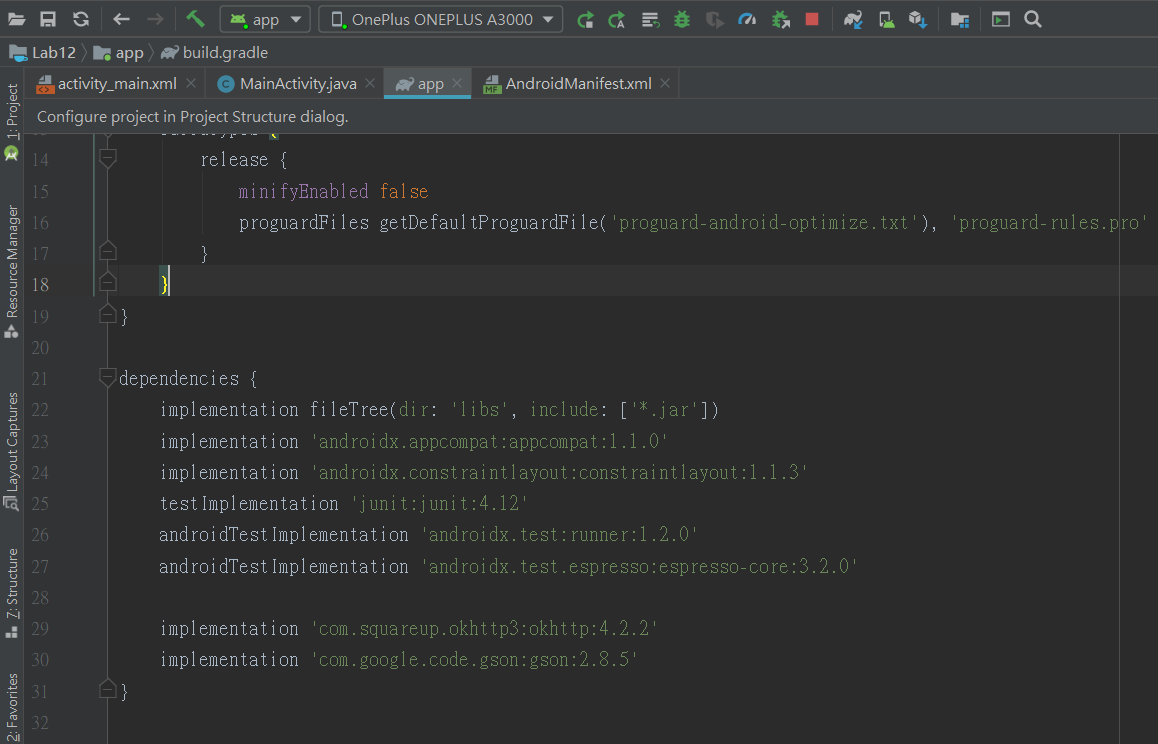
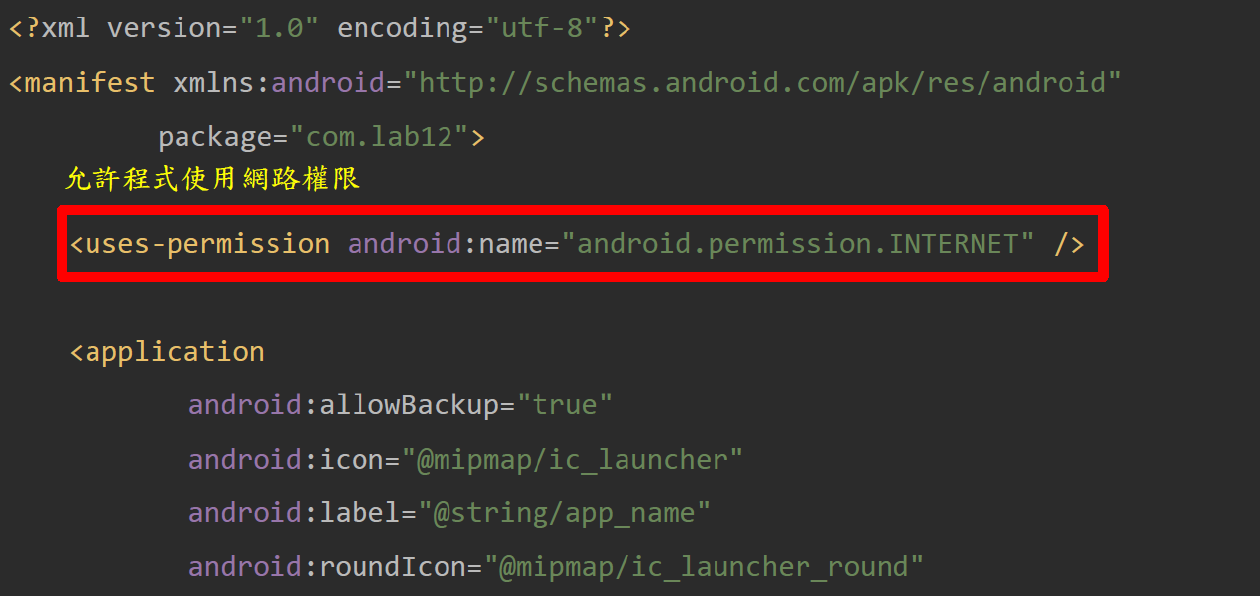


圖 12、加入GSON與OKHttp library

Step4 在AndroidManifest.xml加入網路連線的權限。



Step5 圖 13先以瀏覽器測試Get

[<https://data.taipei/opendata/datalist/apiAccess?scope=resourceAquire&rid=55ec6d6e-dc5c-4268-a725-d04cc262172b>]是否能成功，同時得到Response。



**Response**

圖 13、測試Http GET

Step6 圖 14使用這網頁：<https://jsoneditoronline.org/>，將Response做排版，然後可以得到右邊的資料結構。

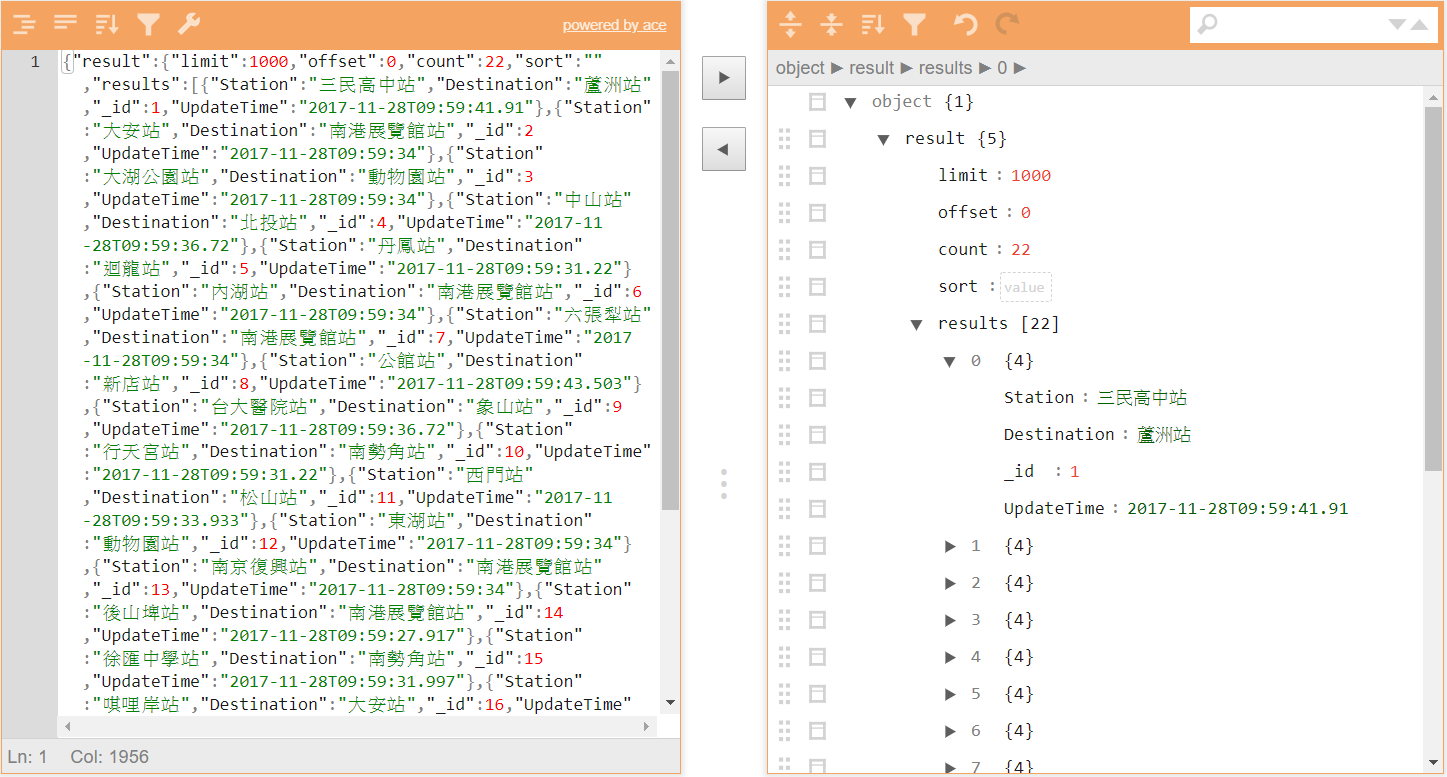
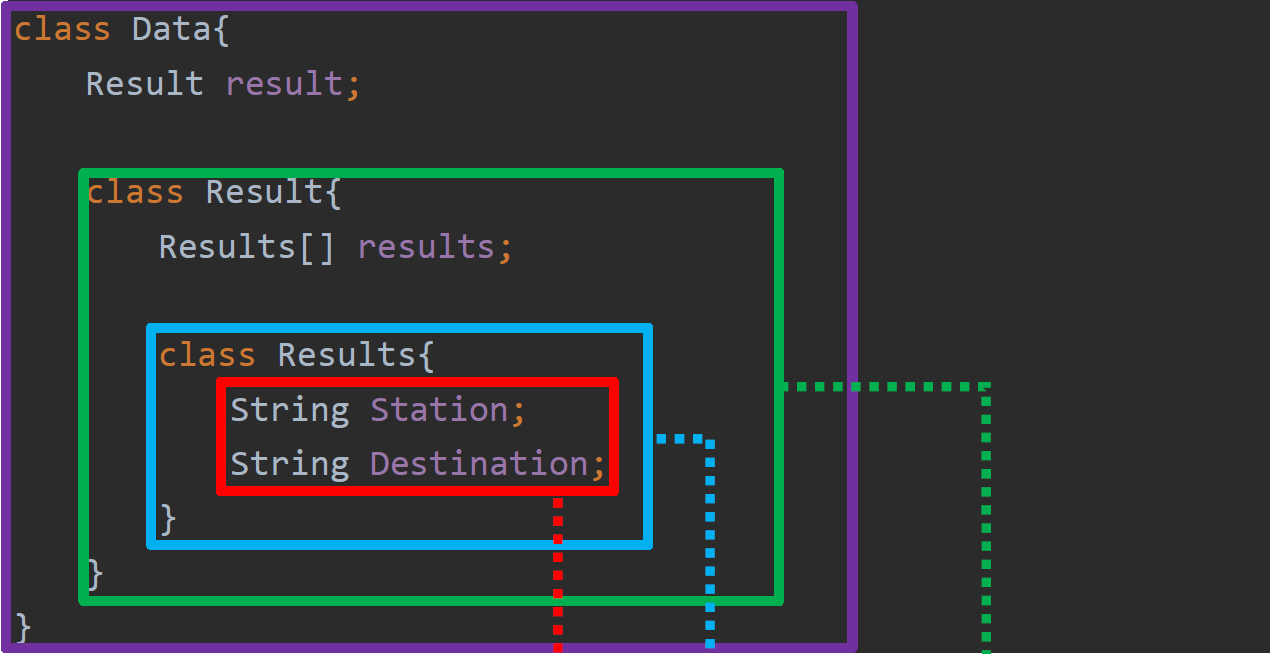
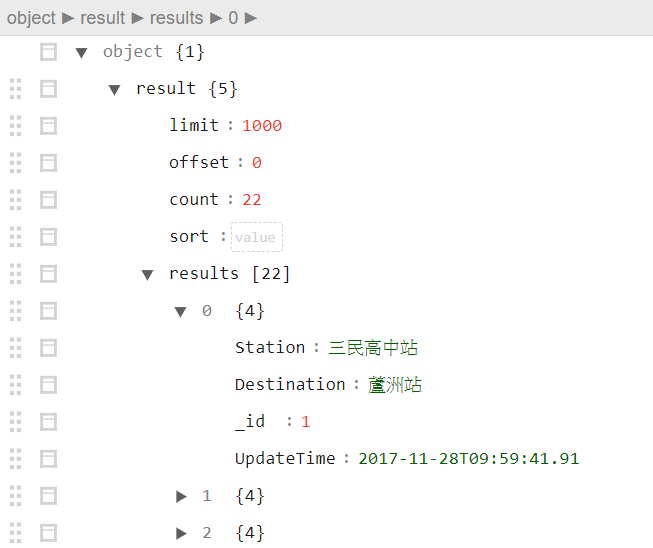


圖 14、排版後的資料

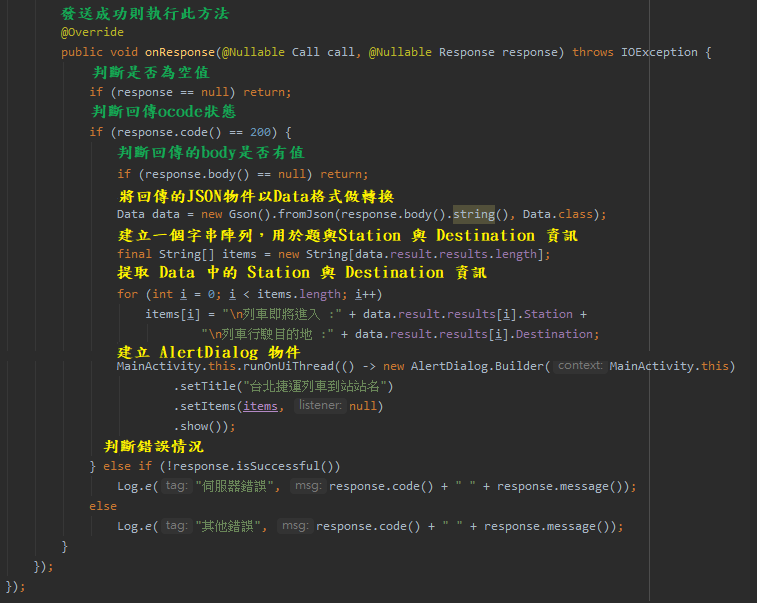
Step7 根據Step6資料結構的JSON，在MainActivity中客製化Data類別，該資料結構必須依照Response來做設計，因此要比對網頁的Response結構去規劃類別內容。由於我們只需要Station與Destination的資訊，因此只需要提取兩者變數即可。





Step8 編寫查詢按鈕按下後，使用OkHttpClient來發出Http Get Request至<https://data.taipei/opendata/datalist/apiAccess?scope=resourceAquire&rid=55ec6d6e-dc5c-4268-a725-d04cc262172b>。



Step9 接收到Response之後，將JSON字串經由GSON轉換至Data物件之中，再將Data物件 轉換放入 AlertDialog 之中