Android Advance 2 Lesson 3 Retrofit and OKhttp

Outline

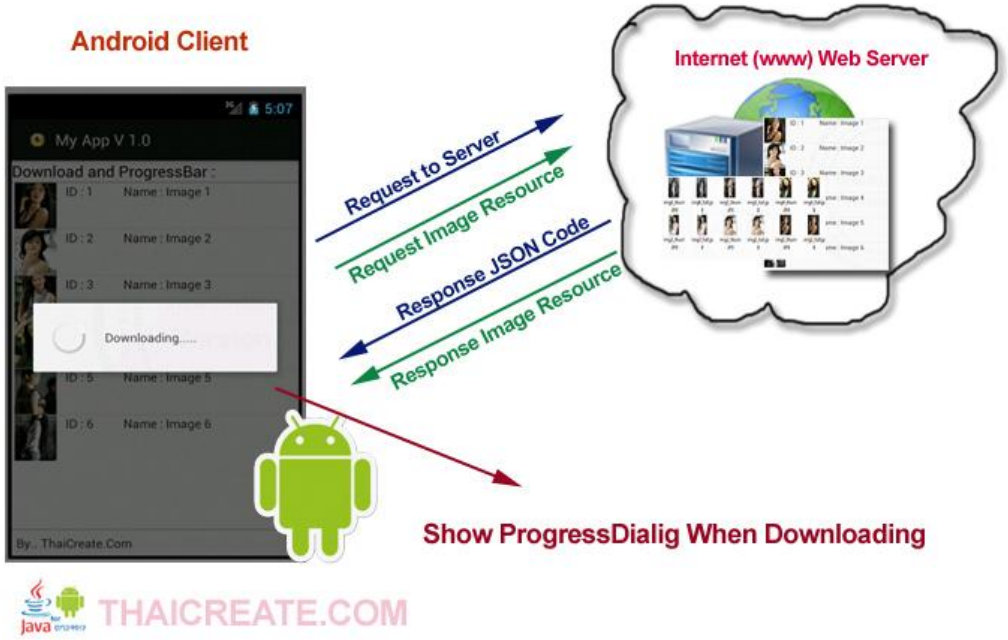
I. Introduction to Retrofit

II. Introduction to OKHttp

I. Introduction to Retrofit

1. What is retrofit?

A type-safe HTTP client for Android and Java



2. Config

- Build.gradle

```
compile 'com.squareup.retrofit2:retrofit:2.2.0'
```

- AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
package="toandoan.framgia.com.android_12_menu_searchview_dialog">

<uses-permission
android:name="android.permission.INTERNET"/
>
</manifest
```

3. Request Methods

GET	Requests data from a specified resource
POST	Submits data to be processed to a specified resource
PUT	Uploads a representation of the specified URI
DELETE	Deletes the specified resource
HEAD	Same as GET but returns only HTTP headers and no document body

4. Converter

Gson	com.squareup.retrofit2:converter-gson
Jackson:	com.squareup.retrofit2:converter-jackson
Moshi:	com.squareup.retrofit2:converter-moshi
Protobuf:	com.squareup.retrofit2:converter-protobuf
Wire:	com.squareup.retrofit2:converter-wire
Simple XML:	com.squareup.retrofit2:converter-simplexml
Scalars (primitives, boxed, and String):	com.squareup.retrofit2:converter-scalars

5. Annotation

Path	Use in case of dynamic url (i.e. username will be swapped for {username} in the URL endpoint).
Query	Add query string to request url
QueryMap	In the case of query string has more value using query map more convenient. Instead of each pair of keys, the value above will correspond to an infix parameter. Here, only one HasMap is passed. Each key (value, value) in the HasMap will be the corresponding pair in the query string
FormUrlEncoded	Field: encoded when passing data to the server, this is the default enctype of the POST
Body	Transmit data implicitly in the body of a request
MultiPart-Part	Used when uploading files.

II. Introduction to OKHttp

What is OKHttp?

- OKHttp is an open source project designed to be an efficient HTTP client. It supports the SPDY protocol. SPDY is the basis for HTTP 2.0 and allows multiple HTTP requests to be multiplexed over one socket connection.

Why we using OKhttp

- OkHttp is an HTTP client that's efficient by default:
  - HTTP/2 support allows all requests to the same host to share a socket.
  - Connection pooling reduces request latency (if HTTP/2 isn't available).
  - Transparent GZIP shrinks download sizes.
  - Response caching avoids the network completely for repeat requests.

Sample

https://github.com/DoanVanToan/OpenEducation1703/tree/develop/Android_15_Retrofit

Q&A

Exercise

- Let’s do a sample search user github apply MVP, Retrofit & OKhttp.

Exercise

- Continue implement project search users on github apply Retrofit & OKhttp instead of HttpUrlConnection.