Retrofit

การทำงาน Client-Server

Mobile App

Internet
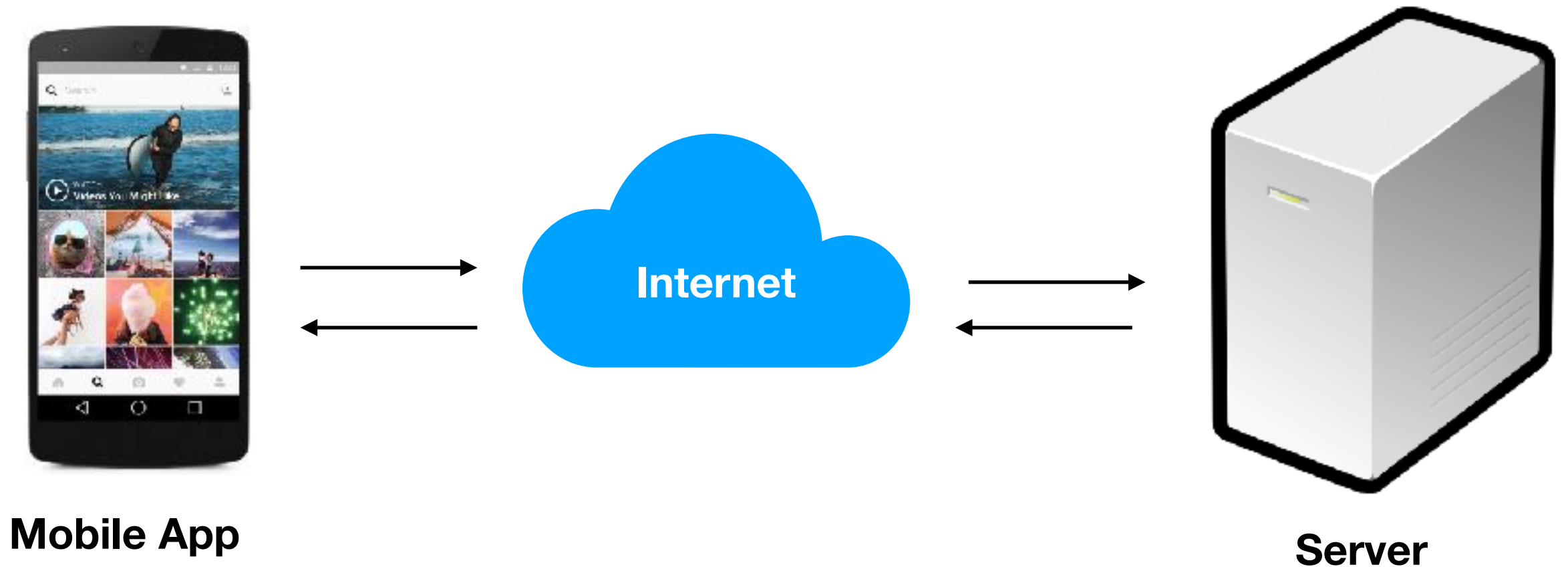
Server

Android Network Library

- HttpsURLConnection (Official)

- Volley

- Retrofit + OKHttp

- Fast Android Networking

Best Android Network Library

	Retrofit + OKHttp	Volley	HttpUrlConnection
ผู้พัฒนา	Square	Google	Google (official)
Support Protocol	Http/Https	Http/Https	Http/Https
Support Header	Support	Support	Support
Support Method	GET/POST/DELETE/PUT	GET/POST/DELETE/PUT	GET/POST/DELETE/PUT
Include Run in Background Thread	Support	No	No
Support GSON/JACKSON	Support	No	No
Support RxJava	Support	No	No
Support REST Api	Support (**Design for Rest Api**)	Support	Support

แลกเปลี่ยนข้อมูลกันอย่างไร

SERVER
https://www.instagram.com

SERVER
https://api.instagram.com

HTML

JSON/XML

WEB
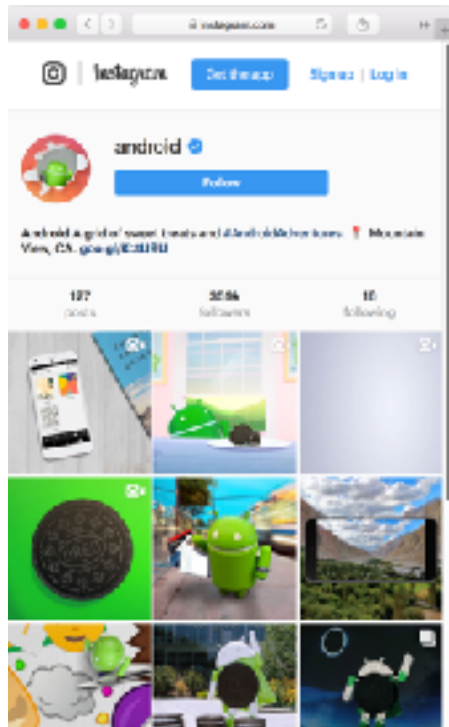
Android App

JSON vs XML

	JSON	XML

ตัวอย่าง	`{`
	`  "name": "John",`
	`  "age": 30`
	`}`
		`<person>`
		`  <name>John</name>`
		`  <age>30</age>`
		`<person>`

โครงสร้าง	เก็บข้อมูล เป็น Key-Value ทำให้อ่านง่าย	เก็บข้อมูล เป็น Hierarchy Tag ทำให้อ่านยาก
Bandwidth	ประหยัด	เปลือง
Parsing	มีตัวช่วย ในการ Parse ข้อมูล	ต้องทำการ Parse ข้อมูลเอง

JSON

Java Script Object Notation

เป็น Plain Text ถูกใช้สำหรับรับส่งข้อมูล

JSON Object

```
{
  "name": "John",
  "age": 30
}
```

JSON Array

```
{
 "cars": [
   {
     "model": "Ford"
   },
   {
     "model": "BMW"
   },
   {
     "model": "Toyota"
   }
 ]
}
```

JSON Object + Array

```
{
  "name": "John",
 "age": 30,
 "cars": [
   {
    "model": "Ford"
   },
   {
    "model": "BMW"
   },
   {
    "model": "Toyota"
   }
 ]
}
```
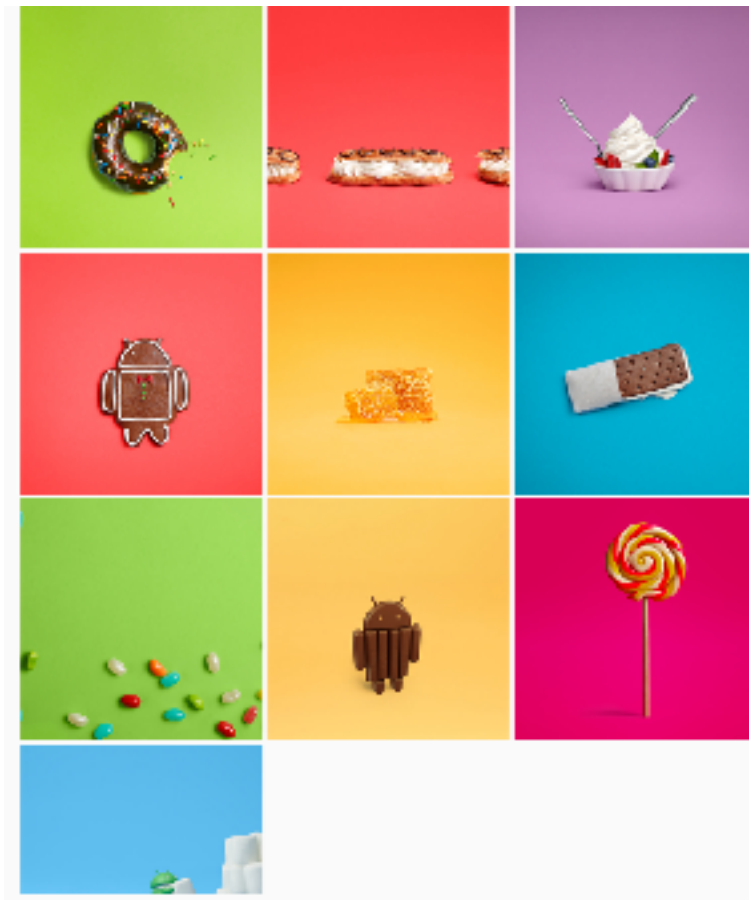
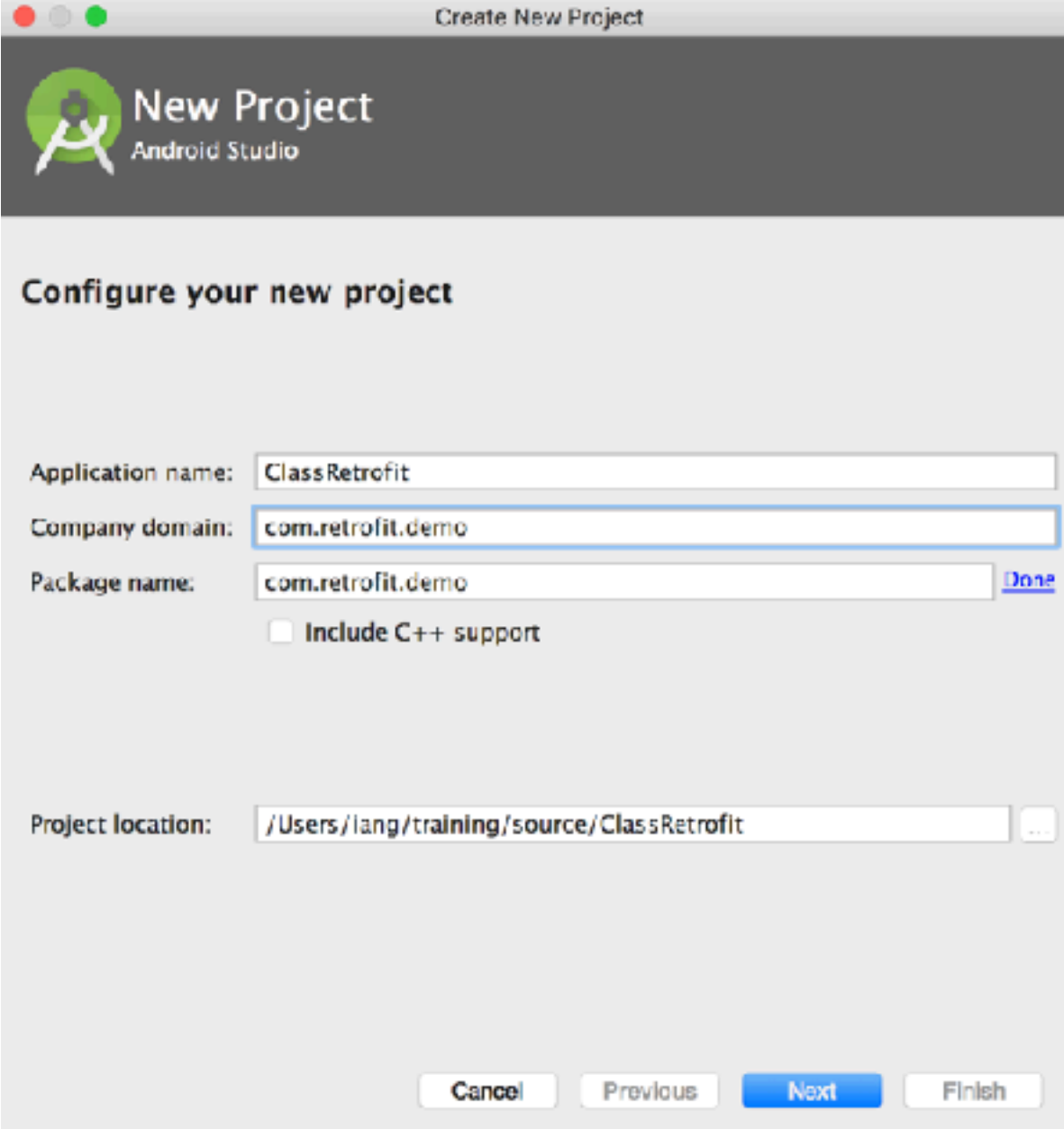# เริ่มลงมือทำ

@android

Post
10

Following
1

Follower
1000000

The official Instagram for Android.

→ **Profile Detail**

→ **Image List**

Create Project

Profile Detail (Top)

**TextView
(id:textUser)**

**TextView
(id:textFollowing)**

**TextView
(id:textFollower)**

@android

Post
10

Following
1

Follower
1000000

The official Instagram for Android.

**TextView
(id:textPost)**

**TextView
(id:textBio)**

**ImageView
(id:imageProfile)**
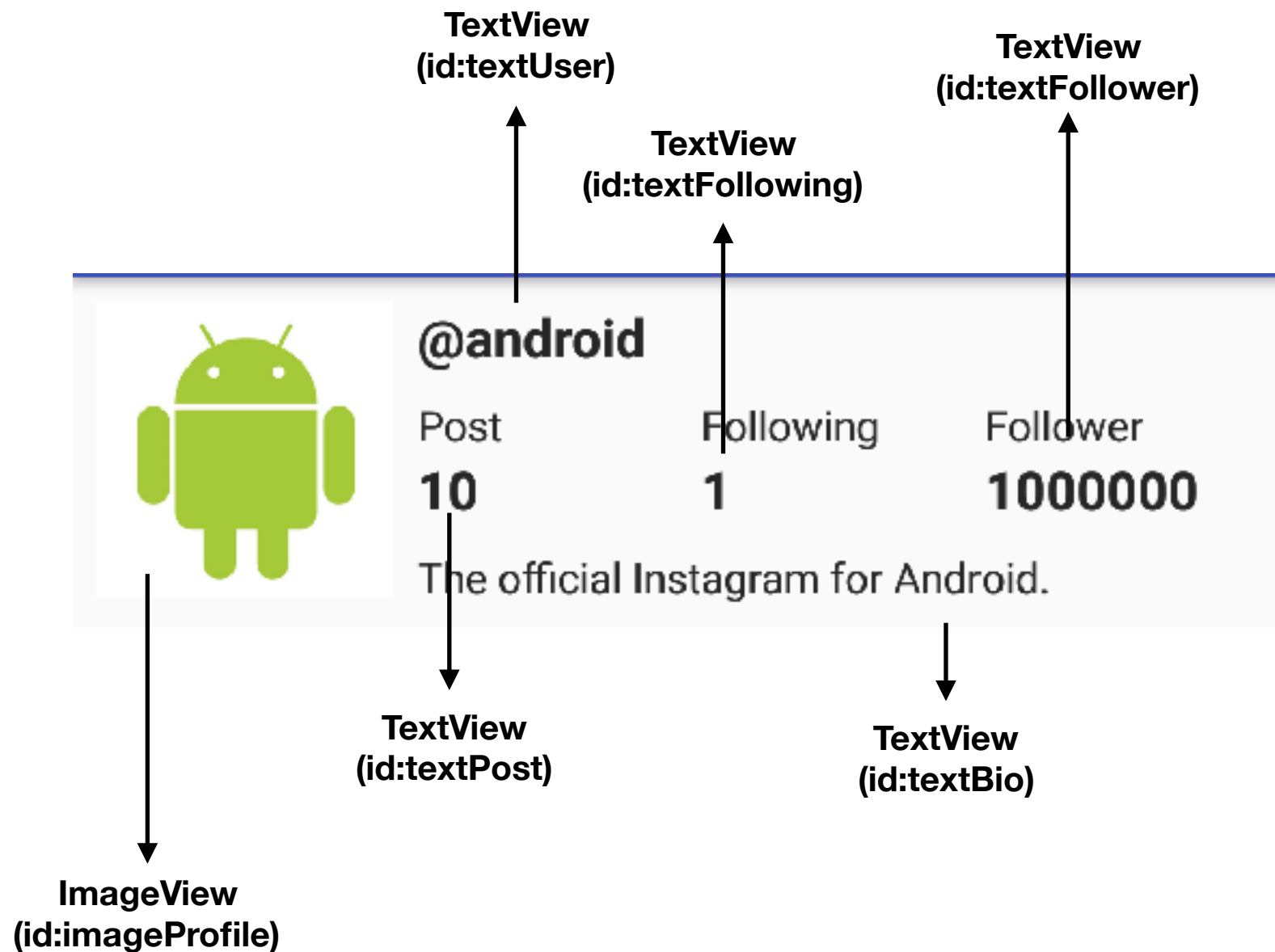
Design UI

activity_main.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="8dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/imageProfile"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_gravity="center_vertical"
            android:background="#2f2f2f" />


        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="8dp"
            android:orientation="vertical">

            <TextView
                android:id="@+id/textUser"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="name" />

            <include layout="@layout/sub_profile_detail" />


            <TextView
                android:id="@+id/textBio"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginTop="8dp"
                android:singleLine="true"
                android:text="Bio : bla bla bla" />

        </LinearLayout>

    </LinearLayout>

</LinearLayout>
```

sub_profile_detail.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/textPost"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Post\n100" />

    <TextView
        android:id="@+id/textFollower"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Follower\n100" />

    <TextView
        android:id="@+id/textFollwing"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Following\n100" />

</LinearLayout>
```

Internet Permission

เพิ่มขออนุญาตเชื่อมต่อ Internet ใน Manifest file

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

↓

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

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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Add Retrofit Library ใน Cradle

```
compile 'com.squareup.retrofit2:retrofit:2.3.0'
compile 'com.google.code.gson:gson:2.8.1'
compile 'com.squareup.retrofit2:converter-scalars:2.3.0'
compile 'com.squareup.retrofit2:converter-gson:2.3.0'
```

Get Profile API

Request

```
Method          :GET
URL.             :https://us-central1-retrofit-course.cloudfunctions.net/getProfile
Query String   :user
```

Response

ContentType JSON

```
{
  "profileUrl": "https://example/com/profile.jpg",
  "user": "android",
  "bio": "The official Instagram for Android.",
  "post": 10,
  "follower": 1000000,
  "following": 1,
  "isFollow": false,
  "posts": [
   {
     "comment": 20,
     "like": 800,
     "url": "https://example/com/image1.jpg"
   },{
     "comment": 99,
     "like": 2323,
     "url": "https://example/com/image2.jpg"
   }
  ]
}
```

Retrofit

เชื่อมต่อ api ด้วย Retrofit (ผลลัพธ์ เป็น String)

สร้าง Interface Api

```java
public interface Api {

    String BASE = "https://us-central1-retrofit-course.cloudfunctions.net";

    @GET("/getProfile")
    Call<String> getProfile(@Query("user") String user);

}
```

สร้าง Method สำหรับดึงข้อมูลจาก Api

```java
private void getUserProfile(String name) {
    OkHttpClient client = new OkHttpClient.Builder().build();
    Retrofit retrofit = new Retrofit.Builder()
            .baseUrl(Api.BASE)
            .client(client)
            .addConverterFactory(ScalarsConverterFactory.create())
            .build();

    Api api = retrofit.create(Api.class);
    api.getProfile(name).enqueue(new retrofit2.Callback<String>() {
        @Override
        public void onResponse(retrofit2.Call<String> call, retrofit2.Response<String> response) {
            Log.d(TAG, "onResponse: " + response.body());
        }

        @Override
        public void onFailure(retrofit2.Call<String> call, Throwable t) {
            Log.d(TAG, "onFailure");
        }
    });

}
```

Retrofit

เชื่อมต่อ api ด้วย Retrofit (ผลลัพธ์ เป็น Object)

สร้าง Interface Api

```java
public interface Api {

    String BASE = "https://us-central1-retrofit-course.cloudfunctions.net";

    @GET("/getProfile")
    Call<UserProfile> getProfile(@Query("user") String user);

}
```

สร้าง Method สำหรับดึงข้อมูลจาก Api

```java
private void getUserProfile(String name) {
    OkHttpClient client = new OkHttpClient.Builder().build();
    Retrofit retrofit = new Retrofit.Builder()
            .baseUrl(Api.BASE)
            .client(client)
            .addConverterFactory(GsonConverterFactory.create())
            .build();

    Api api = retrofit.create(Api.class);
    api.getProfile(name).enqueue(new retrofit2.Callback<UserProfile>() {
        @Override
        public void onResponse(retrofit2.Call<UserProfile> call, retrofit2.Response<UserProfile> response) {
            Log.d(TAG, "onResponse: " + response.body());
        }

        @Override
        public void onFailure(retrofit2.Call<UserProfile> call, Throwable t) {
            Log.d(TAG, "onFailure");
        }
    });

}
```

GSON Mapping

JAVA Class

```java
public class UserProfile {

    private String bio;
    private int follower;
    private int following;
    private boolean isFollow;
    private int post;

    private String urlProfile;
    private String user;


    //Setter & Getter
}
```

JSON Data

```json
{

  "bio": "The official Instagram for Android.",
  "follower": 1412,
  "following": 7643,
  "isFollow": "false",
  "post": 23,
  "posts": [],
  "urlProfile": "https://lh3.googleusercontent.com/DE1k9KAl2teMTpbb1",
  "user": "android"


}
```

Map Response to UI

```java
private void display(UserProfile userProfile){
    TextView textUser = findViewById(R.id.textUser);
    textUser.setText("@"+userProfile.getUser());

    TextView textPost = findViewById(R.id.textPost);
    textPost.setText("Post\n"+userProfile.getPost());

    TextView textFollower = findViewById(R.id.textFollower);
    textFollower.setText("Follower\n"+userProfile.getFollower());

    TextView textFollwing = findViewById(R.id.textFollwing);
    textFollwing.setText("Following\n"+userProfile.getFollowing());

    TextView textBio = findViewById(R.id.textBio);
    textBio.setText(userProfile.getBio());

}
```

OnResponse call display

```java
public void onResponse(retrofit2.Call<UserProfile> call, retrofit2.Response<UserProfile> response) {
    display(response.body());
}
```

แสดงรูป Profile

ใช้ Library (Glide) ช่วยในการ โหลดรูปจาก Internet

เพิ่ม library repo

```
allprojects {
    repositories {
        jcenter()
        mavenCentral()
        maven { url 'https://maven.google.com' }
    }
}
```

Add dependency

```
compile 'com.github.bumptech.glide:glide:4.1.1'
annotationProcessor 'com.github.bumptech.glide:compiler:4.1.1'
```

เรียกใช้

```
ImageView imageProfile = findViewById(R.id.imageProfile);
Glide.with(this).load(userProfile.getUrlProfile()).into(imageProfile);
```

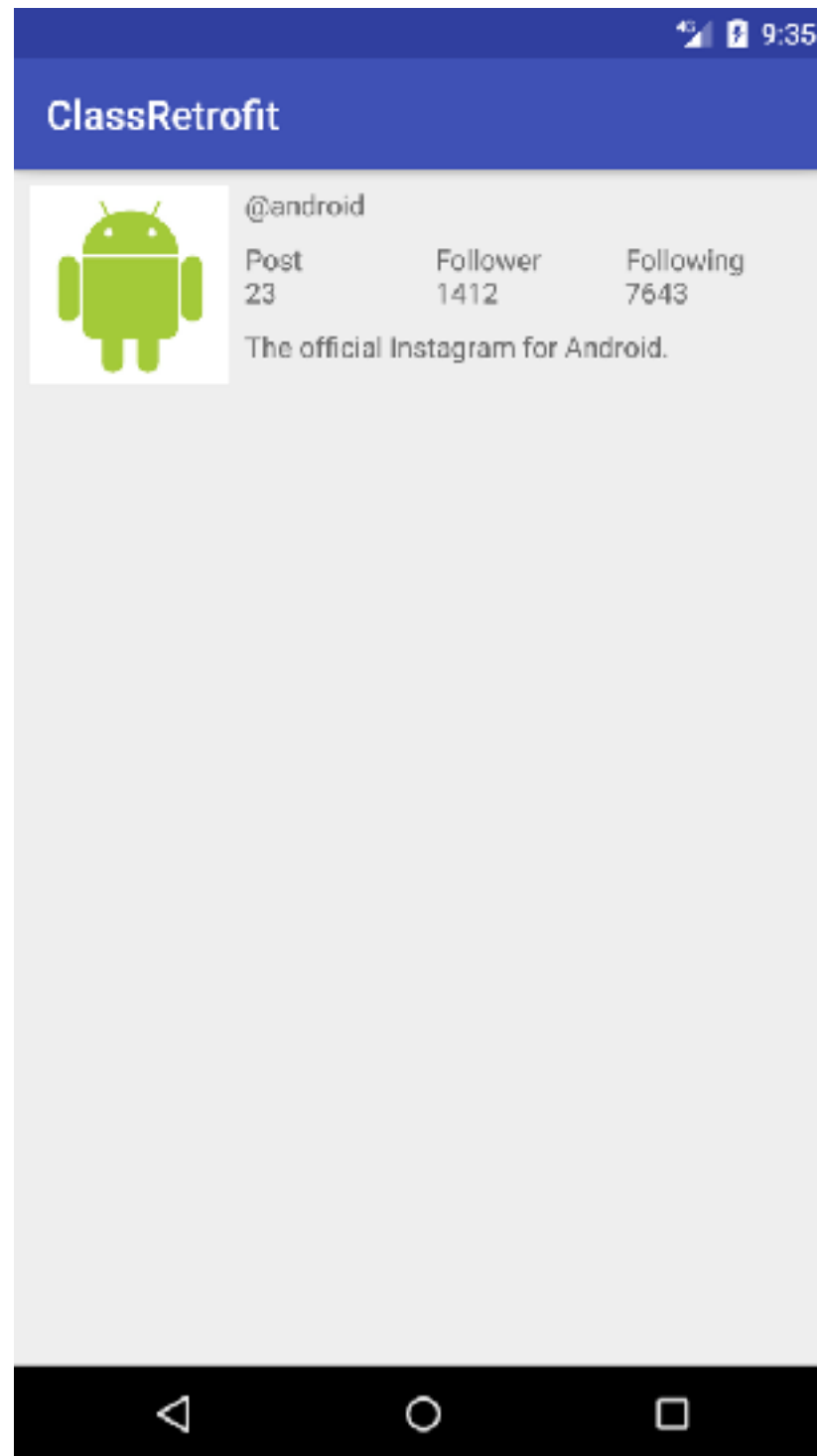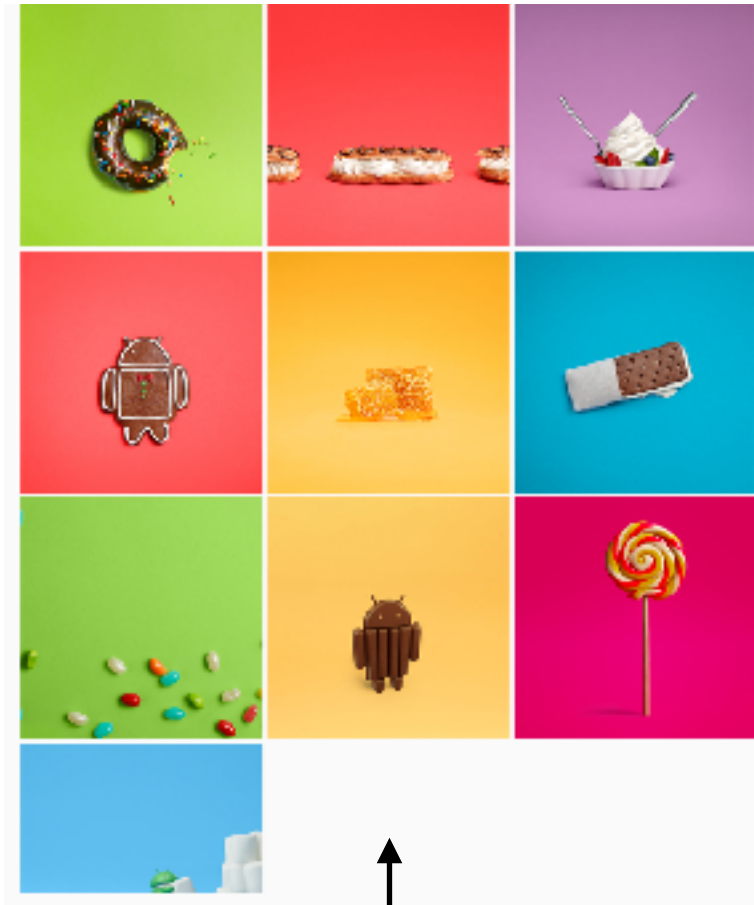# Run

Image List (Bottom)

**RecyclerView
(id:list)**

RecyclerView

เป็น View สำหรับแสดงข้อมูลเป็น List

Add dependency

```
compile 'com.android.support:recyclerview-v7:26.1.0'
```

เพิ่ม Layout

```xml
<android.support.v7.widget.RecyclerView
    android:id="@+id/list"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_marginTop="8dp"
    android:layout_weight="1" />
```

RecyclerView

เพิ่ม Layout ใหม่ item.xml

```xml
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_margin="1dp"
        android:background="#eff0f1"
        android:scaleType="centerCrop"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintDimensionRatio="H,1:1"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>
```

Adapter

```java
public class PostAdapter extends RecyclerView.Adapter<PostAdapter.Holder>{

    private Activity activity;
    private List<PostModel> data;

    public PostAdapter(Activity activity) {
        this.activity = activity;
        data = new ArrayList<>();
    }

    public void setData(List<PostModel> data) {
        this.data = data;
    }

    @Override
    public Holder onCreateViewHolder(ViewGroup parent, int viewType) {
        View itemView = LayoutInflater.from(parent.getContext()).inflate(R.layout.item, null);
        Holder holder = new Holder(itemView);
        return holder;
    }

    @Override
    public void onBindViewHolder(Holder holder, int position) {
        String imageUrl = data.get(position).getUrl();
        Glide.with(activity).load(imageUrl).into(holder.imageView);
    }

    @Override
    public int getItemCount() {
        return data.size();
    }

    static class Holder extends RecyclerView.ViewHolder {
        ImageView imageView;
        public Holder(View itemView) {
            super(itemView);
            imageView = itemView.findViewById(R.id.imageView);
        }
    }
}
```

RecyclerView

เรียกใช้ RecyclerView ใน method display

```java
RecyclerView list = findViewById(R.id.list);
list.setLayoutManager(new GridLayoutManager(this, 3));
PostAdapter adapter = new PostAdapter(this);
adapter.setData(userProfile.getPosts());
list.setAdapter(adapter);
```

QA