**Get API: Android**

**1. จัดเตรียม Recycler View**

ที่ไฟล์ Gradle Scripts > build.gradle (Module : app): เพิ่ม

implementation **'androidx.recyclerview:recyclerview:1.1.0'**

implementation **'com.github.bumptech.glide:glide:4.10.0'**

ที่ไฟล์ app > manifests > AndroidManifests.xml: เพิ่ม

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

**2. สร้าง Fragment ที่ชื่อ RecyclerView**

ทำการเพิ่ม recycler view layout ที่ไฟล์ fragment\_recycler\_view.xml

   <**androidx.recyclerview.widget.RecyclerView**

**android:id="@+id/recyLayout"**

**android:layout\_width="match\_parent"**

**android:layout\_height="match\_parent"**/>

ที่ไฟล์ activity\_main.xml

ทำการกำหนด ID ให้กับ Layout

**android:id="@+id/layout"**

และทำการเพิ่ม ในไฟล์ MainActivity.kt

**val** fragmentRecyclerView = **RecyclerView**()  
**val** transaction = *supportFragmentManager*.beginTransaction()  
transaction.replace(R.id.*layout*, fragmentRecyclerView,**"fragmentRecyclerView"**)  
transaction.addToBackStack(**"fragmentRecyclerView"**)  
transaction.commit()

**3. สร้าง Layout File ชื่อ recy\_layout**

ที่ไฟล์ recy\_layout.xml

--- หน้าต่อไป ---

แก้ไขในไฟล์

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:id="@+id/recyview\_layout"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="@android:color/white"  
 android:padding="16dp"**>  
  
 <**ImageView  
 android:id="@+id/image"  
 android:layout\_width="100dp"  
 android:layout\_height="100dp"  
 android:src="@mipmap/ic\_launcher"  
 android:contentDescription="@string/todo"** />  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:padding="10dp"**>  
  
 <**TextView  
 android:id="@+id/text1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/text1"  
 android:textSize="22sp"**/>  
 <**TextView  
 android:id="@+id/text2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/text2"  
 android:textSize="18sp"**/>  
  
 </**LinearLayout**>  
  
</**LinearLayout**>

**\*\*\*\*แล้วแต่ข้อมูลที่ต้องการจะแสดงออกมา\*\*\*\***

**4. สร้าง Kotlin Class ที่ชื่อ MyRecyclerAdapter**

**class** MyRecyclerAdapter(context: Context, **val dataSource**: JSONArray) : RecyclerView.Adapter<MyRecyclerAdapter.Holder>() {

**private val thiscontext** : Context = context

**class** Holder(view : View) : RecyclerView.ViewHolder(view) {

**private val View** = view;

**lateinit var layout** : LinearLayout

**lateinit var text1**: TextView

**lateinit var text2**: TextView

**lateinit var image**: ImageView

**fun** Holder(){

**layout** = **View**.findViewById<View>(R.id.*recyview\_layout*) **as** LinearLayout

**text1** = **View**.findViewById<View>(R.id.*text1*) **as** TextView

**text2** = **View**.findViewById<View>(R.id.*text2*) **as** TextView

**image** = **View**.findViewById<View>(R.id.*image*) **as** ImageView

   }

}

**override fun** onCreateViewHolder(parent : ViewGroup, viewType: Int): Holder {

**return** Holder(LayoutInflater.from(parent.*context*).inflate(R.layout.*recy\_layout*, parent, **false**))

}

**override fun** getItemCount(): Int {

**return dataSource**.length()

}

**override fun** onBindViewHolder(holder: Holder, position: Int) {

   holder.Holder()

  holder.**text1**.text = (**dataSource**.getJSONObject(position).getString(**"title"**).toString())

  holder.**text2**.text = (**dataSource**.getJSONObject(position).getString(**"description"**).toString())

   Glide.with(**thiscontext**)

       .load(**dataSource**.getJSONObject(position).getString(**"image"**).toString())

       .into(holder.**image**)

   holder.**layout**.setOnClickListener**{**

Toast.makeText(**thiscontext**,holder.**titleTextView**.*text*.toString(),Toast.*LENGTH\_SHORT*).show()

**}**

}

}

**\*\*\*\*แล้วแต่ข้อมูลที่ต้องการจะแสดงออกมา\*\*\*\***

**5. จัดเตรียม AsyncTask & Http get**

ที่ไฟล์ Gradle Script > build.gradle (Module:app) ข้างล่าง buildTypes

packagingOptions {

   exclude **'META-INF/DEPENDENCIES'**

exclude **'META-INF/LICENSE'**

exclude **'META-INF/LICENSE.txt'**

exclude **'META-INF/license.txt'**

exclude **'META-INF/NOTICE'**

exclude **'META-INF/NOTICE.txt'**

exclude **'META-INF/notice.txt'**

exclude **'META-INF/ASL2.0'**

}

และ

implementation **'org.apache.httpcomponents:httpclient:4.5.6'**

**6. สร้างคลาส HttpGetRequest**

**class** HttpGetRequest : AsyncTask<String, Void, String>() {

**override fun** doInBackground(**vararg** urls: String?): String {

**var** result: String? = **""**

**try** {

**val** httpGet = HttpGet(urls.get(0))

**val** client: HttpClient = DefaultHttpClient()

**val** response: HttpResponse = client.execute(httpGet)

**val** statusCode: Int = response.getStatusLine().getStatusCode()

**if** (statusCode == 200) {

**val** inputStream: InputStream = response.getEntity().getContent()

**val** reader = BufferedReader(InputStreamReader(inputStream))

**var** line: String?

**while** (reader.readLine().*also* **{** line = **it }** != **null**) {

                   result += line

               }

           }

       } **catch** (e: ClientProtocolException) {

       } **catch** (e: IOException) {

       }

**return** result!!

   }

**override fun** onPostExecute(result: String?) {

**super**.onPostExecute(result)

   }

}

**7. ส่วนของการแสดงข้อมูล**

*// Some url endpoint that you may have*

**val** myUrl = **"URL"**

*// String to place our result in*

**val** result : String

*// Instantiate new instance of our class*

**val** getRequest = HttpGetRequest()

*// Perform the doInBackground method, passing in our url*

result = getRequest.execute(myUrl).get()

และทำการจัดการกับข้อมูล