**Loading Remote Data with Android Application**

**1. Add Retrofit Dependencies**

Add the following dependencies to your build.gradle file:

kotlin

Copy code

implementation 'com.squareup.retrofit2:retrofit:2.9.0'

implementation 'com.squareup.retrofit2:converter-gson:2.9.0'

implementation 'com.google.code.gson:gson:2.10'

**2. Create the API Model**

Create a data class for the API response, e.g., User.kt:

kotlin

Copy code

data class User(

val id: Int,

val name: String,

val email: String

)

**3. Define the Retrofit Interface**

Create a Retrofit interface, ApiService.kt:

kotlin

Copy code

import retrofit2.Call

import retrofit2.http.GET

interface ApiService {

@GET("users")

fun getUsers(): Call<List<User>>

}

**4. Create the Retrofit Client**

Create a singleton object for Retrofit, RetrofitClient.kt:

kotlin

Copy code

import retrofit2.Retrofit

import retrofit2.converter.gson.GsonConverterFactory

object RetrofitClient {

private const val BASE\_URL = "https://jsonplaceholder.typicode.com/"

val instance: ApiService by lazy {

Retrofit.Builder()

.baseUrl(BASE\_URL)

.addConverterFactory(GsonConverterFactory.create())

.build()

.create(ApiService::class.java)

}

}

**5. Set Up the Activity**

In your activity (e.g., MainActivity.kt), fetch and display the data:

kotlin

Copy code

import android.os.Bundle

import android.widget.Toast

import androidx.appcompat.app.AppCompatActivity

import androidx.recyclerview.widget.LinearLayoutManager

import androidx.recyclerview.widget.RecyclerView

import retrofit2.Call

import retrofit2.Callback

import retrofit2.Response

class MainActivity : AppCompatActivity() {

private lateinit var recyclerView: RecyclerView

private lateinit var userAdapter: UserAdapter

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

recyclerView = findViewById(R.id.recyclerView)

recyclerView.layoutManager = LinearLayoutManager(this)

fetchUsers()

}

private fun fetchUsers() {

RetrofitClient.instance.getUsers().enqueue(object : Callback<List<User>> {

override fun onResponse(call: Call<List<User>>, response: Response<List<User>>) {

if (response.isSuccessful) {

response.body()?.let { users ->

userAdapter = UserAdapter(users)

recyclerView.adapter = userAdapter

}

} else {

Toast.makeText(this@MainActivity, "Error: ${response.code()}", Toast.LENGTH\_SHORT).show()

}

}

override fun onFailure(call: Call<List<User>>, t: Throwable) {

Toast.makeText(this@MainActivity, "Failure: ${t.message}", Toast.LENGTH\_SHORT).show()

}

})

}

}

**6. Create the RecyclerView Adapter**

Create an adapter for the RecyclerView, UserAdapter.kt:

kotlin

Copy code

import android.view.LayoutInflater

import android.view.View

import android.view.ViewGroup

import android.widget.TextView

import androidx.recyclerview.widget.RecyclerView

class UserAdapter(private val users: List<User>) : RecyclerView.Adapter<UserAdapter.UserViewHolder>() {

class UserViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {

val nameTextView: TextView = itemView.findViewById(R.id.nameTextView)

val emailTextView: TextView = itemView.findViewById(R.id.emailTextView)

}

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

val view = LayoutInflater.from(parent.context).inflate(R.layout.item\_user, parent, false)

return UserViewHolder(view)

}

override fun onBindViewHolder(holder: UserViewHolder, position: Int) {

val user = users[position]

holder.nameTextView.text = user.name

holder.emailTextView.text = user.email

}

override fun getItemCount() = users.size

}

**7. Define Layouts**

**activity\_main.xml**

xml

Copy code

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/recyclerView"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:listitem="@layout/item\_user" />

**item\_user.xml**

xml

Copy code

<LinearLayout

xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

android:padding="16dp">

<TextView

android:id="@+id/nameTextView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textStyle="bold"

android:textSize="16sp" />

<TextView

android:id="@+id/emailTextView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="14sp" />

</LinearLayout>

**8. Testing the Application**

1. Ensure your BASE\_URL in RetrofitClient points to a valid API.
2. Run the app. The list of users should display in a RecyclerView.

This approach directly managing network calls and UI updates within the activity.