**Retrofit2 CRUD Android Example :**

Add the following dependencies to your ***build.gradle*** file.

implementation 'com.squareup.retrofit2:retrofit:2.4.0'

implementation 'com.squareup.retrofit2:converter-gson:2.4.0'

implementation 'com.google.code.gson:gson:2.6.1'

***build.gradle***

apply plugin: 'com.android.application'

android {

compileSdkVersion 26

defaultConfig {

applicationId "com.jackrutorial.androidretrofit2crud"

minSdkVersion 21

targetSdkVersion 26

versionCode 1

versionName "1.0"

testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"

}

buildTypes {

release {

minifyEnabled false

proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'

}

}

}

dependencies {

implementation fileTree(dir: 'libs', include: ['\*.jar'])

implementation 'com.android.support:appcompat-v7:26.1.0'

implementation 'com.android.support.constraint:constraint-layout:1.1.0'

testImplementation 'junit:junit:4.12'

androidTestImplementation 'com.android.support.test:runner:1.0.2'

androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'

implementation 'com.squareup.retrofit2:retrofit:2.4.0'

implementation 'com.squareup.retrofit2:converter-gson:2.4.0'

implementation 'com.google.code.gson:gson:2.6.1'

}

Add the permission to access internet in the ***AndroidManifest.xml*** file.

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

***AndroidManifest.xml***

<?xml version="1.0" encoding="utf-8"?><manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.jackrutorial.androidretrofit2crud">

<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>

<activity

android:name=".UserActivity"

android:parentActivityName=".MainActivity"

>

<!-- Parent activity meta-data to support 4.0 and lower -->

<meta-data

android:name="android.support.PARENT\_ACTIVITY"

android:value=".MainActivity" />

</activity>

</application>

</manifest>

## **User Model**

Create a new package to ***src/main/java*** with the name ***com.jackrutorial.androidretrofit2crud.mode***l.

Create a ***User*** class under ***com.jackrutorial.androidretrofit2crud.model*** package and write the following code in it.

package com.jackrutorial.androidretrofit2crud.model;

import com.google.gson.annotations.Expose;

import com.google.gson.annotations.SerializedName;

public class User {

@SerializedName("id")

@Expose

private int id;

@SerializedName("name")

@Expose

private String name;

public User() {

}

public User(int id, String name) {

this.id = id;

this.name = name;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

## **Retrofit Interface**

### **UserService**

Create a new package to ***src/main/java*** with the name ***com.jackrutorial.androidretrofit2crud.remote***.

Create a ***UserService*** interface under ***com.jackrutorial.androidretrofit2crud.remote*** package. In this interface, we'll define ***GET, POST, PUT, DELETE*** method that perform HTTP requests with ***@GET, @POST, @PUT, @DELETE*** annotations and write the following code in it.

package com.jackrutorial.androidretrofit2crud.remote;

import com.jackrutorial.androidretrofit2crud.model.User;

import java.util.List;

import retrofit2.Call;

import retrofit2.http.Body;

import retrofit2.http.DELETE;

import retrofit2.http.GET;

import retrofit2.http.POST;

import retrofit2.http.PUT;

import retrofit2.http.Path;

public interface UserService {

@GET("user/")

Call<List<User>> getUsers();

@POST("add/")

Call<User> addUser(@Body User user);

@PUT("update/{id}")

Call<User> updateUser(@Path("id") int id, @Body User user);

@DELETE("delete/{id}")

Call<User> deleteUser(@Path("id") int id);

}

### **RetrofitClient**

Create a RetrofitClient class under ***com.jackrutorial.androidretrofit2crud.remote*** package and write the following code in it.

package com.jackrutorial.androidretrofit2crud.remote;

import retrofit2.Retrofit;

import retrofit2.converter.gson.GsonConverterFactory;

public class RetrofitClient {

private static Retrofit retrofit = null;

public static Retrofit getClient(String url){

if(retrofit == null){

retrofit = new Retrofit.Builder().baseUrl(url)

.addConverterFactory(GsonConverterFactory.create())

.build();

}

return retrofit;

}

}

### **APIUtils**

Create a ***ApiUtils*** class under ***com.jackrutorial.androidretrofit2crud.remote*** package and write the following code in it.

package com.jackrutorial.androidretrofit2crud.remote;

public class APIUtils {

private APIUtils(){

};

public static final String API\_URL = "http://169.254.35.189:8080/demo/";

public static UserService getUserService(){

return RetrofitClient.getClient(API\_URL).create(UserService.class);

}

}

## **Adjust activity**

Change the ***activity\_main.xml*** layout to the following.

<?xml version="1.0" encoding="utf-8"?>

<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

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

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context="com.jackrutorial.androidretrofit2crud.MainActivity">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

>

<Button

android:id="@+id/btnAddUser"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Add User"

android:background="#BA55D3"

android:textColor="#ffffff"

android:layout\_marginTop="5dp"

android:layout\_marginBottom="5dp"

/>

<Button

android:id="@+id/btnGetUsersList"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Fetch users"

android:background="#32CD32"

android:textColor="#ffffff"

android:layout\_marginBottom="5dp"

/>

<ListView

android:id="@+id/listView"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

</ListView>

</LinearLayout>

</android.support.constraint.ConstraintLayout>

Change ***MainActivity*** activity code to the following.

package com.jackrutorial.androidretrofit2crud;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.ListView;

import com.jackrutorial.androidretrofit2crud.model.User;

import com.jackrutorial.androidretrofit2crud.remote.APIUtils;

import com.jackrutorial.androidretrofit2crud.remote.UserService;

import java.util.ArrayList;

import java.util.List;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

public class MainActivity extends AppCompatActivity {

Button btnAddUser;

Button btnGetUsersList;

ListView listView;

UserService userService;

List<User> list = new ArrayList<User>();

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

setTitle("Retrofit 2 CRUD Demo");

btnAddUser = (Button) findViewById(R.id.btnAddUser);

btnGetUsersList = (Button) findViewById(R.id.btnGetUsersList);

listView = (ListView) findViewById(R.id.listView);

userService = APIUtils.getUserService();

btnGetUsersList.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

//get users list

getUsersList();

}

});

btnAddUser.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent = new Intent(MainActivity.this, UserActivity.class);

intent.putExtra("user\_name", "");

startActivity(intent);

}

});

}

public void getUsersList(){

Call<List<User>> call = userService.getUsers();

call.enqueue(new Callback<List<User>>() {

@Override

public void onResponse(Call<List<User>> call, Response<List<User>> response) {

if(response.isSuccessful()){

list = response.body();

listView.setAdapter(new UserAdapter(MainActivity.this, R.layout.list\_user, list));

}

}

@Override

public void onFailure(Call<List<User>> call, Throwable t) {

Log.e("ERROR: ", t.getMessage());

}

});

}

}

## **Create User Activity**

* Right Click the app folder choose ***File > New > Activity > Empty Activity***
* Name the new activity "***UserActivity***".
* Click Finish.

Change the ***activity\_user.xml*** layout to the following.

<?xml version="1.0" encoding="utf-8"?>

<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

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

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context="com.jackrutorial.androidretrofit2crud.UserActivity">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

>

<TextView

android:id="@+id/txtUId"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="ID"

android:textSize="20dp"

android:layout\_marginTop="5dp"

/>

<EditText

android:id="@+id/edtUId"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:inputType="number"

/>

<TextView

android:id="@+id/txtUUsername"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="USER NAME"

android:textSize="20dp"

android:layout\_marginTop="5dp"

/>

<EditText

android:id="@+id/edtUsername"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

/>

<Button

android:id="@+id/btnSave"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Save"

android:background="#32CD32"

android:textColor="#ffffff"

android:layout\_marginBottom="5dp"

/>

<Button

android:id="@+id/btnDel"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Delete User"

android:background="#FF0000"

android:textColor="#ffffff"

/>

</LinearLayout>

</android.support.constraint.ConstraintLayout>

Change ***UserActivity*** activity code to the following.

package com.jackrutorial.androidretrofit2crud;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.util.Log;

import android.view.MenuItem;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

import com.jackrutorial.androidretrofit2crud.model.User;

import com.jackrutorial.androidretrofit2crud.remote.APIUtils;

import com.jackrutorial.androidretrofit2crud.remote.UserService;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

public class UserActivity extends AppCompatActivity {

UserService userService;

EditText edtUId;

EditText edtUsername;

Button btnSave;

Button btnDel;

TextView txtUId;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_user);

setTitle("Users");

getSupportActionBar().setDisplayHomeAsUpEnabled(true);

txtUId = (TextView) findViewById(R.id.txtUId);

edtUId = (EditText) findViewById(R.id.edtUId);

edtUsername = (EditText) findViewById(R.id.edtUsername);

btnSave = (Button) findViewById(R.id.btnSave);

btnDel = (Button) findViewById(R.id.btnDel);

userService = APIUtils.getUserService();

Bundle extras = getIntent().getExtras();

final String userId = extras.getString("user\_id");

String userName = extras.getString("user\_name");

edtUId.setText(userId);

edtUsername.setText(userName);

if(userId != null && userId.trim().length() > 0 ){

edtUId.setFocusable(false);

} else {

txtUId.setVisibility(View.INVISIBLE);

edtUId.setVisibility(View.INVISIBLE);

btnDel.setVisibility(View.INVISIBLE);

}

btnSave.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

User u = new User();

u.setName(edtUsername.getText().toString());

if(userId != null && userId.trim().length() > 0){

//update user

updateUser(Integer.parseInt(userId), u);

} else {

//add user

addUser(u);

}

}

});

btnDel.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

deleteUser(Integer.parseInt(userId));

Intent intent = new Intent(UserActivity.this, MainActivity.class);

startActivity(intent);

}

});

}

public void addUser(User u){

Call<User> call = userService.addUser(u);

call.enqueue(new Callback<User>() {

@Override

public void onResponse(Call<User> call, Response<User> response) {

if(response.isSuccessful()){

Toast.makeText(UserActivity.this, "User created successfully!", Toast.LENGTH\_SHORT).show();

}

}

@Override

public void onFailure(Call<User> call, Throwable t) {

Log.e("ERROR: ", t.getMessage());

}

});

}

public void updateUser(int id, User u){

Call<User> call = userService.updateUser(id, u);

call.enqueue(new Callback<User>() {

@Override

public void onResponse(Call<User> call, Response<User> response) {

if(response.isSuccessful()){

Toast.makeText(UserActivity.this, "User updated successfully!", Toast.LENGTH\_SHORT).show();

}

}

@Override

public void onFailure(Call<User> call, Throwable t) {

Log.e("ERROR: ", t.getMessage());

}

});

}

public void deleteUser(int id){

Call<User> call = userService.deleteUser(id);

call.enqueue(new Callback<User>() {

@Override

public void onResponse(Call<User> call, Response<User> response) {

if(response.isSuccessful()){

Toast.makeText(UserActivity.this, "User deleted successfully!", Toast.LENGTH\_SHORT).show();

}

}

@Override

public void onFailure(Call<User> call, Throwable t) {

Log.e("ERROR: ", t.getMessage());

}

});

}

@Override

public boolean onOptionsItemSelected(MenuItem item) {

switch (item.getItemId()) {

case android.R.id.home:

finish();

return true;

}

return super.onOptionsItemSelected(item);

}

}

## **Create User Adapter**

Create a ***UserAdapter*** class under ***com.jackrutorial.androidretrofit2crud*** package and write the following code in it.

package com.jackrutorial.androidretrofit2crud;

import android.content.Context;

import android.content.Intent;

import android.support.annotation.LayoutRes;

import android.support.annotation.NonNull;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.ArrayAdapter;

import android.widget.TextView;

import com.jackrutorial.androidretrofit2crud.model.User;

import java.util.List;

public class UserAdapter extends ArrayAdapter<User> {

private Context context;

private List<User> users;

public UserAdapter(@NonNull Context context, @LayoutRes int resource, @NonNull List<User> objects) {

super(context, resource, objects);

this.context = context;

this.users = objects;

}

@Override

public View getView(final int pos, View convertView, ViewGroup parent){

LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.LAYOUT\_INFLATER\_SERVICE);

View rowView = inflater.inflate(R.layout.list\_user, parent, false);

TextView txtUserId = (TextView) rowView.findViewById(R.id.txtUserId);

TextView txtUsername = (TextView) rowView.findViewById(R.id.txtUsername);

txtUserId.setText(String.format("#ID: %d", users.get(pos).getId()));

txtUsername.setText(String.format("USER NAME: %s", users.get(pos).getName()));

rowView.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

//start Activity User Form

Intent intent = new Intent(context, UserActivity.class);

intent.putExtra("user\_id", String.valueOf(users.get(pos).getId()));

intent.putExtra("user\_name", users.get(pos).getName());

context.startActivity(intent);

}

});

return rowView;

}

}

Create the ***list\_user.xml*** file under res/layout/ folder to the following.

Click on app > res > layout > Right Click on layout. Select ***New > XML > Layout XML File***. In the dialog that appears, enter list\_user as name for the file. Then change the ***list\_user.xml*** layout to the following.

<?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:orientation="vertical">

<TextView

android:id="@+id/txtUserId"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="3dp"/>

<TextView

android:id="@+id/txtUsername"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="3dp"/>

</LinearLayout>