



# PENGEMBANGAN APLIKASI MOBILE

1207050026-DIAN HASNA RAMADHANI

1207050035-FARRAS RANGGA WIDAGDO

1207050148-FANY RISTI FATONAH

1207050037-FATIH FAUZAN KARTAMANAH

1207050042-GADUH HARTAWAN

1207050137-JESSY FAUJIYYAH KHAIRANI





## **Retrofit adalah library Rest Client untuk android dan java dari squareup.**

Retrofit membuat relatif mudah untuk mengambil dan mengunggah JSON (atau struktur data lainnya) melalui webservice berbasis REST. Retrofit akan memberi standar yang waras untuk platform tetapi memungkinkan untuk penyesuaian.

2022  
KEELOMPOK 3



Di Retrofit mengonfigurasi konverter mana yang digunakan untuk serialisasi data. Biasanya untuk JSON menggunakan GSON, tetapi dapat menambahkan konverter khusus untuk memproses XML atau protokol lain.



Retrofit dapat dikonfigurasi untuk menggunakan converter. Converter ini menangani de(serialization) data. Beberapa converter sudah tersedia untuk berbagai format serialisasinya.

Format	Library	Converter
JSON	Gson	com.squareup.retrofit2:converter-gson
JSON	Jackson	com.squareup.retrofit2:converter-jackson
JSON	Moshi	com.squareup.retrofit2:converter-moshi
Protobuf	Protobuf	com.squareup.retrofit2:converter-protobuf
Wire	Wire	com.squareup.retrofit2:converter-wire
XML	Simple XML	com.squareup.retrofit2:converter-simplexml

# Cara Menggunakan Retrofit

1. PERGI KE FILE > NEW PROJECT PILIH SESUAI DEFAULT YANG ADA, LALU PILIH EMPTY ACTIVITY.



2. IMPORT DEPENDENCY UNTUK RETROFIT DAN GSON



```
dependencies {  
    ...  
    // Retrofit Library  
    implementation 'com.squareup.retrofit2:converter-gson:2.9.0'  
    implementation 'com.squareup.retrofit2:retrofit:2.9.0'  
    ...  
}
```



# Cara Menggunakan Retrofit

## 3. TAMBAHKAN INTERNET PERMISSION DI ANDROIDMANIFEST.XML



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

    <!--Internet Permission-->
    <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=".activity.MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>
```



# Cara Menggunakan Retrofit

---

4. KITA AKAN MEMBUAT STRUKTUR MODEL DARI API YANG KITA DAPAT. KALIAN JUGA BISA MENGGUNAKAN API DIBAWAH.

[HTTP://JSONPLACEHOLDER.TYPICODE.COM/POSTS](http://jsonplaceholder.typicode.com/posts)

---

BUATLAH SEBUAH CLASS POSTS.JAVA DIDALAM PACKAGE MODEL SEPERTI CONTOH BERIKUT.

```
package lat.pam.retrofitapp;

public class Posts {
    private String title;
    private String body;

    public Posts (String title, String body){
        this.title = title;
        this.body = body;
    }

    public String getTitle() {
        return title;
    }

    public String getBody() {
        return body;
    }
}
```

# Cara Menggunakan Retrofit

## 5. BUAT SEBUAH RETROFIT INSTANCE



UNTUK MELAKUKAN SEBUAH NETWORK REQUEST KE REST API DENGAN RETROFIT, KITA PERLU MEMBUAT SEBUAH INSTANCE DENGAN RETROFIT. BUILDER CLASS LALU MENGKONFIGURASIKANNYA DENGAN BASE\_URL

BUATLAH SEBUAH CLASS BERNAMA RETROFITCLIENT.JAVA DIDALAM PACKAGE SERVICES

```
package lat.pam.retrofitapp;

import retrofit2.Retrofit;
import retrofit2.converter.gson.GsonConverterFactory;

public class RetrofitClient {

    private static final String BASE_URL = "https://jsonplaceholder.typicode.com/";
    private static Retrofit retrofit = null;

    public static ApiInterface getRetrofitClient(){
        if (retrofit == null){
            retrofit = new Retrofit.Builder()
                .baseUrl(BASE_URL)
                .addConverterFactory(GsonConverterFactory.create())
                .build();
        }
        return retrofit.create(ApiInterface.class);
    }
}
```



# Cara Menggunakan Retrofit

## 6. MENENTUKAN ENDPOINT



SEBUAH ENDPOINT  
DIDEFINISIKAN DIDALAM  
SEBUAH INTERFACE  
MENGUNAKAN ANOTASI  
SPESIAL RETROFIT UNTUK  
MELAKUKAN ENCODE DETAIL  
PARAMETER DAN REQUEST  
METHOD.

BUATLAH SEBUAH INTERFACE  
BERNAMA APIINTERFACE.JAVA

```
package lat.pam.retrofitapp;

import java.util.List;

import retrofit2.Call;
import retrofit2.http.GET;

public interface ApiInterface {

    @GET("/posts")
    Call<List<Posts>> getPosts();
}
```

NOTE : “@GET” MENDEFINISIKAN ALAMAT DARI BASE\_URL MENUJU LOKASI JSON.





# Cara Menggunakan Retrofit

## 7. MEMBUAT SEBUAH ADAPTER UNTUK MEMASUKAN DATA KE DALAM RECYCLEVIEW



```
package lat.pam.retrofitapp;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import java.util.List;

public class PostsAdapter extends
RecyclerView.Adapter<PostsAdapter.ViewHolder> {

    private List<Posts> postsList;

    public PostsAdapter(List<Posts> postsList){
        this.postsList = postsList;
    }

    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull
ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(parent.getContext())
            .inflate(R.layout.item_list, parent, false);
        return new ViewHolder(view);
    }
```

```
@Override
    public void onBindViewHolder(@NonNull ViewHolder holder, int
position) {

        holder.title.setText(postsList.get(position).getTitle());
        holder.body.setText(postsList.get(position).getBody());
    }

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

    public class ViewHolder extends RecyclerView.ViewHolder {
        TextView title;
        TextView body;

        public ViewHolder(@NonNull View itemView){
            super(itemView);

            title = itemView.findViewById(R.id.title);
            body = itemView.findViewById(R.id.body);
        }
    }
}
```

# Cara Menggunakan Retrofit

## 8. MEMASUKAN DATA KEDALAM MAINACTIVITY.JAVA



DIDALAM METHOD ONCREATE() YANG BERADA DI MAINACTIVITY.JAVA, KITA AKAN MENAMPILKAN RECYCLERVIEW, ADAPTER DAN PROGRESSBAR. SERTA AKAN MELAKUKAN FETCHPOST PADA METHOD FETCHPOST YANG DIBUAT DIBAWAHNYA.

```
public class MainActivity extends AppCompatActivity {
```

```
    RecyclerView recyclerView;  
    ProgressBar progressBar;  
    LinearLayoutManager layoutManager;  
    PostsAdapter adapter;  
    List<Posts> postsList = new ArrayList<>();
```

```
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);
```

```
        recyclerView = findViewById(R.id.recyclerView);  
        progressBar = findViewById(R.id.progressBar);  
        layoutManager = new LinearLayoutManager(this);  
        recyclerView.setLayoutManager(layoutManager);  
        adapter = new PostsAdapter(postsList);  
        recyclerView.setAdapter(adapter);
```

```
        fetchPosts();  
    }
```

```
    private void fetchPosts(){  
        progressBar.setVisibility(View.VISIBLE);  
        RetrofitClient.getClient().getPosts().enqueue(new  
        Callback<List<Posts>>() {  
            @Override  
            public void onResponse(Call<List<Posts>> call,  
            Response<List<Posts>> response) {  
                if (response.isSuccessful() && response.body() != null){  
                    postsList.addAll(response.body());  
                    adapter.notifyDataSetChanged();  
                    progressBar.setVisibility(View.GONE);  
                }  
            }  
  
            @Override  
            public void onFailure(Call<List<Posts>> call, Throwable t)  
            {  
                progressBar.setVisibility(View.GONE);  
                Toast.makeText(MainActivity.this, "Error: "+  
                t.getMessage(), Toast.LENGTH_SHORT).show();  
            }  
        });  
    }  
}
```

# Hasil Akhir

## Contoh Data Json

```
{
  "userId": 1,
  "id": 1,
  "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
  "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto"
},
{
  "userId": 1,
  "id": 2,
  "title": "qui est esse",
  "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque nisi nulla"
},
```

### Retrofit App

**sunt aut facere repellat provident occaecati excepturi optio reprehenderit**

quia et suscipit  
suscipit recusandae consequuntur expedita et cum  
reprehenderit molestiae ut ut quas totam  
nostrum rerum est autem sunt rem eveniet architecto

**qui est esse**

est rerum tempore vitae  
sequi sint nihil reprehenderit dolor beatae ea dolores neque  
fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis  
qui aperiam non debitis possimus qui neque nisi nulla

**ea molestias quasi exercitationem repellat qui ipsa sit aut**

et iusto sed quo iure  
voluptatem occaecati omnis eligendi aut ad  
voluptatem doloribus vel accusantium quis pariatur  
molestiae porro eius odio et labore et velit aut

**eum et est occaecati**

ullam et saepe reiciendis voluptatem adipisci  
sit amet autem assumenda provident rerum culpa  
quis hic commodi nesciunt rem tenetur doloremque ipsam iure  
quis sunt voluptatem rerum illo velit

**nesciunt quas odio**

repudiandae veniam quaerat sunt sed  
alias aut fugiat sit autem sed est  
voluptatem omnis possimus esse voluptatibus quis

---

# Referensi

- <http://sistem-informasi-s1.stekom.ac.id/informasi/baca/Penggunaan-Retrofit-sebagai-Konektor-Aplikasi-Android-Studio-dengan-Database/af1335dbf4b385abbc89fa77b818700ccccaeece9>
- <https://ibnuard.medium.com/tutorial-simpel-menggunakan-retrofit-android-studio-retrofit-basic-34648061825a>
- <https://polinema-mobile.github.io/jobsheet/10/00.html>

