
## The Simplest Way to Get WordPress Post in Android App

Posted on September 14, 2017 by kamal bunkar

**Deprecated**: Function get_magic_quotes_gpc() is deprecated in **/home2/bustebp2/public_html/blueappsoftware.in/android/wp-includes/formatting.php** on line **4382**

Post Views: 25,900

From last few days, people asked me- How to Get WordPress Post in Android App? WordPress is a most popular platform to host a blog. It has monthly 409+ million *People* view more than 19.6 billion pages hosted by WordPress platform. And this count is kept increasing day by day.

So In this blog, I will show you step by step process to get WordPress post in Android App. I will take help of two plugins that make all the content and resources on my WordPress blog accessible. WP REST API is an amazing feature on WordPress. It allows developers to create apps that can easily integrate with WordPress platform.

# WordPress REST API for Android

Plugins–> Click ADD NEW –> type keyword WordPress rest. It is the official WP REST API plugin that will be used to fetch data from the blog.

WordPress REST API (Version 2)

Second, You have to install Rest API filter. Using this filter you can short your result like of you want only post id and title, so you can pass filter parameter as id, title.

WordPress REST API – Filter Fields

**Fetching All Post of Your Website**

- You can fetch information about all the post on your blog by the following URL. It will return a JSON response that contains all the information about your blog.

> ***http://your-blog-url/wp-json/wp/v2/posts***
>
> ***for example http://www.blueappsoftware.in/android/wp-json/wp/v2/posts***

**Fetching Specified Number of Post**

- For fetching a specified number of posts you can use per-page. The below URL will fetch only 3 posts.

> **http://your-blog-url/wp-json/wp/v2/posts?per_page=3**
>
> **for example http://www.blueappsoftware.in/android/wp-json/wp/v2/posts?per_page=1**

**Fetching Particular Post**

- You can fetch information about a particular post by using post ID.

> ***http://your-blog-url/wp-json/wp/v2/posts/1179***
>
> ***for example http://www.blueappsoftware.in/android/wp-json/wp/v2/posts/1179***

**Use Filtering Fields**

of the android app. As you have seen in above JSON data that there are several fields that we don't require. So with the help of REST API – Filter Fields plugin you can filter few fields. For example, you want to fetch only post's id and title then it can be done by using the following URL.

> ***http://your-blog-url/wp-json/wp/v2/posts?fields=id,title***
>
> ***for example http://www.blueappsoftware.in/android/wp-json/wp/v2/posts/1179?fields=id,title,date***

To know more about WordPress Rest API and Filter, Please check here

# Watch Video Get WordPress post in Android App


# Login & Download source code

Name

Email

DOWNLOAD CODE




# Install WordPress Plugins

You have Installed both the WordPress REST API and WordPress REST filter API on you WordPress blog. Please remember to activate it.

# Add Dependency on Gradle file

Create a new project on android studio. We have to app Internet Permission on AndroidManifest File. Also, add the dependency for Volley or Retrofit lib. If you don't know how to use Retrofit to get data from web server please check how to get JSON data using Retrofit

### Add Gradle Dependency

```
compile 'com.android.support:cardview-v7:25.3.1'
```

```
    // retrofit
    compile 'com.google.code.gson:gson:2.6.2'
    compile 'com.squareup.retrofit2:retrofit:2.0.2'
    compile 'com.squareup.retrofit2:converter-gson:2.0.2'
    // Glide
    compile 'com.github.bumptech.glide:glide:3.7.0'
```

# Add RecyclerView and CardView Layout

Now add RecyclerView on activity-main.xml and create an adapter for RecyclerView. Here you can see my blog post about RecyclerView. Create another layout name as postdetails.xml for adapter which will display Post Image, post title and post short description inside a cardview.

### activity-main.xml

```
<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.and
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true"
    tools:context=".MainActivity">


    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <android.support.v7.widget.RecyclerView

            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:id="@+id/recycler_view">

        </android.support.v7.widget.RecyclerView>

        <ProgressBar
```

```
            android:id="@+id/progressbar"
            android:layout_alignParentBottom="true"
            android:layout_centerHorizontal="true" />

    </RelativeLayout>



</android.support.design.widget.CoordinatorLayout>
```

## PostDetails.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.CardView xmlns:android="http://schemas.android.com/apk/
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    app:cardElevation="5dp">

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

        <ImageView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:src="@mipmap/ic_launcher"
            android:id="@+id/Icon"
            android:layout_margin="5dp"
            android:layout_weight="9"
            android:layout_gravity="center_vertical"/>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:gravity="center_vertical"
            android:paddingTop="5dp"
            android:layout_weight="4">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textStyle="bold"
                android:gravity="left"
```

```
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:gravity="left"
                android:id="@+id/subtitle"
                android:padding="5dp"
                android:layout_marginBottom="5dp"
                android:maxLines="5"
                android:lineSpacingExtra="2dp"/>

        </LinearLayout>

    </LinearLayout>
</android.support.v7.widget.CardView>
```

# Make Retrofit Request to get JSON

Create POJO model for Retrofit and a Retrofit API method to send a request to URL. To create POJO model you can use this online tool that auto create POJO model from JSON string. You can use this JSON Editor to see you JSON string in proper structure. Just past your complete JSON string in this editor and click on right arrow. **Get all java file and layout by downloading the project for free.** I added onclick listener on cardview, so when you will click on any post, I will open new activity with webview. The webview will display your complete post in mobile view, Because my WordPress blog has mobile view campatibility.

MainActivity.java

```
package blueappsoftware.wordpressinandroid;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.util.Log;
```

```
import java.util.ArrayList;
import java.util.List;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import retrofit2.Retrofit;
import retrofit2.converter.gson.GsonConverterFactory;

public class MainActivity extends AppCompatActivity {

    private RecyclerView recyclerView;
    private ProgressBar progressBar;
    private LinearLayoutManager mLayoutManager;
    private ArrayList<Model> list;
    private RecyclerViewAdapter adapter;

    private String baseURL = "http://www.blueappsoftware.in";

    public static List<WPPost> mListPost;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        recyclerView = (RecyclerView) findViewById(R.id.recycler_view);

        progressBar = (ProgressBar) findViewById(R.id.progressbar);

        mLayoutManager = new LinearLayoutManager(MainActivity.this, LinearLayoutMa
        recyclerView.setLayoutManager(mLayoutManager);

        list = new ArrayList<Model>();
        /// call retrofill
        getRetrofit();

        adapter = new RecyclerViewAdapter( list, MainActivity.this);

        recyclerView.setAdapter(adapter);

    }
    public void getRetrofit(){

        Retrofit retrofit = new Retrofit.Builder()
                .baseUrl(baseURL)
                .addConverterFactory(GsonConverterFactory.create())
                .build();

        RetrofitArrayApi service = retrofit.create(RetrofitArrayApi.class);
       Call<List<WPPost>> call = service.getPostInfo();
```

```
        // to make call to dynamic URL

        // String yourURL = yourURL.replace(BaseURL,"");
        // Call<List<WPPost>>  call = service.getPostInfo( yourURL);

        /// to get only 6 post from your blog
        // http://your-blog-url/wp-json/wp/v2/posts?per_page=2

        // to get any specific blog post, use id of post
        //  http://www.blueappsoftware.in/wp-json/wp/v2/posts/1179

        // to get only title and id of specific
        // http://www.blueappsoftware.in/android/wp-json/wp/v2/posts/1179?fields=i



        call.enqueue(new Callback<List<WPPost>>() {
            @Override
            public void onResponse(Call<List<WPPost>> call, Response<List<WPPost>>
                Log.e("mainactivyt", " response "+ response.body());
                mListPost = response.body();
                progressBar.setVisibility(View.GONE);
                for (int i=0; i<response.body().size();i++){
                    Log.e("main ", " title "+ response.body().get(i).getTitle().ge
                            response.body().get(i).getId());

                    String tempdetails =  response.body().get(i).getExcerpt().getR
                    tempdetails = tempdetails.replace("<p>","");
                    tempdetails = tempdetails.replace("</p>","");
                    tempdetails = tempdetails.replace("[&hellip;]","");

                    list.add( new Model( Model.IMAGE_TYPE,  response.body().get(i)
                            tempdetails,
                            response.body().get(i).getLinks().getWpFeaturedmedia()

                }
                adapter.notifyDataSetChanged();

                }

            @Override
            public void onFailure(Call<List<WPPost>> call, Throwable t) {

            }
        });

    }
    public static List<WPPost> getList(){
        return  mListPost;
    }
}
```

### Retrofit API call

```
package blueappsoftware.wordpressinandroid;

import java.util.List;

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

/**
 * Created by Jaink on 14-09-2017.
 */

public interface RetrofitArrayApi {

    @GET("android/wp-json/wp/v2/posts")
    Call<List<WPPost>> getPostInfo();
    /// to make call to dynamic URL
    //  @GET
    //  Call<List<WPPost>> getPostInfo(@Url String url);
    //

}
```

# Run App on device

Now Run this code on the real device. You will see all your blog post title and description is in cardview. You can click on any one of the post to see complete post.