



Virtual Internship Experience

API (Application Programing Interface)

- Menggunakan API yang tersedia

1. Menggunakan API yang tersedia

Sebuah API adalah sekumpulan definisi dan protokol untuk membangun dan mengintegrasikan perangkat lunak aplikasi. API adalah singkatan dari antarmuka pemrograman aplikasi (Application Programming Interface). API memungkinkan produk atau layanan Anda berkomunikasi dengan produk dan layanan lain tanpa harus tahu bagaimana penerapannya.

Cara ini dapat menyederhanakan pengembangan aplikasi, menghemat waktu dan uang. Saat Anda merancang alat dan produk baru atau mengelola yang sudah ada, API memberi Anda fleksibilitas dalam menyederhanakan desain, administrasi, penggunaan, serta memberikan peluang untuk inovasi. API juga kadang-kadang dianggap sebagai kontrak, dengan dokumentasi yang mewakili perjanjian antara para pihak.

Contohnya jika pihak 1 mengirim permintaan jarak jauh yang disusun dengan cara tertentu, maka pihak 1 ingin mengetahui bagaimana perangkat lunak pihak 2 akan merespons. Karena API menyederhanakan cara pengembang mengintegrasikan komponen aplikasi baru ke dalam arsitektur yang ada, API mampu membantu bisnis dan tim IT untuk melakukan kolaborasi. Untuk lebih lanjutnya, mari mengenal API dan semua kelebihanannya.

Contoh Studi kasus dalam mengambil data melalui URL dengan konsep API

Membuat Proyek Android Studio

Buka Android Studio -> Buat Project baru -> Give itu nama

apa pun -> Tick Kotlin sebagai Language.-> Select Empty Activity.

Setelah proyek Anda disinkronkan dengan benar.

Kami akan mulai dengan menambahkan sejumlah tanggungan yang kami perlukan untuk proyek kami.

Buka skrip gradle (Module:app)

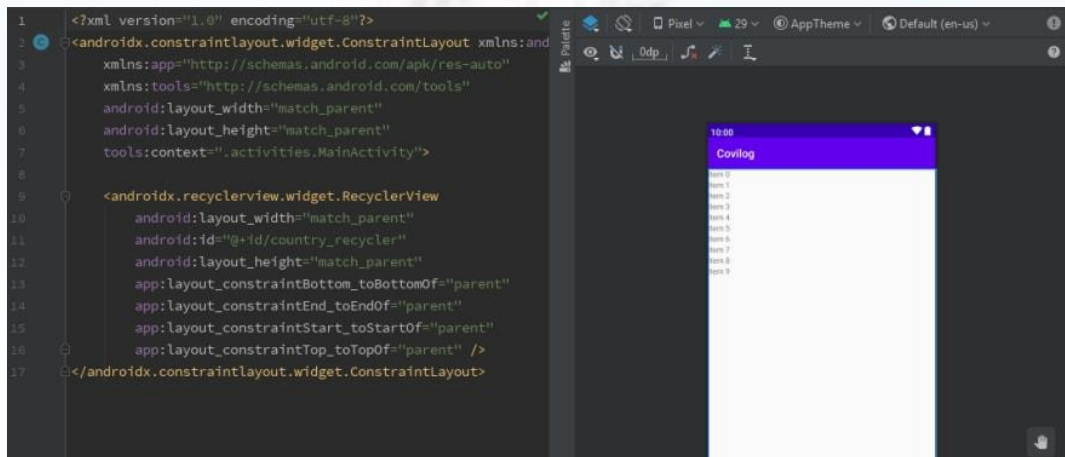
tambahkan yang berikut

```
//for material themes
implementation 'com.google.android.material:material:1.1.0'
implementation 'com.squareup.picasso:picasso:2.71828'
//retrofit requisites retro+gson
implementation 'com.squareup.retrofit2:retrofit:2.5.0'
implementation 'com.squareup.retrofit2:converter-gson:2.5.0'
```

Kami akan menuju ke activity_main.xml kami dan menambahkan tampilan pendaur ulang ke dalamnya. Ini akan digunakan untuk menampilkan daftar negara. Seharusnya seperti ini:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".activities.MainActivity">
```

```
    <androidx.recyclerview.widget.RecyclerView
        android:layout_width="match_parent"
        android:id="@+id/country_recycler"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```



Gambar 1. Tampilan Menu

Karena kami akan menggunakan komponen material (Untuk membuat aplikasi kami terlihat lebih baik), Anda harus mengubah tema Anda untuk mewarisi dari komponen material. Jadi edit gaya Anda yang ada.xml (dalam gaya/styles.xml) sebagai berikut:

```
<resources>
  <!-- Base application theme. -->
  <style name="AppTheme"
    parent="Theme.MaterialComponents.Light.DarkActionBar.Bridge">
    <!-- Customize your theme here. -->
    <item name="colorPrimary">@color/colorPrimary</item>
    <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
    <item name="colorAccent">@color/colorAccent</item>
  </style>
</resources>
```

Perhatikan bahwa tema induk sekarang adalah Theme.MaterialComponents.Light.DarkActionBar.Bridge Anda dapat memilih salah satu, tetapi harus memiliki komponen Material.

Selanjutnya kita perlu mengizinkan aplikasi kita mengakses Sumber Daya dari Internet, jadi kita akan menambahkan permission ke Access Internet.

Buka manifest.xml file Anda dan tambahkan yang berikut di bagian atas tag aplikasi.

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

Setelah semua diatur,
kami sekarang siap untuk mulai menerapkan fungsionalitas.

Ruang lingkup kami dalam tutorial ini adalah mengambil data dari API yang memiliki beberapa objek data,

respons json seperti ini:

```
{  
  "updated":1592127440420,  
  "country":"Afghanistan",  
  "countryInfo"  
  {  
    "_id":4,  
    "iso2":"AF",  
    "iso3":"AFG",  
    "lat":33,"long":65,  
    "flag":"https://disease.sh/assets/img/flags/af.png"  
  },  
  "cases":24766,  
  "todayCases":664,  
  "deaths":471,  
  "todayDeaths":20,  
  "recovered":4725,  
  "todayRecovered":524,
```



```
"active":19570,  
"critical":19,  
}
```

Respons di atas memiliki dua objek json yaitu :Info Negara dan Detail Halus Negara Jika Anda telah bekerja dengan permintaan dan tanggapan HTTP, Anda akan membuktikan bagaimana ini biasanya merupakan langkah sibuk untuk memetakan data Anda. Kami akan menggunakan Retrofit yang sebelumnya diimpor.

Daftar Pustaka

<https://dev.to/bensalcie/android-kotlin-get-data-from-restful-api-having-multiple-json-objects-o5a>

<https://idcloudhost.com/mengenal-apa-itu-api-application-programming-interface-fungsi-dan-cara-kerjanya/>