

Menggunakan Firebase di Aplikasi Android dengan Firebase Assistant

Firebase adalah sebuah kumpulan library dari Google yang sangat berguna untuk pengembangan aplikasi Android. Ada banyak kegunaan dari Firebase, di antaranya adalah untuk menambah user base, develop high quality apps, analytics, bahkan untuk mendapatkan uang dari aplikasi yg kita buat. Dalam bab ini kita akan menggunakan Firebase Assistant untuk menambahkan Firebase ke aplikasi Android kita, sehingga prosesnya akan lebih mudah karena dibantu dengan GUI.

Pre-requisites

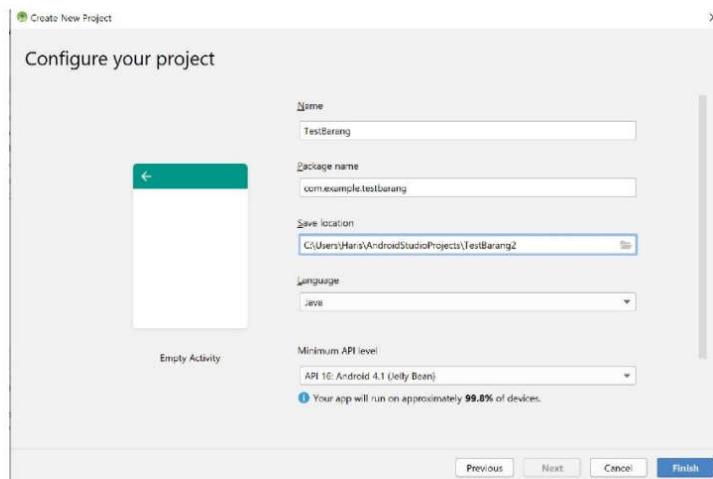
- Untuk bisa menggunakan Firebase maka harus memiliki akun [google](#) dan memiliki koneksi Internet

Menambahkan Firebase lewat Firebase Assistant

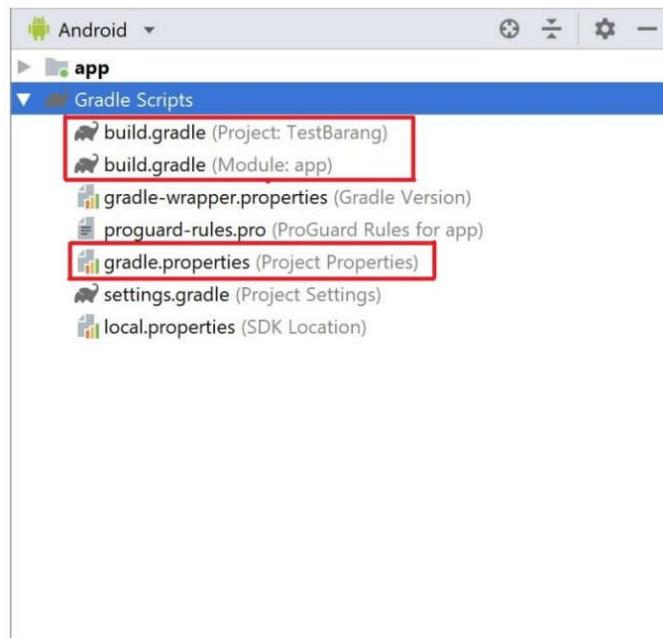
Pada dasarnya ada 3 langkah yang harus dilakukan jika ingin membuat aplikasi yang menggunakan Firebase untuk menyimpan data, yaitu :

- Siapkan Project baru atau Buka project yg sudah pernah dibuat. Kali ini kita mulai dengan membuat project baru.
- Lengkapi library-library dalam Gradle Script. Ada 3 file utama gradle yang akan dilengkapi, yaitu `build.gradle (project)`, `build.gradle (modul:app)`, dan `gradle.properties (project properties)`. Library yang harus ditambahkan ada 2 jenis, jenis yang pertama berkaitan dengan pembuatan layout aplikasi dan jenis kedua adalah untuk dapat berkoneksi dengan firebase.
- Koneksi dengan firebase dengan bantuan Firebase Assistant.

Langkah 1. Membuat Project baru dengan nama ‘TestBarang’



Langkah 2, Kelola Gradle Script



- Buka file **gradle.properties (project properties)** tambahkan dua baris berikut, gunanya adalah agar kita bisa menggunakan ‘`androidx`’.

```
android.enableJetifier=true  
android.useAndroidX=true
```

- Buka **build.gradle (project)**, tambahkan classpath yang belum ada berikut kedalam dependencies dan maven kedalam allprojects. Fungsinya adalah agar bisa berkomunikasi dengan firebase.

```
dependencies {  
    classpath 'com.android.tools.build:gradle:3.3.2'  
    classpath 'com.google.gms:google-services:4.3.3'  
    classpath 'com.google.firebaseio.firebaseio-plugins:2.0.0'  
  
    // NOTE: Do not place your application dependencies here; they belong  
    // in the individual module build.gradle files  
}  
  
allprojects {  
    repositories {  
        google()  
        jcenter()  
        maven { url "https://maven.google.com" }  
    }
```

- Buka ***build.gradle (Module:app)***, tambahkan 3 hal, yaitu apply plugin, library androidx yang akan digunakan dalam membuat antarmuka aplikasi, dan library untuk database firebase.

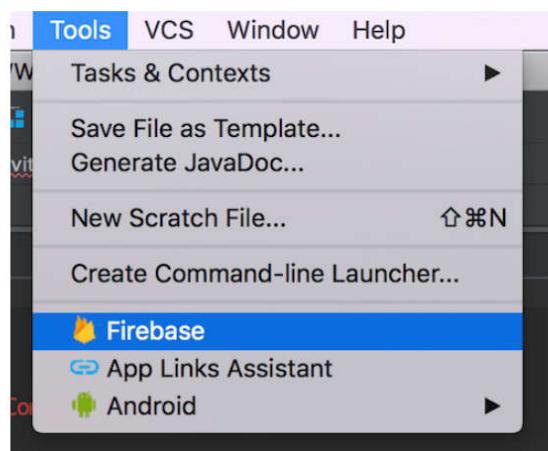
```
apply plugin: 'com.google.gms.google-services'

dependencies {
    implementation 'com.google.firebase:firebase-core:17.4.0'
    implementation 'com.google.firebase:firebase-database:19.3.0'

    implementation 'androidx.multidex:multidex:2.0.1'
    implementation 'androidx.appcompat:appcompat:1.1.0'
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
}
```

- Jika sudah semua, kemudian klik Sync Now untuk mensikronisasi gradle di Android Studio.

Langkah 3, Koneksi firebase dengan menggunakan Firebase Assistant. Pada menu atas Android Studio, klik **Tools > Firebase**.



Firebase Assistant

Jika sudah, maka akan muncul kolom Firebase Assistant baru pada sebelah kanan layar. Seperti gambar di bawah ini.

The screenshot shows the Firebase console interface. At the top, there's a navigation bar with tabs for 'Assistant' and 'Firebase'. Below the navigation bar is the Firebase logo. A descriptive text block says: 'Firebase gives you the tools and infrastructure from Google to help you develop, grow and earn money from your app. [Learn more](#)'. A vertical sidebar on the left lists several services with icons: Analytics (selected), Cloud Messaging, Authentication, Realtime Database, and Storage. Each service has a brief description and a 'More info' link.

- ▶ **Analytics**
Measure user activity and engagement with free, easy, and unlimited analytics. [More info](#)
- ▶ **Cloud Messaging**
Deliver and receive messages and notifications reliably across cloud and device. [More info](#)
- ▶ **Authentication**
Sign in and manage users with ease, accepting emails, Google Sign-In, Facebook and other login providers. [More info](#)
- ▶ **Realtime Database**
Store and sync data in realtime across all connected clients. [More info](#)
- ▶ **Storage**
Store and retrieve large files like images, audio, and video without writing server-side code. [More info](#)

Menu utama Firebase

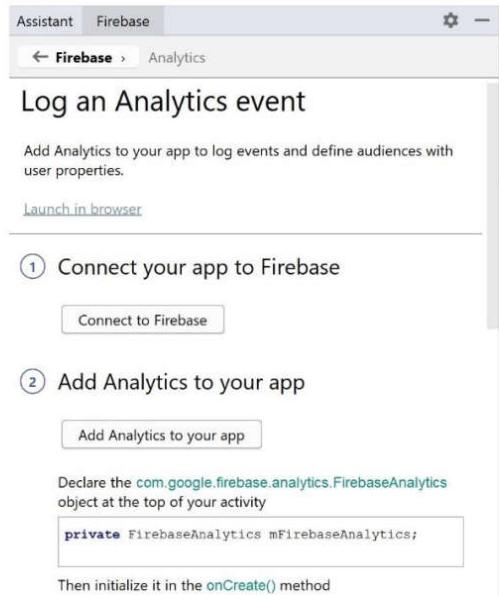
Di sana kalian bisa memilih layanan-layanan dari Firebase yang ingin digunakan. Untuk keperluan tutorial ini kita coba pilih **Analytics**.

This screenshot shows the 'Analytics' service page within the Firebase console. The top navigation bar and Firebase logo are visible. The main content area displays the same introductory text as the main menu. Under the 'Analytics' heading, there is a sub-section titled 'Log an Analytics event' with a circular arrow icon and a link. The other service items (Cloud Messaging, Authentication, Realtime Database, Storage) are listed below in their respective sections.

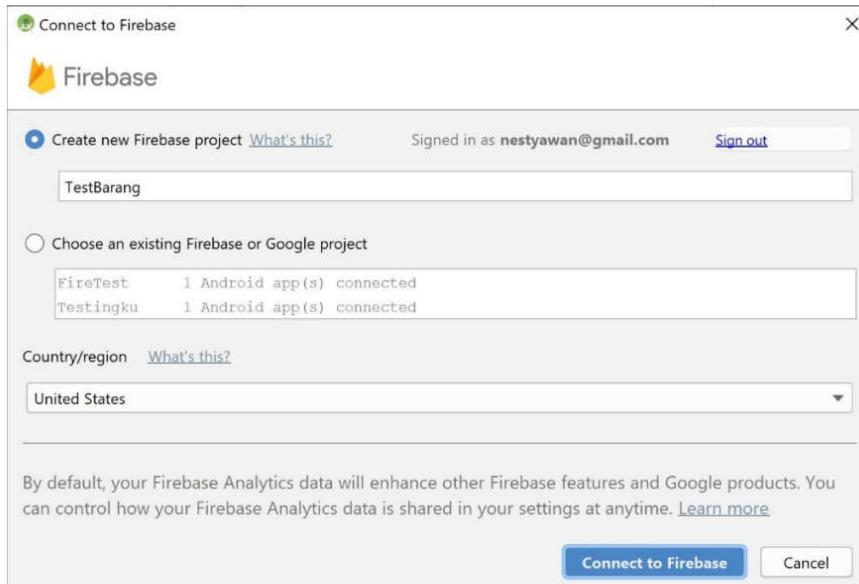
- ▼ **Analytics**
Measure user activity and engagement with free, easy, and unlimited analytics. [More info](#)
 - ▶ [Log an Analytics event](#)
- ▶ **Cloud Messaging**
Deliver and receive messages and notifications reliably across cloud and device. [More info](#)
- ▶ **Authentication**
Sign in and manage users with ease, accepting emails, Google Sign-In, Facebook and other login providers. [More info](#)
- ▶ **Realtime Database**
Store and sync data in realtime across all connected clients. [More info](#)
- ▶ **Storage**

Firebase Analytics

Kemudian akan muncul tampilan seperti di atas. Klik Log An Analytic Even, akan muncul tampilan berikut.



Pilih untuk **Connect to Firebase** terlebih dahulu. Kalian akan diarahkan untuk membuka browser dan login ke akun Google kalian. Jika kalian sudah login dan mengauthentifikasi Firebase, buka kembali aplikasi Android Studio nya. Maka akan muncul dialog untuk connect ke Firebase dan membuat project Firebase baru seperti gambar di bawah.



Firebase Create New Project

Masukkan nama project Firebase baru kalian, kemudian klik **Connect to Firebase**. Kemudian, kalian akan menunggu beberapa saat. Jika sudah selesai maka akan muncul notifikasi bahwa project Android kalian telah terkoneksi ke Firebase seperti gambar berikut.

Firebase > Analytics

Log an Analytics event

Add Analytics to your app to log events and define audiences with user properties.

[Launch in browser](#)

① Connect your app to Firebase

Connected

② Add Analytics to your app

Add Analytics to your app

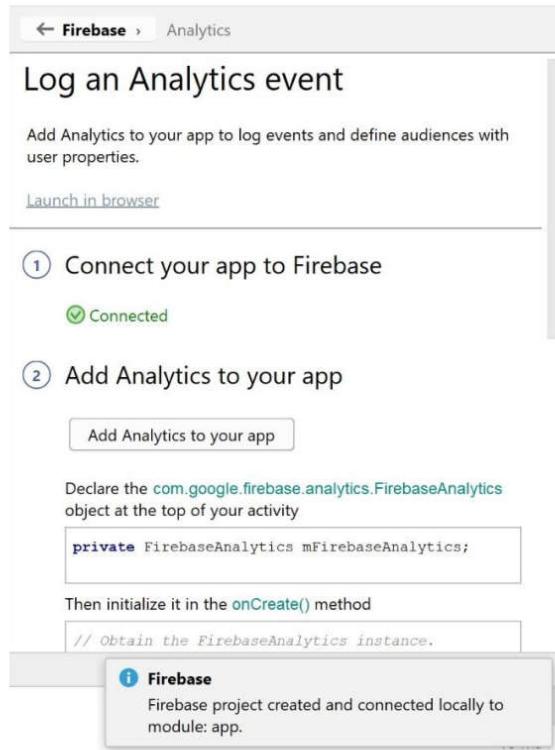
Declare the `com.google.firebaseio.analytics.FirebaseAnalytics` object at the top of your activity

```
private FirebaseAnalytics mFirebaseAnalytics;
```

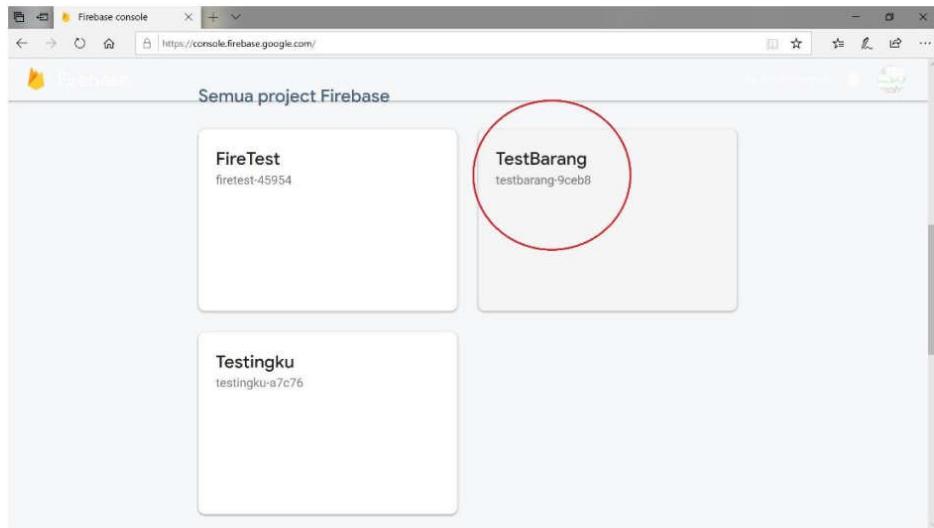
Then initialize it in the `onCreate()` method

```
// Obtain the FirebaseAnalytics instance.
```

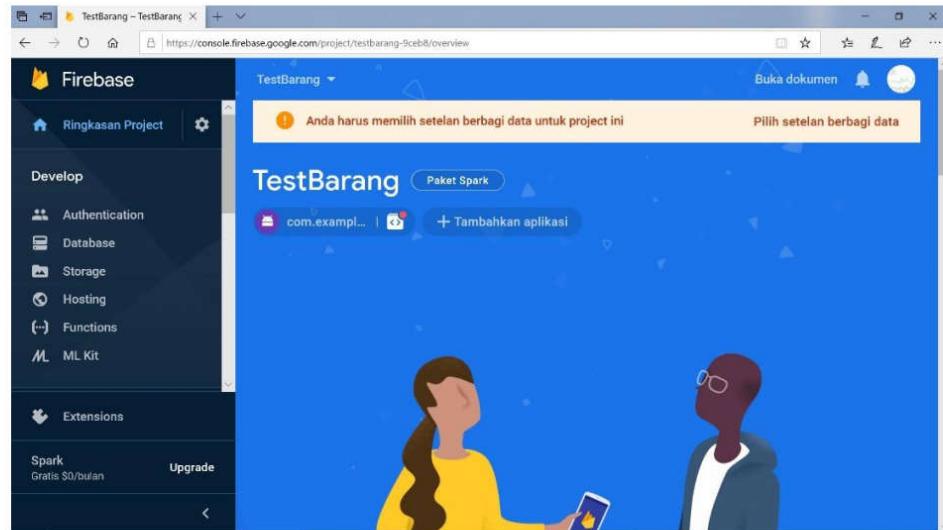
i Firebase
Firebase project created and connected locally to module: app.



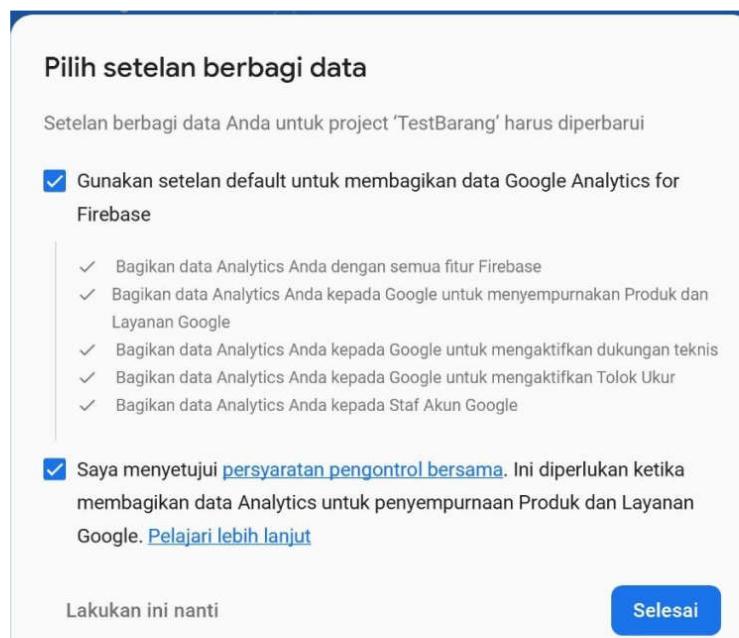
Jika kita check di Console Firebase maka akan kelihatan bahwa project yang dibuat telah ada di Firebase.



Jika di klik pada project TestBarang maka akan masuk ke jendela TestBarang. Disini kita bisa setting segala keperluan kita.

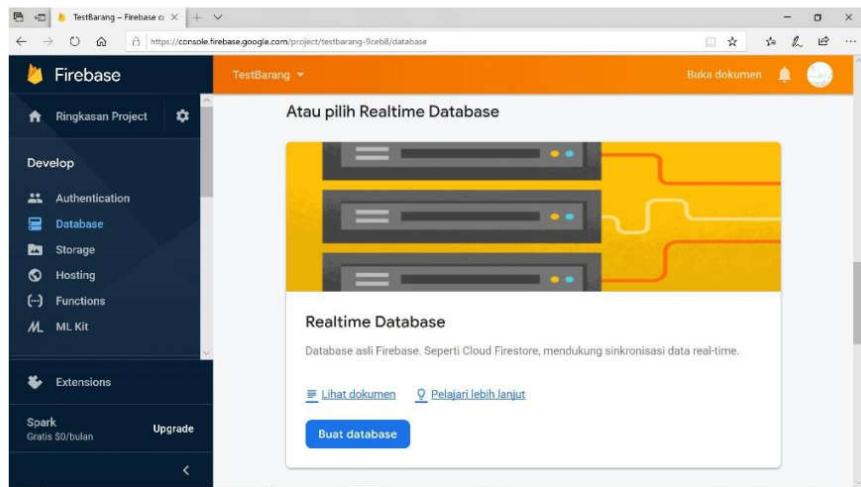


Pilih Setelan Berbagi Data, lalu klik persyaratan yang ada dan tekan selesai.

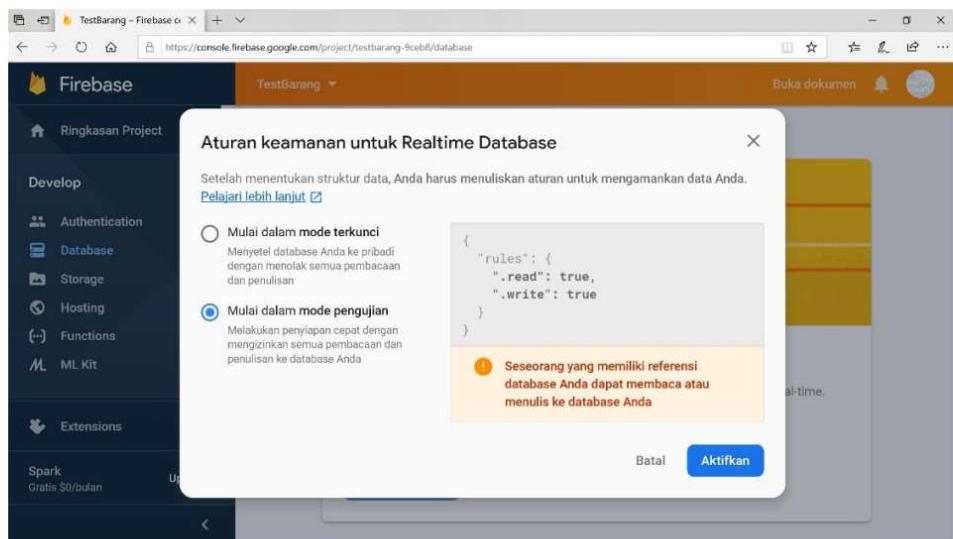


Perlu diketahui, proses ini hanya untuk mendapatkan konfigurasi `google-services.json`, meng-create Firebase Project baru pada Firebase Dashboard dan menyambungkan project Android kalian ke Dashboard Firebase.

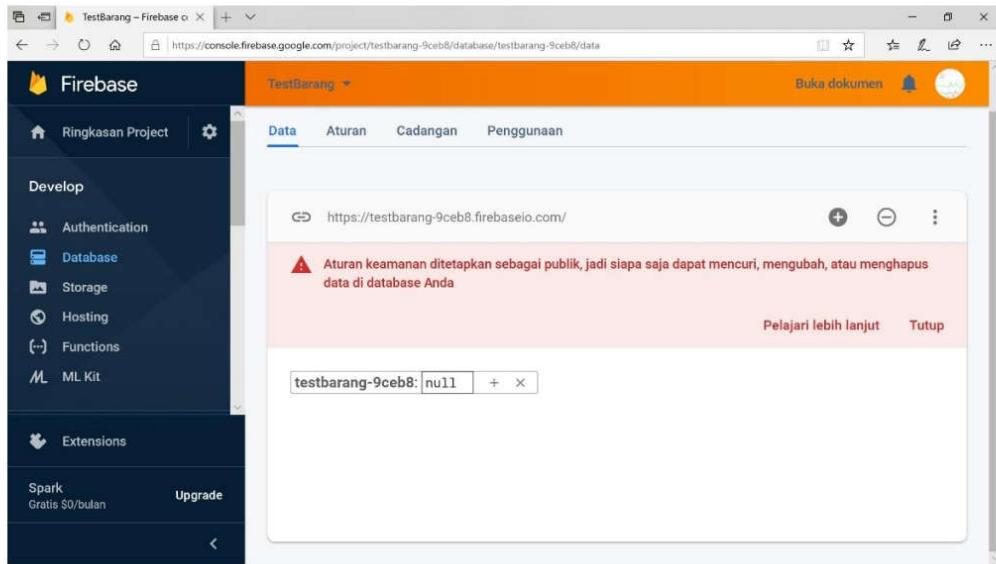
Setelah itu klik Database, pilihlah di Realtime Database seperti dalam gambar berikut ini.



Kemudian klik ‘Buat database’ hingga muncul tampilan aturan keamanan berikut. Pilihlah ‘Mulai dalam mode pengujian’, karena aplikasi ini hanya untuk Latihan terlebih dahulu. Pengaturan keamanan lebih lanjut bisa kalian pelajari.



Setelah klik ‘Aktifkan’ , kita akan dibawa masuk ke project kita dan siap untuk diuji database nya.



Now what ?

Kita Kembali lagi ke Android Studio untuk meneruskan pembuatan aplikasi.

1. Mulai dari menyiapkan string dalam `string.xml`

```
<resources>
    <string name="namaApp">Testingku</string>
    <string name="btTambah">Tambah Data</string>
    <string name="btLihat">Lihat Data</string>
    <string name="bOk">Submit Data</string>
    <string name="eno">Kode Barang</string>
    <string name="enama">Nama Barang</string>
    <string name="txtJudul1">ACTIVITAS</string>
    <string name="txtJudul2">Penambahan Data</string>
    <string name="txtJudul3">Data Barang</string>
    <string name="txtKosong"></string>
</resources>
```

2. Buat tampilan utama `activity_main.xml`

```
<?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=".MainActivity" >

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
```

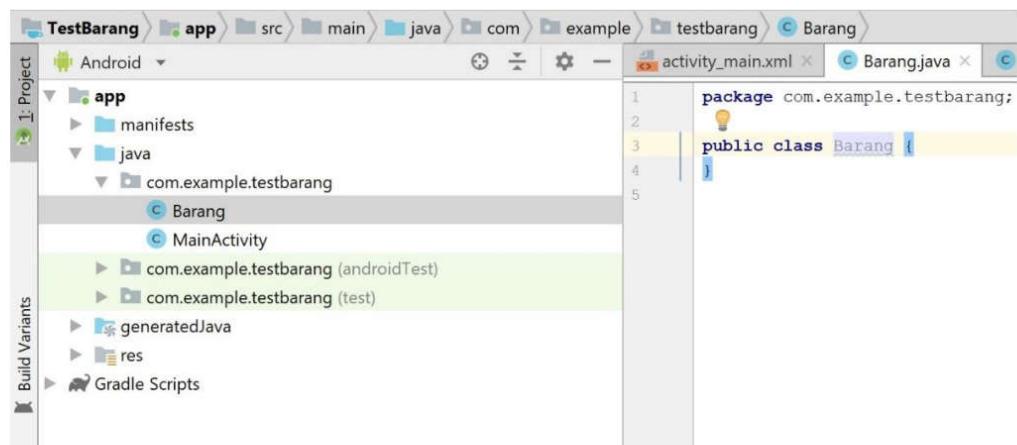
```
        android:layout_height="42dp"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginTop="10dp"
        android:layout_marginEnd="8dp"
        android:layout_marginRight="8dp"
        android:text="@string/txtJudul1"
        android:textSize="24dp"
        app:layout_constraintBottom_toTopOf="@+id	btnTambah"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.423"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id	btnTambah"
        android:layout_width="238dp"
        android:layout_height="54dp"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginBottom="8dp"
        android:text="@string/btTambah"
        app:layout_constraintBottom_toTopOf="@+id	btnLihat"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.414"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView"
        app:layout_constraintVertical_bias="0.252" />

    <Button
        android:id="@+id	btnLihat"
        android:layout_width="234dp"
        android:layout_height="59dp"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginBottom="400dp"
        android:text="@string/btLihat"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.428"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id	btnTambah"
        app:layout_constraintVertical_bias="0.0" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

3. Buat klas baru untuk model databasenya, **Barang.java**



```
package com.example.testbarang;

import java.io.Serializable;

public class Barang implements Serializable {
    private String kode;
    private String nama;

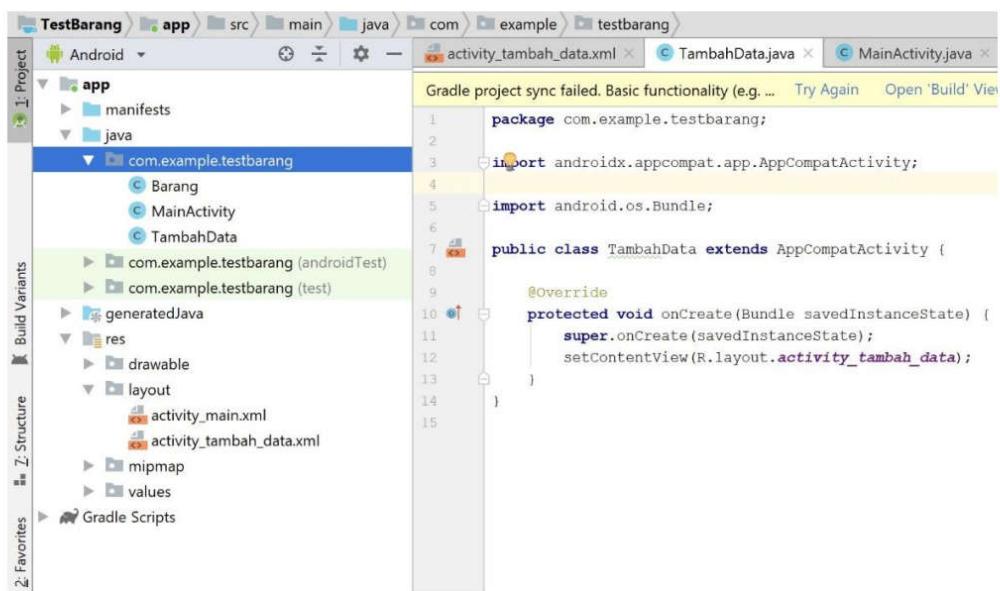
    public Barang(){
    }

    public String getKode() {
        return kode;
    }
    public void setKode(String kd) {
        this.kode = kd;
    }

    public String getNama() {
        return nama;
    }
    public void setNama(String nama) {
        this.nama = nama;
    }

    @Override
    public String toString() {
        return " "+kode+"\n" +
            " "+nama;
    }
    public Barang( String kd, String nm){
        kode = kd;
        nama = nm;
    }
}
```

4. Buat activity baru dengan nama TambahData



Lalu edit layout `activity_tambah_data.xml` berikut.

```
<?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=".TambahData">
    <EditText
        android:id="@+id/editNo"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginBottom="8dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:hint="@string/eno"
        app:layout_constraintBottom_toTopOf="@+id/editNama"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView2"
        app:layout_constraintVertical_bias="0.255" />
```

```
<EditText
    android:id="@+id/editNama"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:layout_marginBottom="200dp"
    android:ems="10"
    android:hint="@string/enama"
    android:inputType="textPersonName"
    app:layout_constraintBottom_toTopOf="@+id	btnOk"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/editNo"
    app:layout_constraintVertical_bias="0.089" />

<Button
    android:id="@+id	btnOk"
    android:layout_width="125dp"
    android:layout_height="72dp"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:layout_marginBottom="100dp"
    android:text="@string/bOk"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/editNama"
    app:layout_constraintVertical_bias="0.557" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:layout_marginBottom="8dp"
    android:text="@string/txtKosong"
    app:layout_constraintBottom_toTopOf="@+id/editNo"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

5. Edit `MainActivity.java` menjadi seperti berikut :

```
package com.example.testbarang;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    private Button bTambah;
    private Button bLihat;

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

        bTambah = (Button) findViewById(R.id.btnAdd);
        bLihat = (Button) findViewById(R.id.btnLihat);

        bTambah.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(TambahData.getActIntent(MainActivity.this));
            }
        });

        bLihat.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                //besok dilanjut
            }
        });
    }
}
```

6. Tambahkan kode berikut kedalam `TambahData.java`

```
package com.example.testbarang;

import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.inputmethod.InputMethodManager;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.android.gms.tasks.OnSuccessListener;
```

```
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;

import androidx.appcompat.app.AppCompatActivity;

public class TambahData extends AppCompatActivity {
    private DatabaseReference database;

    // variable fields EditText dan Button
    private Button btSubmit;
    private EditText etKode;
    private EditText etNama;

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

        etKode = (EditText) findViewById(R.id.editNo);
        etNama = (EditText) findViewById(R.id.editNama);
        btSubmit = (Button) findViewById(R.id.btnOk);

        // mengambil referensi ke Firebase Database
        database = FirebaseDatabase.getInstance().getReference();

        btSubmit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (!(etKode.getText().toString().isEmpty()) &&
                !(etNama.getText().toString().isEmpty()))
                    submitBrg(new Barang(etKode.getText().toString(),
etNama.getText().toString()));
                else
                    Toast.makeText(getApplicationContext(), "Data tidak boleh
kosong", Toast.LENGTH_LONG).show();
            }
        });

        InputMethodManager imm = (InputMethodManager)
            getSystemService(Context.INPUT_METHOD_SERVICE);
        imm.hideSoftInputFromWindow(etKode.getWindowToken(), 0);
    }

    public void submitBrg(Barang brg){
        /* Berikut ini adalah kode yang digunakan untuk mengirimkan data ke
         * Firebase Realtime Database dan juga kita set onSuccessListener yang
         * berisi kode yang akan dijalankan ketika data berhasil ditambahkan
         */
        database.child("Barang").push().setValue(brg).addOnSuccessListener(this,
new OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void aVoid) {
                etKode.setText("");
                etNama.setText("");
                Toast.makeText(getApplicationContext(),"Data berhasil

```

```

ditambahkan", Toast.LENGTH_LONG).show();
        }
    });
}

public static Intent getActIntent(Activity activity) {
    // kode untuk pengambilan Intent
    return new Intent(activity, TambahData.class);
}
}

```

7. Jangan lupa ijin penggunaan internet dalam AndroidManifest.xml

```

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

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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/namaApp"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".TambahData"
            android:label="@string/txtJudul2">
            </activity>
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>

```

HASIL PENGUJIAN APLIKASI

Setelah aplikasi selesai dibuat dan tidak ada error, maka kita bisa mencoba untuk menjalankan aplikasi, sekaligus membuka console firebase untuk memantau hasilnya.

Perlu kalian ketahui bahwa Firebase ini adalah ‘NoSQL’ yang artinya kita tidak perlu repot menggunakan bahasa SQL untuk mendefinisikan database dan table—tabelnya ataupun manipulasi data. Database akan otomatis dibuat manakala kita jalankan aplikasinya.

Semua data akan disajikan dalam bentuk pohon (Tree) JSON seperti terlihat dalam gambar berikut ini.

