

Firebase Realtime Database

1. AndroidManifest.xml

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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/Theme.PJR8B">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

2. build.gradle (Project)

```
buildscript {
    repositories {
        google()
        mavenCentral()
    }
    dependencies {
        classpath 'com.android.tools.build:gradle:8.1.1'
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:1.6.21"
        classpath 'com.google.gms:google-services:4.3.15'
    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}
```

3. build.gradle (module)

```
plugins {
    id 'com.android.application'
    id 'kotlin-android'
    id 'com.google.gms.google-services'
}

android {
    compileSdk 33
```

```

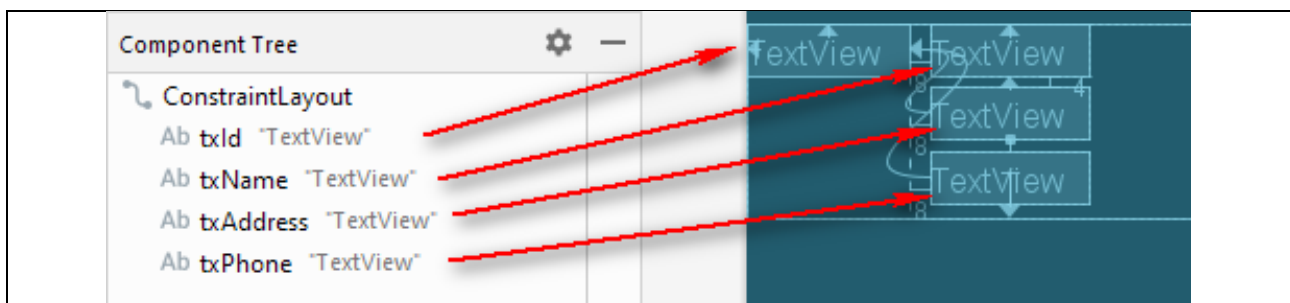
defaultConfig {
    applicationId "polinema.mi.pjr8b"
    minSdk 21
    targetSdk 33
    versionCode 1
    versionName "1.0"

    testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
}
buildTypes {
    release {
        minifyEnabled false
        proguardFiles getDefaultProguardFile(
            'proguard-android-optimize.txt'), 'proguard-rules.pro'
    }
}
compileOptions {
    sourceCompatibility JavaVersion.VERSION_1_8
    targetCompatibility JavaVersion.VERSION_1_8
}
kotlinOptions {
    jvmTarget = '1.8'
}
buildFeatures{
    viewBinding true
}
namespace 'polinema.mi.pjr8b'
}

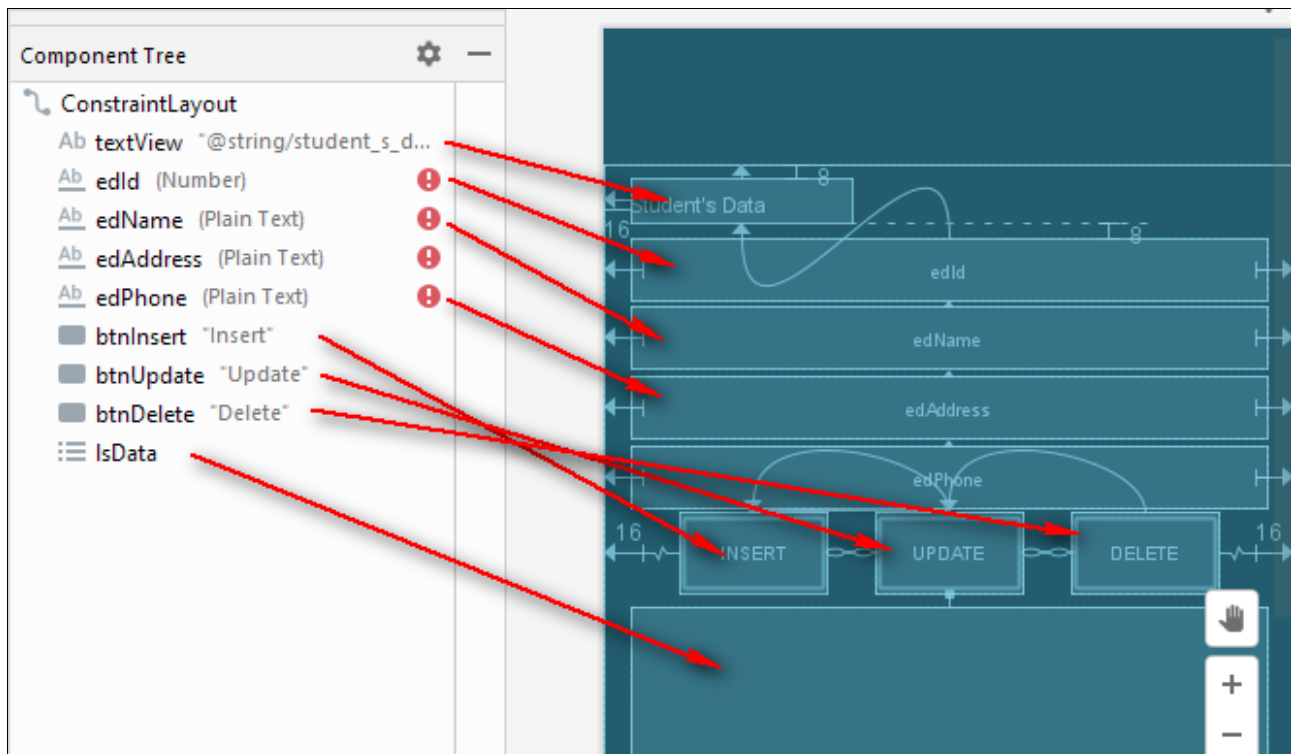
dependencies {
    implementation platform('com.google.firebase:firebase-bom:32.2.3')
    implementation 'com.google.firebase:firebase-database-ktx'
    implementation 'androidx.core:core-ktx:1.9.0'
    implementation 'androidx.appcompat:appcompat:1.6.1'
    implementation 'com.google.android.material:material:1.9.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
    testImplementation 'junit:junit:'
    androidTestImplementation 'androidx.test.ext:junit:1.1.5'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.5.1'
}

```

4. row_data.xml



5. activity_main.xml



6. MainActivity.kt (class)

```
package polinema.mi.pjr8b

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.widget.AdapterView
import android.widget.AdapterView.OnItemClickListener
import android.widget.ListAdapter
import android.widget.SimpleAdapter
import android.widget.Toast
import com.google.firebase.database.*
import com.google.firebase.database.ktx.getValue
import polinema.mi.pjr8b.databinding.ActivityMainBinding

class MainActivity : AppCompatActivity(), View.OnClickListener {

    lateinit var db : DatabaseReference
    lateinit var adapter : ListAdapter
    var alMhs = ArrayList<HashMap<String,Any>>()
    var mhs = Mahasiswa()
    var hm = HashMap<String,Any>()
    lateinit var b : ActivityMainBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        b = ActivityMainBinding.inflate(layoutInflater)
        setContentView(b.root)

        FirebaseDatabase.getInstance().setPersistenceEnabled(true)

        b.btnInsert.setOnClickListener(this)
        b.btnDelete.setOnClickListener(this)
    }
}
```

```

        b.btnUpdate.setOnClickListener(this)
        b.lvMahasiswa.setOnItemClickListener(itemClickListener)
    }

    override fun onStart() {
        super.onStart()
        db = FirebaseDatabase.getInstance().getReference("TabelMahasiswa")
        showData()
    }

    fun showData(){
        db.addValueEventListener(object : ValueEventListener{
            override fun onDataChange(snapshot: DataSnapshot) {

                val dataSnapshotIterable = snapshot.children
                val iterator = dataSnapshotIterable.iterator()
                alMhs.clear()
                while (iterator.hasNext()){
                    mhs = iterator.next().getValue<Mahasiswa>()!!
                    hm = HashMap()
                    hm.put("nim", mhs.nim!!)
                    hm.put("namaMhs", mhs.namaMhs!!)
                    hm.put("alamatMhs", mhs.alamatMhs!!)
                    alMhs.add(hm)
                }

                //binding data didalam arrayllist dengan layout row_mahasiswa.xml
                //untuk ditampilkan melalui listview
                adapter = SimpleAdapter(
                    this@MainActivity,
                    alMhs,
                    R.layout.row_mahasiswa,
                    arrayOf("nim", "namaMhs", "alamatMhs"),
                    intArrayOf(R.id.txNim, R.id.txNama, R.id.txAlamat)
                )
                b.lvMahasiswa.setAdapter(adapter)
            }

            override fun onCancelled(error: DatabaseError) {
                Toast.makeText(this@MainActivity,
                    "Connection to database error : ${error.message}",
                    Toast.LENGTH_SHORT).show()
            }
        })
    }

    val itemClickListener = AdapterView.OnItemClickListener { adapterView, view, i, l ->
        hm = HashMap()
        hm = alMhs.get(i)
        b.edNim.setText(hm.get("nim").toString())
        b.edNama.setText(hm.get("namaMhs").toString())
        b.edAlamat.setText(hm.get("alamatMhs").toString())
    }

    override fun onClick(p0: View?) {
        when(p0!!.id){
            R.id.btnInsert -> {
                //Toast.makeText(this, "hello world", Toast.LENGTH_SHORT).show()
            }
        }
    }

```

```

        mhs.nim = b.edNim.text.toString()
        mhs.namaMhs = b.edNama.text.toString()
        mhs.alamatMhs = b.edAlamat.text.toString()
        db.child(mhs.nim!!).setValue(mhs)
    }
    R.id.btnDelete -> {
        db.child("234").removeValue()
    }
    R.id.btnUpdate -> {
        db.child(mhs.nim!!).child("NamaMhs").setValue(b.edNama.text.toString())
        db.child(mhs.nim!!).child("alamatMhs").setValue(b.edAlamat.text.toString())
    }
}
b.edNim.setText(""); b.edNama.setText(""); b.edAlamat.setText("")
}
}

```

7. Mahasiswa.kt (class)

```

package polinema.mi.pjr8b

class Mahasiswa {
    var namaMhs: String? = null
    var alamatMhs: String? = null
    var nim: String? = null
}

```