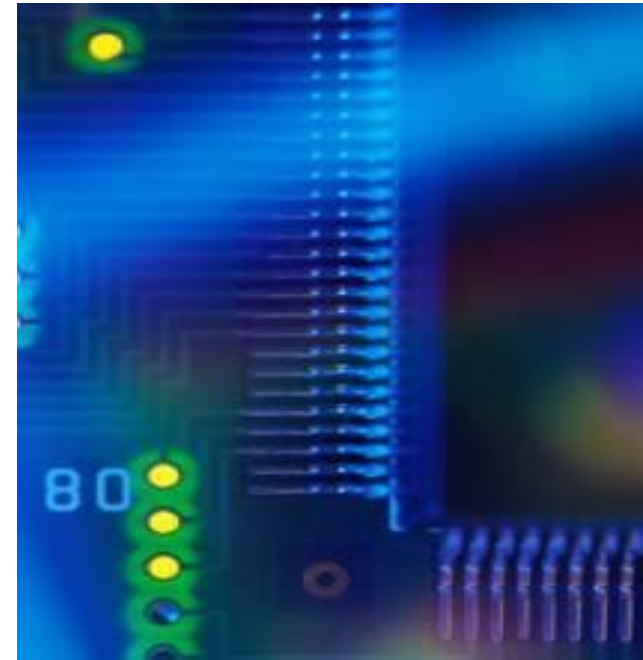




KEMENTERIAN KOMUNIKASI DAN INFORMATIKA
REPUBLIK INDONESIA
Menuju Masyarakat Informasi Indonesia



JUNIOR MOBILE PROGRAMMER

PROJECT: APLIKASI DATABASE DENGAN SQLITE + BACKEND (Aplikasi Database SQLite & Backend API)

Deskripsi Singkat

Deskripsi Singkat mengenai Topik

Peserta pelatihan melakukan praktek mengerjakan project membuat aplikasi database dengan backend

Tujuan Pelatihan

1. Peserta pelatihan dapat mengerjakan tugas yang diberikan sesuai dengan persyaratan yang ditentukan.

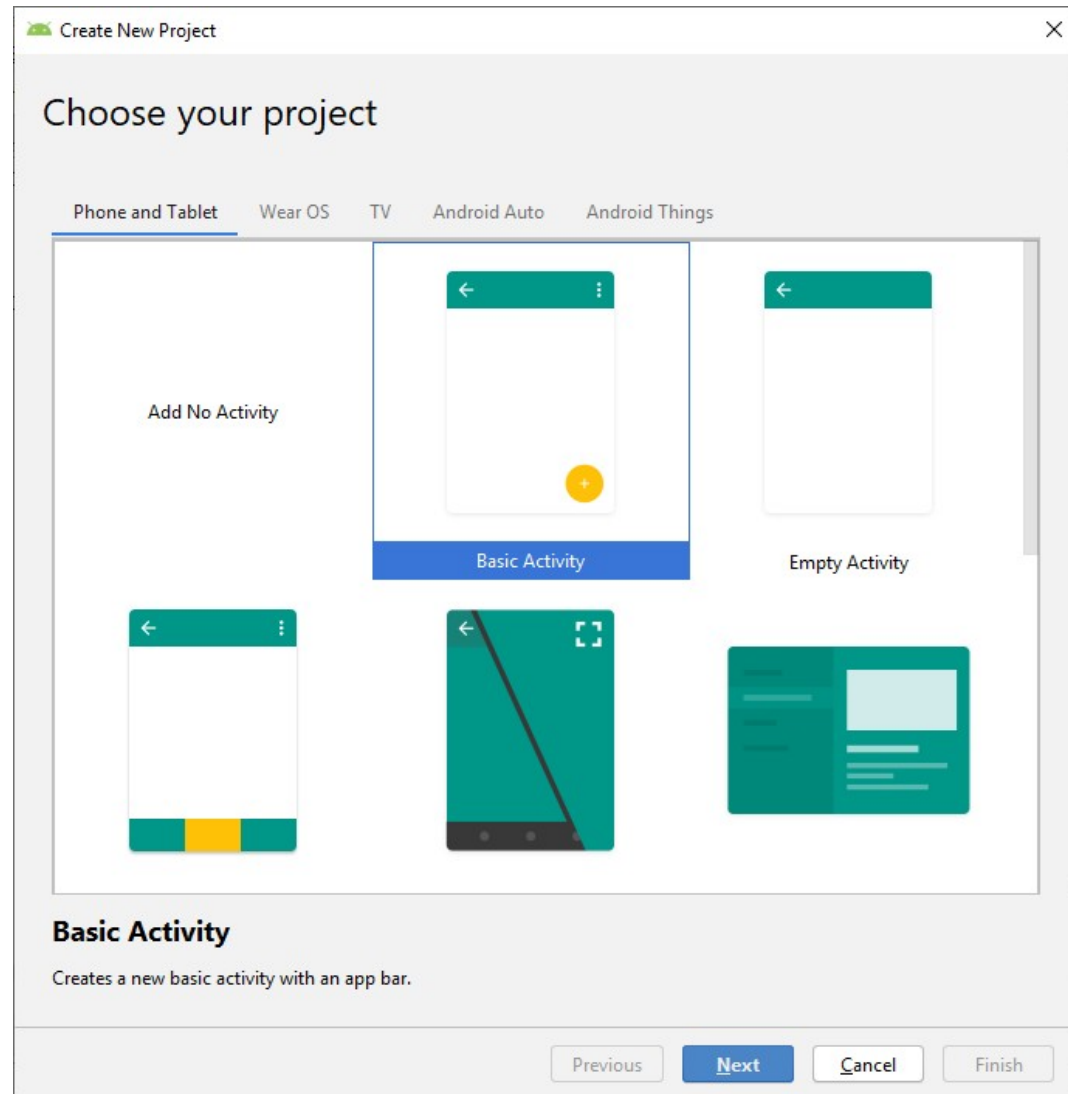
Materi Yang akan disampaikan:

1. -

Tugas :

Membuat aplikasi database dengan backend menggunakan alat bantu yang telah diinstalasi sesuai dengan spesifikasi projek yang diberikan.

Aplikasi Database SQLite



- ❖ Buat Project Baru dengan cara klik menu File → New → New Project.
- ❖ Pilih Basic Activity.

Aplikasi Database SQLite

Create New Project

Configure your project

Name
Aplikasi SQLite

Package name
com.sugiartha.aplikasisqlite

Save location
Jsers\I Komang Sugiartha\AndroidStudioProjects\AplikasiSQLite

Language
Java

Minimum API level
API 15: Android 4.0.3 (IceCreamSandwich)

i Your app will run on approximately **100%** of devices.
[Help me choose](#)

☐ This project will support instant apps

☒ Use androidx.* artifacts

⚠ project location should not contain whitespace, as this can cause problems with the NDK tools.

Previous Next Cancel Finish

- ❖ Isi Name dengan nama Aplikasi SQLite
- ❖ Kolom yang lain biarkan secara default.
- ❖ Klik Finish

Aplikasi Database SQLite

- ❖ Buka layout activity_main.xml dan ganti bagian ic_dialog_email menjadi ic_input_add

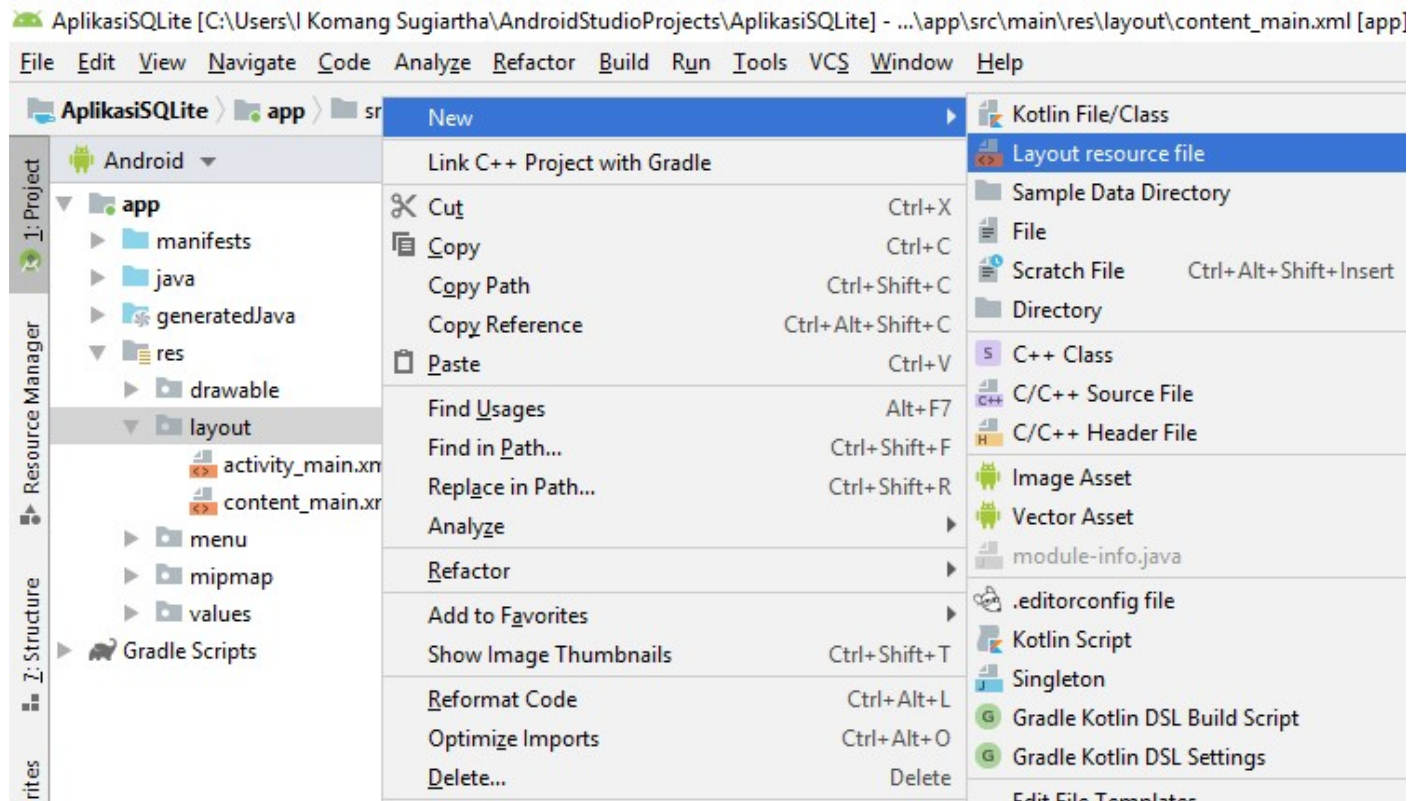
```
25 <com.google.android.material.floatingactionbutton.FloatingActionButton
26     android:id="@+id/fab"
27     android:layout_width="wrap_content"
28     android:layout_height="wrap_content"
29     android:layout_gravity="bottom|end"
30     android:layout_margin="16dp"
31     app:srcCompat="@android:drawable/ic_input_add" />
32
33 </androidx.coordinatorlayout.widget.CoordinatorLayout>
```

Aplikasi Database SQLite

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     app:layout_behavior="@string/appbar_scrolling_view_behavior"
9     tools:context=".MainActivity"
10    tools:showIn="@layout/activity_main">
11
12    <ListView
13        android:id="@+id/list_view"
14        android:layout_width="match_parent"
15        android:layout_height="match_parent" />
16
17 </RelativeLayout>
```

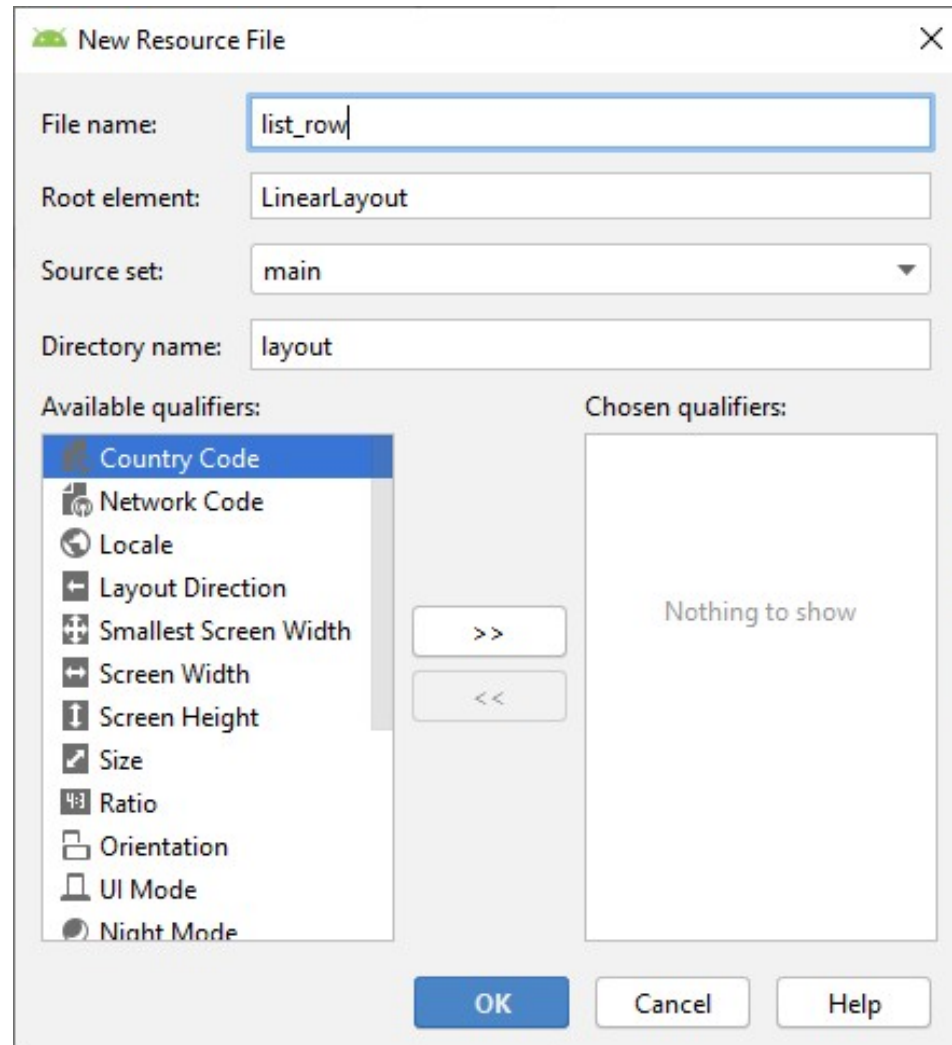
- ❖ Buka layout content_main.xml
- ❖ Ubah layout menjadi RelativeLayout
- ❖ Tambahkan blok program ListView

Aplikasi Database SQLite



- ❖ Buat layout baru dengan cara klik kanan pada package layout → New → Layout resource file

Aplikasi Database SQLite



- ❖ Isi File Name dengan nama list_row dan pastikan Root Element adalah LinearLayout.
- ❖ Selanjutnya klik OK.

Aplikasi Database SQLite

- ❖ Buka layout list_row.xml
- ❖ Biat 3 buah Text View seperti pada gambar dibawah ini.

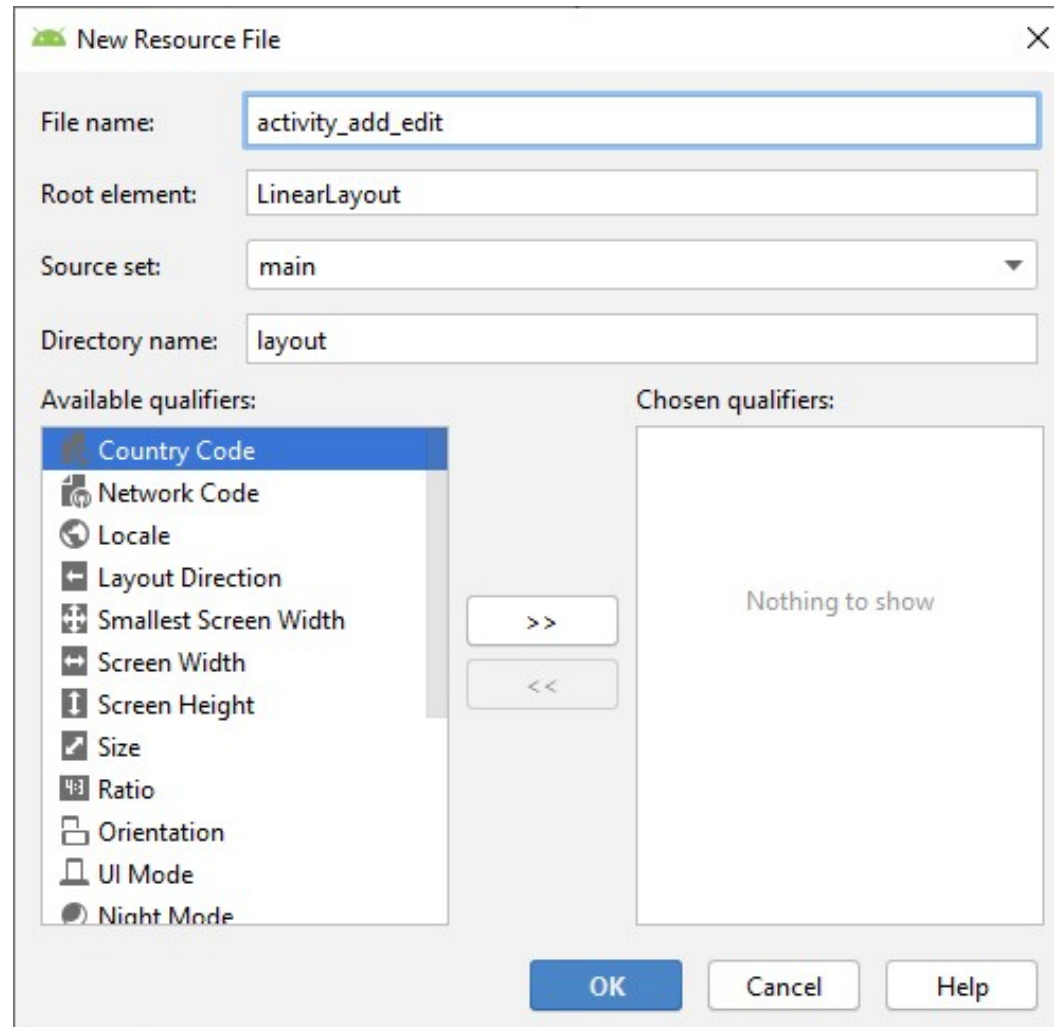
```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:orientation="vertical"
4     android:padding="16dp"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent">
7
8     <TextView
9         android:id="@+id/id"
10        android:layout_width="wrap_content"
11        android:layout_height="wrap_content"
12        android:visibility="gone"/>
```

Aplikasi Database SQLite

```
14 <TextView
15     android:id="@+id/name"
16     android:layout_marginBottom="5dp"
17     android:textStyle="bold"
18     android:textSize="16dp"
19     android:layout_width="wrap_content"
20     android:layout_height="wrap_content" />
21
22 <TextView
23     android:id="@+id/address"
24     android:textSize="16dp"
25     android:layout_width="wrap_content"
26     android:layout_height="wrap_content" />
27
28 </LinearLayout>
```

- ❖ Lanjutan 2 Text View pada program sebelumnya.
- ❖ Fungsi dari Text View ini adalah untuk menampilkan Nama dan Alamat yang telah diinput.

Aplikasi Database SQLite



- ❖ Buat layout baru dengan nama activity_add_edit
- ❖ Selanjutnya klik OK

Aplikasi Database SQLite

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:orientation="vertical"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent">
6
7     <EditText
8         android:id="@+id/txt_id"
9         android:layout_width="match_parent"
10        android:layout_height="wrap_content"
11        android:inputType="number"
12        android:visibility="gone"/>
13
14    <EditText
15        android:id="@+id/txt_name"
16        android:layout_width="match_parent"
17        android:layout_height="wrap_content"
18        android:hint="Input Nama"/>
19
20    <EditText
21        android:id="@+id/txt_address"
22        android:layout_marginTop="10dp"
23        android:layout_width="match_parent"
24        android:layout_height="wrap_content"
25        android:hint="Input Alamat"/>
```

- ❖ Buka layout activity_add_edit.xml
- ❖ Buat 3 buah Edit Text.
- ❖ Fungsi Edit Text ini adalah untuk menginput Nama dan Alamat.
- ❖ Buat script seperti gambar disamping.

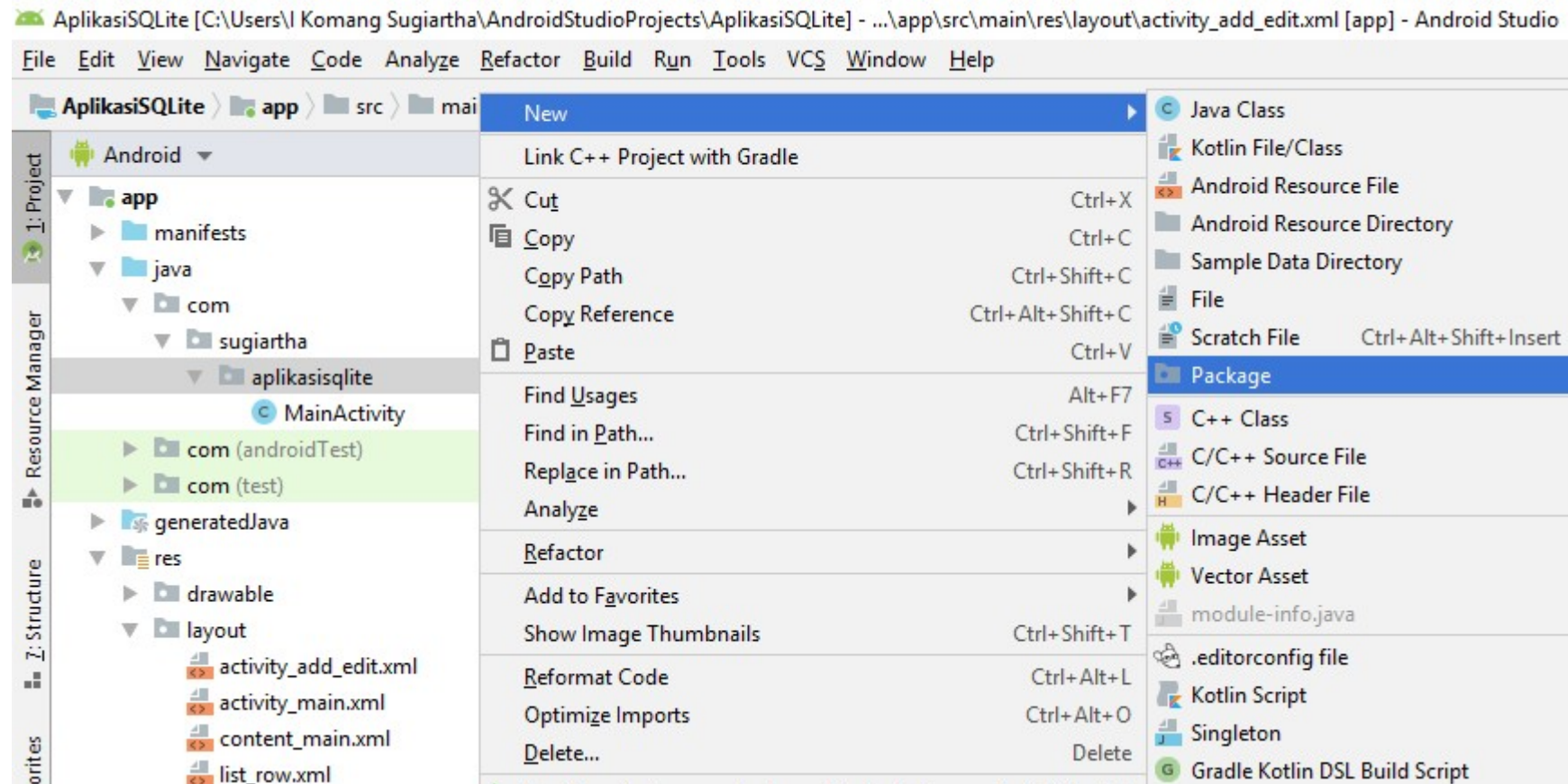
Aplikasi Database SQLite

```
27 <LinearLayout
28     android:orientation="horizontal"
29     android:layout_marginTop="10dp"
30     android:layout_width="match_parent"
31     android:layout_height="wrap_content">
32
33     <Button
34         android:id="@+id/btn_submit"
35         android:layout_weight="0.5"
36         android:layout_width="match_parent"
37         android:layout_height="wrap_content"
38         android:text="Submit"/>
39
40     <Button
41         android:id="@+id/btn_cancel"
42         android:layout_weight="0.5"
43         android:layout_width="match_parent"
44         android:layout_height="wrap_content"
45         android:text="Cancel"/>
46
47 </LinearLayout>
48
49 </LinearLayout>
```

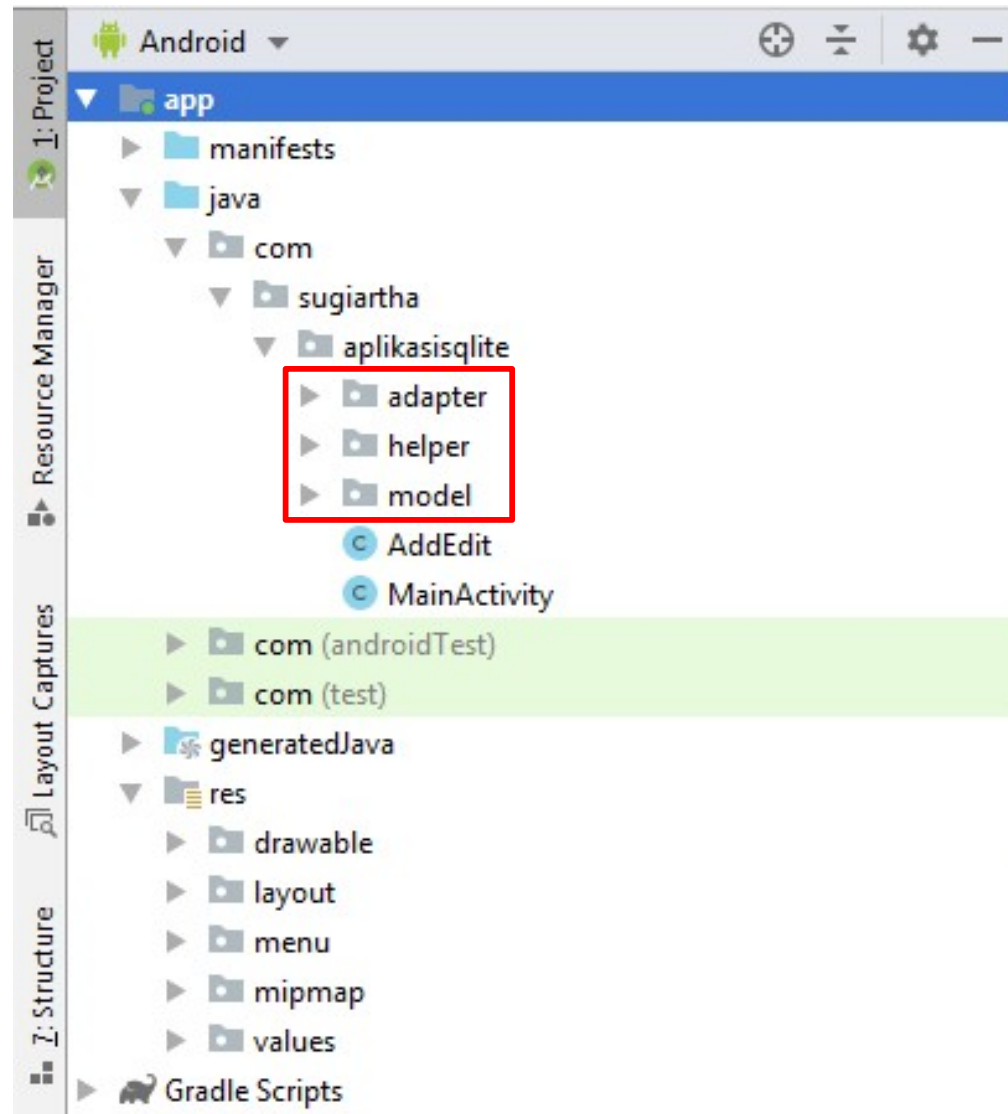
- ❖ Lanjutan script sebelumnya, yaitu membuat layout baru. Didalam Linear Layout ada Linear Layout (sub layout).
- ❖ Didalam Linear Layout tambahkan 2 Button yaitu, Submit dan Cancel.
- ❖ Seperti gambar disamping.

Aplikasi Database SQLite

- ❖ Agar project terstruktur dan terorganisir, buat 3 paket dengan nama adapter, helper, dan model.
- ❖ Dengan cara klik kanan pada aplikasisqlite → New → Package.



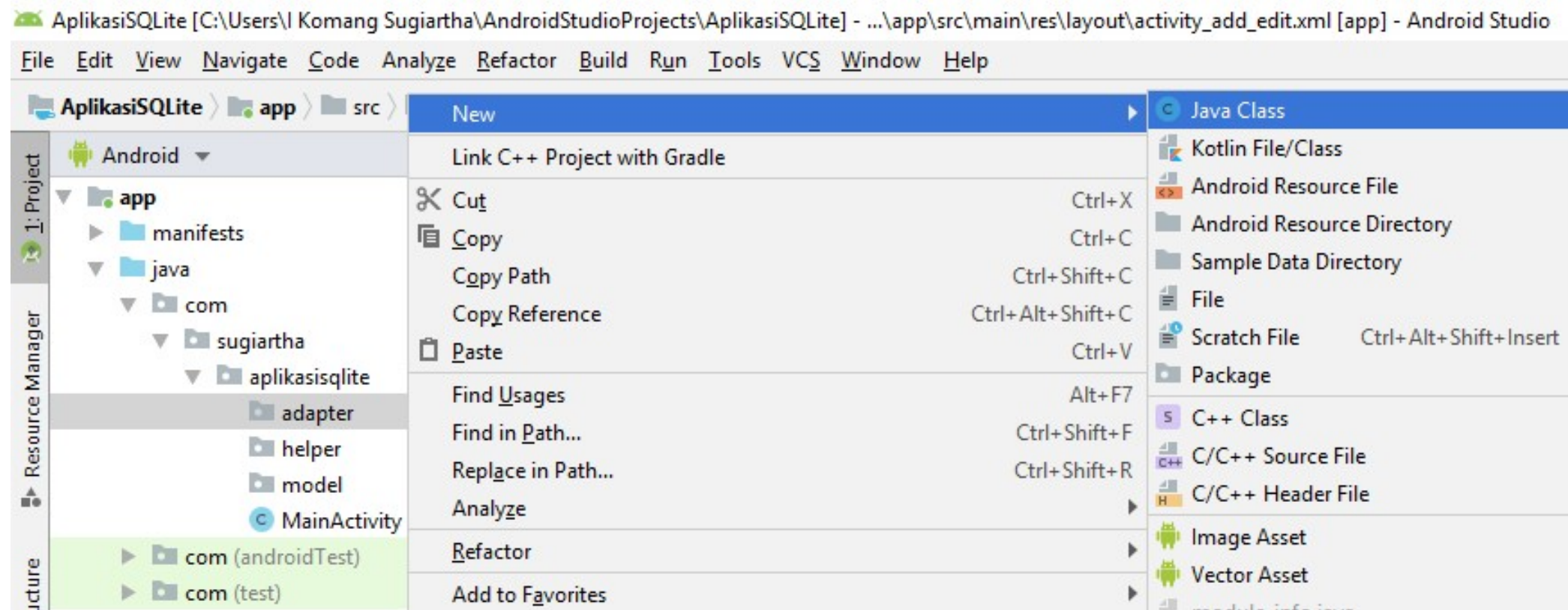
Aplikasi Database SQLite



- ❖ Berikut tampilan package adapter, helper dan model yang telah dibuat.

Aplikasi Database SQLite

- ❖ Buat class baru didalam package adapter.
- ❖ Dengan cara klik kanan pada package adapter, pilih New → Java Class.



Aplikasi Database SQLite

Create New Class

Name:

Kind:

Superclass:

Interface(s):

Package:

Visibility: ☒ Public ☐ Package Private

Modifiers: ☒ None ☐ Abstract ☐ Final

☐ Show Select Overrides Dialog

OK Cancel Help

- ❖ Isi Name dengan nama Adapter.
- ❖ Class ini berfungsi untuk menampilkan data seperti id, nama, dan alamat kemudian ditampilkan ke dalam listview.

Aplikasi Database SQLite

```
1 package com.sugiartha.aplikasisqlite.adapter;
2
3 import android.app.Activity;
4 import android.content.Context;
5 import android.view.LayoutInflater;
6 import android.view.View;
7 import android.view.ViewGroup;
8 import android.widget.BaseAdapter;
9 import android.widget.TextView;
10
11 public class Adapter extends BaseAdapter {
12     private Activity activity;
13     private LayoutInflater inflater;
14     private List<Data> items;
15
16     public Adapter(Activity activity, List<Data> items) {
17         this.activity = activity;
18         this.items = items;
19     }
20
21     @Override
22     public int getCount() {
23         return items.size();
24     }
```

- ❖ Buka file Adapter.java
- ❖ Ketik program seperti pada gambar di samping ini.
- ❖ Untuk blok program yang masih error (berwarna merah) diabaikan saja, karena program tersebut ada yang memanggil program selanjutnya.

Aplikasi Database SQLite

```
26      @Override
27      public Object getItem(int location) {
28          return items.get(location);
29      }
30
31      @Override
32      public long getItemId(int position) {
33          return position;
34      }
35
36      @Override
37      public View getView(int position, View convertView, ViewGroup parent) {
38
39          if (inflater == null)
40              inflater = (LayoutInflater) activity
41                  .getSystemService(Context.LAYOUT_INFLATER_SERVICE);
42
43          if (convertView == null)
44              convertView = inflater.inflate(R.layout.list_row, null);
45
46          TextView id = (TextView) convertView.findViewById(R.id.id);
47          TextView name = (TextView) convertView.findViewById(R.id.name);
48          TextView address = (TextView) convertView.findViewById(R.id.address);
```

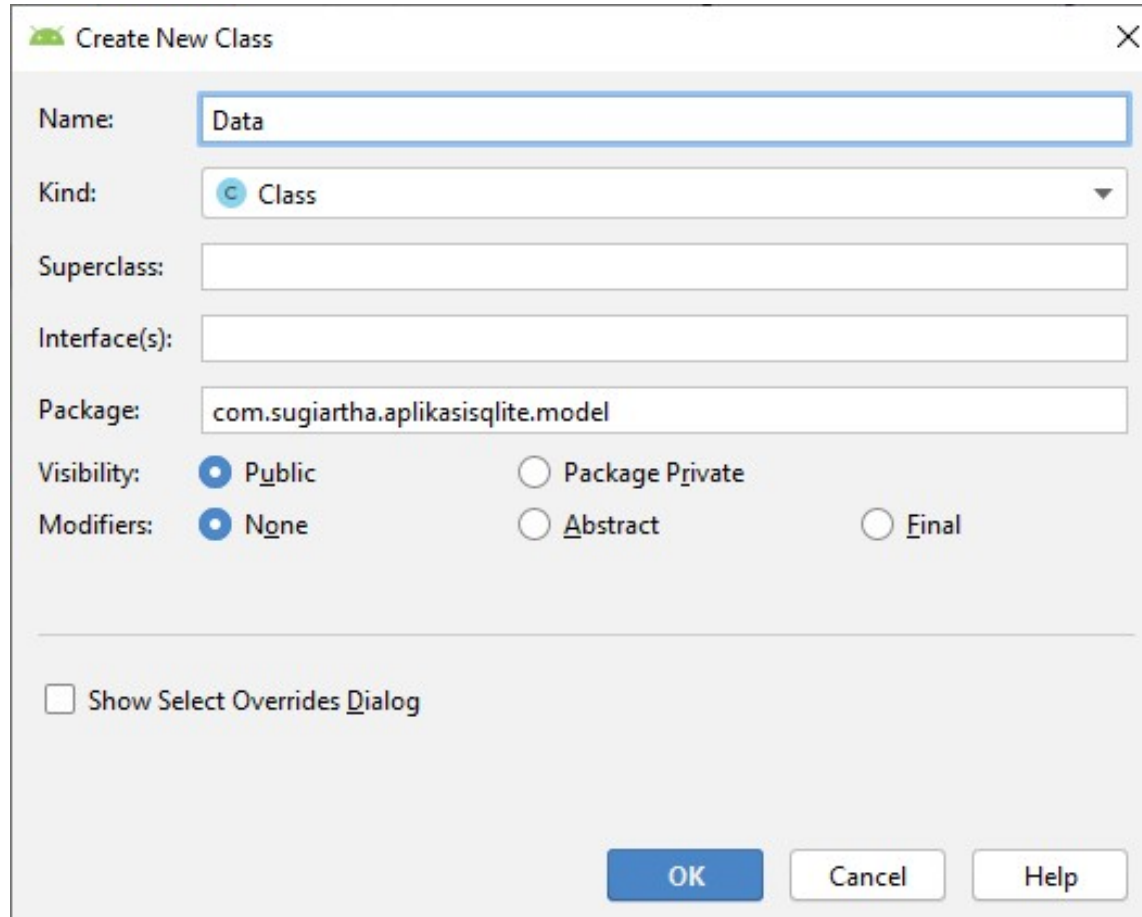
- ❖ Lanjutan program sebelumnya.
- ❖ Untuk blok program yang masih error (berwarna merah) diabaikan saja, karena program tersebut ada yang memanggil program selanjutnya.
- ❖ Jadi harus membuat program lanjutan, baru error tersebut bisa diatasi.

Aplikasi Database SQLite

```
50      Data data = items.get(position);  
51  
52      id.setText(data.getId());  
53      name.setText(data.getName());  
54      address.setText(data.getAddress());  
55  
56      return convertView;  
57  }  
58 }
```

- ❖ Lanjutan program terakhir untuk Adapter.java
- ❖ Untuk blok program yang masih error (berwarna merah) diabaikan saja, karena program tersebut ada yang memanggil program selanjutnya.
- ❖ Jadi harus membuat program lanjutan, baru error tersebut bisa diatasi.

Aplikasi Database SQLite



Create New Class

Name:

Kind:

Superclass:

Interface(s):

Package:

Visibility: ☒ Public ☐ Package Private

Modifiers: ☒ None ☐ Abstract ☐ Final

☐ Show Select Overrides Dialog

OK Cancel Help

- ❖ Buat class baru didalam package model.
- ❖ Isi Name dengan nama Data.

Aplikasi Database SQLite

```
1 package com.sugiartha.aplikasisqlite.model;
2
3 public class Data {
4     private String id, name, address;
5
6     public Data() {
7     }
8
9     public Data(String id, String name, String address) {
10         this.id = id;
11         this.name = name;
12         this.address = address;
13     }
14
15     public String getId() {
16         return id;
17     }
18
19     public void setId(String id) {
20         this.id = id;
21     }
22
23     public String getName() {
24         return name;
25     }
```

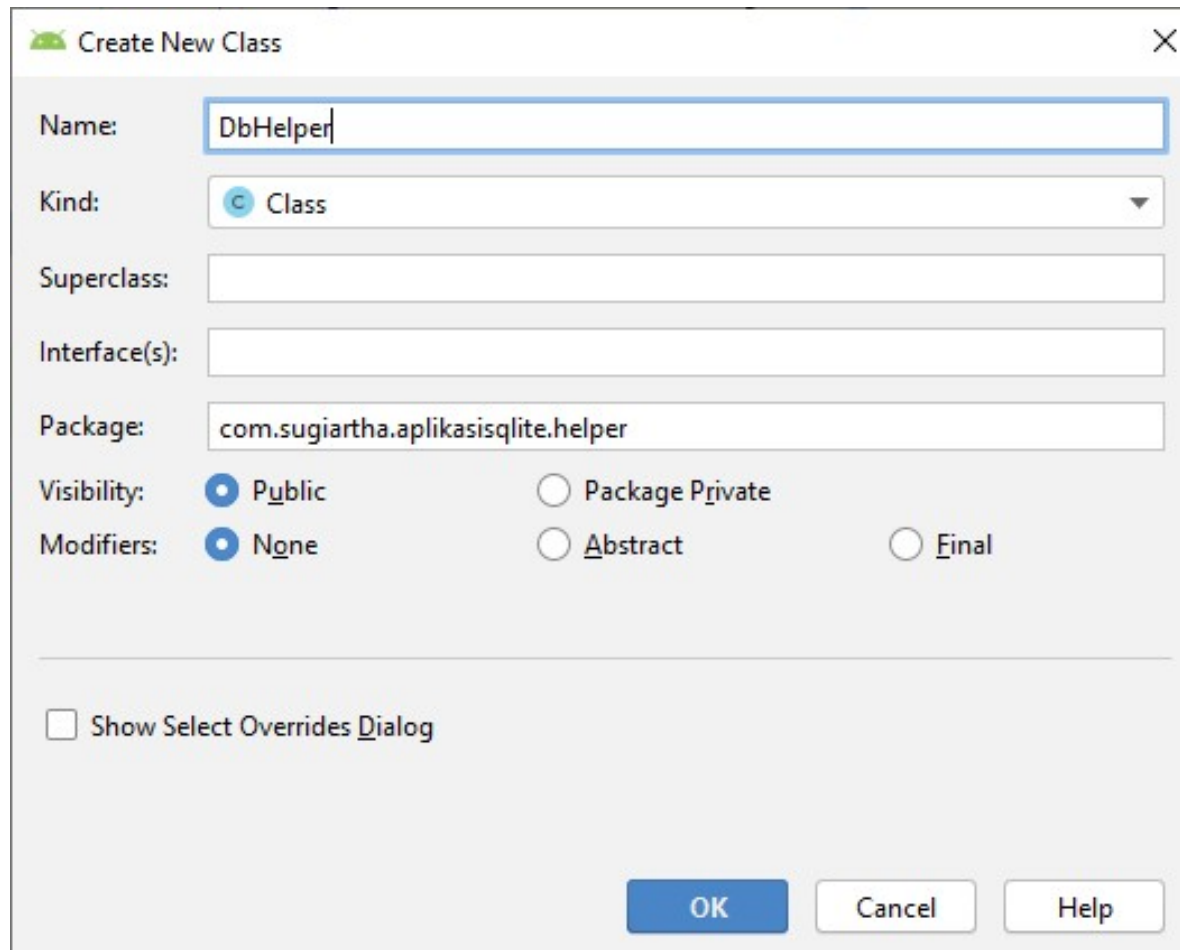
- ❖ Buka file Data.java
- ❖ Ketik program seperti pada gambar di samping ini.
- ❖ Class ini berfungsi untuk membuat objek.
- ❖ Objek ini berisi informasi seperti id, nama, dan alamat.

Aplikasi Database SQLite

```
27 public void setName(String name) {  
28     this.name = name;  
29 }  
30  
31 public String getAddress() {  
32     return address;  
33 }  
34  
35 public void setAddress(String address) {  
36     this.address = address;  
37 }  
38 }
```

❖ Lanjutan program terakhir untuk Data.java

Aplikasi Database SQLite



Create New Class

Name: DbHelper

Kind: Class

Superclass:

Interface(s):

Package: com.sugiartha.aplikasisqlite.helper

Visibility: ☒ Public ☐ Package Private

Modifiers: ☒ None ☐ Abstract ☐ Final

☐ Show Select Overrides Dialog

OK Cancel Help

- ❖ Buat class baru didalam package helper.
- ❖ Isi Name dengan nama DbHelper.

Aplikasi Database SQLite

```
1 package com.sugiartha.aplikasisqlite.helper;
2
3 import android.content.Context;
4 import android.database.Cursor;
5 import android.database.sqlite.SQLiteDatabase;
6 import android.database.sqlite.SQLiteOpenHelper;
7 import android.util.Log;
8
9 import java.util.ArrayList;
10 import java.util.HashMap;
11
12 public class DbHelper extends SQLiteOpenHelper {
13     private static final int DATABASE_VERSION = 2;
14
15     static final String DATABASE_NAME = "digitaltalent.db";
16
17     public static final String TABLE_SQLite = "sqlite";
18
19     public static final String COLUMN_ID = "id";
20     public static final String COLUMN_NAME = "name";
21     public static final String COLUMN_ADDRESS = "address";
22
23     public DbHelper(Context context) {
24         super(context, DATABASE_NAME, factory: null, DATABASE_VERSION);
25     }
```

- ❖ Buka file DbHelper.java
- ❖ Ketik program seperti pada gambar di samping ini.
- ❖ Class ini berfungsi sebagai pengelola query SQL CRUD.
- ❖ Pembuatan Database SQLite.

Aplikasi Database SQLite

- ❖ Lanjutan program DbHelper.java
- ❖ Pada blok program ini menjelaskan mengenai pembuatan table dengan kolom dan tipe datanya.

```
27      @Override
28      public void onCreate(SQLiteDatabase db) {
29          final String SQL_CREATE_MOVIE_TABLE = "CREATE TABLE " + TABLE_SQLite + " (" +
30              COLUMN_ID + " INTEGER PRIMARY KEY autoincrement, " +
31              COLUMN_NAME + " TEXT NOT NULL, " +
32              COLUMN_ADDRESS + " TEXT NOT NULL" +
33              " )";
34
35          db.execSQL(SQL_CREATE_MOVIE_TABLE);
36      }
37
38      @Override
39      public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
40          db.execSQL("DROP TABLE IF EXISTS " + TABLE_SQLite);
41          onCreate(db);
42      }
```


Aplikasi Database SQLite

```
44 public ArrayList<HashMap<String, String>> getAllData() {  
45     ArrayList<HashMap<String, String>> wordList;  
46     wordList = new ArrayList<HashMap<String, String>>();  
47     String selectQuery = "SELECT * FROM " + TABLE_SQLite;  
48     SQLiteDatabase database = this.getWritableDatabase();  
49     Cursor cursor = database.rawQuery(selectQuery, selectionArgs: null);  
50     if (cursor.moveToFirst()) {  
51         do {  
52             HashMap<String, String> map = new HashMap<String, String>();  
53             map.put(COLUMN_ID, cursor.getString(0));  
54             map.put(COLUMN_NAME, cursor.getString(1));  
55             map.put(COLUMN_ADDRESS, cursor.getString(2));  
56             wordList.add(map);  
57         } while (cursor.moveToNext());  
58     }  
59  
60     Log.e(tag: "select sqlite ", msg: "" + wordList);  
61  
62     database.close();  
63     return wordList;  
64 }
```

- ❖ Lanjutan program DbHelper.java
- ❖ Blok program ini berfungsi untuk mengambil semua data yang ada pada table.

Aplikasi Database SQLite

```
66 public void insert(String name, String address) {
67     SQLiteDatabase database = this.getWritableDatabase();
68     String queryValues = "INSERT INTO " + TABLE_SQLite + " (name, address) " +
69         "VALUES ('" + name + "', '" + address + "')";
70
71     Log.e( tag: "insert sqlite ", msg: "" + queryValues);
72     database.execSQL(queryValues);
73     database.close();
74 }
75
76 public void update(int id, String name, String address) {
77     SQLiteDatabase database = this.getWritableDatabase();
78
79     String updateQuery = "UPDATE " + TABLE_SQLite + " SET "
80         + COLUMN_NAME + "='" + name + "', "
81         + COLUMN_ADDRESS + "='" + address + "'"
82         + " WHERE " + COLUMN_ID + "='" + id + "'";
83     Log.e( tag: "update sqlite ", updateQuery);
84     database.execSQL(updateQuery);
85     database.close();
86 }
```

- ❖ Lanjutan program DbHelper.java
- ❖ Blok program ini berfungsi untuk menginput data dan mengupdate data.
- ❖ Yaitu fungsi public void insert dan public void update.

Aplikasi Database SQLite

- ❖ Blok program terakhir pada DbHelper.java
- ❖ Membuat fungsi public void delete, dimana fungsi ini digunakan untuk menghapus data yang telah diinput.

```
88     public void delete(int id) {  
89         SQLiteDatabase database = this.getWritableDatabase();  
90  
91         String updateQuery = "DELETE FROM " + TABLE_SQLite + " WHERE " + COLUMN_ID + "=" + "'" + id + "'";  
92         Log.e("update sqlite ", updateQuery);  
93         database.execSQL(updateQuery);  
94         database.close();  
95     }  
96 }
```

Aplikasi Database SQLite

```
15 public class MainActivity extends AppCompatActivity {  
16  
17     ListView listView;  
18     AlertDialog.Builder dialog;  
19     List<Data> itemList = new ArrayList<Data>();  
20     Adapter adapter;  
21     DBHelper SQLite = new DBHelper(this);  
22  
23     public static final String TAG_ID = "id";  
24     public static final String TAG_NAME = "name";  
25     public static final String TAG_ADDRESS = "address";
```

- ❖ Buka file MainActivity.java
- ❖ Ketik program seperti pada gambar di samping ini.
- ❖ Blok program ini berfungsi untuk deklarasi variable.

Aplikasi Database SQLite

```
39      @Override
40      protected void onCreate(Bundle savedInstanceState) {
41          super.onCreate(savedInstanceState);
42          setContentView(R.layout.activity_main);
43          Toolbar toolbar = findViewById(R.id.toolbar);
44          setSupportActionBar(toolbar);
45
46          //Tambah SQLite
47          SQLite = new DBHelper(getApplicationContext());
48
49          FloatingActionButton fab = findViewById(R.id.fab);
50
51          //Tambah List View
52          listView = (ListView) findViewById(R.id.list_view);
53
54          fab.setOnClickListener(new View.OnClickListener() {
55              @Override
56              public void onClick(View view) {
57                  //Tambah Intent untuk pindah ke halaman Add dan Edit
58                  Intent intent = new Intent( packageContext: MainActivity.this, AddEdit.class);
59                  startActivity(intent);
60              }
61          });
62
63          //Tambah adapter dan listview
64          adapter = new Adapter( activity: MainActivity.this, itemList);
65          listView.setAdapter(adapter);
```

- ❖ Lanjutan program MainActivity.java
- ❖ Tambahkan blok program seperti pada gambar disamping.

Aplikasi Database SQLite

- ❖ Lanjutan program MainActivity.java
- ❖ Blok Program ini berfungsi untuk melakukan edit dan hapus data pada saat daftar list view ditekan lama.
- ❖ Tambahkan blok program seperti pada gambar dibawah ini.

```
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80

// tekan lama daftar listview untuk menampilkan edit dan hapus
listView.setOnItemLongClickListener(new AdapterView.OnItemLongClickListener() {

    @Override
    public boolean onItemLongClick(final AdapterView<?> parent, View view,
                                   final int position, long id) {

        // TODO Auto-generated method stub
        final String idx = itemList.get(position).getId();
        final String name = itemList.get(position).getName();
        final String address = itemList.get(position).getAddress();

        final CharSequence[] dialogitem = {"Edit", "Delete"};
        dialog = new AlertDialog.Builder(context: MainActivity.this);
        dialog.setCancelable(true);
        dialog.setItems(dialogitem, new DialogInterface.OnClickListener() {
```


Aplikasi Database SQLite

- ❖ Lanjutan program MainActivity.java
- ❖ Tambahkan blok program seperti pada gambar dibawah ini.

```
82
83 
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105

@Override
public void onClick(DialogInterface dialog, int which) {
    // TODO Auto-generated method stub
    switch (which) {
        case 0:
            Intent intent = new Intent( packageContext: MainActivity.this, AddEdit.class);
            intent.putExtra(TAG_ID, idx);
            intent.putExtra(TAG_NAME, name);
            intent.putExtra(TAG_ADDRESS, address);
            startActivity(intent);
            break;
        case 1:
            SQLite.delete(Integer.parseInt(idx));
            itemList.clear();
            getAllData();
            break;
    }
}

}).show();
return false;
}

});
getAllData();
}
```

Aplikasi Database SQLite

```
107 private void getAllData() {  
108     ArrayList<HashMap<String, String>> row = SQLite.getAllData();  
109  
110     for (int i = 0; i < row.size(); i++) {  
111         String id = row.get(i).get(TAG_ID);  
112         String poster = row.get(i).get(TAG_NAME);  
113         String title = row.get(i).get(TAG_ADDRESS);  
114  
115         Data data = new Data();  
116  
117         data.setId(id);  
118         data.setName(poster);  
119         data.setAddress(title);  
120  
121         itemList.add(data);  
122     }  
123     adapter.notifyDataSetChanged();  
124 }  
125  
126 @Override  
127 protected void onResume() {  
128     super.onResume();  
129     itemList.clear();  
130     getAllData();  
131 }
```

- ❖ Lanjutan program MainActivity.java
- ❖ Buat fungsi dengan nama getAllData dan onResume
- ❖ Fungsi ini digunakan untuk mengambil semua data yang ada pada database.

Aplikasi Database SQLite

- ❖ Cara mengatasi blok program yang masih error atau yang berwarna merah dengan cara tekan Alt+Enter pada blok yang berwarna merah.

AplikasiSQLite [C:\Users\I Komang Sugiarta\AndroidStudioProjects\AplikasiSQLite] - ...app\src\main\java\com\sugiarta\aplikasisqlite\MainActivity.java [app] - Ar

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

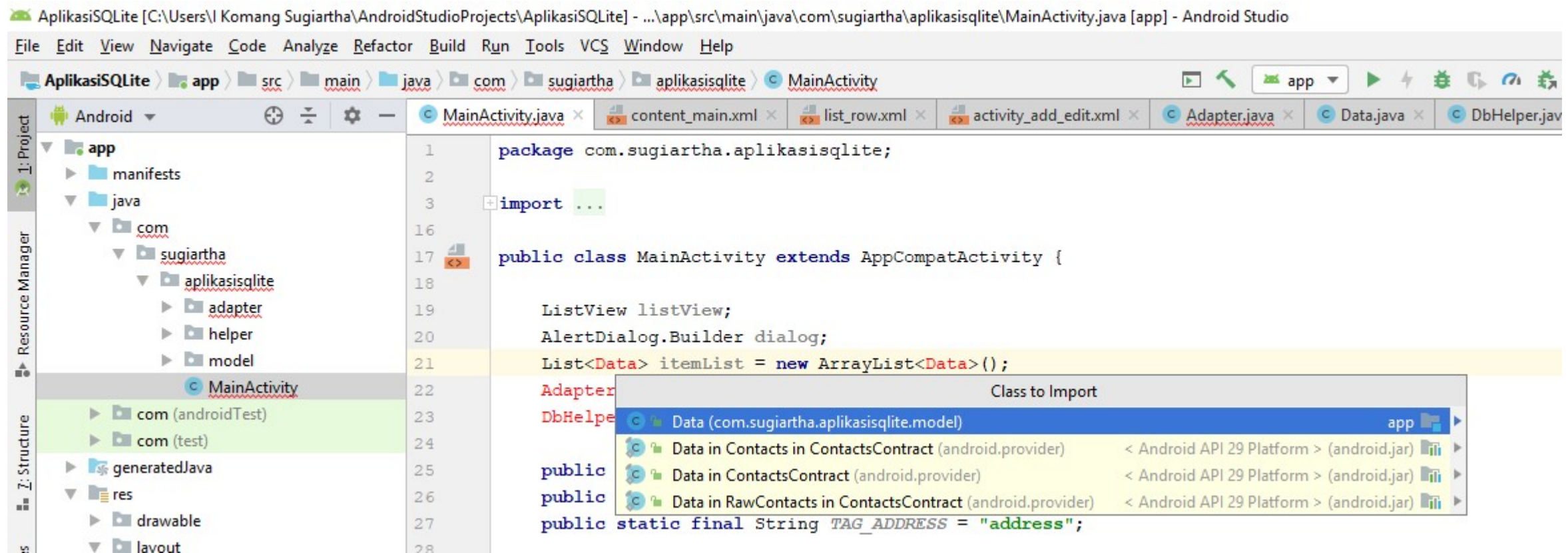
AplikasiSQLite > app > src > main > java > com > sugiarta > aplikasisqlite > MainActivity

Android > 1: Project > Resource Manager > 2: Structure

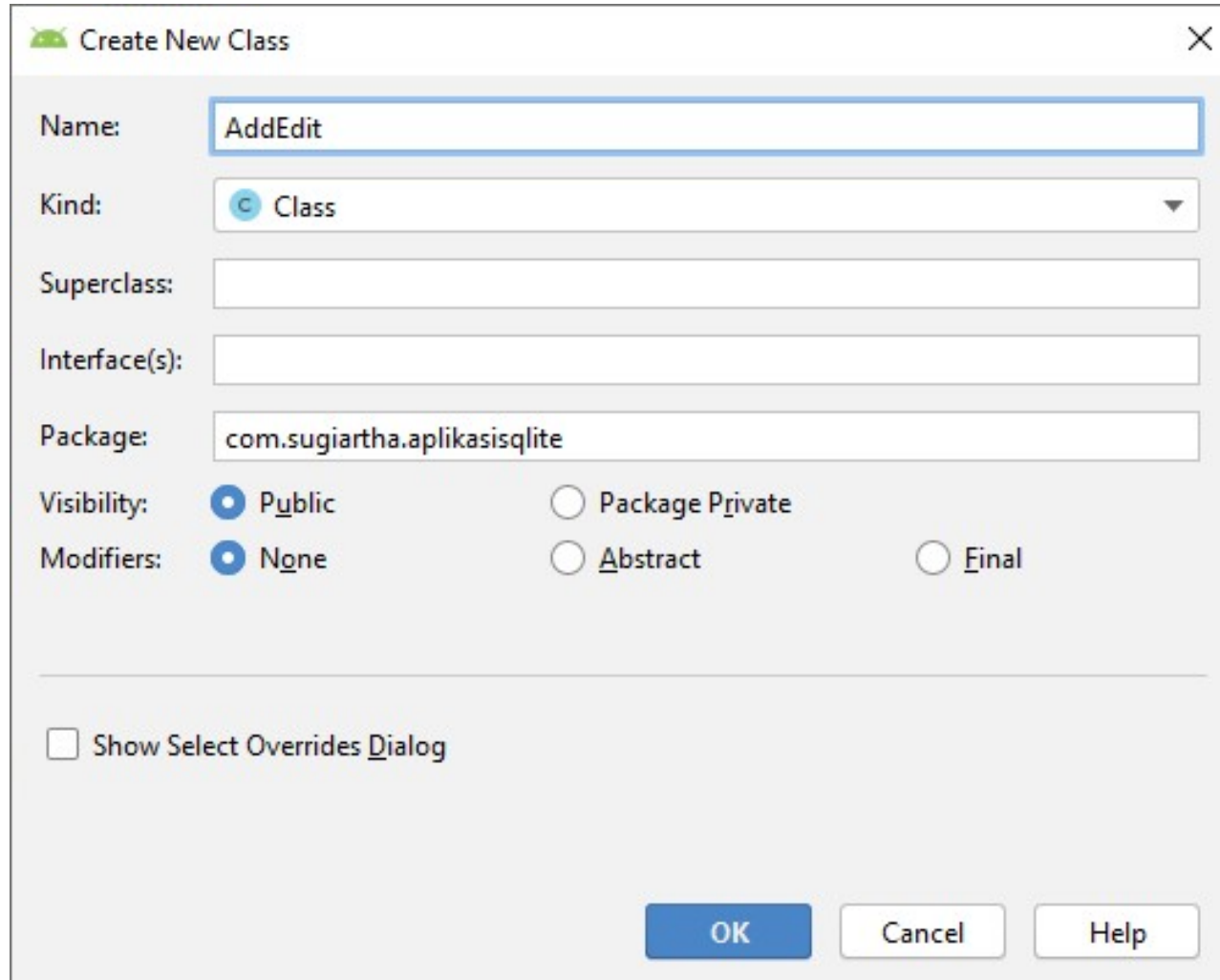
```
1 package com.sugiarta.aplikasisqlite;
2
3 import ...
14
15 ?android.widget.ListView? Alt+Enter activity extends AppCompatActivity {
16
17 ListView listView;
18 AlertDialog.Builder dialog;
19 List<Data> itemList = new ArrayList<Data>();
20 Adapter adapter;
21 DBHelper SQLite = new DBHelper(this);
22
23 public static final String TAG_ID = "id";
24 public static final String TAG_NAME = "name";
25 public static final String TAG_ADDRESS = "address";
26
```


Aplikasi Database SQLite

- ❖ Pilih Import Class.
- ❖ Jika muncul option seperti pada gambar dibawah, pilih nama class yang dibuat sebelumnya dan nama packagenya (Data(com.sugiartha.aplikasisqlite.model))



Aplikasi Database SQLite



Create New Class

Name: AddEdit

Kind: Class

Superclass:

Interface(s):

Package: com.sugiartha.aplikasisqlite

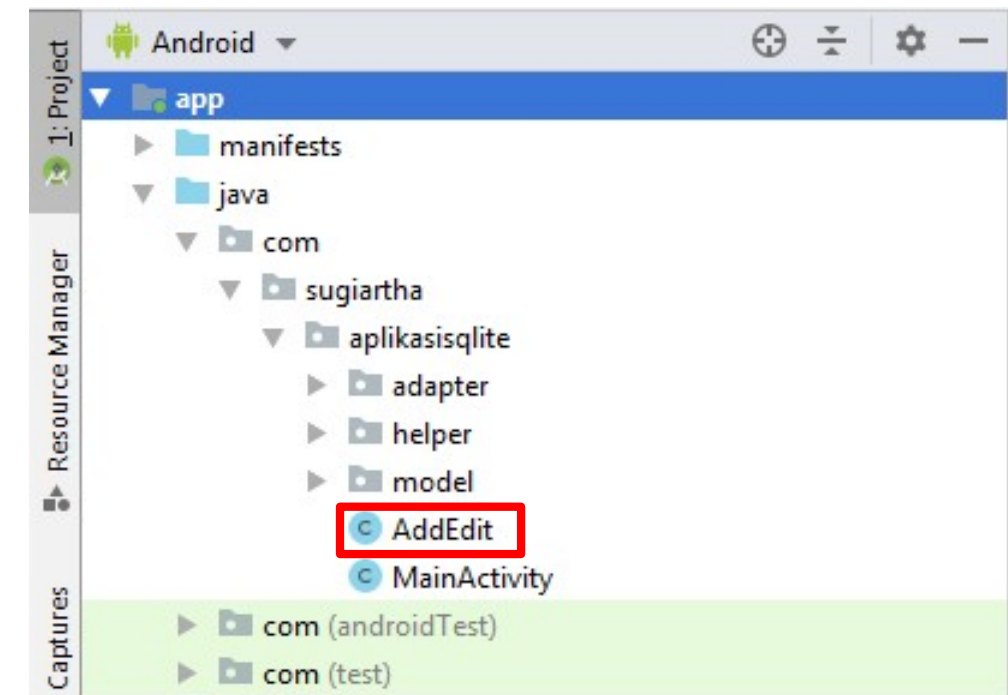
Visibility: ☒ Public ☐ Package Private

Modifiers: ☒ None ☐ Abstract ☐ Final

☐ Show Select Overrides Dialog

OK Cancel Help

- ❖ Buat class baru didalam package utama.
- ❖ Isi Name dengan nama AddEdit.
- ❖ Seperti pada gambar berikut.



Aplikasi Database SQLite

```
15 public class AddEdit extends AppCompatActivity {  
16     EditText txt_id, txt_name, txt_address;  
17     Button btn_submit, btn_cancel;  
18     DBHelper SQLite = new DBHelper(context: this);  
19     String id, name, address;
```

- ❖ Buka file AddEdit.java
- ❖ Ketik program seperti pada gambar di samping ini.
- ❖ Blok program ini deklarasi variable.

Aplikasi Database SQLite

```
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_add_edit);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);

    txt_id = (EditText) findViewById(R.id.txt_id);
    txt_name = (EditText) findViewById(R.id.txt_name);
    txt_address = (EditText) findViewById(R.id.txt_address);
    btn_submit = (Button) findViewById(R.id.btn_submit);
    btn_cancel = (Button) findViewById(R.id.btn_cancel);

    id = getIntent().getStringExtra(MainActivity.TAG_ID);
    name = getIntent().getStringExtra(MainActivity.TAG_NAME);
    address = getIntent().getStringExtra(MainActivity.TAG_ADDRESS);

    if (id == null || id == "") {
        setTitle("Add Data");
    } else {
        setTitle("Edit Data");
        txt_id.setText(id);
        txt_name.setText(name);
        txt_address.setText(address);
    }
}
```

- ❖ Lanjutan program AddEdit.java
- ❖ Blok program ini adalah untuk memanggil variable berdasarkan id.
- ❖ Class ini berfungsi untuk menampilkan EditText nama dan alamat.
- ❖ Dan membuat kondisi Add Data dan Edit Data.

Aplikasi Database SQLite

```
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
    btn_submit.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            try {  
                if (txt_id.getText().toString().equals("")) {  
                    save();  
                } else {  
                    edit();  
                }  
            } catch (Exception e){  
                Log.e( tag: "Submit", e.toString());  
            }  
        }  
    });  
  
    btn_cancel.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            blank();  
            finish();  
        }  
    });  
}
```

- ❖ Lanjutan program AddEdit.java
- ❖ Blok program ini digunakan untuk membuat fungsi pada Button Submit dan Button Cancel.

Aplikasi Database SQLite

```
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92

@Override
public void onBackPressed() {
    finish();
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case android.R.id.home:
            blank();
            this.finish();
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}

// Kosongkan semua Edit Teks
private void blank() {
    txt_name.requestFocus();
    txt_id.setText(null);
    txt_name.setText(null);
    txt_address.setText(null);
}
```

- ❖ Lanjutan program AddEdit.java
- ❖ Blok program ini digunakan untuk memilih item yang dipilih dan mengosongkan semua Edit Text.

Aplikasi Database SQLite

- ❖ Lanjutan program pada AddEdit.java.
- ❖ Didalam class ini terdapat fungsi untuk menyimpan data ke SQLite database.

```
94 // Menyimpan Data ke Database SQLite
95 private void save() {
96     if (String.valueOf(txt_name.getText()).equals(null) || String.valueOf(txt_name.getText()).equals("") ||
97         String.valueOf(txt_address.getText()).equals(null) || String.valueOf(txt_address.getText()).equals("")) {
98         Toast.makeText(getApplicationContext(),
99             text: "Please input name or address ...", Toast.LENGTH_SHORT).show();
100     } else {
101         SQLite.insert(txt_name.getText().toString().trim(), txt_address.getText().toString().trim());
102         blank();
103         finish();
104     }
105 }
```


Aplikasi Database SQLite

- ❖ Blok program terakhir pada AddEdit.java.
- ❖ Didalam class ini terdapat fungsi untuk megupdate data ke SQLite database.

```
1106 // Update data kedalam Database SQLite
1107 private void edit() {
1108     if (String.valueOf(txt_name.getText()).equals(null) || String.valueOf(txt_name.getText()).equals("") ||
1109         String.valueOf(txt_address.getText()).equals(null) || String.valueOf(txt_address.getText()).equals("")) {
1110         Toast.makeText(getApplicationContext(),
1111             text: "Please input name or address ...", Toast.LENGTH_SHORT).show();
1112     } else {
1113         SQLite.update(Integer.parseInt(txt_id.getText().toString().trim()), txt_name.getText().toString().trim(),
1114             txt_address.getText().toString().trim());
1115         blank();
1116         finish();
1117     }
1118 }
1119 }
```

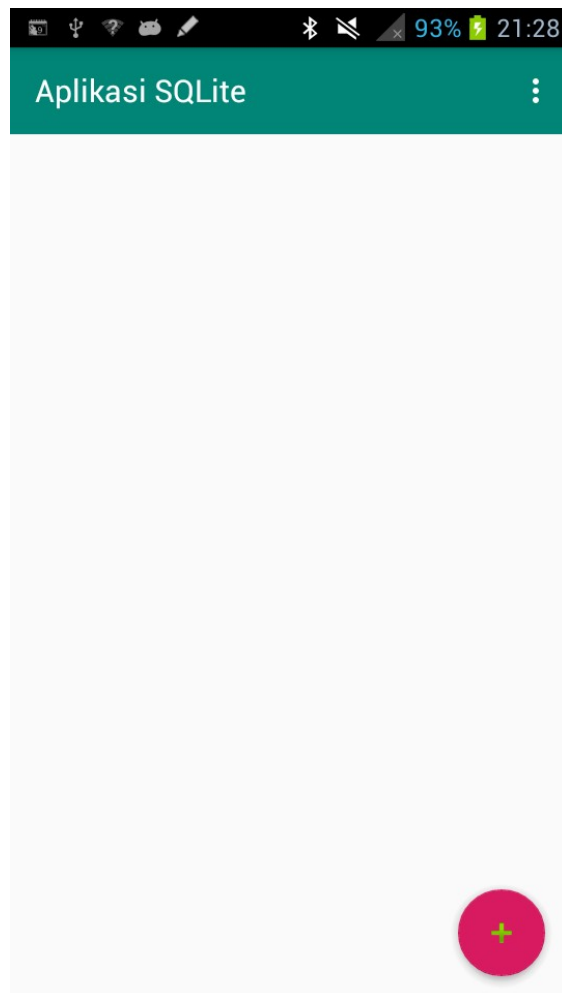

Aplikasi Database SQLite

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="com.sugiartha.aplikasisqlite">
4
5      <application
6          android:allowBackup="true"
7          android:icon="@mipmap/ic_launcher"
8          android:label="Aplikasi SQLite"
9          android:roundIcon="@mipmap/ic_launcher_round"
10         android:supportRtl="true"
11         android:theme="@style/AppTheme">
12         <activity android:name=".AddEdit"></activity>
13         <activity
14             android:name=".MainActivity"
15             android:label="Aplikasi SQLite"
16             android:theme="@style/AppTheme.NoActionBar">
17             <intent-filter>
18                 <action android:name="android.intent.action.MAIN" />
19
20                 <category android:name="android.intent.category.LAUNCHER" />
21             </intent-filter>
22         </activity>
23     </application>
24
25 </manifest>
```

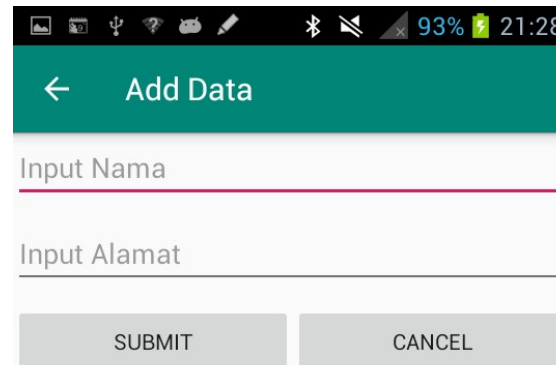
- ❖ Langkah terakhir adalah menambahkan activity AddEdit kedalam AndroidManifest.xml
- ❖ Seperti pada gambar disamping.

Aplikasi Database SQLite

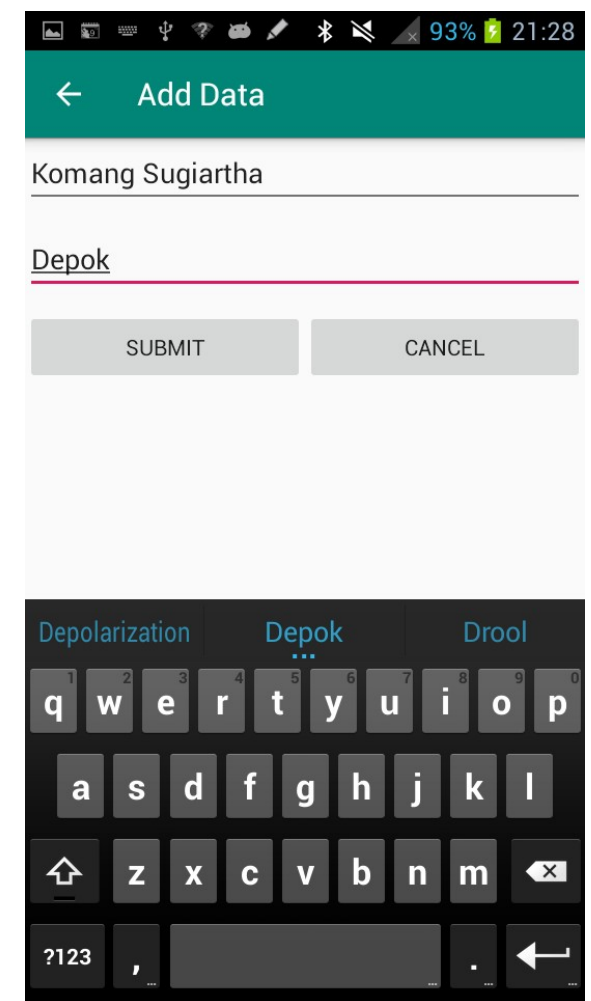
Tampilan Awal



Tampilan Add Data



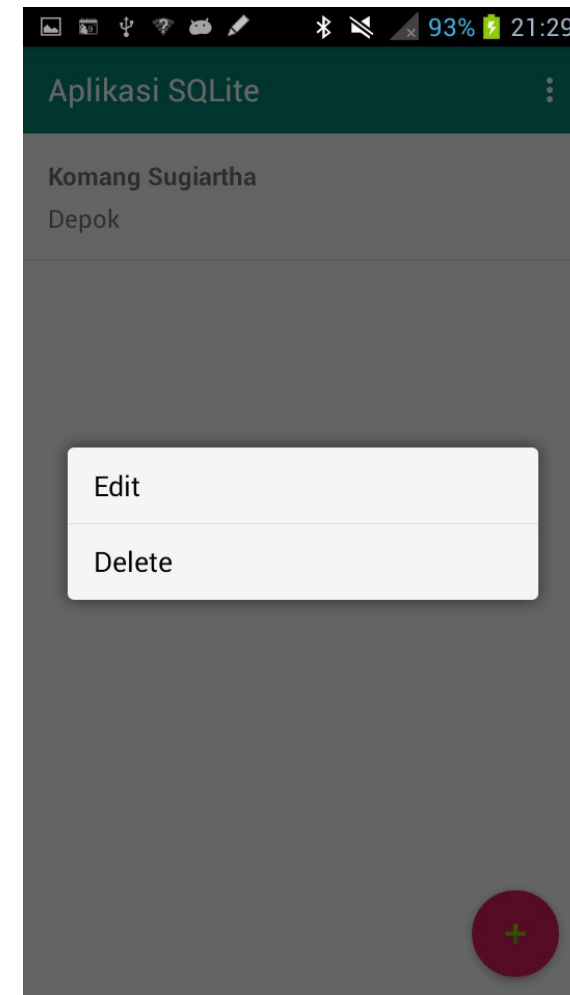
Tampilan Input Data



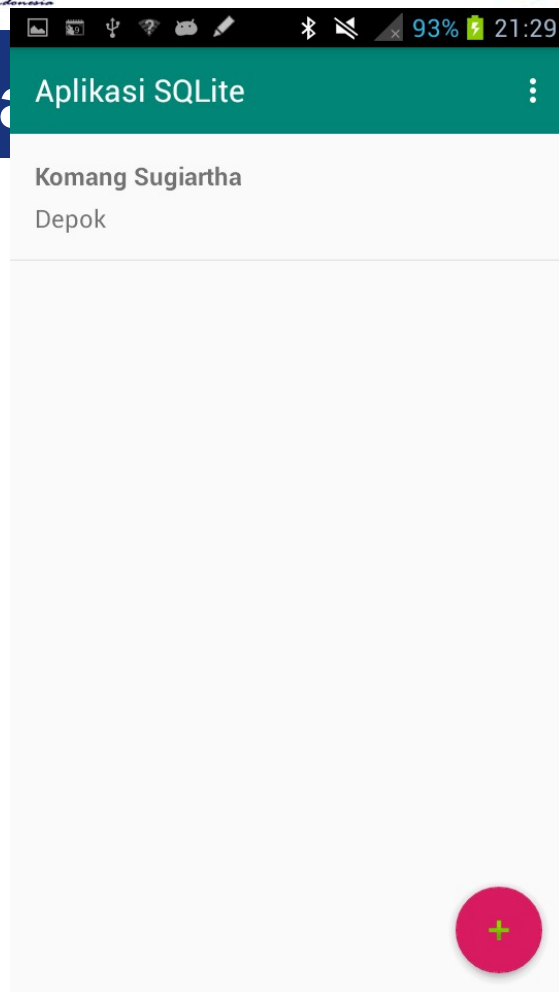
Aplikasi Database SQLite

Data berhasil disimpan dan ditampilkan pada List View

Tekan lama pada daftar List View maka option Edit dan Delete akan muncul

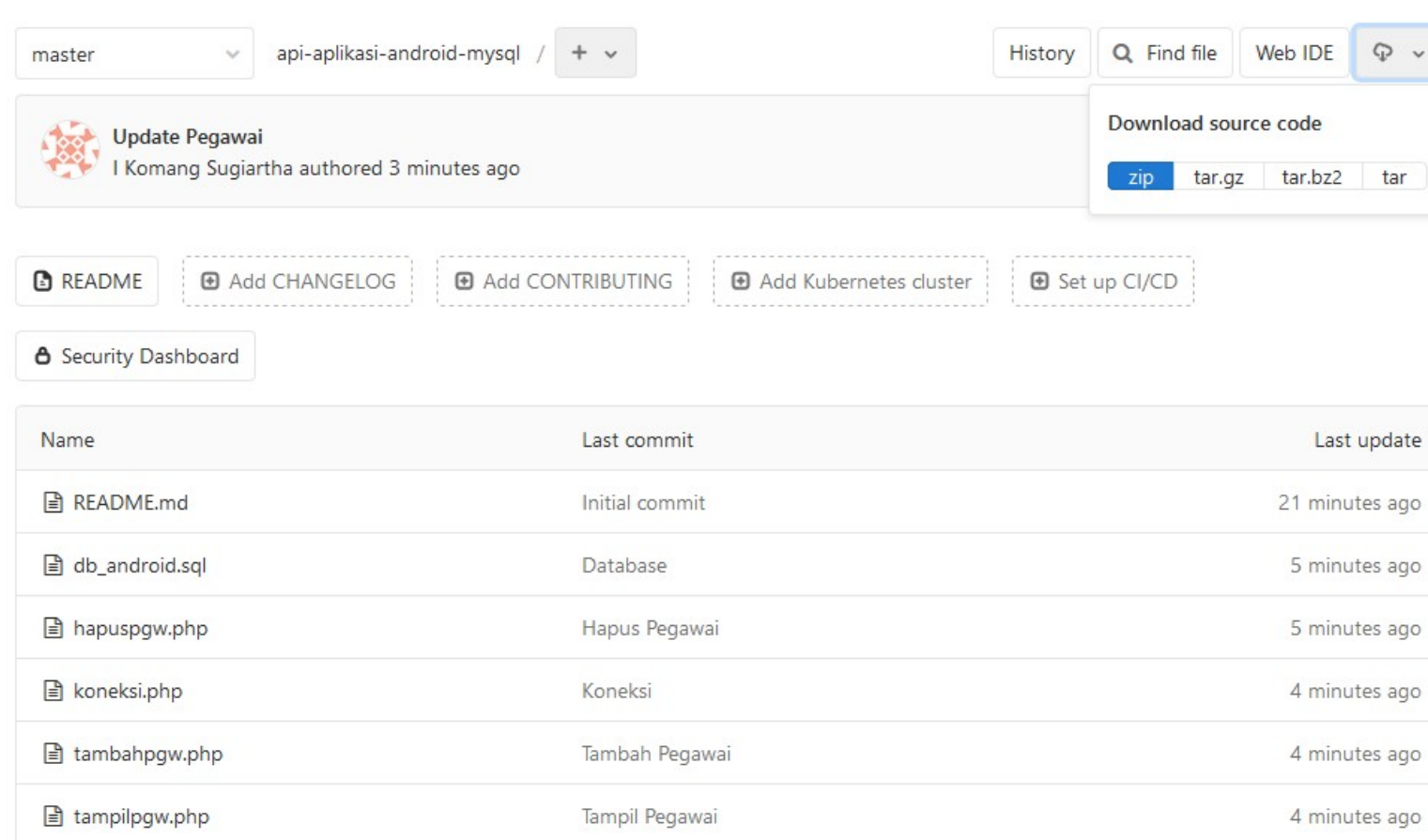


Aplikasi SQLite



Backend API Aplikasi Android

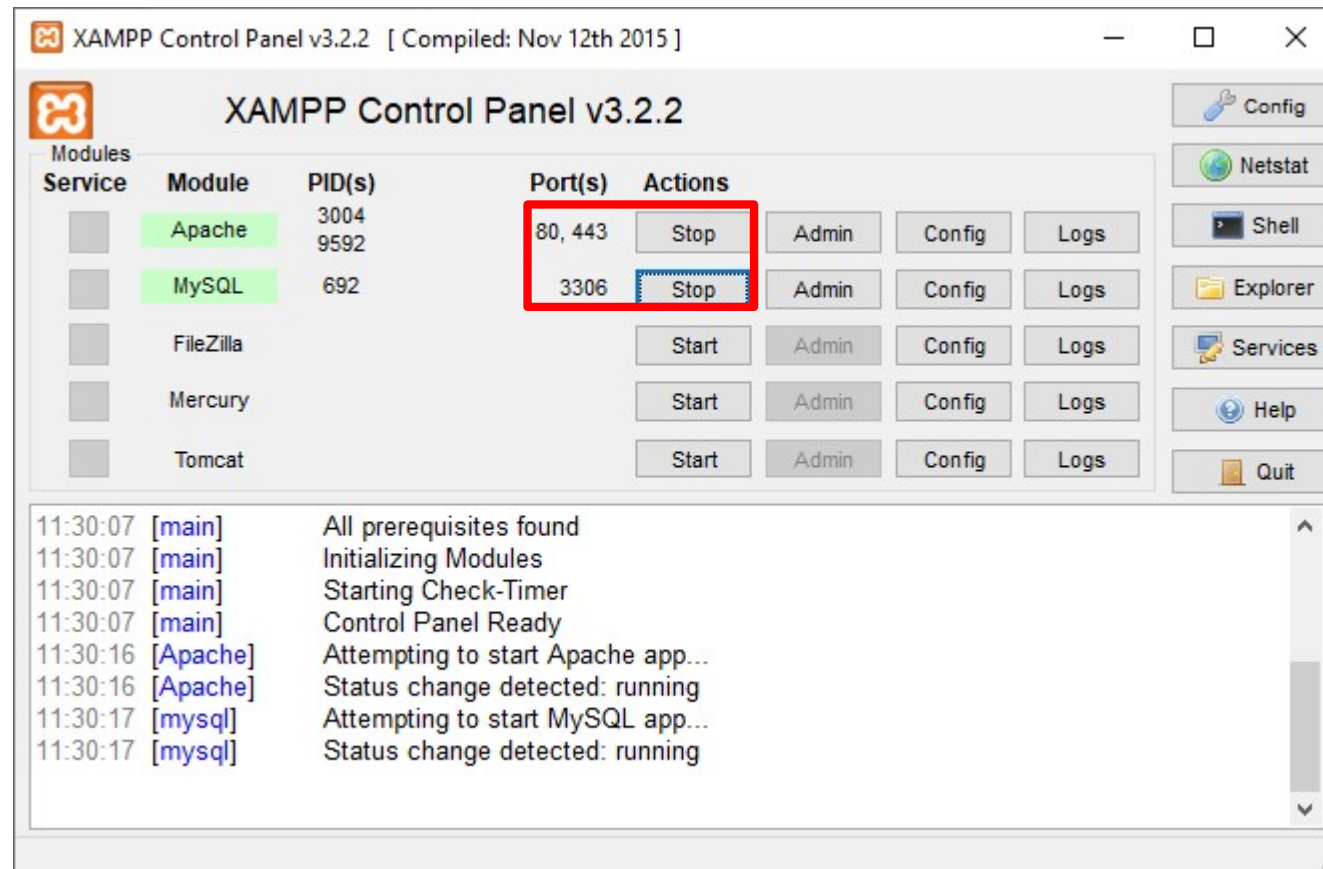
- ❖ Download backend API melalui link berikut : <https://gitlab.com/sugiartha/api-aplikasi-android-mysql>



The screenshot shows the GitLab repository page for 'api-aplikasi-android-mysql'. The 'Download source code' dropdown menu is open, showing options: zip, tar.gz, tar.bz2, and tar. A blue arrow points to the dropdown button. Below the dropdown, there are buttons for README, Add CHANGELOG, Add CONTRIBUTING, Add Kubernetes cluster, and Set up CI/CD. A Security Dashboard button is also visible. A table lists the repository files and their last commit details.

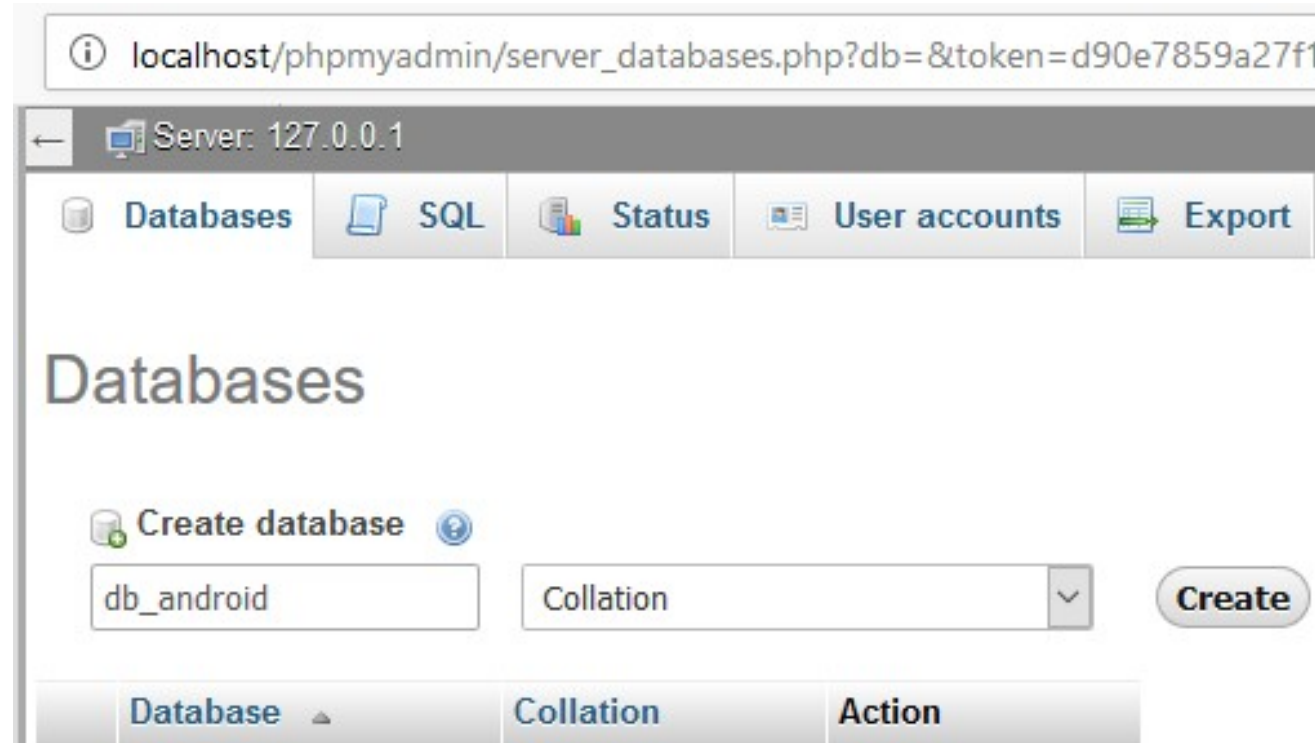
Name	Last commit	Last update
README.md	Initial commit	21 minutes ago
db_android.sql	Database	5 minutes ago
hapuspgw.php	Hapus Pegawai	5 minutes ago
koneksi.php	Koneksi	4 minutes ago
tambahpgw.php	Tambah Pegawai	4 minutes ago
tampilpgw.php	Tampil Pegawai	4 minutes ago

Backend API Aplikasi Android



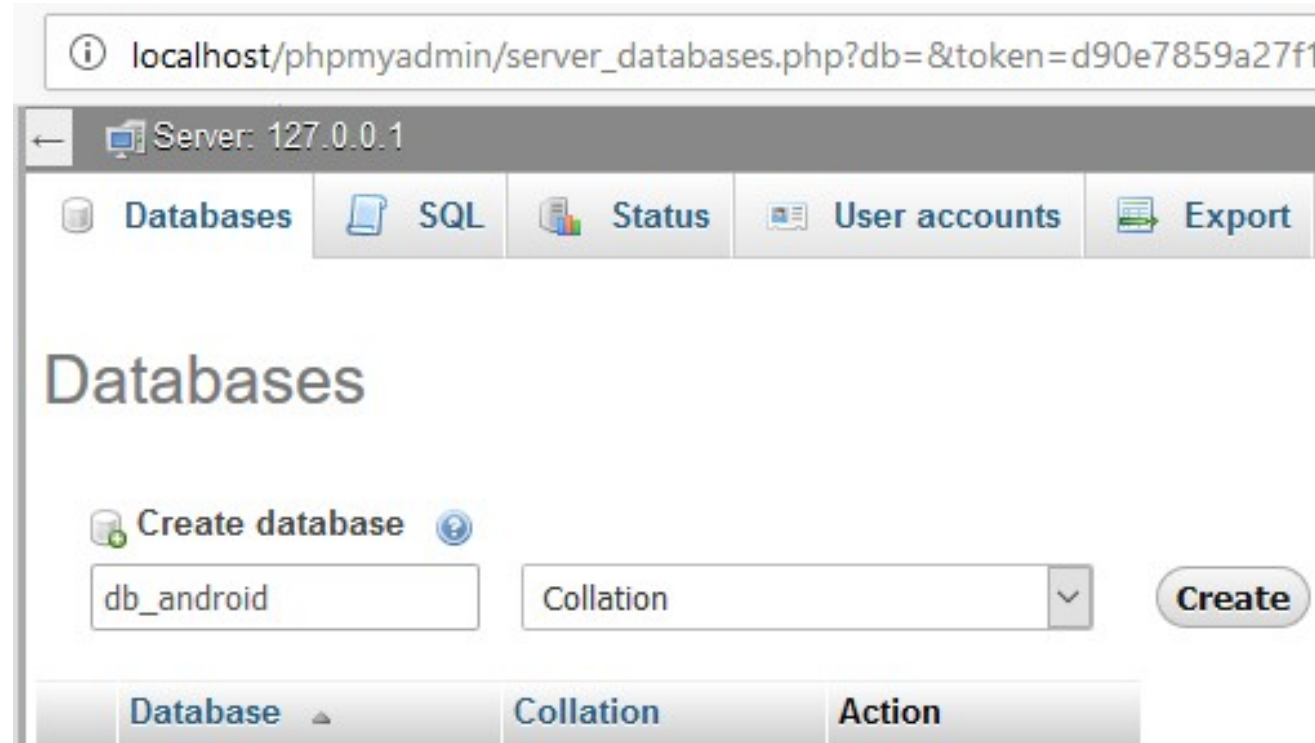
- ❖ Buka aplikasi XAMPP
- ❖ Start Apache dan MySQL

Backend API Aplikasi Android



- ❖ Buka web browser (Firefox, Chrome, dll)
- ❖ Ketik di URL : <http://localhost/phpmyadmin>
- ❖ Buat database dengan nama db_android

Backend API Aplikasi Android



- ❖ Buka web browser (Firefox, Chrome, dll)
- ❖ Ketik di URL : <http://localhost/phpmyadmin>
- ❖ Buat database dengan nama db_android
- ❖ Buka db_android kemudian import database yang ada di folder pegawai.

Backend API Aplikasi Android

❖ Berikut cara pemanggilan API pada Aplikasi Android

```
public static final String URL_ADD="http://192.168.1.102/pegawai/tambahpgw.php";  
public static final String URL_GET_ALL = "http://192.168.1.102/pegawai/tampilsemuapgw.php";  
public static final String URL_GET_EMP = "http://192.168.1.102/pegawai/tampilpgw.php?id=";  
public static final String URL_UPDATE_EMP = "http://192.168.1.102/pegawai/updatepgw.php";  
public static final String URL_DELETE_EMP = "http://192.168.1.102/pegawai/hapuspgw.php?id=";
```

❖ IP disesuaikan dengan IP pada computer/laptop masing-masing peserta.

Backend API Aplikasi Android

- ❖ Cara mengecek IP dengan cara buka CMD dan ketik IPCONFIG

```
C:\Windows\system32\cmd.exe

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::e0ca:f92f:eb87:3668%19
    IPv4 Address. . . . . : 10.10.3.79
    Subnet Mask . . . . . : 255.255.248.0
    Default Gateway . . . . . : 10.10.0.1

Ethernet adapter Bluetooth Network Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : 

C:\Users\I Komang Sugiarta>
```


< Topik_Silabus >

Pelatihan

Kesimpulan Pertemuan

1. <Kesimpulan materi 1>
2. <Kesimpulan materi 2>
3. <Kesimpulan materi 3>
4. <dst>

Referensi:

1. “ “, Java™ Programming Language, Oracle America
2. Android Cook Book, McGraw-Hill/Osborne, 2013
3. Herbert Schildt, *Java2 : A beginner's Guide*, Second Edition, McGraw-Hill/Osborne
4. Matthew Mathias, Swift Programming, 2nd edition, Big Nerd Ranch
5. <https://developer.apple.com/library/archive/referencelibrary/GettingStarted/DevelopiOSAppsSwift/index.html/>
6. <https://developer.android.com/topic/libraries/architecture>
7. <https://www.oracle.com/technetwork/java/javase/overview/index.html>

Tim Penyusun:

- Alif Akbar Fitrawan, S.Pd, M. Kom (Politeknik Negeri Banyuwangi);
- Anwar, S.Si, MCs. (Politeknik Negeri Lhokseumawe);
- Eddo Fajar Nugroho (BPPTIK Cikarang);
- Eddy Tungadi, S.T., M.T. (Politeknik Negeri Ujung Pandang);
- Fitri Wibowo (Politeknik Negeri Pontianak);
- Ghifari Munawar (Politeknik Negeri Bandung);
- Hetty Meileni, S.Kom., M.T. (Politeknik Negeri Sriwijaya) ;
- I Wayan Candra Winetra, S.Kom., M.Kom (Politeknik Negeri Bali) ;
- Irkham Huda (Vokasi UGM) ;
- Josseano Amakora Koli Parera, S.Kom., M.T. (Politeknik Negeri Ambon) ;
- I Komang Sugiarta, S.Kom., MMSI (Universitas Gunadarma) ;
- Lucia Sri Istiyowati, M.Kom (Institut Perbanas) ;
- Maksy Sendiang, ST, MIT (Politeknik Negeri Manado) ;
- Medi Noviana (Universitas Gunadarma) ;
- Muhammad Nashrullah (Politeknik Negeri Batam) ;
- Nat. I Made Wiryana, S.Si., S.Kom., M.Sc. (Universitas Gunadarma) ;
- Rika Idmayanti, ST, M.Kom (Politeknik Negeri Padang) ;
- Rizky Yuniar Hakkun (Politeknik Elektronik Negeri Surabaya) ;
- Robinson A.Wadu, ST., MT (Politeknik Negeri Kupang) ;
- Roslina. M.IT (Politeknik Negeri Medan) ;
- Sukamto, SKom., MT. (Politeknik Negeri Semarang) ;
- Syamsi Dwi Cahya, M.Kom. (Politeknik Negeri Jakarta) ;
- Syamsul Arifin, S.Kom, M.Cs (Politeknik Negeri Jember) ;
- Usmanudin (Universitas Gunadarma) ;
- Wandy Alifha Saputra (Politeknik Negeri Banjarmasin) ;