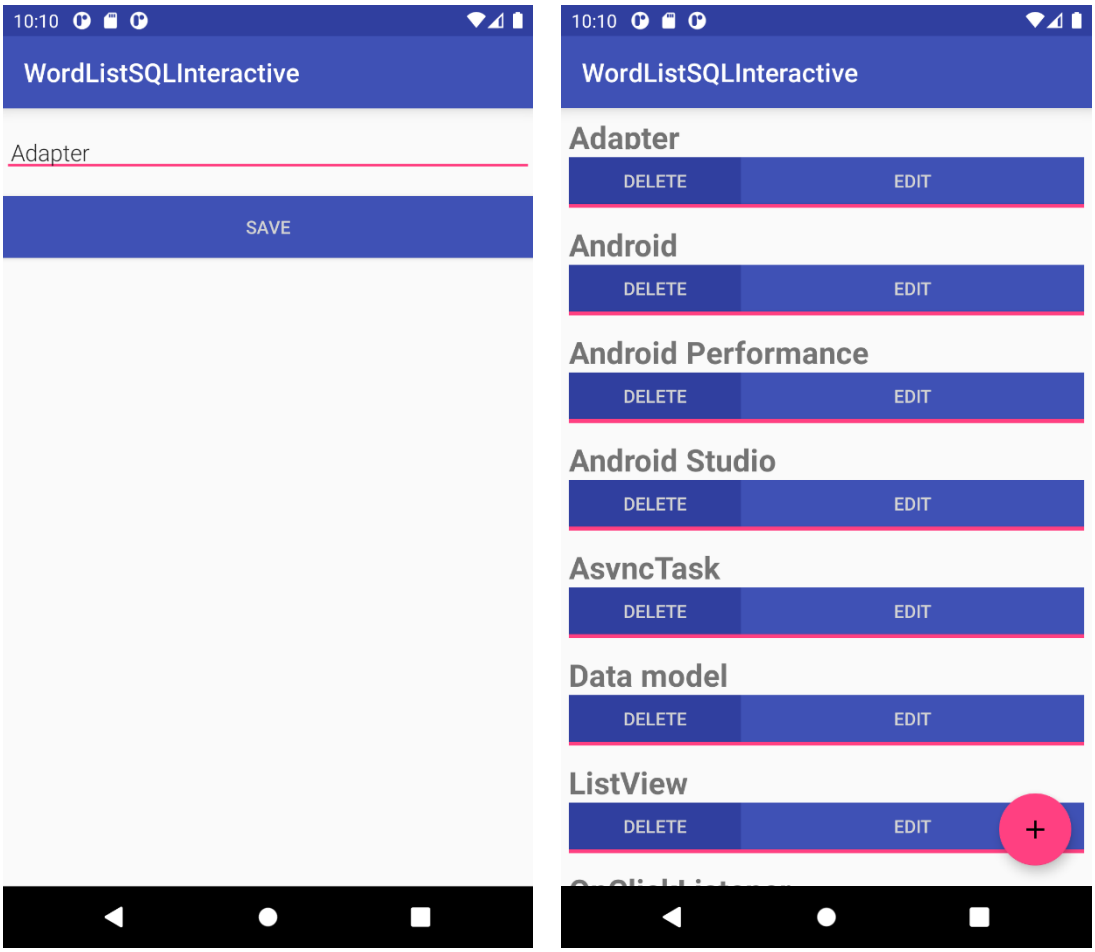
Matkul : **Pemrograman Perangkat Bergerak - TI – S1**

NIM : **A11.2019.11688**

Nama : **Bayu Prasetya Adji Sugiyarto**

Pertemuan : **8**

Word List SQL finished



String.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">WordListSQLInteractive</string>
    <string name="hint_word">Word...</string>
    <string name="hint_definition">Definition...</string>
    <string name="button_save">Save</string>
    <string name="button_new">New</string>
    <string name="button_edit">Edit</string>
    <string name="button_delete">Delete</string>
```

```
    <string name="empty_not_saved">Word not saved because it is empty.</string>
</resources>
```

Layout

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <android.support.v7.widget.RecyclerView
        android:id="@+id/recyclerview"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
    </android.support.v7.widget.RecyclerView>

    <android.support.design.widget.FloatingActionButton
        android:id="@+id/fab"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|end"
        android:layout_margin="16dp"
        android:clickable="true"
        android:src="@drawable/ic_add_24dp" />

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

Activity_edit_word.xml

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/edit_word"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="sans-serif-light"
        android:hint="@string/hint_word"
        android:inputType="textAutoComplete"
        android:padding="@dimen/small_padding"
        android:layout_marginBottom="@dimen/big_padding"
        android:layout_marginTop="@dimen/big_padding"
        android:textSize="18sp" />

    <Button
        android:id="@+id/button_save"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/colorPrimary"
        android:onClick="returnReply"
        android:text="@string/button_save"
        android:textColor="@color/buttonLabel" />
```

```
</LinearLayout>
```

Wordlist_item

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="6dp">

    <TextView
        android:id="@+id/word"
        android:layout_width="match_parent"
        style="@style/word_title" />

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

        <Button
            android:id="@+id/delete_button"
            android:layout_width="match_parent"
            android:layout_height="@dimen/button_height"
            android:layout_weight="2"
            android:background="@color/colorPrimaryDark"
            android:text="@string/button_delete"
            android:textColor="@color/buttonLabel"/>

        <Button
            android:id="@+id/edit_button"
            android:layout_width="match_parent"
            android:layout_height="@dimen/button_height"
            android:layout_weight="1"
            android:background="@color/colorPrimary"
            android:text="@string/button_edit"
            android:textColor="@color/buttonLabel"/>

    </LinearLayout>

    <Button
        android:layout_width="match_parent"
        android:layout_height="@dimen/divider_height"
        android:background="@color/colorAccent" />

</LinearLayout>
```

Java

Mainactivity.java

```
package com.android.example.wordlistsql;

import android.content.Intent;
import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.LinearLayoutManager;
```

```
import android.support.v7.widget.RecyclerView;
import android.text.TextUtils;
import android.view.View;
import android.widget.Toast;


public class MainActivity extends AppCompatActivity {

    private static final String TAG = MainActivity.class.getSimpleName();

    public static final int WORD_EDIT = 1;
    public static final int WORD_ADD = -1;

    private WordListOpenHelper mDB;
    private RecyclerView mRecyclerView;
    private WordListAdapter mAdapter;
    private int mLastPosition;

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

        mDB = new WordListOpenHelper(this);

        // Create recycler view.
        mRecyclerView = (RecyclerView) findViewById(R.id.recyclerview);
        // Create an mAdapter and supply the data to be displayed.
        mAdapter = new WordListAdapter(this, /* mDB.getAllEntries(),*/ mDB);
        // Connect the mAdapter with the recycler view.
        mRecyclerView.setAdapter(mAdapter);
        // Give the recycler view a default layout manager.
        mRecyclerView.setLayoutManager(new LinearLayoutManager(this));

        // Add a floating action click handler for creating new entries.
        FloatingActionButton fab = (FloatingActionButton)
findViewById(R.id.fab);
        fab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                // Starts empty edit activity.
                Intent intent = new Intent(getBaseContext(),
EditWordActivity.class);
                startActivityForResult(intent, WORD_EDIT);
            }
        });
    }

    public void onActivityResult(int requestCode, int resultCode, Intent data)
{
        super.onActivityResult(requestCode, resultCode, data);

        if (requestCode == WORD_EDIT) {
            if (resultCode == RESULT_OK) {
                String word =
data.getStringExtra(EditWordActivity.EXTRA_REPLY);

                // Update the database.
                if (!TextUtils.isEmpty(word)) {
                    int id = data.getIntExtra(WordListAdapter.EXTRA_ID, -99);

                    if (id == WORD_ADD) {
                        mDB.insert(word);
                    } else if (id >= 0) {
```

```
                        mDB.update(id, word);
                    }
                    // Update the UI.
                    mAdapter.notifyDataSetChanged();
                } else {
                    Toast.makeText(
                            getApplicationContext(),
                            R.string.empty_not_saved,
                            Toast.LENGTH_LONG).show();
                }
            }
        }
    }
}
```

EditWordActivity.java

```
package com.android.example.wordlistsql;

import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.EditText;

public class EditWordActivity extends AppCompatActivity {

    private static final String TAG = EditWordActivity.class.getSimpleName();

    private static final int NO_ID = -99;
    private static final String NO_WORD = "";

    private EditText mEditWordView;

    // Unique tag for the intent reply.
    public static final String EXTRA_REPLY =
"com.example.android.wordlistsql.REPLY";

    int mId = MainActivity.WORD_ADD;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_edit_word);

        mEditWordView = (EditText) findViewById(R.id.edit_word);

        // Get data sent from calling activity.
        Bundle extras = getIntent().getExtras();

        // If we are passed content, fill it in for the user to edit.
        if (extras != null) {
            int id = extras.getInt(WordListAdapter.EXTRA_ID, NO_ID);
            String word = extras.getString(WordListAdapter.EXTRA_WORD,
NO_WORD);
            if ((id != NO_ID) && (word != NO_WORD)) {
                mId = id;
                mEditWordView.setText(word);
            }
        } // Otherwise, start with empty fields.
    }

    public void returnReply(View view) {
```

```
        String word = ((EditText)
findViewById(R.id.edit_word)).getText().toString();

        Intent replyIntent = new Intent();
        replyIntent.putExtra(EXTRA_REPLY, word);
        replyIntent.putExtra(WordListAdapter.EXTRA_ID, mId);
        setResult(RESULT_OK, replyIntent);
        finish();
    }
}
```

WordItem.java

```
package com.android.example.wordlistsql;


public class WordItem {

    private int mId;
    private String mWord;

    public WordItem() {}

    public int getId() {
        return this.mId;
    }

    public String getWord() {
        return this.mWord;
    }

    public void setId(int id) {
        this.mId = id;
    }

    public void setWord(String word) {
        this.mWord = word;
    }
}
```

WordListAdapter.java

```
package com.android.example.wordlistsql;

import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.support.v7.widget.RecyclerView;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.TextView;


public class WordListAdapter extends
RecyclerView.Adapter<WordListAdapter.WordViewHolder> {

    class WordViewHolder extends RecyclerView.ViewHolder {
        public final TextView wordItemView;
        Button delete_button;
        Button edit_button;
```

```
        public WordViewHolder(View itemView) {
            super(itemView);
            wordItemView = (TextView) itemView.findViewById(R.id.word);
            delete_button = (Button)itemView.findViewById(R.id.delete_button);
            edit_button = (Button)itemView.findViewById(R.id.edit_button);
        }
    }

    private static final String TAG = WordListAdapter.class.getSimpleName();

    public static final String EXTRA_ID = "ID";
    public static final String EXTRA_WORD = "WORD";
    public static final String EXTRA_POSITION = "POSITION";

    private final LayoutInflater mInflater;
    WordListOpenHelper mDB;
    Context mContext;

    public WordListAdapter(Context context, WordListOpenHelper db) {
        mInflater = LayoutInflater.from(context);
        mContext = context;
        mDB = db;
    }

    @Override
    public WordViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        View itemView = mInflater.inflate(R.layout.wordlist_item, parent,
false);
        return new WordViewHolder(itemView);
    }

    @Override
    public void onBindViewHolder(WordViewHolder holder, int position) {
        WordItem current = mDB.query(position);
        holder.wordItemView.setText(current.getWord());
        // Keep a reference to the view holder for the click listener
        final WordViewHolder h = holder; // needs to be final for use in
callback
        // Attach a click listener to the DELETE button.
        holder.delete_button.setOnClickListener(new MyButtonOnClickListener(
                current.getId(), null)  {


            @Override
            public void onClick(View v ) {
                // You have to get the position like this, you can't hold a
reference
                Log.d (TAG + "onClick", "VHPos " + h.getAdapterPosition() + "
ID " + id);
                int deleted = mDB.delete(id);
                if (deleted >= 0)
                    notifyItemRemoved(h.getAdapterPosition());
            }
        });

        // Attach a click listener to the EDIT button.
        holder.edit_button.setOnClickListener(new MyButtonOnClickListener(
                current.getId(), current.getWord()) {

            @Override
            public void onClick(View v) {
                Intent intent = new Intent(mContext, EditWordActivity.class);
```

```
                intent.putExtra(EXTRA_ID, id);
                intent.putExtra(EXTRA_POSITION, h.getAdapterPosition());
                intent.putExtra(EXTRA_WORD, word);

                // Start an empty edit activity.
                ((Activity) mContext).startActivityForResult(intent,
MainActivity.WORD_EDIT);
            }
        });
    }

    @Override
    public int getItemCount() {
        return (int) mDB.count();
    }
}
```

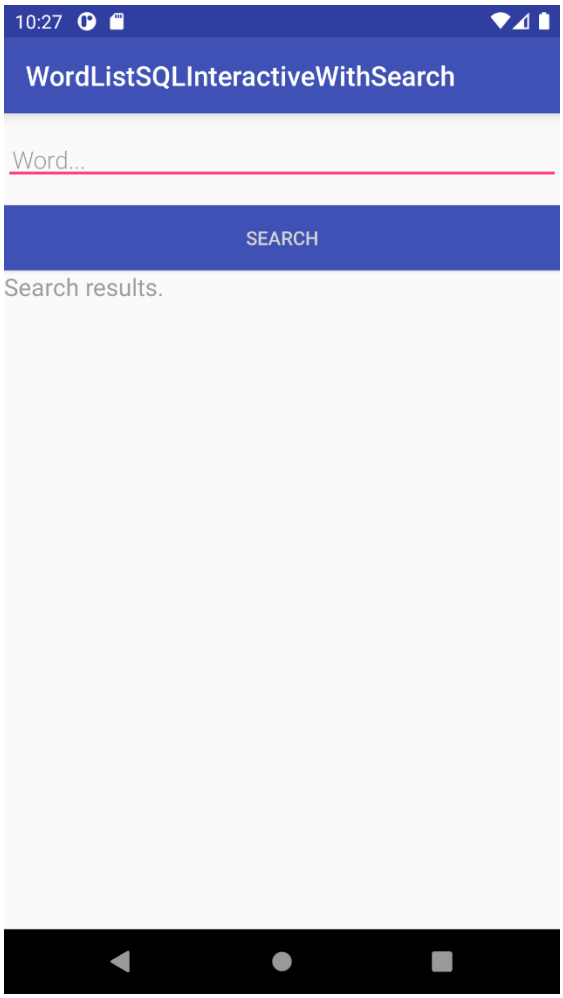
WordListOpenHelper.java

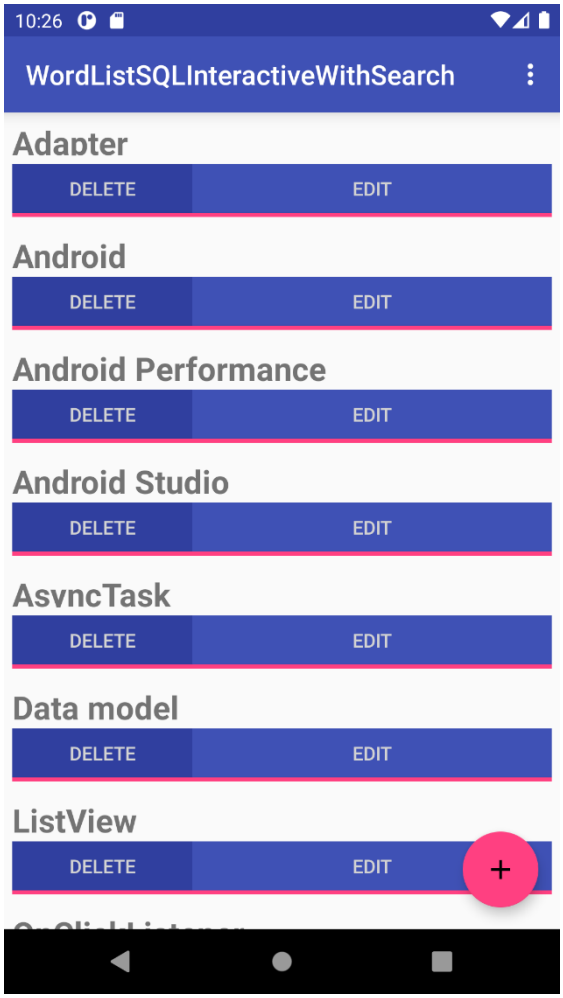
```
package com.android.example.wordlistsql;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.DatabaseUtils;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;

public class WordListOpenHelper extends SQLiteOpenHelper {

    private static final String TAG = WordListOpenHelper.class.getSimpleName();

    /

    private static final int DATABASE_VERSION = 1;
    private static final String WORD_LIST_TABLE = "word_entries";
    private static final String DATABASE_NAME = "wordlist";

    // Column names...
    public static final String KEY_ID = "_id";
    public static final String KEY_WORD = "word";

    // ... and a string array of columns.
    private static final String[] COLUMNS =
            {KEY_ID, KEY_WORD};

    // Build the SQL query that creates the table.
    private static final String WORD_LIST_TABLE_CREATE =
            "CREATE TABLE " + WORD_LIST_TABLE + " (" +
                    KEY_ID + " INTEGER PRIMARY KEY, " + // will auto-increment
if no value passed
                    KEY_WORD + " TEXT );";

    private SQLiteDatabase mWritableDB;
    private SQLiteDatabase mReadableDB;

    public WordListOpenHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
        Log.d(TAG, "Construct WordListOpenHelper");
    }

    @Override
```

```
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(WORD_LIST_TABLE_CREATE);
        fillDatabaseWithData(db);

    }

    public void fillDatabaseWithData(SQLiteDatabase db) {

        String[] words = {"Android", "Adapter", "ListView", "AsyncTask",
"Android Studio",
                "SQLiteDatabase", "SQLOpenHelper", "Data model", "ViewHolder",
                "Android Performance", "OnClickListener"};

        // Create a container for the data.
        ContentValues values = new ContentValues();

        for (int i=0; i < words.length;i++) {
            // Put column/value pairs into the container. put() overwrites
existing values.
            values.put(KEY_WORD, words[i]);
            db.insert(WORD_LIST_TABLE, null, values);
        }
    }

    public WordItem query(int position) {
        String query = "SELECT  * FROM " + WORD_LIST_TABLE +
                " ORDER BY " + KEY_WORD + " ASC " +
                "LIMIT " + position + ",1";

        Cursor cursor = null;
        WordItem entry = new WordItem();

        try {
            if (mReadableDB == null) {mReadableDB = getReadableDatabase();}
            cursor = mReadableDB.rawQuery(query, null);
            cursor.moveToFirst();
            entry.setId(cursor.getInt(cursor.getColumnIndex(KEY_ID)));
            entry.setWord(cursor.getString(cursor.getColumnIndex(KEY_WORD)));
        } catch (Exception e) {
            Log.d(TAG, "QUERY EXCEPTION! " + e.getMessage());
        } finally {
            // Must close cursor and db now that we are done with it.
            cursor.close();
            return entry;
        }
    }


    public long count() {
        if (mReadableDB == null) {mReadableDB = getReadableDatabase();}
        return DatabaseUtils.queryNumEntries(mReadableDB, WORD_LIST_TABLE);
    }


    public long insert(String word) {
        long newId = 0;
        ContentValues values = new ContentValues();
        values.put(KEY_WORD, word);
        try {
            if (mWritableDB == null) {mWritableDB = getWritableDatabase();}
            newId = mWritableDB.insert(WORD_LIST_TABLE, null, values);
        } catch (Exception e) {
            Log.d(TAG, "INSERT EXCEPTION! " + e.getMessage());
        }
```

```
        return newId;
    }


    public int update(int id, String word) {
        int mNumberOfRowsUpdated = -1;
        try {
            if (mWritableDB == null) {mWritableDB = getWritableDatabase();}
            ContentValues values = new ContentValues();
            values.put(KEY_WORD, word);

            mNumberOfRowsUpdated = mWritableDB.update(WORD_LIST_TABLE, //table
to change
                    values, // new values to insert
                    KEY_ID + " = ?", // selection criteria for row (in this
case, the _id column)
                    new String[]{String.valueOf(id)}); //selection args; the
actual value of the id

        } catch (Exception e) {
            Log.d (TAG, "UPDATE EXCEPTION! " + e.getMessage());
        }
        return mNumberOfRowsUpdated;
    }

    public int delete(int id) {
        int deleted = 0;
        try {
            if (mWritableDB == null) {mWritableDB = getWritableDatabase();}
            deleted = mWritableDB.delete(WORD_LIST_TABLE, //table name
                    KEY_ID + " = ? ", new String[]{String.valueOf(id)});
        } catch (Exception e) {
            Log.d (TAG, "DELETE EXCEPTION! " + e.getMessage());        }
        return deleted;
    }


    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        Log.w(WordListOpenHelper.class.getName(),
                "Upgrading database from version " + oldVersion + " to "
                        + newVersion + ", which will destroy all old data");
        db.execSQL("DROP TABLE IF EXISTS " + WORD_LIST_TABLE);
        onCreate(db);
    }
}
```

Word List SQL Searchable





String.xml

```
<?xml version="1.0" encoding="utf-8"?>

<resources>
    <string name="app_name">WordListSQLInteractiveWithSearch</string>
    <string name="hint_word">Word...</string>
    <string name="hint_definition">Definition...</string>
    <string name="button_save">Save</string>
    <string name="button_new">New</string>
    <string name="button_edit">Edit</string>
    <string name="button_delete">Delete</string>
    <string name="button_search">Search</string>
    <string name="search_results">Search results.</string>
    <string name="menu_search">Search...</string>
    <string name="empty_word_not_saved">Word not saved because it is
empty.</string>
    <string name="no_result">No result.</string>
</resources>
```

Layout

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <android.support.v7.widget.RecyclerView
        android:id="@+id/recyclerview"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
    </android.support.v7.widget.RecyclerView>

    <android.support.design.widget.FloatingActionButton
        android:id="@+id/fab"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|end"
        android:layout_margin="16dp"
        android:clickable="true"
        android:src="@drawable/ic_add_24dp" />

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

Activity_edit_word.xml

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/edit_word"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="sans-serif-light"
        android:hint="@string/hint_word"
        android:inputType="textAutoComplete"
        android:padding="@dimen/small_padding"
        android:layout_marginBottom="@dimen/big_padding"
        android:layout_marginTop="@dimen/big_padding"
        android:textSize="18sp" />

    <Button
        android:id="@+id/button_save"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/colorPrimary"
        android:onClick="returnReply"
        android:text="@string/button_save"
        android:textColor="@color/buttonLabel" />

</LinearLayout>
```

Activity_search.xml

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
```

```
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/search_word"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="sans-serif-light"
        android:hint="@string/hint_word"
        android:inputType="textAutoComplete"
        android:padding="@dimen/small_padding"
        android:layout_marginBottom="@dimen/big_padding"
        android:layout_marginTop="@dimen/big_padding"
        android:textSize="18sp" />

    <Button
        android:id="@+id/button_search"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/colorPrimary"
        android:onClick="showResult"
        android:text="@string/button_search"
        android:textColor="@color/buttonLabel" />

    <TextView
        android:id="@+id/search_result"
        android:layout_width="match_parent"
        android:layout_height="300dp"
        android:textSize="18sp"
        android:hint="@string/search_results"/>

</LinearLayout>
```

Wordlist_item.xml

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="6dp">

    <TextView
        android:id="@+id/word"
        android:layout_width="match_parent"
        style="@style/word_title" />

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

        <Button
            android:id="@+id/delete_button"
            android:layout_width="match_parent"
            android:layout_height="@dimen/button_height"
            android:layout_weight="2"
            android:background="@color/colorPrimaryDark"
            android:text="@string/button_delete"
            android:textColor="@color/buttonLabel"/>

        <Button
            android:id="@+id/edit_button"
```

```
            android:layout_width="match_parent"
            android:layout_height="@dimen/button_height"
            android:layout_weight="1"
            android:background="@color/colorPrimary"
            android:text="@string/button_edit"
            android:textColor="@color/buttonLabel"/>

    </LinearLayout>

    <Button
        android:layout_width="match_parent"
        android:layout_height="@dimen/divider_height"
        android:background="@color/colorAccent" />

</LinearLayout>
```

Mainactivity

```
package com.android.example.wordlistsqlsearchable;

import android.content.Intent;
import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.text.TextUtils;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    private static final String TAG = MainActivity.class.getSimpleName();

    public static final int WORD_EDIT = 1;
    public static final int WORD_ADD = -1;

    private WordListOpenHelper mDB;
    private RecyclerView mRecyclerView;
    private WordListAdapter mAdapter;

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

        mDB = new WordListOpenHelper(this);

        // Create recycler view.
        mRecyclerView = (RecyclerView) findViewById(R.id.recyclerview);
        // Create an mAdapter and supply the data to be displayed.
        mAdapter = new WordListAdapter(this, /* mDB.getAllEntries(),*/ mDB);
        // Connect the mAdapter with the recycler view.
        mRecyclerView.setAdapter(mAdapter);
        // Give the recycler view a default layout manager.
        mRecyclerView.setLayoutManager(new LinearLayoutManager(this));

        // Add a floating action click handler for creating new entries.
        FloatingActionButton fab = (FloatingActionButton)
findViewById(R.id.fab);
        fab.setOnClickListener(new View.OnClickListener() {
```

```
            @Override
            public void onClick(View view) {
                // Starts empty edit activity.
                Intent intent = new Intent(getBaseContext(),
EditWordActivity.class);
                startActivityForResult(intent, WORD_EDIT);
            }
        });
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case R.id.action_search:
                // Starts search activity.
                Intent intent = new Intent(getBaseContext(),
com.android.example.wordlistsqlsearchable.SearchActivity.class);
                startActivity(intent);
                return true;
        }
        return super.onOptionsItemSelected(item);
    }

    public void onActivityResult(int requestCode, int resultCode, Intent data)
{
        super.onActivityResult(requestCode, resultCode, data);

        if (requestCode == WORD_EDIT) {
            if (resultCode == RESULT_OK) {
                String word =
data.getStringExtra(EditWordActivity.EXTRA_REPLY);

                // Update the database.
                if (!TextUtils.isEmpty(word)) {
                    int id = data.getIntExtra(WordListAdapter.EXTRA_ID, -99);

                    if (id == WORD_ADD) {
                        mDB.insert(word);
                    } else if (id >= 0) {
                        mDB.update(id, word);
                    }
                    // Update the UI.
                    mAdapter.notifyDataSetChanged();
                } else {
                    Toast.makeText(
                            getApplicationContext(),
                            R.string.empty_word_not_saved,
                            Toast.LENGTH_LONG).show();
                }
            }
        }
    }
}
```

EditWordActivity.java

```
package com.android.example.wordlistsqlsearchable;

import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.EditText;

public class EditWordActivity extends AppCompatActivity {

    private static final String TAG = EditWordActivity.class.getSimpleName();

    private static final int NO_ID = -99;
    private static final String NO_WORD = "";

    private EditText mEditWordView;

    // Unique tag for the intent reply.
    public static final String EXTRA_REPLY =
"com.example.android.wordlistsql.REPLY";

    int mId = MainActivity.WORD_ADD;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_edit_word);

        mEditWordView = (EditText) findViewById(R.id.edit_word);

        // Get data sent from calling activity.
        Bundle extras = getIntent().getExtras();

        // If we are passed content, fill it in for the user to edit.
        if (extras != null) {
            int id = extras.getInt(WordListAdapter.EXTRA_ID, NO_ID);
            String word = extras.getString(WordListAdapter.EXTRA_WORD,
NO_WORD);
            if (id != NO_ID && word != NO_WORD) {
                mId = id;
                mEditWordView.setText(word);
            }
        } // Otherwise, start with empty fields.
    }

    public void returnReply(View view) {
        String word = ((EditText)
findViewById(R.id.edit_word)).getText().toString();

        Intent replyIntent = new Intent();
        replyIntent.putExtra(EXTRA_REPLY, word);
        replyIntent.putExtra(WordListAdapter.EXTRA_ID, mId);
        setResult(RESULT_OK, replyIntent);
        finish();
    }
}
```

MyButtonOnClickListener.java

```
package com.android.example.wordlistsqlsearchable;

import android.view.View;
```

```
public class MyButtonOnClickListener implements View.OnClickListener {
    private static final String TAG =
View.OnClickListener.class.getSimpleName();

    int id;
    String word;

    public MyButtonOnClickListener(int id, String word) {
        this.id = id;
        this.word = word;
    }

    public void onClick(View v) {
        // Implemented in WordListAdapter
    }

}
```

SearchActiviy.java

```
package com.android.example.wordlistsqlsearchable;

import android.database.Cursor;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;


public class SearchActivity extends AppCompatActivity {

    private static final String TAG = EditWordActivity.class.getSimpleName();

    private WordListOpenHelper mDB;

    private EditText mEditWordView;
    private TextView mTextView;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_search);

        mDB = new WordListOpenHelper(this);

        mEditWordView = ((EditText) findViewById(R.id.search_word));
        mTextView = ((TextView) findViewById(R.id.search_result));
    }

    // Click handler for Search button.
    public void showResult(View view) {
        String word = mEditWordView.getText().toString();
        mTextView.setText("Result for " + word + ":\n\n");

        // Search for the word in the database.
        Cursor cursor = mDB.search(word);
        // You must move the cursor to the first item.
        cursor.moveToFirst();
        // Only process a non-null cursor with rows.
        if (cursor != null & cursor.getCount() > 0) {
            int index;
            String result;
```

```
            // Iterate over the cursor, while there are entries.
            do {
                // Don't guess at the column index. Get the index for the named
column.
                index = cursor.getColumnIndex(WordListOpenHelper.KEY_WORD);
                // Get the value from the column for the current cursor.
                result = cursor.getString(index);
                // Add result to what's already in the text view.
                mTextView.append(result + "\n");
            } while (cursor.moveToNext());
            cursor.close();
        } else {
            mTextView.append(getString(R.string.no_result));
        }
    }
}
```

WordItem.java

```
package com.android.example.wordlistsqlsearchable;

public class WordItem {

    private int mId;
    private String mWord;

    public WordItem() {}

    public int getId() {
        return this.mId;
    }

    public String getWord() {
        return this.mWord;
    }

    public void setId(int id) {
        this.mId = id;
    }

    public void setWord(String word) {
        this.mWord = word;
    }
}
```

Wordlistadapter.java

```
package com.android.example.wordlistsqlsearchable;

import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.TextView;

public class WordListAdapter extends
RecyclerView.Adapter<WordListAdapter.WordViewHolder> {
```

```
    /**
     *  Custom view holder with a text view and two buttons.
     */
    class WordViewHolder extends RecyclerView.ViewHolder {
        public final TextView wordItemView;
        Button delete_button;
        Button edit_button;

        public WordViewHolder(View itemView) {
            super(itemView);
            wordItemView = (TextView) itemView.findViewById(R.id.word);
            delete_button = (Button)itemView.findViewById(R.id.delete_button);
            edit_button = (Button)itemView.findViewById(R.id.edit_button);
        }
    }

    private static final String TAG = WordListAdapter.class.getSimpleName();

    public static final String EXTRA_ID = "ID";
    public static final String EXTRA_WORD = "WORD";
    public static final String EXTRA_POSITION = "POSITION";

    private final LayoutInflater mInflater;
    WordListOpenHelper mDB;
    Context mContext;

    public WordListAdapter(Context context, WordListOpenHelper db) {
        mInflater = LayoutInflater.from(context);
        mContext = context;
        mDB = db;
    }

    @Override
    public WordViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        View itemView = mInflater.inflate(R.layout.wordlist_item, parent,
false);
        return new WordViewHolder(itemView);
    }

    @Override
    public void onBindViewHolder(WordViewHolder holder, int position) {
        // Keep a reference to the view holder for the click listener
        final WordViewHolder h = holder; // needs to be final for use in
callback

        WordItem current = mDB.query(position);
        holder.wordItemView.setText(current.getWord());

        // Attach a click listener to the DELETE button.
        holder.delete_button.setOnClickListener(new MyButtonOnClickListener(
                current.getId(), null)  {

            @Override
            public void onClick(View v ) {
                // Remove from the database.
                int deleted = mDB.delete(id);
                if (deleted >= 0) {
                    // Redisplay the view.
                    notifyItemRemoved(h.getAdapterPosition());
                }
            }
        });

        // Attach a click listener to the EDIT button.
```

```
        holder.edit_button.setOnClickListener(new MyButtonOnClickListener(
                current.getId(), current.getWord()) {

            @Override
            public void onClick(View v) {
                Intent intent = new Intent(mContext, EditWordActivity.class);

                intent.putExtra(EXTRA_ID, id);
                intent.putExtra(EXTRA_POSITION, h.getAdapterPosition());
                intent.putExtra(EXTRA_WORD, word);

                // Start an empty edit activity.
                ((Activity) mContext).startActivityForResult(intent,
MainActivity.WORD_EDIT);
            }
        });
    }

    @Override
    public int getItemCount() {
        return (int) mDB.count();
    }
}
```

WordListOpenHelper.java

```
package com.android.example.wordlistsqlsearchable;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.DatabaseUtils;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;

public class WordListOpenHelper extends SQLiteOpenHelper {

    private static final String TAG = WordListOpenHelper.class.getSimpleName();

    // Version has to be 1 first time or app will crash.
    private static final int DATABASE_VERSION = 1;
    private static final String WORD_LIST_TABLE = "word_entries";
    private static final String DATABASE_NAME = "wordlist";

    // Column names...
    public static final String KEY_ID = "_id";
    public static final String KEY_WORD = "word";

    // ... and a string array of columns.
    private static final String[] COLUMNS =
            {KEY_ID, KEY_WORD};

    // Build the SQL query that creates the table.
    private static final String WORD_LIST_TABLE_CREATE =
            "CREATE TABLE " + WORD_LIST_TABLE + " (" +
                    KEY_ID + " INTEGER PRIMARY KEY, " + // will auto-increment
if no value passed
                    KEY_WORD + " TEXT );";

    private SQLiteDatabase mWritableDB;
    private SQLiteDatabase mReadableDB;
```

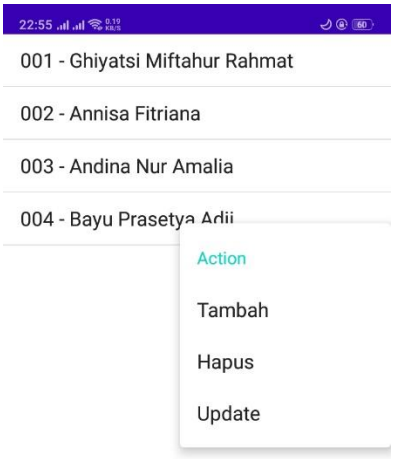
```
    public WordListOpenHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
        Log.d(TAG, "Construct WordListOpenHelper");
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(WORD_LIST_TABLE_CREATE);
        fillDatabaseWithData(db);
    }

    public void fillDatabaseWithData(SQLiteDatabase db) {

        String[] words = {"Android", "Adapter", "ListView", "AsyncTask",
"Android Studio",
                "SQLiteDatabase", "SQLOpenHelper", "Data model", "ViewHolder",
                "Android Performance", "OnClickListener"};

        // Create a container for the data.
        ContentValues values = new ContentValues();

        for (int i=0; i < words.length; i++) {
            // Put column/value pairs for current row into the container.
            values.put(KEY_WORD, words[i]); // put() overrides existing values.
            // Insert the row.
            db.insert(WORD_LIST_TABLE, null, values);
        }
    }

    public Cursor search(String searchString) {
        String[] columns = new String[]{KEY_WORD};
        String where =  KEY_WORD + " LIKE ?";
        searchString = "%" + searchString + "%";
        String[] whereArgs = new String[]{searchString};

        Cursor cursor = null;
        try {
            if (mReadableDB == null) {
                mReadableDB = getReadableDatabase();
            }
            cursor = mReadableDB.query(WORD_LIST_TABLE, columns, where,
whereArgs, null, null, null);
        } catch (Exception e) {
            Log.d(TAG, "SEARCH EXCEPTION! " + e); // Just log the exception
        }
        return cursor;
    }


    public WordItem query(int position) {
        String query = "SELECT  * FROM " + WORD_LIST_TABLE +
                " ORDER BY " + KEY_WORD + " ASC " +
                "LIMIT " + position + ",1";

        Cursor cursor = null;
        WordItem entry = new WordItem();

        try {
            if (mReadableDB == null) {
                mReadableDB = getReadableDatabase();
            }
            cursor = mReadableDB.rawQuery(query, null);
            cursor.moveToFirst();
            entry.setId(cursor.getInt(cursor.getColumnIndex(KEY_ID)));
```

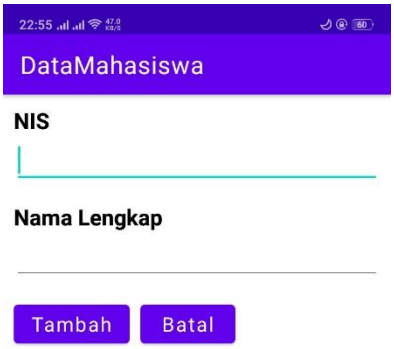
```
            entry.setWord(cursor.getString(cursor.getColumnIndex(KEY_WORD)));
        } catch (Exception e) {
            Log.d(TAG, "QUERY EXCEPTION! " + e); // Just log the exception
        } finally {
            // Must close cursor and db now that we are done with it.
            cursor.close();
            return entry;
        }
    }

    public long count() {
        if (mReadableDB == null) {
            mReadableDB = getReadableDatabase();
        }
        return DatabaseUtils.queryNumEntries(mReadableDB, WORD_LIST_TABLE);
    }

    public long insert(String word) {
        long newId = 0;
        ContentValues values = new ContentValues();
        values.put(KEY_WORD, word);
        try {
            if (mWritableDB == null) {
                mWritableDB = getWritableDatabase();
            }
            newId = mWritableDB.insert(WORD_LIST_TABLE, null, values);
        } catch (Exception e) {
            Log.d(TAG, "INSERT EXCEPTION! " + e);
        }
        return newId;
    }

    public int update(int id, String word) {
        int mNumberOfRowsUpdated = -1;
        try {
            if (mWritableDB == null) {
                mWritableDB = getWritableDatabase();
            }
            ContentValues values = new ContentValues();
            values.put(KEY_WORD, word);

            mNumberOfRowsUpdated = mWritableDB.update(WORD_LIST_TABLE, //table
to change
                    values, // new values to insert
                    KEY_ID + " = ?", // selection criteria for row (in this
case, the _id column)
                    new String[]{String.valueOf(id)}); //selection args; the
actual value of the id

        } catch (Exception e) {
            Log.d (TAG, "UPDATE EXCEPTION! " + e);
        }
        return mNumberOfRowsUpdated;
    }

    public int delete(int id) {
        int deleted = 0;
        try {
            if (mWritableDB == null) {
                mWritableDB = getWritableDatabase();
            }
            deleted = mWritableDB.delete(WORD_LIST_TABLE, //table name
                    KEY_ID + " =? ", new String[]{String.valueOf(id)});
        } catch (Exception e) {
```

```
            Log.d (TAG, "DELETE EXCEPTION! " + e);         }
        return deleted;
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        Log.w(WordListOpenHelper.class.getName(),
                "Upgrading database from version " + oldVersion + " to "
                        + newVersion + ", which will destroy all old data");
        db.execSQL("DROP TABLE IF EXISTS " + WORD_LIST_TABLE);
        onCreate(db);
    }
}
```

DataSiswa


001 - Ghiyatsi Miftahur Rahmat
002 - Annisa Fitriana
003 - Andina Nur Amalia
004 - Bayu Prasetya Adji


001 - Ghiyatsi Miftahur Rahmat
002 - Annisa Fitriana
003 - Andina Nur Amalia
004 - Bayu Prasetya Adji
Action
Tambah
Hapus
Update


DataMahasiswa
NIS
Nama Lengkap
Tambah
Batal


DataMahasiswa
NIS
004
Nama Lengkap
Bayu Prasetya Adji
Update
Batal

String.xml

```
<resources>
    <string name="app_name">DataMahasiswa</string>
    <string name="batal">Batal</string>
    <string name="nama_lengkap">Nama Lengkap</string>
    <string name="nis">NIS</string>
    <string name="update">Update</string>
    <string name="tambah">Tambah</string>
</resources>
```

Layout

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:layout_marginVertical="10dp"
    android:layout_marginHorizontal="10dp"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <ListView
        android:id="@+id/simple_list_item_1"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>

</LinearLayout>
```

Activity_tambah.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:orientation="vertical"
    android:layout_marginHorizontal="10dp"
    android:layout_marginVertical="10dp"
    tools:context=".Tambah">

    <TextView
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:text="@string/nis"
        android:textColor="@color/black"
        android:textSize="18sp"
        android:textStyle="bold"/>

    <EditText
        android:id="@+id/inputnis"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="16sp"
        android:layout_marginBottom="15dp"/>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/nama_lengkap"
        android:textColor="@color/black"
        android:textSize="18sp"
        android:textStyle="bold"/>

    <EditText
        android:id="@+id/inputnama"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="16sp"
        android:layout_marginBottom="15dp"/>

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

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/btnTambah"
            android:text="@string/tambah"
            android:textAllCaps="false"
            android:layout_marginRight="10dp"
            android:textSize="16sp"/>

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/btnCancel"
            android:text="@string/batal"
            android:textAllCaps="false"
```

```
            android:textSize="16sp"/>

    </LinearLayout>

</LinearLayout>
```

Activity_update.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:orientation="vertical"
    android:layout_marginHorizontal="10dp"
    android:layout_marginVertical="10dp"
    tools:context=".Update">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/nis"
        android:textColor="@color/black"
        android:textSize="18sp"
        android:textStyle="bold"/>

    <EditText
        android:id="@+id/updatenis"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="16sp"
        android:layout_marginBottom="15dp"
        android:enabled="false"/>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/nama_lengkap"
        android:textColor="@color/black"
        android:textSize="18sp"
        android:textStyle="bold"/>

    <EditText
        android:id="@+id/updatenama"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="16sp"
```

```
        android:layout_marginBottom="15dp"/>

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

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/btnUpdate"
            android:text="@string/update"
            android:textAllCaps="false"
            android:layout_marginRight="10dp"
            android:textSize="16sp"/>

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/btnCancel"
            android:text="@string/batal"
            android:textAllCaps="false"
            android:textSize="16sp"/>

    </LinearLayout>

</LinearLayout>
```

Java

DatabaseHendler.java

```
package com.example.datamahasiswa;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import java.util.ArrayList;
import java.util.List;

public class DatabaseHandler extends SQLiteOpenHelper {

    private static final int DATABASE_VERSION= 1;
    // NamaDatabase
    private static final String DATABASE_NAME= "Sekolah";
    // NamaTable
```

```
    private static final String TABLE_SISWA= "Siswa";
    // NamaKolomTable Siswa
    private static final String KEY_NIS= "nis";
    private static final String KEY_NAMA= "nama";

    public DatabaseHandler(Context context) {
    // TODOAuto-generated constructor stub
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }


    // Create Table
    @Override
    public void onCreate(SQLiteDatabase db) {
    // TODOAuto-generated method stub
        String query_table_siswa = "CREATE TABLE "+
                TABLE_SISWA+ "("
                + KEY_NIS+ " TEXT PRIMARY KEY,"+ KEY_NAMA+ " TEXT)";
        db.execSQL(query_table_siswa);
    }


    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    // TODOAuto-generated method stub
    // Drop older table if existed
        db.execSQL("DROP TABLE IF EXISTS "+ TABLE_SISWA);
    // Create tables again
        onCreate(db);
    }


    // add new siswa
    public void addSiswa(Siswa siswa) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(KEY_NIS, siswa.getNis());
        values.put(KEY_NAMA, siswa.getNama());
// Inserting Row
        db.insert(TABLE_SISWA, null, values);
        db.close();
    }

    // read siswa
    public Siswa getSiswa(String nis) {
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.query(TABLE_SISWA, new String[] {KEY_NIS,
KEY_NAMA}, KEY_NIS+ "=?", new String[] {nis}, null, null, null, null);
        if(cursor != null)
```

```
            cursor.moveToFirst();
        Siswa siswa = new Siswa(cursor.getString(0),
                cursor.getString(1));
        return siswa;
    }

    //read all siswa
    public List<Siswa> getSemuaSiswa() {
        List<Siswa> siswaList = new ArrayList<Siswa>();
        String query_select_siswa = "SELECT * FROM "+ TABLE_SISWA;
        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor = db.rawQuery(query_select_siswa,null);
        if(cursor.moveToFirst()) {
            do{
                Siswa siswa = new Siswa(cursor.getString(0),
                        cursor.getString(1));
                siswaList.add(siswa);
            } while(cursor.moveToNext());
        }
        return siswaList;
    }


    //delete data siswa
    public void deleteSiswa(Siswa siswa) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_SISWA, KEY_NIS+ "='"+ siswa.getNis()+"'",null);
        db.close();
        System.out.println("Data terhapus "+siswa.getNis());
    }


    public void deleteRow(String xnis) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_SISWA, KEY_NIS+ "='"+ xnis+"'",null);
        db.close();
        System.out.println("Data terhapus "+xnis);
    }

    public void updateMethod(String nis, String nama){
        SQLiteDatabase db = this.getWritableDatabase();
        db.execSQL("update "+TABLE_SISWA+" set nama='"+nama+ "' where
nis='"+nis+"'");
        db.close();
        System.out.println("Data sudah di update "+nis);
    }

}
```

Mainactivity.java

```
package com.example.datamahasiswa;

import androidx.appcompat.app.AppCompatActivity;

import android.app.ListActivity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.ContextMenu;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;

import java.util.List;

public class MainActivity extends ListActivity {
    String dataSiswa[] = null;
    String dS[] = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        //setContentView(R.layout.activity_main);
        // Tambah Siswa
        DatabaseHandler db = new DatabaseHandler(this);
        // Membaca Semua Siswa
        Log.d("Baca Siswa: ", "Membaca Semua Data Siswa..");
        List<Siswa> siswa = db.getSemuaSiswa();
        dataSiswa= new String[siswa.size()];
        dS= new String[siswa.size()];
        int i=0;
        for(Siswa s : siswa) {
            String log = "NIS: "+ s.getNis() + ",Nama: "+
                    s.getNama();
            Log.d("Name: ", log);
            dataSiswa[i] = s.getNis() + " - "+ s.getNama();
            dS[i] = s.getNis();
            i++;
        }

        // check data
        if(i==0)
        {
            Log.d("Tambah Siswa: ", "Menambah Data Siswa..");
```

```
            db.addSiswa(new Siswa("001", "Ghiyatsi Miftahur Rahmat"));
            db.addSiswa(new Siswa("002", "Annisa Fitriana"));
            db.addSiswa(new Siswa("003", "Andina Nur Amalia"));
            db.addSiswa(new Siswa("004", "Najwa Aulia Dhofiroh"));
        }
        setListAdapter(new ArrayAdapter<Object>(this,
android.R.layout.simple_list_item_1, dataSiswa));
        registerForContextMenu(getListView());
    }


    @Override
    public void onCreateContextMenu(ContextMenu menu, View v,
ContextMenu.ContextMenuInfo menuInfo) {
// TODOAuto-generated method stub
        super.onCreateContextMenu(menu, v, menuInfo);
        menu.setHeaderTitle("Action");
        menu.add(0,0,0,"Tambah");
        menu.add(0,1,1,"Hapus");
        menu.add(0,2,2,"Update");
    }

    @Override
    public boolean onContextItemSelected(MenuItem item) {
// TODOAuto-generated method stub
        try{
            switch(item.getItemId()){
                case 0:{
                    Class c =
Class.forName("com.example.datamahasiswa.Tambah");
                    Intent i = new Intent(MainActivity.this, c);
                    startActivity(i);break;
                }
                case 1:{
                    DatabaseHandler db = new DatabaseHandler(this);
                    AdapterView.AdapterContextMenuInfo info =
(AdapterView.AdapterContextMenuInfo) item.getMenuInfo();
                    String[] args = {String.valueOf(info.id) };
                    int xpos=Integer.parseInt(args[0]);
                    db.deleteRow(dS[xpos]);
                    Class c =
Class.forName("com.example.datamahasiswa.MainActivity");
                    Intent i = new Intent(MainActivity.this, c);
                    startActivity(i);
                    break;
                }
                case 2:{
                    DatabaseHandler db = new DatabaseHandler(this);
```

```
                    AdapterView.AdapterContextMenuInfo info
=(AdapterView.AdapterContextMenuInfo) item.getMenuInfo();
                    String[] args ={String.valueOf(info.id)};
                    Log.d("args0 : ",args[0]);
                    int xpos=Integer.parseInt(args[0]);
                    db.getSiswa(dS[xpos]);
                    String namax=db.getSiswa(dS[xpos]).getNama();
                    Intent i = new Intent(this, Update.class);
                    Bundle bun = new Bundle();
                    bun.putString("nis", dS[xpos]);
                    bun.putString("nama", namax);
                    i.putExtras(bun);
                    startActivity(i); break;
                }
            }
        }catch(ClassNotFoundException e) {
            // TODOAuto-generated catch block
            e.printStackTrace();
        }
        return true;
    }

}
```

Siswa.java

```
package com.example.datamahasiswa;

public class Siswa {
    private String nis;
    private String nama;
    public Siswa() {
    }
    public Siswa(String nis, String nama) {
        this.nis= nis;
        this.nama= nama;
    }
    public String getNis() {
        return nis;
    }
    public void setNis(String nis) {
        this.nis= nis;
    }
    public String getNama() {
        return nama;
    }
    public void setNama(String nama) {
        this.nama= nama;
```

```
    }
}
```

Tambah.java

```
package com.example.datamahasiswa;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class Tambah extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tambah);
        final DatabaseHandler db = new DatabaseHandler(this);
        final EditText editNis = (EditText) findViewById(R.id.inputnis);
        final EditText editNama = (EditText) findViewById(R.id.inputnama);
        Button btnTambah = (Button) findViewById(R.id.btnTambah);
        Button btnBatal = (Button) findViewById(R.id.btnCancel);
        btnTambah.setOnClickListener(new View.OnClickListener()
        {
            @Override
            public void onClick(View v) {
                String nis = editNis.getText().toString();
                String nama = editNama.getText().toString();
                db.addSiswa(new Siswa(nis, nama));
                editNis.setText("");
                editNama.setText("");
                try{
                    Class c =
Class.forName("com.example.datamahasiswa.MainActivity");
                    Intent i = new Intent(Tambah.this, c);
                    startActivity(i);
                } catch(ClassNotFoundException e) {
                    e.printStackTrace();
                }
            }
        });
        btnBatal.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
```

```
                try {
                    Class c =
Class.forName("com.example.datamahasiswa.MainActivity");
                    Intent i=new Intent(Tambah.this,c);
                    startActivity(i);
                } catch (ClassNotFoundException e) {
                    e.printStackTrace();
                }
            }
        });
    }
}
```

Update.java

```
package com.example.datamahasiswa;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class Update extends AppCompatActivity {
    private String xnis,xnama;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_update);
        final DatabaseHandler db = new DatabaseHandler(this);
        final EditText editNis = (EditText)findViewById(R.id.updatenis);
        final EditText editNama = (EditText) findViewById(R.id.updatenama);
        Button btnTambah = (Button) findViewById(R.id.btnUpdate);
        Button btnBatal = (Button) findViewById(R.id.btnCancel);

        // ambil data siswa
        Bundle bun = this.getIntent().getExtras();
        xnis = bun.getString("nis");
        xnama = bun.getString("nama");

        //masukkan data siswa
        editNama.setText(xnama);
        editNis.setText(xnis);

        btnTambah.setOnClickListener(new View.OnClickListener()
        {
```

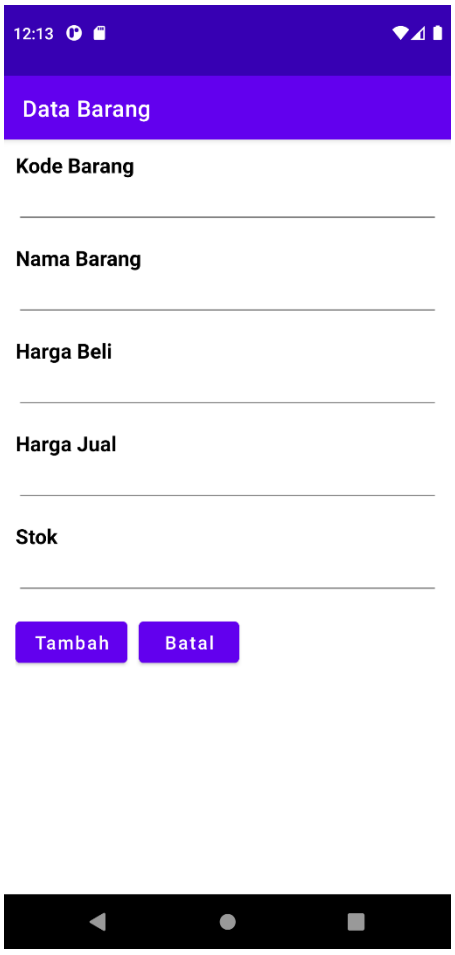
```
            @Override
            public void onClick(View v) {
                String nis = editNis.getText().toString();
                String nama = editNama.getText().toString();
                //db.addSiswa(new Siswa(nis, nama));
                //db.updateData(nis,nama);
                db.updateMethod(nis,nama);
                editNis.setText("");
                editNama.setText("");
                try{
                    Class c=
Class.forName("com.example.datamahasiswa.MainActivity");
                    Intent i = new Intent(Update.this, c);
                    startActivity(i);
                } catch(ClassNotFoundException e) {
                    e.printStackTrace();
                }
            }
        });

        btnBatal.setOnClickListener(new View.OnClickListener()
        {public void onClick(View v) {
            try {
                Class c =
Class.forName("com.example.datamahasiswa.MainActivity");
                Intent i=new Intent(Update.this,c);
                startActivity(i);
            } catch (ClassNotFoundException e) {
                e.printStackTrace();
            }
        }
        });
    }
}
```

AndroidManifest

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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.DataMahasiswa">
```

```
        <activity
            android:name=".Tambah"
            android:exported="false" />
        <activity
            android:name=".Update"
            android:exported="false" />
        <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>
```

Data Barang



String.xml

```
<resources>
    <string name="app_name">Data Barang</string>
    <string name="batal">Batal</string>
    <string name="nama_lengkap">Nama Barang</string>
    <string name="nis">Kode Barang</string>
    <string name="update">Update</string>
    <string name="tambah">Tambah</string>
    <string name="harga_beli">Harga Beli</string>
    <string name="harga_jual">Harga Jual</string>
    <string name="stok">Stok</string>
</resources>
```

Layout

Activity_main

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:layout_marginVertical="10dp"
    android:layout_marginHorizontal="10dp"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <ListView
        android:id="@+id/simple_list_item_1"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>

</LinearLayout>
```

Activity_tambah

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:orientation="vertical"
    android:layout_marginHorizontal="10dp"
    android:layout_marginVertical="10dp"
    tools:context=".Tambah">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/nis"
        android:textColor="@color/black"
        android:textSize="18sp"
        android:textStyle="bold"/>

    <EditText
        android:id="@+id/inputkdbrg"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="16sp"
        android:layout_marginBottom="15dp" />
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/nama_lengkap"
    android:textColor="@color/black"
    android:textSize="18sp"
    android:textStyle="bold"/>

<EditText
    android:id="@+id/inputnama"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="16sp"
    android:layout_marginBottom="15dp"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/harga_beli"
    android:textColor="@color/black"
    android:textSize="18sp"
    android:textStyle="bold"/>

<EditText
    android:id="@+id/inputbeli"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="16sp"
    android:layout_marginBottom="15dp"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/harga_jual"
    android:textColor="@color/black"
    android:textSize="18sp"
    android:textStyle="bold"/>

<EditText
    android:id="@+id/inputjual"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="16sp"
    android:layout_marginBottom="15dp"/>

<TextView
    android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:text="@string/stok"
        android:textColor="@color/black"
        android:textSize="18sp"
        android:textStyle="bold"/>

    <EditText
        android:id="@+id/inputstok"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="16sp"
        android:layout_marginBottom="15dp"/>

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

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/btnTambah"
            android:text="@string/tambah"
            android:textAllCaps="false"
            android:layout_marginRight="10dp"
            android:textSize="16sp"/>

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/btnCancel"
            android:text="@string/batal"
            android:textAllCaps="false"
            android:textSize="16sp"/>

    </LinearLayout>

</LinearLayout>
```

Activity_update

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:orientation="vertical"
```

```
android:layout_marginHorizontal="10dp"
android:layout_marginVertical="10dp"
tools:context=".Update">

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/nis"
    android:textColor="@color/black"
    android:textSize="18sp"
    android:textStyle="bold"/>

<EditText
    android:id="@+id/updatekdbrg"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="16sp"
    android:layout_marginBottom="15dp"
    android:enabled="false"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/nama_lengkap"
    android:textColor="@color/black"
    android:textSize="18sp"
    android:textStyle="bold"/>

<EditText
    android:id="@+id/updatenama"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="16sp"
    android:layout_marginBottom="15dp"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/harga_beli"
    android:textColor="@color/black"
    android:textSize="18sp"
    android:textStyle="bold"/>

<EditText
    android:id="@+id/updatebeli"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="16sp"
```

```
        android:layout_marginBottom="15dp"/>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/harga_jual"
        android:textColor="@color/black"
        android:textSize="18sp"
        android:textStyle="bold"/>

    <EditText
        android:id="@+id/updatejual"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="16sp"
        android:layout_marginBottom="15dp"/>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/stok"
        android:textColor="@color/black"
        android:textSize="18sp"
        android:textStyle="bold"/>

    <EditText
        android:id="@+id/updatestok"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="16sp"
        android:layout_marginBottom="15dp"/>

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

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/btnUpdate"
            android:text="@string/update"
            android:textAllCaps="false"
            android:layout_marginRight="10dp"
            android:textSize="16sp"/>

        <Button
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:id="@+id/btnCancel"
            android:text="@string/batal"
            android:textAllCaps="false"
            android:textSize="16sp"/>

    </LinearLayout>

</LinearLayout>
```

Java

DatabaseHandler.java

```
package com.example.databarang;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import java.util.ArrayList;
import java.util.List;

public class DatabaseHandler extends SQLiteOpenHelper {

    private static final int DATABASE_VERSION= 1;
    // NamaDatabase
    private static final String DATABASE_NAME= "DataBarang";
    // NamaTable
    private static final String TABLE_BARANG= "Barang";
    // NamaKolomTable Siswa
    private static final String KEY_KDBRG= "kdbarang";
    private static final String KEY_NAMA= "nama";
    private static final String KEY_BELI= "hgbeli";
    private static final String KEY_JUAL= "hgjual";
    private static final String KEY_STOK= "jmlstok";

    public DatabaseHandler(Context context) {
    // TODOAuto-generated constructor stub
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }


    // Create Table
    @Override
    public void onCreate(SQLiteDatabase db) {
    // TODOAuto-generated method stub
```

```
        String query_table_barang = "CREATE TABLE "+ TABLE_BARANG +
"("+KEY_KDBRG + " TEXT PRIMARY KEY, "+ KEY_NAMA + " TEXT NOT NULL, "+ KEY_BELI
+" TEXT NOT NULL, "+ KEY_JUAL +" TEXT NOT NULL, "+ KEY_STOK +" TEXT NOT
NULL)";
        db.execSQL(query_table_barang);
    }


    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    // TODOAuto-generated method stub
    // Drop older table if existed
        db.execSQL("DROP TABLE IF EXISTS "+ TABLE_BARANG);
    // Create tables again
        onCreate(db);
    }


    // add new barang
    public void addBarang(Barang barang) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(KEY_KDBRG, barang.getKdbarang());
        values.put(KEY_NAMA, barang.getNama());
        values.put(KEY_BELI, barang.getBeli());
        values.put(KEY_JUAL, barang.getJual());
        values.put(KEY_STOK, barang.getStok());
// Inserting Row
        db.insert(TABLE_BARANG, null, values);
        db.close();
    }

    // read barang
    public Barang getBarang(String kdbarang) {
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.query(TABLE_BARANG, new String[] {KEY_KDBRG,
KEY_NAMA}, KEY_KDBRG+ "=?", new String[] {kdbarang}, null, null, null, null);
        if(cursor != null)
            cursor.moveToFirst();
        Barang barang = new Barang(cursor.getString(0), cursor.getString(1),
cursor.getInt(2), cursor.getInt(3), cursor.getInt(4) );
        return barang;
    }

    //read all siswa
    public List<Barang> getSemuaBarang() {
        List<Barang> barangList = new ArrayList<Barang>();
        String query_select_barang = "SELECT * FROM "+ TABLE_BARANG;
        SQLiteDatabase db = this.getWritableDatabase();
```

```
        Cursor cursor = db.rawQuery(query_select_barang,null);
        if(cursor.moveToFirst()) {
            do{
                Barang barang = new Barang(cursor.getString(0),
cursor.getString(1), cursor.getInt(2), cursor.getInt(3), cursor.getInt(4) );
                barangList.add(barang);
            } while(cursor.moveToNext());
        }
        return barangList;
    }


    //delete data barang
    public void deleteBarang(Barang barang) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_BARANG, KEY_KDBRG+ "='"+
barang.getKdbarang()+"'",null);
        db.close();
        System.out.println("Data terhapus "+barang.getKdbarang());
    }


    public void deleteRow(String xkdbarang) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_BARANG, KEY_KDBRG+ "='"+ xkdbarang+"'",null);
        db.close();
        System.out.println("Data terhapus "+xkdbarang);
    }

    public void updateMethod(String kdbarang, String nama, int beli, int jual,
int stok){
        SQLiteDatabase db = this.getWritableDatabase();
        db.execSQL("update "+TABLE_BARANG + " set nama='" + nama + "'" + "
,beli=" + beli + " ,jual=" + jual + " ,stok=" + stok +" where kdbarang='" +
kdbarang + "'") ;
        db.close();
        System.out.println("Data sudah di update "+kdbarang);
    }

}
```

Barang.java

```
package com.example.databarang;

public class Barang {
    private String kdbarang;
```

```
    private String nama;
    private int beli;
    private int jual;
    private int stok;

    public Barang(String kdbarang, String nama, int beli, int jual, int stok)
{
        this.kdbarang= kdbarang;
        this.nama= nama;
        this.beli= beli;
        this.jual= jual;
        this.stok= stok;
    }
    public String getKdbarang() {
        return kdbarang;
    }
    public void setKdbarang(String kdbarang) {
        this.kdbarang= kdbarang;
    }
    public String getNama() {
        return nama;
    }
    public void setNama(String nama) {
        this.nama= nama;
    }

    public int getBeli() {
        return beli;
    }

    public void setBeli(int beli) {
        this.beli = beli;
    }

    public int getJual() {
        return jual;
    }

    public void setJual(int jual) {
        this.jual = jual;
    }

    public int getStok() {
        return stok;
    }

    public void setStok(int stok) {
        this.stok = stok;
```

```
    }
}
```

Mainactivity.java

```
package com.example.databarang;

import androidx.appcompat.app.AppCompatActivity;
import android.app.ListActivity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.ContextMenu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;

import java.util.List;

public class MainActivity extends ListActivity {
    String dataBarang[] = null;
    String dS[] = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        //setContentView(R.layout.activity_main);
        // Tambah Siswa
        DatabaseHandler db = new DatabaseHandler(this);
        // Membaca Semua Siswa
        Log.d("Baca Siswa: ", "Membaca Semua Data Siswa..");
        List<Barang> barang = db.getSemuaBarang();
        dataBarang= new String[barang.size()];
        dS= new String[barang.size()];
        int i=0;
        for(Barang s : barang) {
            /*String log = "Kode Barang: "+ s.getKdbarang() + ",Nama: "+
s.getNama() + ",Harga Beli: "+ s.getJual() + ",Harga Jual: "+ s.getJual() +
",Stok: "+ s.getStok() ;
            Log.d("Name: ", log);*/
            dataBarang[i] = s.getKdbarang() + " - "+ s.getNama() + " - "+
s.getBeli() + " - "+ s.getJual() + " - "+ s.getStok();
            dS[i] = s.getKdbarang();
            i++;
        }

        // check data
        if(i==0)
```

```
        {
            Log.d("Tambah Barang: ", "Menambah Data Barang..");
            db.addBarang(new Barang("001", "Minyak goreng", 15000, 18000,
200));
            db.addBarang(new Barang("002", "Meja", 200000, 300000, 200));
            db.addBarang(new Barang("003", "Kursi", 150000, 200000, 200));
            db.addBarang(new Barang("004", "Mouse", 300000, 400000, 200));
        }
        setListAdapter(new ArrayAdapter<Object>(this,
android.R.layout.simple_list_item_1, dataBarang));
        registerForContextMenu(getListView());
    }


    @Override
    public void onCreateContextMenu(ContextMenu menu, View v,
ContextMenu.ContextMenuInfo menuInfo) {
// TODOAuto-generated method stub
        super.onCreateContextMenu(menu, v, menuInfo);
        menu.setHeaderTitle("Action");
        menu.add(0,0,0,"Tambah");
        menu.add(0,1,1,"Hapus");
        menu.add(0,2,2,"Update");
    }

    @Override
    public boolean onContextItemSelected(MenuItem item) {
// TODOAuto-generated method stub
        try{
            switch(item.getItemId()){
                case 0:{
                    Class c = Class.forName("com.example.databarang.Tambah");
                    Intent i = new Intent(MainActivity.this, c);
                    startActivity(i);break;
                }
                case 1:{
                    DatabaseHandler db = new DatabaseHandler(this);
                    AdapterView.AdapterContextMenuInfo info =
(AdapterView.AdapterContextMenuInfo) item.getMenuInfo();
                    String[] args = {String.valueOf(info.id) };
                    int xpos=Integer.parseInt(args[0]);
                    db.deleteRow(dS[xpos]);
                    Class c =
Class.forName("com.example.databarang.MainActivity");
                    Intent i = new Intent(MainActivity.this, c);
                    startActivity(i);
                    break;
                }
                case 2:{
```

```
                    DatabaseHandler db = new DatabaseHandler(this);
                    AdapterView.AdapterContextMenuInfo info
=(AdapterView.AdapterContextMenuInfo) item.getMenuInfo();
                    String[] args ={String.valueOf(info.id)};
                    Log.d("args0 : ",args[0]);
                    int xpos=Integer.parseInt(args[0]);
                    db.getBarang(dS[xpos]);
                    String namax=db.getBarang(dS[xpos]).getNama();
                    Intent i = new Intent(this, Update.class);
                    Bundle bun = new Bundle();
                    bun.putString("kdbarang", dS[xpos]);
                    bun.putString("nama", namax);
                    i.putExtras(bun);
                    startActivity(i); break;
                }
            }
        }catch(ClassNotFoundException e) {
            // TODOAuto-generated catch block
            e.printStackTrace();
        }
        return true;
    }

}
```

Tambah.java

```
package com.example.databarang;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class Tambah extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tambah);
        final DatabaseHandler db = new DatabaseHandler(this);
        final EditText editKdbrg = (EditText)findViewById(R.id.inputkdbrg);
        final EditText editNama = (EditText) findViewById(R.id.inputnama);
        final EditText editBeli = (EditText) findViewById(R.id.inputbeli);
        final EditText editJual = (EditText) findViewById(R.id.inputjual);
```

```
        final EditText editStok = (EditText) findViewById(R.id.inputstok);
        Button btnTambah = (Button) findViewById(R.id.btnTambah);
        Button btnBatal = (Button) findViewById(R.id.btnCancel);
        btnTambah.setOnClickListener(new View.OnClickListener()
        {
            @Override
            public void onClick(View v) {
                String kdbarang = editKdbrg.getText().toString();
                String nama = editNama.getText().toString();
                int beli = Integer.valueOf(editBeli.getText().toString());
                int jual = Integer.valueOf(editJual.getText().toString());
                int stok = Integer.valueOf(editStok.getText().toString());
                db.addBarang(new Barang(kdbarang,nama,beli,jual,stok));
                editKdbrg.setText("");
                editNama.setText("");
                editBeli.setText("");
                editJual.setText("");
                editStok.setText("");
                try{
                    Class c =
Class.forName("com.example.databarang.MainActivity");
                    Intent i = new Intent(Tambah.this, c);
                    startActivity(i);
                } catch(ClassNotFoundException e) {
                    e.printStackTrace();
                }
            }
        });
        btnBatal.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {

                try {
                    Class c =
Class.forName("com.example.databarang.MainActivity");
                    Intent i=new Intent(Tambah.this,c);
                    startActivity(i);
                } catch (ClassNotFoundException e) {
                    e.printStackTrace();
                }
            }
        });
    }
}
```

Update.java

```
package com.example.databarang;
```

```java
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class Update extends AppCompatActivity {
    private String xkdbarang,xnama;
    private int xbeli,xjual,xstok;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_update);
        final DatabaseHandler db = new DatabaseHandler(this);
        final EditText editKdbrg = (EditText)findViewById(R.id.updatekdbrg);
        final EditText editNama = (EditText) findViewById(R.id.updatenama);
        final EditText editBeli = (EditText) findViewById(R.id.updatebeli);
        final EditText editJual = (EditText) findViewById(R.id.updatejual);
        final EditText editStok = (EditText) findViewById(R.id.updatestok);
        Button btnTambah = (Button) findViewById(R.id.btnUpdate);
        Button btnBatal = (Button) findViewById(R.id.btnCancel);

        // ambil data barang
        Bundle bun = this.getIntent().getExtras();
        xkdbarang = bun.getString("kdbarang");
        xnama = bun.getString("nama");
        xbeli = bun.getInt("hgbeli");
        xjual = bun.getInt("hgjual");
        xstok = bun.getInt("stok");

        //masukkan data barang
        editNama.setText(xnama);
        editKdbrg.setText(xkdbarang);
        editBeli.setText(xbeli);
        editJual.setText(xjual);
        editStok.setText(xstok);

        btnTambah.setOnClickListener(new View.OnClickListener()
        {
            @Override
            public void onClick(View v) {
                String kdbarang = editKdbrg.getText().toString();
                String nama = editNama.getText().toString();
                int beli = Integer.valueOf(editBeli.getText().toString());
                int jual = Integer.valueOf(editJual.getText().toString());
```

```
                int stok = Integer.valueOf(editStok.getText().toString());

                db.updateMethod(kdbarang,nama,beli,jual,stok);
                editKdbrg.setText("");
                editNama.setText("");
                editBeli.setText("");
                editJual.setText("");
                editStok.setText("");

                try{
                    Class c=
Class.forName("com.example.databarang.MainActivity");
                    Intent i = new Intent(Update.this, c);
                    startActivity(i);
                } catch(ClassNotFoundException e) {
                    e.printStackTrace();
                }
            }
        });

        btnBatal.setOnClickListener(new View.OnClickListener()
        {public void onClick(View v) {
            try {
                Class c =
Class.forName("com.example.databarang.MainActivity");
                Intent i=new Intent(Update.this,c);
                startActivity(i);
            } catch (ClassNotFoundException e) {
                e.printStackTrace();
            }
        }
        });
    }
}
```