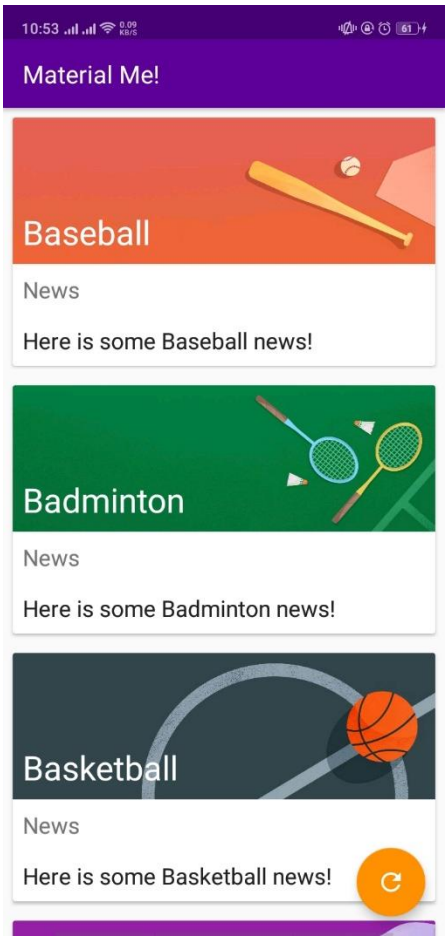
Nama : **Bayu Prasetya Adji Sugiyarto**

NIM : **A11.2019.11688**

Pertemuan : **5**

Kelompok : **A11.4304**

MaterialMe!



String.xml

```
<resources>
    <string name="app_name">Material Me!</string>
    <string name="title_placeholder">Title</string>
    <string name="news_label">News</string>
    <string name="sports_info_placeholder">Here is some news</string>
    <string name="sports_image">Sports Image</string>
    <string name="filler_text">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Maecenas
        vitae semper quam. In a metus ut nisl pharetra hendrerit et a leo.
Curabitur nec sapien
        odio. Vestibulum a mollis felis. Cras molestie felis nibh, ut maximus
mauris feugiat
        tincidunt. Curabitur ultricies eros sed ipsum pulvinar vehicula.
```

```
Maecenas volutpat,
        massa sit amet aliquam eleifend, massa lorem tempor sapien, eget
finibus massa dolor
        molestie lorem. Suspendisse at purus mauris. Proin nibh ligula,
suscipit vel pharetra
        posuere, bibendum id lorem.</string>

    <string-array name="sports_titles">
        <item>Baseball</item>
        <item>Badminton</item>
        <item>Basketball</item>
        <item>Bowling</item>
        <item>Cycling</item>
        <item>Golf</item>
        <item>Running</item>
        <item>Soccer</item>
        <item>Swimming</item>
        <item>Table Tennis</item>
        <item>Tennis</item>
    </string-array>

    <string-array name="sports_info">
        <item>Here is some Baseball news!</item>
        <item>Here is some Badminton news!</item>
        <item>Here is some Basketball news!</item>
        <item>Here is some Bowling news!</item>
        <item>Here is some Cycling news!</item>
        <item>Here is some Golf news!</item>
        <item>Here is some Running news!</item>
        <item>Here is some Soccer news!</item>
        <item>Here is some Swimming news!</item>
        <item>Here is some Table Tennis news!</item>
        <item>Here is some Tennis news!</item>
    </string-array>

    <array name="sports_images">
        <item>@drawable/img_baseball</item>
        <item>@drawable/img_badminton</item>
        <item>@drawable/img_basketball</item>
        <item>@drawable/img_bowling</item>
        <item>@drawable/img_cycling</item>
        <item>@drawable/img_golf</item>
        <item>@drawable/img_running</item>
        <item>@drawable/img_soccer</item>
        <item>@drawable/img_swimming</item>
        <item>@drawable/img_tabletennis</item>
        <item>@drawable/img_tennis</item>
    </array>

</resources>
```

Mainactivity.java

```
package com.example.android.materialme;

import android.content.res.TypedArray;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.helper.ItemTouchHelper;
import android.view.View;
```

```
import java.util.ArrayList;
import java.util.Collections;


public class MainActivity extends AppCompatActivity {

    //Member variables
    private RecyclerView mRecyclerView;
    private ArrayList<Sport> mSportsData;
    private SportsAdapter mAdapter;

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

        //Initialize the RecyclerView
        mRecyclerView = (RecyclerView)findViewById(R.id.recyclerView);

        //Set the Layout Manager
        mRecyclerView.setLayoutManager(new LinearLayoutManager(this));


        //Initialize the ArrayList that will contain the data
        mSportsData = new ArrayList<>();

        //Initialize the adapter and set it ot the RecyclerView
        mAdapter = new SportsAdapter(this, mSportsData);
        mRecyclerView.setAdapter(mAdapter);

        //Get the data
        initializeData();

        //Helper class for creating swipe to dismiss and drag and drop
functionality
        ItemTouchHelper helper = new ItemTouchHelper(new
ItemTouchHelper.SimpleCallback
                (ItemTouchHelper.LEFT | ItemTouchHelper.RIGHT |
ItemTouchHelper.DOWN
                        | ItemTouchHelper.UP, ItemTouchHelper.LEFT |
ItemTouchHelper.RIGHT) {


            @Override
            public boolean onMove(RecyclerView recyclerView,
RecyclerView.ViewHolder viewHolder,
                                  RecyclerView.ViewHolder target) {

                //Get the from and to position
                int from = viewHolder.getAdapterPosition();
                int to = target.getAdapterPosition();

                //Swap the items and notify the adapter
                Collections.swap(mSportsData, from, to);
                mAdapter.notifyItemMoved(from, to);
                return true;
            }

            @Override
            public void onSwiped(RecyclerView.ViewHolder viewHolder, int
direction) {

                //Remove the item from the dataset
                mSportsData.remove(viewHolder.getAdapterPosition());
```

```
                //Notify the adapter
                mAdapter.notifyItemRemoved(viewHolder.getAdapterPosition());
            }
        });

        //Attach the helper to the RecyclerView
        helper.attachToRecyclerView(mRecyclerView);
    }

    /**
     * Method for initializing the sports data from resources.
     */
    private void initializeData() {
        //Get the resources from the XML file
        String[] sportsList =
getResources().getStringArray(R.array.sports_titles);
        String[] sportsInfo =
getResources().getStringArray(R.array.sports_info);
        TypedArray sportsImageResources =
getResources().obtainTypedArray(R.array.sports_images);
        //Clear the existing data (to avoid duplication)
        mSportsData.clear();


        //Create the ArrayList of Sports objects with the titles, images
        // and information about each sport
        for(int i=0; i<sportsList.length; i++){
            mSportsData.add(new Sport(sportsList[i], sportsInfo[i],
                    sportsImageResources.getResourceId(i,0)));
        }

        //Recycle the typed array
        sportsImageResources.recycle();

        //Notify the adapter of the change
        mAdapter.notifyDataSetChanged();
    }

    public void resetSports(View view) {
        initializeData();
    }
}
```

detailactivity.java

```
package com.example.android.materialme;

import android.graphics.Color;
import android.graphics.drawable.Drawable;
import android.graphics.drawable.GradientDrawable;
import android.os.Bundle;
import android.support.v4.content.ContextCompat;
import android.support.v7.app.AppCompatActivity;
import android.widget.ImageView;
import android.widget.TextView;

import com.bumptech.glide.Glide;


public class DetailActivity extends AppCompatActivity {

    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detail);

        //Initialize the views
        TextView sportsTitle = (TextView)findViewById(R.id.titleDetail);
        ImageView sportsImage =
(ImageView)findViewById(R.id.sportsImageDetail);

        //Get the drawable
        Drawable drawable = ContextCompat.getDrawable
                (this,getIntent().getIntExtra(Sport.IMAGE_KEY, 0));

        //Create a placeholder gray scrim while the image loads
        GradientDrawable gradientDrawable = new GradientDrawable();
        gradientDrawable.setColor(Color.GRAY);

        //Make it the same size as the image
        if(drawable!=null) {
            gradientDrawable.setSize(drawable.getIntrinsicWidth(),
drawable.getIntrinsicHeight());
        }

        //Set the text from the Intent extra
        sportsTitle.setText(getIntent().getStringExtra(Sport.TITLE_KEY));

        //Load the image using the glide library and the Intent extra
        Glide.with(this).load(getIntent().getIntExtra(Sport.IMAGE_KEY,0))
                .placeholder(gradientDrawable).into(sportsImage);
    }
}
```

sport.java

```
package com.example.android.materialme;

import android.content.Context;
import android.content.Intent;
import android.support.annotation.DrawableRes;


class Sport {

    //Member variables representing the title, image and information about the
sport
    private final String title;
    private final String info;
    private final int imageResource;

    static final String TITLE_KEY = "Title";
    static final String IMAGE_KEY = "Image Resource";

    Sport(String title, String info, int imageResource) {
        this.title = title;
        this.info = info;
        this.imageResource = imageResource;
    }


    String getTitle() {
        return title;
    }
```

```
    String getInfo() {
        return info;
    }

    int getImageResource() {
        return imageResource;
    }


    static Intent starter(Context context, String title, @DrawableRes int
imageResId) {
        Intent detailIntent = new Intent(context, DetailActivity.class);
        detailIntent.putExtra(TITLE_KEY, title);
        detailIntent.putExtra(IMAGE_KEY, imageResId);
        return detailIntent;
    }
}
```

sportAdapter.java

```
package com.example.android.materialme;

import android.content.Context;
import android.content.Intent;
import android.graphics.Color;
import android.graphics.drawable.Drawable;
import android.graphics.drawable.GradientDrawable;
import android.support.v4.content.ContextCompat;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import com.bumptech.glide.Glide;

import java.util.ArrayList;


class SportsAdapter extends
RecyclerView.Adapter<SportsAdapter.SportsViewHolder>  {

    //Member variables
    private GradientDrawable mGradientDrawable;
    private ArrayList<Sport> mSportsData;
    private Context mContext;


    SportsAdapter(Context context, ArrayList<Sport> sportsData) {
        this.mSportsData = sportsData;
        this.mContext = context;

        //Prepare gray placeholder
        mGradientDrawable = new GradientDrawable();
        mGradientDrawable.setColor(Color.GRAY);

        //Make the placeholder same size as the images
        Drawable drawable = ContextCompat.getDrawable
                (mContext,R.drawable.img_badminton);
        if(drawable != null) {
            mGradientDrawable.setSize(drawable.getIntrinsicWidth(),
```

```
drawable.getIntrinsicHeight());
        }
    }


    @Override
    public SportsViewHolder onCreateViewHolder(ViewGroup parent, int viewType)
{
        return new SportsViewHolder(mContext, LayoutInflater.from(mContext).
                inflate(R.layout.list_item, parent, false), mGradientDrawable);
    }

    @Override
    public void onBindViewHolder(SportsViewHolder holder, int position) {

        //Get the current sport
        Sport currentSport = mSportsData.get(position);

        //Bind the data to the views
        holder.bindTo(currentSport);

    }


    @Override
    public int getItemCount() {
        return mSportsData.size();
    }


    static class SportsViewHolder extends RecyclerView.ViewHolder
            implements View.OnClickListener {

        //Member Variables for the holder data
        private TextView mTitleText;
        private TextView mInfoText;
        private ImageView mSportsImage;
        private Context mContext;
        private Sport mCurrentSport;
        private GradientDrawable mGradientDrawable;

        SportsViewHolder(Context context, View itemView, GradientDrawable
gradientDrawable) {
            super(itemView);

            //Initialize the views
            mTitleText = (TextView)itemView.findViewById(R.id.title);
            mInfoText = (TextView)itemView.findViewById(R.id.subTitle);
            mSportsImage = (ImageView)itemView.findViewById(R.id.sportsImage);

            mContext = context;
            mGradientDrawable = gradientDrawable;

            //Set the OnClickListener to the whole view
            itemView.setOnClickListener(this);
        }

        void bindTo(Sport currentSport){
            //Populate the textviews with data
            mTitleText.setText(currentSport.getTitle());
            mInfoText.setText(currentSport.getInfo());

            //Get the current sport
            mCurrentSport = currentSport;
```

```
            //Load the images into the ImageView using the Glide library
            Glide.with(mContext).load(currentSport.

getImageResource()).placeholder(mGradientDrawable).into(mSportsImage);
        }

        @Override
        public void onClick(View view) {

            //Set up the detail intent
            Intent detailIntent = Sport.starter(mContext,
mCurrentSport.getTitle(),
                    mCurrentSport.getImageResource());


            //Start the detail activity
            mContext.startActivity(detailIntent);
        }
    }
}
```

activity_main.xml

```
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context="com.example.android.materialme.MainActivity">

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

  <android.support.design.widget.FloatingActionButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom|end|right"
    android:layout_margin="@dimen/activity_horizontal_margin"
    android:src="@drawable/ic_reset"
    android:tint="@android:color/white"
    android:onClick="resetSports"/>

</FrameLayout>
```

Activity_detail.xml

```
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.android.materialme.MainActivity">
```

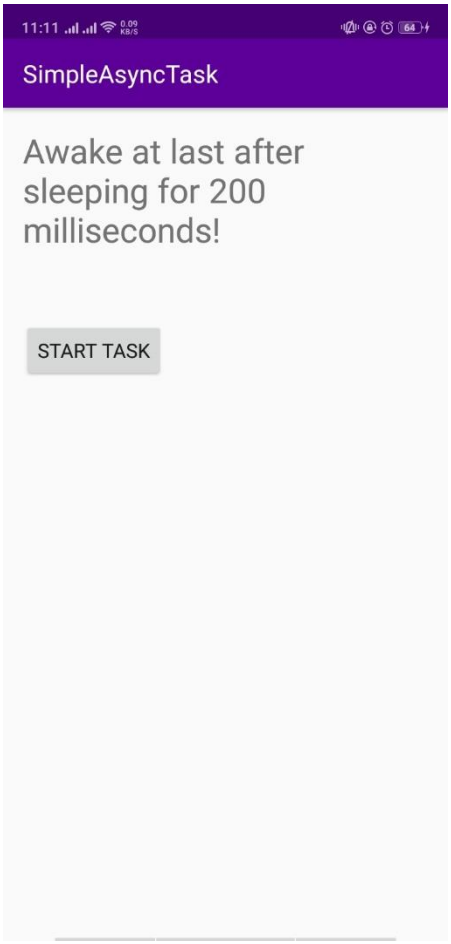
```
    <android.support.v7.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:scrollbars="vertical"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

    <android.support.design.widget.FloatingActionButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|end|right"
        android:layout_margin="@dimen/activity_horizontal_margin"
        android:src="@drawable/ic_reset"
        android:tint="@android:color/white"
        android:onClick="resetSports"/>

</FrameLayout>
```

List_item.xml

```
<android.support.v7.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:layout_margin="@dimen/card_margin"
  android:background="?android:selectableItemBackground">

  <RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <ImageView
      android:id="@+id/sportsImage"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:adjustViewBounds="true"/>

    <TextView
      android:id="@+id/title"
      style="@style/TextAppearance.AppCompat.Headline"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_alignBottom="@id/sportsImage"
      android:padding="@dimen/card_margin"
      android:text="@string/title_placeholder"
      android:theme="@style/ThemeOverlay.AppCompat.Dark"/>

    <TextView
      android:id="@+id/newsTitle"
      style="@style/TextAppearance.AppCompat.Subhead"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_below="@id/sportsImage"
      android:padding="@dimen/card_margin"
      android:text="@string/news_label"
      android:textColor="?android:textColorSecondary" />

    <TextView
```

```
        android:id="@+id/subTitle"
        style="@style/TextAppearance.AppCompat.Subhead"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/newsTitle"
        android:padding="@dimen/card_margin"
        android:text="@string/sports_info_placeholder" />
    </RelativeLayout>
</android.support.v7.widget.CardView>
```

SimpleAsyncTask



String.xml

```
<resources>
    <string name="app_name">SimpleAsyncTask</string>
    <string name="ready_to_start">I am ready to start work!</string>
    <string name="start_task">Start Task</string>
    <string name="napping">"Napping…"</string>
</resources>
```

MainActivity.java

```
package android.example.com.simpleasynctask;

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

public class MainActivity extends AppCompatActivity {

    //Key for saving the state of the TextView
    private static final String TEXT_STATE = "currentText";

    // The TextView where we will show results
    private TextView mTextView;
```

```
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //  Initialize mTextView
        mTextView = (TextView) findViewById(R.id.textView1);

        // Restore TextView if there is a savedInstanceState
        if(savedInstanceState!=null){
           mTextView.setText(savedInstanceState.getString(TEXT_STATE));
        }
    }

    public void startTask (View view) {
        // Put a message in the text view
        mTextView.setText(R.string.napping);

        // Start the AsyncTask.
        // The AsyncTask has a callback that will update the text view.
        new SimpleAsyncTask(mTextView).execute();
    }


    @Override
    protected void onSaveInstanceState(Bundle outState) {
        super.onSaveInstanceState(outState);
        // Save the state of the TextView
        outState.putString(TEXT_STATE, mTextView.getText().toString());
    }
}
```

SimpleAsyncTask.java

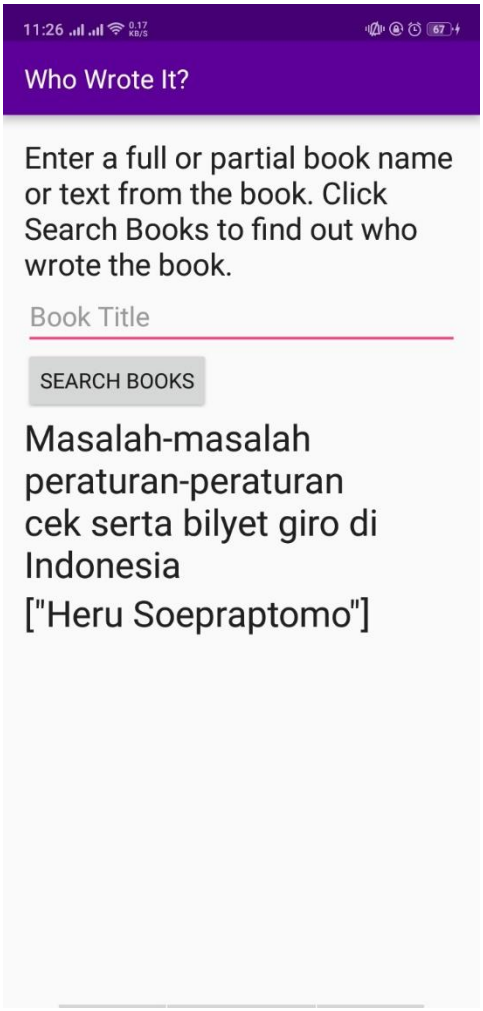
```
package android.example.com.simpleasynctask;

import android.os.AsyncTask;
import android.widget.TextView;

import java.util.Random;

class SimpleAsyncTask extends AsyncTask<Void, Void, String> {

    // The TextView where we will show results
    private TextView mTextView;

    // Constructor that provides a reference to the TextView from the
MainActivity
    public SimpleAsyncTask(TextView tv) {
        mTextView = tv;
    }

    @Override
    protected String doInBackground(Void... voids) {

        // Generate a random number between 0 and 10
        Random r = new Random();
        int n = r.nextInt(11);

        // Make the task take long enough that we have
        // time to rotate the phone while it is running
        int s = n * 200;
```

```
            // Sleep for the random amount of time
            try {
                Thread.sleep(s);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }

            // Return a String result
            return "Awake at last after sleeping for " + s + " milliseconds!";
        }

        protected void onPostExecute(String result) {
            mTextView.setText(result);
        }
    }
```

activity_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:orientation="vertical">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/ready_to_start"
        android:id = "@+id/textView1"
        android:textSize="24sp"/>

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/start_task"
        android:id="@+id/button"
        android:layout_marginTop="56dp"
        android:onClick="startTask" />
</LinearLayout>
```

WhoWroteIt



String.xml

```
<resources>
    <string name="app_name">Who Wrote It?</string>
    <string name="instructions">Enter a full or partial book name or text from
the book. Click Search Books to find out who wrote the book. </string>
    <string name="button_text">Search Books</string>
    <string name="input_hint">Book Title</string>
    <string name="no_search_term">Please enter a search term</string>
    <string name="no_network">Please check your network connection and try
again.</string>
    <string name="no_results">No Results Found</string>
</resources>
```

Mainactivity.java

```
package com.example.android.whowroteit;

import android.content.Context;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
```

```
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.view.inputmethod.InputMethodManager;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    // Variables for the search input field, and results TextViews.
    private EditText mBookInput;
    private TextView mTitleText;
    private TextView mAuthorText;


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

        // Initialize all the view variables.
        mBookInput = (EditText)findViewById(R.id.bookInput);
        mTitleText = (TextView)findViewById(R.id.titleText);
        mAuthorText = (TextView)findViewById(R.id.authorText);
    }

    public void searchBooks(View view) {
        // Get the search string from the input field.
        String queryString = mBookInput.getText().toString();

        // Hide the keyboard when the button is pushed.
        InputMethodManager inputManager = (InputMethodManager)
                getSystemService(Context.INPUT_METHOD_SERVICE);

inputManager.hideSoftInputFromWindow(getCurrentFocus().getWindowToken(),
                InputMethodManager.HIDE_NOT_ALWAYS);

        // Check the status of the network connection.
        ConnectivityManager connMgr = (ConnectivityManager)
                getSystemService(Context.CONNECTIVITY_SERVICE);
        NetworkInfo networkInfo = connMgr.getActiveNetworkInfo();

        // If the network is active and the search field is not empty, start a
FetchBook AsyncTask.
        if (networkInfo != null && networkInfo.isConnected() &&
queryString.length()!=0) {
            new FetchBook(mTitleText, mAuthorText,
mBookInput).execute(queryString);
        }
        // Otherwise update the TextView to tell the user there is no
connection or no search term.
        else {
            if (queryString.length() == 0) {
                mAuthorText.setText("");
                mTitleText.setText(R.string.no_search_term);
            } else {
                mAuthorText.setText("");
                mTitleText.setText(R.string.no_network);
            }
        }
    }
}
```

FetchBook.java

```
package com.example.android.whowroteit;

import android.net.Uri;
import android.os.AsyncTask;
import android.widget.EditText;
import android.widget.TextView;

import org.json.JSONArray;
import org.json.JSONObject;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;

public class FetchBook extends AsyncTask<String,Void,String>{

    // Variables for the search input field, and results TextViews
    private EditText mBookInput;
    private TextView mTitleText;
    private TextView mAuthorText;

    // Class name for Log tag
    private static final String LOG_TAG = FetchBook.class.getSimpleName();

    // Constructor providing a reference to the views in MainActivity
    public FetchBook(TextView titleText, TextView authorText, EditText
bookInput) {
        this.mTitleText = titleText;
        this.mAuthorText = authorText;
        this.mBookInput = bookInput;
    }

    @Override
    protected String doInBackground(String... params) {

        // Get the search string
        String queryString = params[0];


        // Set up variables for the try block that need to be closed in the
finally block.
        HttpURLConnection urlConnection = null;
        BufferedReader reader = null;
        String bookJSONString = null;

        // Attempt to query the Books API.
        try {
            // Base URI for the Books API.
            final String BOOK_BASE_URL =
"https://www.googleapis.com/books/v1/volumes?";

            final String QUERY_PARAM = "q"; // Parameter for the search string.
            final String MAX_RESULTS = "maxResults"; // Parameter that limits
search results.
            final String PRINT_TYPE = "printType"; // Parameter to filter by
print type.

            // Build up your query URI, limiting results to 10 items and
printed books.
```

```
            Uri builtURI = Uri.parse(BOOK_BASE_URL).buildUpon()
                    .appendQueryParameter(QUERY_PARAM, queryString)
                    .appendQueryParameter(MAX_RESULTS, "10")
                    .appendQueryParameter(PRINT_TYPE, "books")
                    .build();

            URL requestURL = new URL(builtURI.toString());

            // Open the network connection.
            urlConnection = (HttpURLConnection) requestURL.openConnection();
            urlConnection.setRequestMethod("GET");
            urlConnection.connect();

            // Get the InputStream.
            InputStream inputStream = urlConnection.getInputStream();

            // Read the response string into a StringBuilder.
            StringBuilder builder = new StringBuilder();

            reader = new BufferedReader(new InputStreamReader(inputStream));

            String line;
            while ((line = reader.readLine()) != null) {
              // Since it's JSON, adding a newline isn't necessary (it won't
affect parsing)
                // but it does make debugging a *lot* easier if you print out the
completed buffer for debugging.
                builder.append(line + "\n");
            }

            if (builder.length() == 0) {
                // Stream was empty.  No point in parsing.
                // return null;
                return null;
            }
            bookJSONString = builder.toString();

        // Catch errors.
        } catch (IOException e) {
            e.printStackTrace();

        // Close the connections.
        } finally {
            if (urlConnection != null) {
                urlConnection.disconnect();
            }
            if (reader != null) {
                try {
                    reader.close();
                } catch (IOException e) {
                    e.printStackTrace();
                }

            }
        }

        // Return the raw response.
        return bookJSONString;
    }

    @Override
    protected void onPostExecute(String s) {
        super.onPostExecute(s);
        try {
```

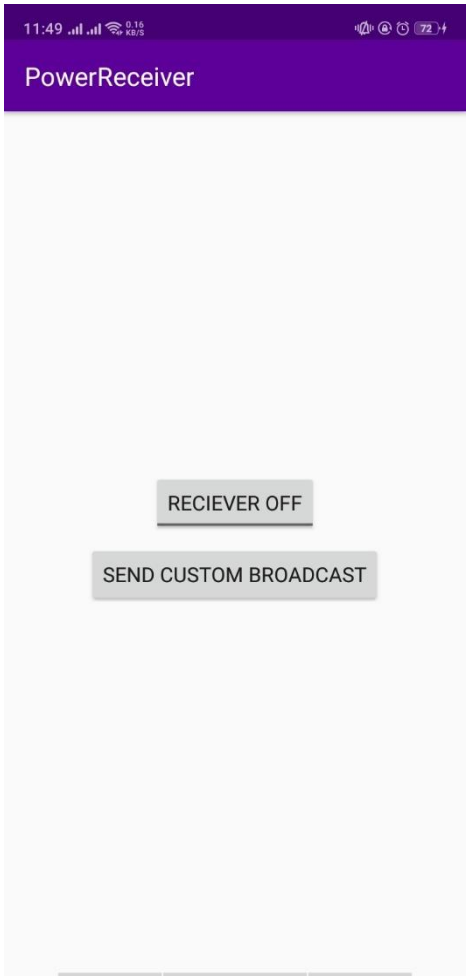
```
            // Convert the response into a JSON object.
            JSONObject jsonObject = new JSONObject(s);
            // Get the JSONArray of book items.
            JSONArray itemsArray = jsonObject.getJSONArray("items");

            // Initialize iterator and results fields.
            int i = 0;
            String title = null;
            String authors = null;

            // Look for results in the items array, exiting when both the title
and author
            // are found or when all items have been checked.
            while (i < itemsArray.length() || (authors == null && title ==
null)) {
                // Get the current item information.
                JSONObject book = itemsArray.getJSONObject(i);
                JSONObject volumeInfo = book.getJSONObject("volumeInfo");

                // Try to get the author and title from the current item,
                // catch if either field is empty and move on.
                try {
                    title = volumeInfo.getString("title");
                    authors = volumeInfo.getString("authors");
                } catch (Exception e){
                    e.printStackTrace();
                }

                // Move to the next item.
                i++;
            }

            // If both are found, display the result.
            if (title != null && authors != null){
                mTitleText.setText(title);
                mAuthorText.setText(authors);
                mBookInput.setText("");
            } else {
                // If none are found, update the UI to show failed results.
                mTitleText.setText(R.string.no_results);
                mAuthorText.setText("");
            }

        } catch (Exception e){
            // If onPostExecute does not receive a proper JSON string,
            // update the UI to show failed results.
            mTitleText.setText(R.string.no_results);
            mAuthorText.setText("");
            e.printStackTrace();
        }
    }
}
```

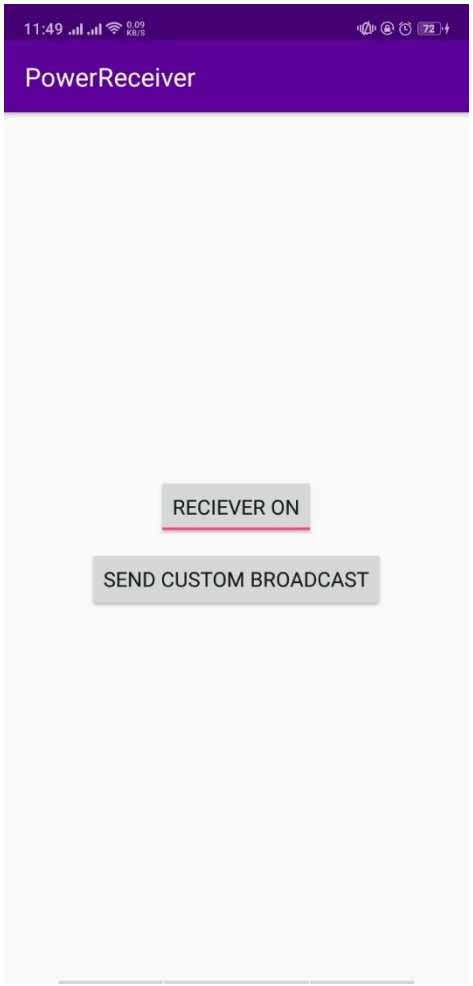
activity_main.xml

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
```

```
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:orientation="vertical"
    tools:context="com.example.android.whowroteit.MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/instructions"
        android:text="@string/instructions"
        android:textAppearance="@style/TextAppearance.AppCompat.Title"/>

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/bookInput"
        android:inputType="text"
        android:hint="@string/input_hint"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/searchButton"
        android:text="@string/button_text"
        android:onClick="searchBooks" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/titleText"
        android:textAppearance="@style/TextAppearance.AppCompat.Headline"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/authorText"
        android:textAppearance="@style/TextAppearance.AppCompat.Headline"/>
</LinearLayout>
```

PowerReciever





Stirng.xml

```
<resources>
    <string name="app_name">PowerReceiver</string>
    <string name="power_connected">Power connected!</string>
    <string name="power_disconnected">Power disconnected!</string>
    <string name="custom_broadcast_toast">Custom Broadcast Received</string>
    <string name="send_custom_broadcast">Send Custom Broadcast</string>
    <string name="receiver_on">Receiver On</string>
    <string name="receiver_off">Receiver Off</string>
</resources>
```

Mainactivity.java

```
package com.example.android.powerreceiver;

import android.content.ComponentName;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.pm.PackageManager;
import android.os.Bundle;
```

```
import android.support.v4.content.LocalBroadcastManager;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;
import android.widget.CompoundButton;
import android.widget.ToggleButton;

import java.util.function.ToLongBiFunction;

public class MainActivity extends AppCompatActivity {

    private CustomReceiver mReceiver = new CustomReceiver();
    private ComponentName mReceiverComponentName;
    private PackageManager mPackageManager;

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

        Button broadcastButton = (Button) findViewById(R.id.broadcastButton);

        //Get the PackageManager and ComponentName so you can toggle to
broadcast receiver.
        mReceiverComponentName = new ComponentName(this, CustomReceiver.class);
        mPackageManager = getPackageManager();

        //Use LocalBroadcastManager so that the broadcast is not received by
other applications.
        LocalBroadcastManager.getInstance(this).registerReceiver
                (mReceiver, new
IntentFilter(CustomReceiver.ACTION_CUSTOM_BROADCAST));

        //onClick method for the button
        broadcastButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                sendCustomBroadcast();
            }
        });

        ToggleButton Simpletogglebutton = (ToggleButton)
findViewById(R.id.tbutton);
        Simpletogglebutton.setText("Reciever Off");
        Simpletogglebutton.setTextOn("Reciever On");
        Simpletogglebutton.setTextOff("Reciever Off");

        Simpletogglebutton.setOnCheckedChangeListener(new
CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton buttonView, boolean
isChecked) {
                if (isChecked) {
                    onStart();
                }else {
                    onStop();
                }
            }
        });

    }


    @Override
```

```
    protected void onStart() {
        //Enable the broadcast receiver when the app is visible
        mPackageManager.setComponentEnabledSetting
                (mReceiverComponentName,
PackageManager.COMPONENT_ENABLED_STATE_ENABLED,
                        PackageManager.DONT_KILL_APP);
        super.onStart();
    }

    @Override
    protected void onStop() {
        //Disable the broadcast receiver when the app is visible
        mPackageManager.setComponentEnabledSetting
                (mReceiverComponentName,
PackageManager.COMPONENT_ENABLED_STATE_DISABLED,
                        PackageManager.DONT_KILL_APP);
        super.onStop();
    }

    @Override
    protected void onDestroy() {
        LocalBroadcastManager.getInstance(this).unregisterReceiver(mReceiver);
        super.onDestroy();
    }

    private void sendCustomBroadcast() {
        Intent customBroadcastIntent = new
Intent(CustomReceiver.ACTION_CUSTOM_BROADCAST);

LocalBroadcastManager.getInstance(this).sendBroadcast(customBroadcastIntent);
    }
}
```

CustomReciever.java

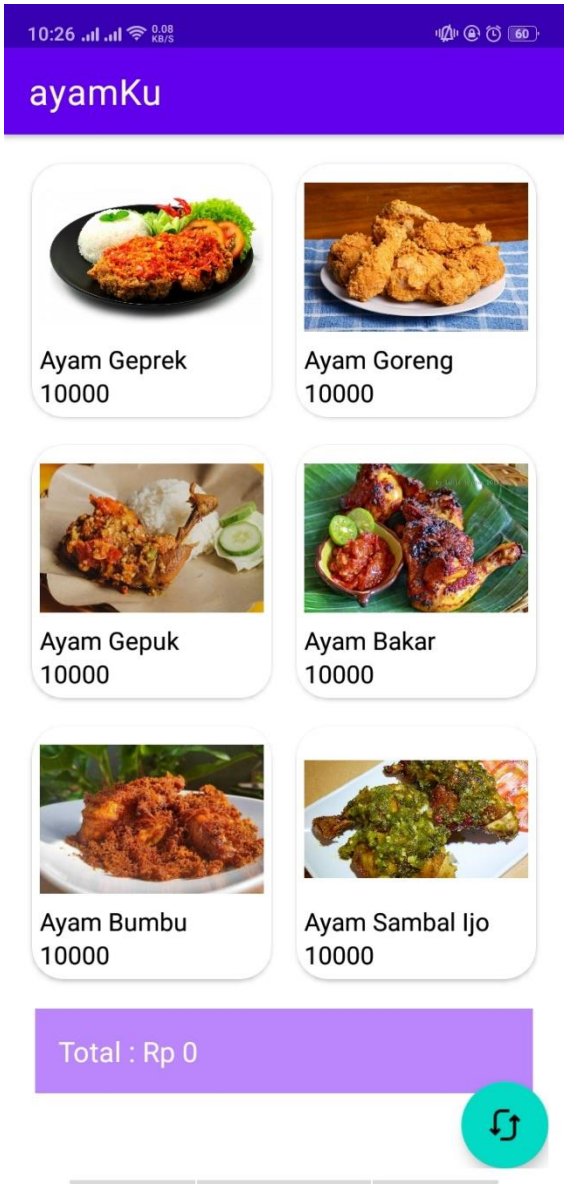
```
package com.example.android.powerreceiver;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.widget.Toast;


public class CustomReceiver extends BroadcastReceiver {

    //String constant that defines the custom Broadcast Action
    static final String ACTION_CUSTOM_BROADCAST =
            "com.example.android.powerreceiver.ACTION_CUSTOM_BROADCAST";


    //Empty constructor
    public CustomReceiver() {
    }


    @Override
    public void onReceive(Context context, Intent intent) {
        String intentAction = intent.getAction();
        String toastMessage = null;
        switch (intentAction){
            case Intent.ACTION_POWER_CONNECTED:
                toastMessage = context.getString(R.string.power_connected);
                break;
```

```
            case Intent.ACTION_POWER_DISCONNECTED:
                toastMessage = context.getString(R.string.power_disconnected);
                break;
            case ACTION_CUSTOM_BROADCAST:
                toastMessage =
context.getString(R.string.custom_broadcast_toast);
                break;
        }

        Toast.makeText(context, toastMessage, Toast.LENGTH_SHORT).show();
    }

}
```

activity_main.java

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.android.powerreceiver.MainActivity">


    <Button
        android:id="@+id/broadcastButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:text="@string/send_custom_broadcast"/>
</LinearLayout>
```

AyamKu



String.xml

```
<resources>
    <string name="app_name">ayamKu</string>

    <string-array name="title">
        <item>Ayam Geprek</item>
        <item>Ayam Goreng</item>
        <item>Ayam Gepuk</item>
        <item>Ayam Bakar</item>
        <item>Ayam Bumbu</item>
        <item>Ayam Sambal Ijo</item>
    </string-array>

    <string-array name="harga">
        <item>10000</item>
        <item>10000</item>
        <item>10000</item>
        <item>10000</item>
```

```
        <item>10000</item>
        <item>10000</item>
    </string-array>

    <string-array name="image">
        <item>@drawable/ayamgeprek</item>
        <item>@drawable/ayamgoreng</item>
        <item>@drawable/ayamgepuk</item>
        <item>@drawable/ayambakar</item>
        <item>@drawable/ayambumbu</item>
        <item>@drawable/ayamsambalijo</item>
    </string-array>
</resources>
```

Mainactivity.java

```
package com.example.ayamku;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.GridLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.content.res.TypedArray;
import android.os.Bundle;
import android.view.MotionEvent;
import android.view.View;
import android.widget.TextView;

import java.lang.reflect.Type;
import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {
    private RecyclerView recyclerView;
    private ayamadapter adapter;
    private ArrayList<ayam> ayamlist;
    private int total = 0;
    private TextView totalView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ayamlist = new ArrayList<>();
        totalView = (TextView)  findViewById(R.id.total);
        setTotal(0);
        addData();

        recyclerView = (RecyclerView) findViewById(R.id.recycle_view);
        adapter = new ayamadapter(ayamlist);
        RecyclerView.LayoutManager layoutManager = new
GridLayoutManager(this,2);

        recyclerView.setLayoutManager(layoutManager);
        recyclerView.setAdapter(adapter);
        recyclerView.addOnItemTouchListener(new
RecyclerView.OnItemTouchListener() {
            @Override
            public boolean onInterceptTouchEvent(@NonNull RecyclerView rv,
@NonNull MotionEvent e) {
```

```
                TextView total = (TextView) rv.findViewById(R.id.harga);

                setTotal(Integer.valueOf(String.valueOf(total.getText())));
                return true;
            }

            @Override
            public void onTouchEvent(@NonNull RecyclerView rv, @NonNull
MotionEvent e) {

            }

            @Override
            public void onRequestDisallowInterceptTouchEvent(boolean
disallowIntercept) {

            }
        });
    }

    public void  addData(){
        String[] title_list = getResources().getStringArray(R.array.title);
        String[] harga_list = getResources().getStringArray(R.array.harga);
        TypedArray image_list =
getResources().obtainTypedArray(R.array.image);

        for (int i = 0; i < title_list.length; i++){
            ayamlist.add(new ayam(title_list[i], harga_list[i],
image_list.getResourceId(i, 0)));
        }
    }

    public void setTotal(int total){
        this.total = this.total + total;
        totalView.setText("Total : Rp "+ String.valueOf(this.total));
    }

    public void reset (View view){
        this.total = 0;
        setTotal(0);
    }
}
```

ayam.java

```
package com.example.ayamku;

public class ayam {
    private String title;
    private String harga;
    private int image;

    public ayam (String title, String harga, int image){
        this.title = title;
        this.harga = harga;
        this.image = image;
    }

    public String getTitle(){
        return title;
```

```
    }

    public String getHarga(){
        return harga;
    }

    public int getImage(){
        return image;
    }


}
```

ayamadapter.java

```
package com.example.ayamku;

import android.text.Layout;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;


public class ayamadapter extends
RecyclerView.Adapter<ayamadapter.AyamViewHolder> {

    private ArrayList<ayam> dataList;
    public ayamadapter(ArrayList<ayam> dataList){
        this.dataList = dataList;
    }
    public static class AyamViewHolder extends RecyclerView.ViewHolder{
        private TextView nama, price;
        private TextView total;
        private ImageView image;

        public AyamViewHolder (View itemView){
            super(itemView);
            image = (ImageView) itemView.findViewById(R.id.image);
            nama = (TextView) itemView.findViewById(R.id.title);
            price = (TextView) itemView.findViewById(R.id.harga);
        }
    }
    @NonNull
    @Override
    public AyamViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType){
        LayoutInflater layoutInflater =
LayoutInflater.from(parent.getContext());
        View view = layoutInflater.inflate(R.layout.list_item, parent,
false);
        return new AyamViewHolder(view);
    }
    @Override
```

```
    public int getItemCount(){

        return (dataList != null) ? dataList.size():0;
    }
    @Override
    public void  onBindViewHolder(@NonNull ayamadapter.AyamViewHolder
holder, int position){
        holder.nama.setText(dataList.get(position).getTitle());
        holder.price.setText(dataList.get(position).getHarga());
        holder.image.setImageResource(dataList.get(position).getImage());
    }


}
```

activity_main.java

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
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_marginHorizontal="10dp"
    android:layout_marginVertical="10dp"
    tools:context=".MainActivity">
<ScrollView
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <include layout="@layout/content_main"/>
</ScrollView>

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/reset"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|end"
        android:onClick="reset"
        android:src="@drawable/refresh"
        />

</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

Content_main.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="match_parent"
    android:orientation="vertical">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycle_view"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

    <TextView
        android:id="@+id/total"
```

```
        android:layout_margin="10dp"
        android:padding="15dp"
        android:background="@color/purple_200"
        android:textSize="18dp"
        android:text="HAI"
        android:textColor="@color/white"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        />

</LinearLayout>
```

List_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    app:cardCornerRadius="20dp"
    app:cardUseCompatPadding="true">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="5dp">

    <ImageView
        android:id="@+id/image"
        android:layout_width="match_parent"
        android:layout_height="110dp"/>
    <TextView
        android:id="@+id/title"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="@color/black" />
    <TextView
        android:id="@+id/harga"
        android:textColor="@color/black"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

</LinearLayout>


</androidx.cardview.widget.CardView>
```