**MAIN ACTIVITY CODE:**  
package com.myproject.fbapplication;  
import android.content.Intent;  
import android.support.v7.app.ActionBarActivity;  
import android.support.v4.app.Fragment;  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.ExpandableListView;  
  
import com.facebook.Session;  
import com.facebook.UiLifecycleHelper;  
import com.facebook.SessionState;  
  
import java.util.ArrayList;  
import java.util.HashMap;  
import java.util.List;  
  
public class MainActivity extends ActionBarActivity {  
  
    private UiLifecycleHelper uiHelper;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity\_main);  
        uiHelper = new UiLifecycleHelper(this, callback);  
        uiHelper.onCreate(savedInstanceState);  
        if (savedInstanceState == null) {  
            getSupportFragmentManager().beginTransaction()  
                    .add(R.id.container, new PlaceholderFragment())  
                    .commit();  
        }  
    }  
  
    private Session.StatusCallback callback = new Session.StatusCallback() {  
        @Override  
        public void call(Session session, SessionState state,  
                         Exception exception) {  
            onSessionStateChange(session, state, exception);  
        }  
    };  
  
    private void onSessionStateChange(Session session, SessionState state,  
                                      Exception exception) {  
        if (state.isOpened()) {  
            getSupportFragmentManager().beginTransaction()  
                    .replace(R.id.container, new UserFragment())  
                    .commit();  
        } else if (state.isClosed()) {  
            getSupportFragmentManager().beginTransaction()  
                    .replace(R.id.container, new PlaceholderFragment())  
                    .commit();  
        }  
    }  
  
    @Override  
    public void onActivityResult(int requestCode, int resultCode, Intent data) {  
        super.onActivityResult(requestCode, resultCode, data);  
        uiHelper.onActivityResult(requestCode, resultCode, data);  
    }  
  
    @Override  
    public void onResume() {  
        super.onResume();  
        uiHelper.onResume();  
    }  
  
    @Override  
    public void onPause() {  
        super.onPause();  
        uiHelper.onPause();  
    }  
  
    @Override  
    public void onDestroy() {  
        super.onDestroy();  
        uiHelper.onDestroy();  
    }  
  
    @Override  
    public void onSaveInstanceState(Bundle outState) {  
        super.onSaveInstanceState(outState);  
        uiHelper.onSaveInstanceState(outState);  
    }  
  
    @Override  
    public boolean onCreateOptionsMenu(Menu menu) {  
        // Inflate the menu; this adds items to the action bar if it is present.  
        getMenuInflater().inflate(R.menu.menu\_main, menu);  
        return true;  
    }  
  
    @Override  
    public boolean onOptionsItemSelected(MenuItem item) {  
        // Handle action bar item clicks here. The action bar will  
        // automatically handle clicks on the Home/Up button, so long  
        // as you specify a parent activity in AndroidManifest.xml.  
        int id = item.getItemId();  
  
        //noinspection SimplifiableIfStatement  
        if (id == R.id.action\_settings) {  
            return true;  
        }  
  
        return super.onOptionsItemSelected(item);  
    }  
  
    public void donateMenu(View view) {  
        getSupportFragmentManager().beginTransaction()  
                .replace(R.id.container, new DonateFragment())  
                .commit();  
    }  
  
    public void receiveMenu(View view) {  
  
    }  
  
    /\*\*  
     \* A placeholder fragment containing a simple view.  
     \*/  
    public static class PlaceholderFragment extends Fragment {  
  
        public PlaceholderFragment() {  
        }  
  
        @Override  
        public View onCreateView(LayoutInflater inflater, ViewGroup container,  
                                 Bundle savedInstanceState) {  
            View rootView = inflater.inflate(R.layout.fragment\_main, container, false);  
  
            return rootView;  
        }  
  
    }  
  
    public static class UserFragment extends Fragment {  
  
        public UserFragment() {  
        }  
  
        @Override  
        public View onCreateView(LayoutInflater inflater, ViewGroup container,  
                                 Bundle savedInstanceState) {  
            View rootView = inflater.inflate(R.layout.user, container, false);  
            return rootView;  
        }  
    }  
  
    public static class DonateFragment extends Fragment {  
  
        ExpandableListView expandableListView;  
        ExpandableListAdapter expandableListAdapter;  
        List<String> expandableListTitle;  
        HashMap<String, List<String>> expandableListDetail;  
  
        public DonateFragment() {  
        }  
  
        @Override  
        public View onCreateView(LayoutInflater inflater, ViewGroup container,  
                                 Bundle savedInstanceState) {  
            View rootView = inflater  
                    .inflate(R.layout.activity\_list\_view\_android\_example, container, false);  
            expandableListView = (ExpandableListView) rootView.findViewById(R.id.expandableListView);  
            expandableListDetail = ExpandableListData.getData();  
            expandableListTitle = new ArrayList<String>(expandableListDetail.keySet());  
            expandableListAdapter = new ExpandableListAdapter(  
                    getActivity().getApplicationContext(),  
                    expandableListTitle,  
                    expandableListDetail  
            );  
            expandableListView.setAdapter(expandableListAdapter);  
            /\*expandableListView.setOnGroupExpandListener(  
                    new ExpandableListView.OnGroupExpandListener() {  
  
                        @Override  
                        public void onGroupExpand(int groupPosition) {  
                            Toast.makeText(getActivity().getApplicationContext(),  
                                expandableListTitle.get(groupPosition) + " List Expanded.",  
                                Toast.LENGTH\_SHORT).show();  
                        }  
                    }  
            );\*/  
  
            /\*expandableListView.setOnGroupCollapseListener(  
                    new ExpandableListView.OnGroupCollapseListener() {  
  
                        @Override  
                        public void onGroupCollapse(int groupPosition) {  
                            Toast.makeText(  
                                    getActivity().getApplicationContext(),  
                                    expandableListTitle.get(groupPosition) + " List Collapsed.",  
                                    Toast.LENGTH\_SHORT  
                            ).show();  
                        }  
                    }  
            );\*/  
  
            /\*expandableListView.setOnChildClickListener(  
                    new ExpandableListView.OnChildClickListener() {  
  
                        @Override  
                        public boolean onChildClick(ExpandableListView parent, View v,  
                                                    int groupPosition, int childPosition, long id) {  
                            Toast.makeText(  
                                    getActivity().getApplicationContext(),  
                                    expandableListTitle.get(groupPosition)  
                                            + " -> "  
                                            + expandableListDetail.get(  
                                                    expandableListTitle.get(groupPosition)  
                                            ).get(childPosition),  
                                    Toast.LENGTH\_SHORT  
                            ).show();  
                    return false;  
                }  
            });\*/  
            /\*// Get ListView object from xml  
            ListView listView = (ListView) rootView.findViewById(R.id.list);  
  
            // Defined Array values to show in ListView  
            String[] values = new String[] {  
                    "Android List View",  
                    "Adapter implementation",  
                    "Simple List View In Android",  
                    "Create List View Android",  
                    "Android Example",  
                    "List View Source Code",  
                    "List View Array Adapter",  
                    "Android Example List View"  
            };  
  
            // Define a new Adapter  
            // First parameter - Context  
            // Second parameter - Layout for the row  
            // Third parameter - ID of the TextView to which the data is written  
            // Forth - the Array of data  
  
            ArrayAdapter<String> adapter = new ArrayAdapter<String>(  
                    getActivity().getApplicationContext(),  
                    android.R.layout.simple\_list\_item\_1,  
                    android.R.id.text1,  
                    values  
            );  
  
            // Assign adapter to ListView  
            listView.setAdapter(adapter);\*/  
  
            return rootView;  
        }  
    }  
}

**JAVA CODE:**

package com.myproject.fbapplication;  
  
import android.content.Context;  
import android.graphics.Typeface;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.BaseExpandableListAdapter;  
import android.widget.TextView;  
  
import java.util.HashMap;  
import java.util.List;  
  
public class ExpandableListAdapter extends BaseExpandableListAdapter {  
  
    private Context context;  
    private List<String> expandableListTitle;  
    private HashMap<String, List<String>> expandableListDetail;  
  
    public ExpandableListAdapter(Context context, List<String> expandableListTitle,  
                                 HashMap<String, List<String>> expandableListDetail) {  
        this.context = context;  
        this.expandableListTitle = expandableListTitle;  
        this.expandableListDetail = expandableListDetail;  
    }  
  
    @Override  
    public Object getChild(int listPosition, int expandedListPosition) {  
        return this.expandableListDetail.get(this.expandableListTitle.get(listPosition))  
                .get(expandedListPosition);  
    }  
  
    @Override  
    public long getChildId(int listPosition, int expandedListPosition) {  
        return expandedListPosition;  
    }  
  
    @Override  
    public View getChildView(int listPosition, final int expandedListPosition,  
                             boolean isLastChild, View convertView, ViewGroup parent) {  
        final String expandedListText = (String) getChild(listPosition, expandedListPosition);  
        if (convertView == null) {  
            LayoutInflater layoutInflater = (LayoutInflater) this.context  
                    .getSystemService(Context.LAYOUT\_INFLATER\_SERVICE);  
            convertView = layoutInflater.inflate(R.layout.list\_item, null);  
        }  
        TextView expandedListTextView = (TextView) convertView  
                .findViewById(R.id.expandedListItem);  
        expandedListTextView.setText(expandedListText);  
        return convertView;  
    }  
  
    @Override  
    public int getChildrenCount(int listPosition) {  
        return this.expandableListDetail.get(this.expandableListTitle.get(listPosition))  
                .size();  
    }  
  
    @Override  
    public Object getGroup(int listPosition) {  
        return this.expandableListTitle.get(listPosition);  
    }  
  
    @Override  
    public int getGroupCount() {  
        return this.expandableListTitle.size();  
    }  
  
    @Override  
    public long getGroupId(int listPosition) {  
        return listPosition;  
    }  
  
    @Override  
    public View getGroupView(int listPosition, boolean isExpanded,  
                             View convertView, ViewGroup parent) {  
        String listTitle = (String) getGroup(listPosition);  
        if (convertView == null) {  
            LayoutInflater layoutInflater = (LayoutInflater) this.context.  
                    getSystemService(Context.LAYOUT\_INFLATER\_SERVICE);  
            convertView = layoutInflater.inflate(R.layout.list\_group, null);  
        }  
        TextView listTitleTextView = (TextView) convertView  
                .findViewById(R.id.listTitle);  
        listTitleTextView.setTypeface(null, Typeface.BOLD);  
        listTitleTextView.setText(listTitle);  
        return convertView;  
    }  
  
    @Override  
    public boolean hasStableIds() {  
        return false;  
    }  
  
    @Override  
    public boolean isChildSelectable(int listPosition, int expandedListPosition) {  
        return true;  
    }  
}