To Do List Application

Aim :

Develop a To Do List Application for Mobile Phone using Android Studio.

Theory :

To Do List is a list of tasks you need to complete, or things that you want to do. Most typically, they're organised in order of priority. Traditionally, they're written on a piece of paper or post it notes and acts as a memory aid. As technology has evolved, we have been able to create a to do lists with excel spreadsheets, word documents, email lists, to do list apps, Microsoft to do and google to do list to name a few. You can use a to do list in your home and personal life, or in the workplace.

Having a list of everything you need to do written down in one place means you shouldn't forget anything important. By prioritising the tasks in the list, you plan the order in which you're going to do them and can quickly see what needs your immediate attention and what tasks you can leave until a little later.

The Benefits of Using a To Do List

One of the most important reasons you should use a to do list is that it will help you stay organised. When you write all your tasks in a list, they seem more manageable. When you've got a clear outline of the tasks you've got to do and those you've completed, it helps you stay focused. While freeing up space in your mind for other more creative tasks.

When you complete a task, you can cross it off your list. This gives you a sense of progress and achievement, something you'll lack if you're always rushing from one task to the next. If you feel a sense of achievement, it spurs you on and motivates you to keep moving forward.

Improves your memory: A to do list acts as an external memory aid. It's only possible to hold a few pieces of information at one time. Keep a to do list and you'll be able to keep track of everything, rather than just a few of the tasks you need to do. Your to do list will also reinforce the information, which makes it less likely you're going to forget something.

Increases productivity: A to do list allows you to prioritize the tasks that are more important. This means you don't waste time on tasks that don't require your immediate attention. Your list will help you stay focused on the tasks that are the most important.

Helps with motivation: To do lists are a great motivational tool because you can use them to clarify your goals. You can divide your long-term goal into smaller, more achievable short-term goals and as you tick each one off your list, your confidence will increase.

Requirements :

- Android Studio
- Knowledge of XML and Java
- Android emulator (or) Android mobile

Features of Application :

- Navigation Drawer
- Floating Action Button
- Material Themes and Animations
- Swipe to Delete
- WebView
- Action Bar

Steps for Creating To Do List Application :

Step 1: Create a **New Project**

Create a new project in Android Studio and select **Java** as the programming language.

Step 2: Working with **activity_main.xml** file

Code :

activity_main.xml file

```
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:id="@+id/drawer_layout"
    android:layout_width="match_parent" android:layout_height="match_parent"
    tools:context=".MainActivity">

    <RelativeLayout android:id="@+id/container" android:layout_width="match_parent"
        android:layout_height="match_parent" android:background="@android:color/white">
        <ListView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:id="@+id/list"
            android:layout_weight="1"
            android:layout_margin="5dp"/>

        <ImageButton
            android:id="@+id/fab_image_button"
            android:layout_width="@dimen/fab_button_diameter"
            android:layout_height="@dimen/fab_button_diameter"
            android:layout_alignParentEnd="true"
            android:layout_alignParentBottom="true"
            android:layout_marginBottom="@dimen/fab_button_margin_bottom"
            android:layout_marginRight="@dimen/fab_button_margin_right"
            android:background="@drawable/oval_ripple"
            android:src="@mipmap/fab_ic_add"
            android:tint="@android:color/white"
            android:elevation="@dimen/elevation_low"
            android:stateListAnimator="@anim/button_elevation"/>

    </RelativeLayout>

    <fragment android:id="@+id/navigation_drawer"
        android:layout_width="@dimen/navigation_drawer_width" android:layo
ut_height="match_parent"
        android:layout_gravity="start" android:name="chaos.list.Navigation
DrawerFragment"
        tools:layout="@layout/fragment_navigation_drawer" />

</android.support.v4.widget.DrawerLayout>
```

Step 3: Working with **MainActivity.java** file

Code :

MainActivity.java file

```
package chaos.list;

import android.app.Activity;
import android.app.AlertDialog;
import android.content.ContentValues;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.support.v7.app.ActionBarActivity;
import android.support.v7.app.ActionBar;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.content.Context;
import android.os.Build;
import android.os.Bundle;
import android.view.Gravity;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.support.v4.widget.DrawerLayout;
import android.widget.ArrayAdapter;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.ListAdapter;
import android.widget.ListView;
import android.widget.SimpleCursorAdapter;
import android.widget.TextView;

import java.util.ArrayList;


public class MainActivity extends ActionBarActivity
        implements NavigationDrawerFragment.NavigationDrawerCallbacks {

    private ListView myList;
    private ListAdapter todoListAdapter;
    private TodoListSQLHelper todoListSQLHelper;

    private NavigationDrawerFragment mNavigationDrawerFragment;
    private CharSequence mTitle;

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

        myList = (ListView) findViewById(R.id.list);
        ImageButton fabImageButton = (ImageButton) findViewById(R.id.fab_i
mage_button);

        mNavigationDrawerFragment = (NavigationDrawerFragment)
     getSupportFragmentManager().findFragmentById(R.id.navigation_drawer);
```

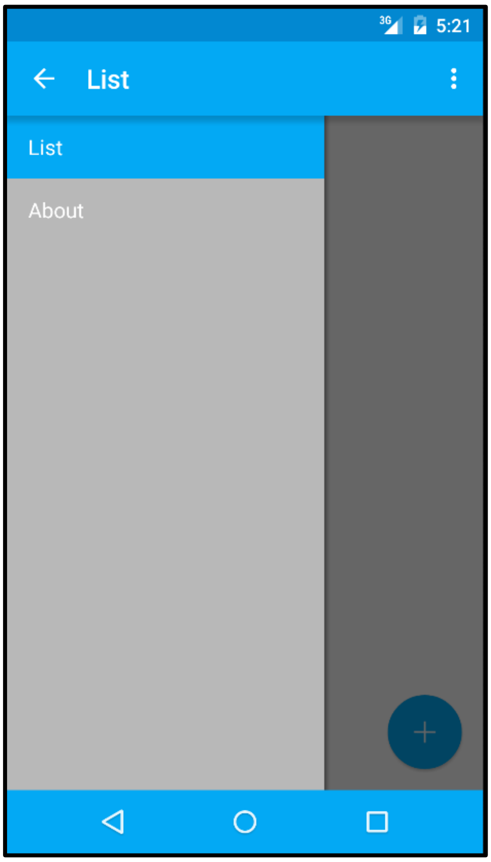
```
        mTitle = getTitle();

        mNavigationDrawerFragment.setUp(
                R.id.navigation_drawer,
                (DrawerLayout) findViewById(R.id.drawer_layout));
        final ArrayList<String> list = new ArrayList<>();
        final MyCustomAdapter adapter = new MyCustomAdapter(this, list);
        SwipeDismissListViewTouchListener touchListener =
                new SwipeDismissListViewTouchListener(
                        (ListView) findViewById(R.id.list),
                new SwipeDismissListViewTouchListener.DismissCallbacks() {
                            @Override
                            public boolean canDismiss(int position) {
                                return true;
                            }

                            @Override
  public void onDismiss(ListView listView, int[] reverseSortedPositions) {
                             for (int position : reverseSortedPositions) {

String deleteTodoItemSql = "DELETE FROM " + TodoListSQLHelper.TABLE_NAME +
                               " WHERE " + TodoListSQLHelper._ID+ " = '" +
 todoListAdapter.getItemId(position) + "'";

             todoListSQLHelper = new TodoListSQLHelper(MainActivity.this);
           SQLiteDatabase sqlDB = todoListSQLHelper.getWritableDatabase();
                                    sqlDB.execSQL(deleteTodoItemSql);
                                    updateTodoList();
                                }
                            }
                        });
        findViewById(R.id.list).setOnTouchListener(touchListener);

        fabImageButton.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                list.add("New Item");
                adapter.notifyDataSetChanged();
                AlertDialog.Builder todoTaskBuilder = new AlertDialog.Buil
der(MainActivity.this);
                todoTaskBuilder.setTitle("Add a List item.");
                todoTaskBuilder.setMessage("Describe the item.");
                final EditText todoET = new EditText(MainActivity.this);
                todoTaskBuilder.setView(todoET);
                todoTaskBuilder.setPositiveButton("Add Item", new DialogIn
terface.OnClickListener() {
```

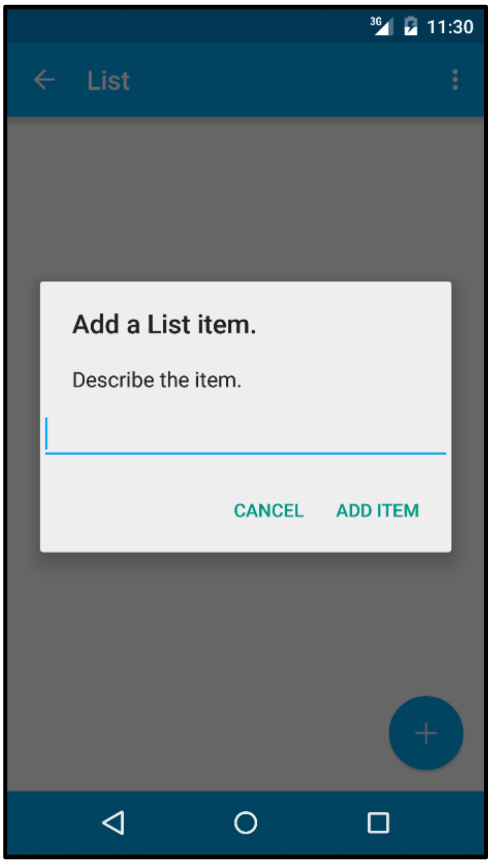
```
                    @Override
             public void onClick(DialogInterface dialogInterface, int i) {
                       String todoTaskInput = todoET.getText().toString();
             todoListSQLHelper = new TodoListSQLHelper(MainActivity.this);
  SQLiteDatabase sqLiteDatabase = todoListSQLHelper.getWritableDatabase();
                        ContentValues values = new ContentValues();
                        values.clear();

                   values.put(TodoListSQLHelper.COL1_TASK, todoTaskInput);
                        sqLiteDatabase.insertWithOnConflict(TodoListSQLHel
per.TABLE_NAME, null, values, SQLiteDatabase.CONFLICT_IGNORE);

                        updateTodoList();
                    }
                });
                todoTaskBuilder.setNegativeButton("Cancel", null);
                todoTaskBuilder.create().show();
            }
        });
        updateTodoList();
    }

    @Override
    public void onNavigationDrawerItemSelected(int position) {
        FragmentManager fragmentManager = getSupportFragmentManager();
        switch (position){
            case 0:
                fragmentManager.beginTransaction()
   .replace(R.id.container, PlaceholderFragment.newInstance(position + 1))
                        .commit();
                break;
            case 1:
                Intent intent = new Intent(this, About.class);
                startActivity(intent);
                break;
        }
    }

    public void onSectionAttached(int number) {
        switch (number) {
            case 1:
                mTitle = getString(R.string.title_section1);
                break;
            case 2:
                mTitle = getString(R.string.title_section2);
                break;
        }
    }

    public void restoreActionBar() {
        ActionBar actionBar = getSupportActionBar();
        actionBar.setNavigationMode(ActionBar.NAVIGATION_MODE_STANDARD);
        actionBar.setDisplayShowTitleEnabled(true);
        actionBar.setTitle(mTitle);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        if (!mNavigationDrawerFragment.isDrawerOpen()) {
            getMenuInflater().inflate(R.menu.main, menu);
            restoreActionBar();
            return true;
        }
        return super.onCreateOptionsMenu(menu);
    }
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.action_settings) {
            ArrayList<String> list = new ArrayList<>();

            MyCustomAdapter adapter = new MyCustomAdapter(this, list);

            list.add("New Item");
            adapter.notifyDataSetChanged();
            AlertDialog.Builder todoTaskBuilder = new AlertDialog.Builder(
MainActivity.this);
            todoTaskBuilder.setTitle("Add List Item.");
            todoTaskBuilder.setMessage("Describe the item.");
            final EditText todoET = new EditText(MainActivity.this);
            todoTaskBuilder.setView(todoET);
            todoTaskBuilder.setPositiveButton("Add Task", new DialogInterf
ace.OnClickListener() {

                @Override
             public void onClick(DialogInterface dialogInterface, int i) {
                    String todoTaskInput = todoET.getText().toString();
             todoListSQLHelper = new TodoListSQLHelper(MainActivity.this);
  SQLiteDatabase sqLiteDatabase = todoListSQLHelper.getWritableDatabase();
                    ContentValues values = new ContentValues();
                    values.clear();

                   values.put(TodoListSQLHelper.COL1_TASK, todoTaskInput);
```

```
                    sqLiteDatabase.insertWithOnConflict(TodoListSQLHelper.
TABLE_NAME, null, values, SQLiteDatabase.CONFLICT_IGNORE);

                    updateTodoList();
                    sqLiteDatabase.close();
                }
            });
            todoTaskBuilder.setNegativeButton("Cancel", null);
            todoTaskBuilder.create().show();

            updateTodoList();
            return true;
        }
        return super.onOptionsItemSelected(item);
    }

    public static class PlaceholderFragment extends Fragment {
        private static final String ARG_SECTION_NUMBER = "section_number";

        public static PlaceholderFragment newInstance(int sectionNumber) {
            PlaceholderFragment fragment = new PlaceholderFragment();
            Bundle args = new Bundle();
            args.putInt(ARG_SECTION_NUMBER, sectionNumber);
            fragment.setArguments(args);
            return fragment;
        }
        public PlaceholderFragment() {
        }

        @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                                 Bundle savedInstanceState) {
            View rootView = inflater.inflate(R.layout.fragment_main, conta
iner, false);
            return rootView;
        }

        @Override
        public void onAttach(Activity activity) {
            super.onAttach(activity);
            ((MainActivity) activity).onSectionAttached(
                    getArguments().getInt(ARG_SECTION_NUMBER));
        }
    }
    private void updateTodoList() {
        todoListSQLHelper = new TodoListSQLHelper(MainActivity.this);
  SQLiteDatabase sqLiteDatabase = todoListSQLHelper.getReadableDatabase();
```

```
        Cursor cursor = sqLiteDatabase.query(TodoListSQLHelper.TABLE_NAME,
         new String[]{TodoListSQLHelper._ID, TodoListSQLHelper.COL1_TASK},
                null, null, null, null, null);

        todoListAdapter = new SimpleCursorAdapter(
                this,
                R.layout.due,
                cursor,
                new String[]{TodoListSQLHelper.COL1_TASK},
                new int[]{R.id.due_text_view},
                0
        );
        myList.setAdapter(todoListAdapter);
    }

    public void onDoneButtonClick(View view) {
        View v = (View) view.getParent();
        TextView todoTV = (TextView) v.findViewById(R.id.due_text_view);
        String todoTaskItem = todoTV.getText().toString();

String deleteTodoItemSql = "DELETE FROM " + TodoListSQLHelper.TABLE_NAME +
    " WHERE " + TodoListSQLHelper.COL1_TASK + " = '" + todoTaskItem + "'";

        todoListSQLHelper = new TodoListSQLHelper(MainActivity.this);
        SQLiteDatabase sqlDB = todoListSQLHelper.getWritableDatabase();
        sqlDB.execSQL(deleteTodoItemSql);
        updateTodoList();
        sqlDB.close();
    }
}
```

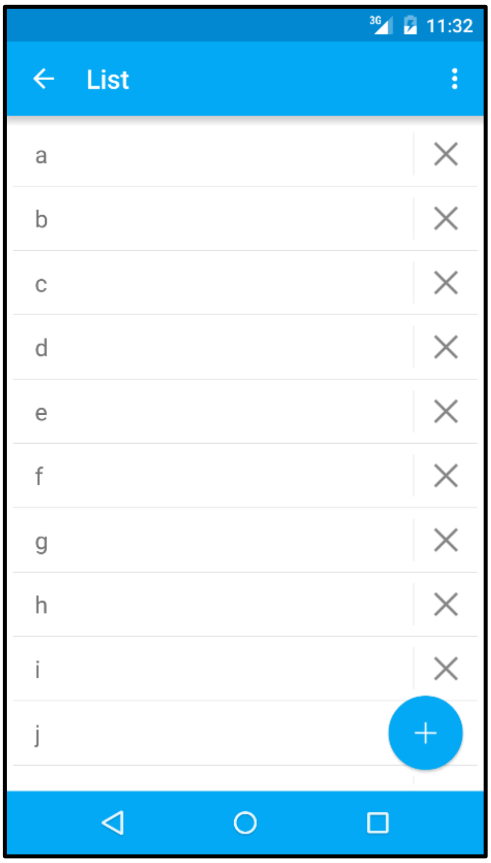
<u>Application Screenshots</u> :





Add a List item.

Describe the item.


11:32
List
a
b
c
d
e
f
g
h
i
j