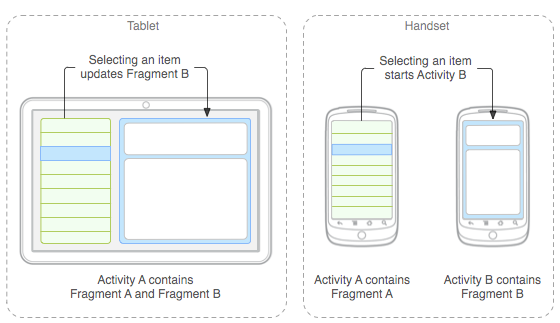
Tutorial 09 – Fragments

In this tutorial, we will learn how to create layouts that can adopt to larger screens using fragments. Assume, we are writing an e-mail application and this is how we want it to behave:



Please, create a new Android Studio project for this tutorial task.

# Task 0 – Add array of data to show in the list

In real life e-mail application, we would gotten email list from the mail server. However, in this tutorial, let us simplify our task and simply create variables that hold an array of emails for us. We will populate our lists from data in them.

Please create a new java class named EmailContent and paste this code.

**public final class** EmailContent {  
 **public static final** String[] ***EMAIL\_TITLES*** = {  
 **"Your email confirmation"**,  
 **"You won lottery ($1M)"**,  
 **"I can't get to you"**,  
 **"The class is cancelled"** };  
  
 **public static final** String[] ***EMAIL\_BODIES*** = {  
 **"This is a very long email body for email with title: Your email confirmation"**,  
 **"Hey, hey, hay! You won lottery $1 000 000!!!\n Please send $2000 for bank transfer fees."**,  
 **"Body: I can't get to you"**,  
 **"Another body: The class is cancelled"** };  
}

# Task 1 – Create main layout

As you can see from the image above, the app has two “sections” to hold our fragments. In addition, our activity\_main.xml file should have different “versions” (portrait and landscape). Change your activity\_main.xml layout file to hold list fragment only and a version for landscape with the list and a details view.

**activity\_main.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="horizontal" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**fragment  
 android:id="@+id/list"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"** />  
</**LinearLayout**>

**activity\_main-land.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="horizontal" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**fragment  
 android:id="@+id/list"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"** />  
 <**FrameLayout  
 android:id="@+id/details"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="2"**/>  
</**LinearLayout**>

# Task 2 – Create Fragment classes

Now, we need to create our fragments. Let’s name them MailListFragment and DetailsFragment.

1. In Android Studio, go to “Project” pane (ALT+1), right click on your java package folder and choose:   
   New > Fragment > Fragment (Blank).
2. Let us call it MailListFragment. Uncheck the factory and callback checkboxes. We will write them ourselves.
3. Since our fragment is a list, let’s change it to extend from ListFragment instead of Fragment class.
4. Let us create the second Fragment class as well. Name it DetailsFragment.

**public class** DetailsFragment **extends** Fragment {  
  
 **public static** DetailsFragment newInstance(**int** index) {  
 DetailsFragment f = **new** DetailsFragment();  
 Bundle args = **new** Bundle();  
 args.putInt(**"index"**, index);  
 f.setArguments(args);  
 **return** f;  
 }  
  
 **public int** getShownIndex() {  
 **return** getArguments().getInt(**"index"**, 0);  
 }  
  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
 ScrollView scroller = **new** ScrollView(getActivity());  
 TextView text = **new** TextView(getActivity());  
 **int** padding = (**int**) TypedValue.*applyDimension*(TypedValue.***COMPLEX\_UNIT\_DIP***,  
 4, getActivity().getResources().getDisplayMetrics());  
 text.setPadding(padding, padding, padding, padding);  
 scroller.addView(text);  
 text.setText(EmailContent.***BODIES***[getShownIndex()]);  
 **return** scroller;  
 }  
}

1. Create an Activity that will show the details on devices in portrait mode. It is denoted as “Activity B” in the picture above. Let’s call it DetailsActivity.

**public class** DetailsActivity **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_details***);  
  
 Intent intent = getIntent();  
 **int** index = intent.getIntExtra(**"index"**, 0);  
 TextView tvBody = findViewById(R.id.***tvBody***);  
 tvBody.setText(EmailContent.***BODIES***[index]);  
 }  
}

1. Don’t forget to put a text view in the corresponding activity layout file.

<**TextView  
 android:id="@+id/tvBody"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"** />

1. Copy-paste the code below into your MailListFragment class. Let’s go through the code line by line.

**public class** MailListFragment **extends** ListFragment {  
  
 **boolean mDualPane**;  
 **int mCurCheckPosition** = 0;  
  
 @Override  
 **public void** onActivityCreated(Bundle savedInstanceState) {  
 **super**.onActivityCreated(savedInstanceState);  
  
 setListAdapter(**new** ArrayAdapter<String>(getActivity(),  
 android.R.layout.***simple\_list\_item\_activated\_1***, EmailContent.***TITLES***));  
  
 View detailsFrame = getActivity().findViewById(R.id.***details***);  
 **mDualPane** = detailsFrame != **null** && detailsFrame.getVisibility() == View.***VISIBLE***;  
  
 **if** (savedInstanceState != **null**) {  
 **mCurCheckPosition** = savedInstanceState.getInt(**"curChoice"**, 0);  
 }  
  
 **if** (**mDualPane**) {  
 getListView().setChoiceMode(ListView.***CHOICE\_MODE\_SINGLE***);  
 showDetails(**mCurCheckPosition**);  
 }  
 }  
  
 @Override  
 **public void** onSaveInstanceState(Bundle outState) {  
 **super**.onSaveInstanceState(outState);  
 outState.putInt(**"curChoice"**, **mCurCheckPosition**);  
 }  
  
 @Override  
 **public void** onListItemClick(ListView l, View v, **int** position, **long** id) {  
 showDetails(position);  
 }  
  
 **void** showDetails(**int** index) {  
 **mCurCheckPosition** = index;  
  
 **if** (**mDualPane**) {  
 getListView().setItemChecked(index, **true**);  
  
 DetailsFragment details = (DetailsFragment)  
 getActivity().getSupportFragmentManager().findFragmentById(R.id.***details***);  
 **if** (details == **null** || details.getShownIndex() != index) {  
 details = DetailsFragment.*newInstance*(index);  
  
 FragmentTransaction ft = getActivity().getSupportFragmentManager().beginTransaction();  
 ft.replace(R.id.***details***, details);  
 ft.setTransition(FragmentTransaction.***TRANSIT\_FRAGMENT\_FADE***);  
 ft.commit();  
 }  
  
 } **else** {  
 Intent intent = **new** Intent();  
 intent.setClass(getActivity(), DetailsActivity.**class**);  
 intent.putExtra(**"index"**, index);  
 startActivity(intent);  
 }  
 }  
}

1. In your main\_activity.xml file, add class property to fragment tag and point it to the newly created MailListFragment class.

<**fragment  
 class="uz.wiut.sabduvaliev.tw10fragments.MailListFragment"  
 android:id="@+id/list"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"** />

1. Run the app.

