CSE2MAD Lab Week 7 – Fragments

Aim: To create an activity that contains two fragments that independently perform an operation and update their UI based on information coming from the parent container.



Ensure your Build.gradle (module) has

implementation 'androidx.appcompat:appcompat:1.2.0'  
implementation 'androidx.legacy:legacy-support-v4:1.0.0'

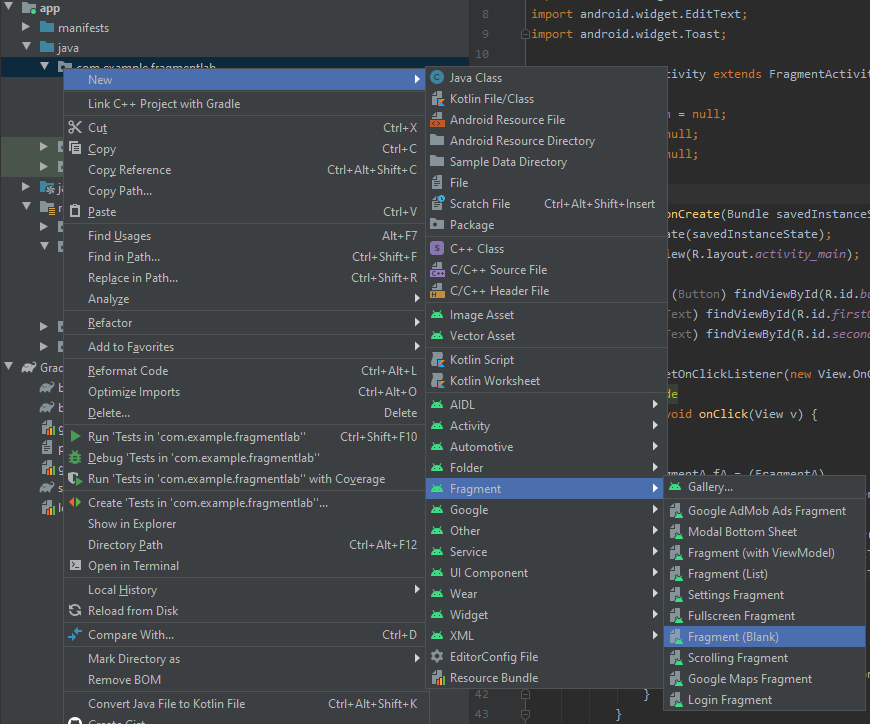
1. Create the parent activity

Ensure you import androidx.fragment.app.FragmentActivity; and extend FragmentActivity as your MainActivity

As above we need two fragments, two EditText’s and a button. Before you go too deep with the UI, lets create the fragments.

1. Let’s create the fragments. Their job is to handle a set of parameters and display the result of a calculation. You will go to the res/layout folder of your project and create the UI. Here is my XML.

Right click on your java package and create two new fragments, I called mine FragementA and FragmentB



These fragments will have a basic UI according to these XML’s. You may replace the existing XML with these for each fragment in the res/layout folder.

fragment\_a.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="200dp" android:layout\_height="200dp"  
 android:orientation="vertical">  
 <TextView  
 android:id="@+id/resultText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_gravity="center\_horizontal|center\_vertical"  
 android:layout\_marginTop="20dip"  
 android:background="@android:color/holo\_red\_dark"  
 android:text="."  
 android:textAppearance="?android:attr/textAppearanceLarge"  
 android:textSize="30dip" />  
</LinearLayout>

fragment\_b.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="200dp" android:layout\_height="200dp"  
 android:orientation="vertical">  
 <TextView  
 android:id="@+id/resultText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_gravity="center\_horizontal|center\_vertical"  
 android:layout\_marginTop="20dip"  
 android:background="@android:color/holo\_blue\_dark"  
 android:text="."  
 android:textAppearance="?android:attr/textAppearanceLarge"  
 android:textSize="30dip" />  
</LinearLayout>

3) You need code to implement the functionality of the fragment.

There will be 2 java files in your code folder.

For FragmentA.java

Modify as per below

@Override  
public View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
  
 View view = inflater.inflate(R.layout.*fragment\_a*, container, false);  
 tv = view.findViewById(R.id.*resultText*);  
 // Inflate the layout for this fragment  
 return view;  
}  
  
public void doCalcDisplay(int a, int b) {  
 // perform the multiplication operation and update the textview in this fragment  
 tv.setText("" + (a\*b));  
}

For FragmentB.java

Modify as per below

public View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
 View view = inflater.inflate(R.layout.*fragment\_b*,  
 container, false);  
 tv = view.findViewById(R.id.*resultText*);  
 // Inflate the layout for this fragment  
 return view;  
}  
  
public void doCalcDisplay(int a, int b) {  
 // perform the addition operation and update the textview in this fragment  
 tv.setText("" + (a+b));  
}

As you can see we have to extend Fragment and override ‘onCreateView’, this is a pre‐req for making fragments. If you look carefully it’s like a mini Activity. The doCalcDisplay() method performs a calc and updates the textView.

Note: We inflate the fragment using is resource ID, remember your filenames of the XML.

4) Revisit the layout for your main activity.

Your XML for the main activity will be similar to.

<?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:id="@+id/container" android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 android:orientation="vertical"  
 tools:context="com.example.fragmentlab.MainActivity">  
 <fragment  
 android:id="@+id/fragmentA"  
 android:name="com.example.fragmentlab.FragmentA"  
 android:layout\_width="match\_parent"  
 android:layout\_height="100dp"  
 tools:layout\_editor\_absoluteX="102dp"  
 tools:layout\_editor\_absoluteY="177dp" />  
 <fragment  
 android:id="@+id/fragmentB"  
 android:name="com.example.fragmentlab.FragmentB"  
 android:layout\_width="match\_parent"  
 android:layout\_height="100dp"  
 tools:layout\_editor\_absoluteX="102dp"  
 tools:layout\_editor\_absoluteY="305dp" />  
 <EditText  
 android:id="@+id/firstOperandET"  
 android:layout\_width="match\_parent" android:layout\_height="44dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text=""  
 tools:layout\_editor\_absoluteX="16dp"  
 tools:layout\_editor\_absoluteY="16dp" />  
 <EditText  
 android:id="@+id/secondOperandET"  
 android:layout\_width="match\_parent" android:layout\_height="44dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text=""  
 tools:layout\_editor\_absoluteX="130dp"  
 tools:layout\_editor\_absoluteY="16dp" />  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" android:text="Pass  
to Fragments" tools:layout\_editor\_absoluteX="240dp"  
 tools:layout\_editor\_absoluteY="16dp" />  
</LinearLayout>

5) This is the final stage, in the MainActivity java file we need to send input to the fragments.

package com.example.fragmentlab;  
  
import androidx.fragment.app.FragmentActivity;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
public class MainActivity extends FragmentActivity {  
  
 Button resButton = null;  
 EditText opA = null;  
 EditText opB = null;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 resButton = (Button) findViewById(R.id.*button*);  
 opA = (EditText) findViewById(R.id.*firstOperandET*);  
 opB = (EditText) findViewById(R.id.*secondOperandET*);  
  
 resButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
  
  
 FragmentA fA = (FragmentA)  
 getSupportFragmentManager().findFragmentById(R.id.*fragmentA*);  
 FragmentB fB = (FragmentB)  
 getSupportFragmentManager().findFragmentById(R.id.*fragmentB*);  
 int a = Integer.*parseInt*(opA.getText().toString());  
 int b = Integer.*parseInt*(opB.getText().toString());  
 if(a !=0 && b!=0) {  
 fA.doCalcDisplay(a, b);  
 fB.doCalcDisplay(a, b);  
 } else {  
 Toast.*makeText*(getApplicationContext(),"Please enter non-zero integers", Toast.*LENGTH\_SHORT*);  
 }  
 }  
 });  
 }  
}

Note\_ We obtain the framgents in a simlar way to the UI widgets, instead of using findViewById() we use findFragmentById().

You may wish to do some range and format checking on the input data obtained from the EditText’s.

END OF PART A