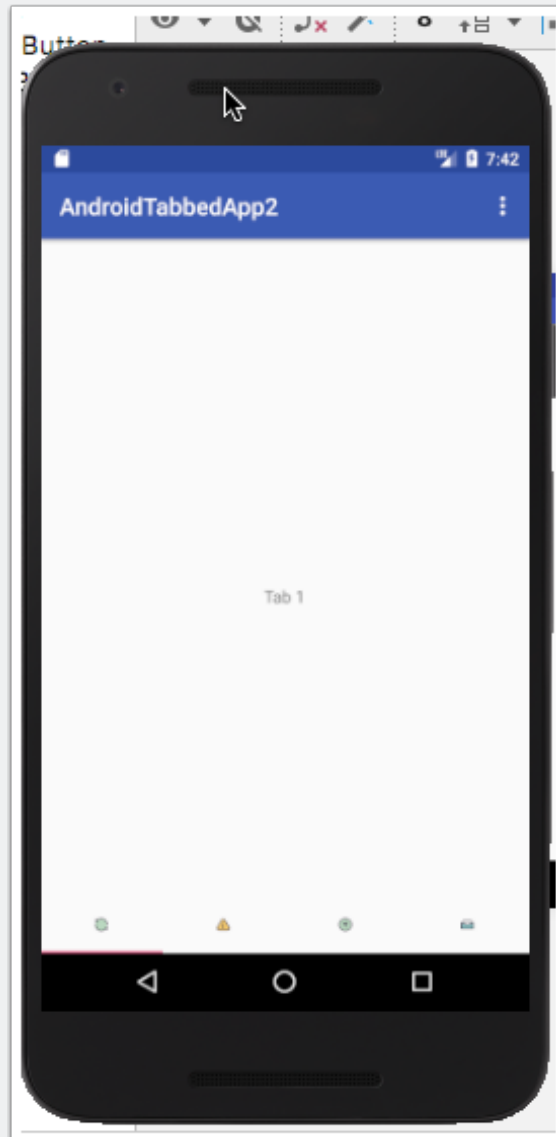


# Tabbed App with Custom Icons

## Step by Step

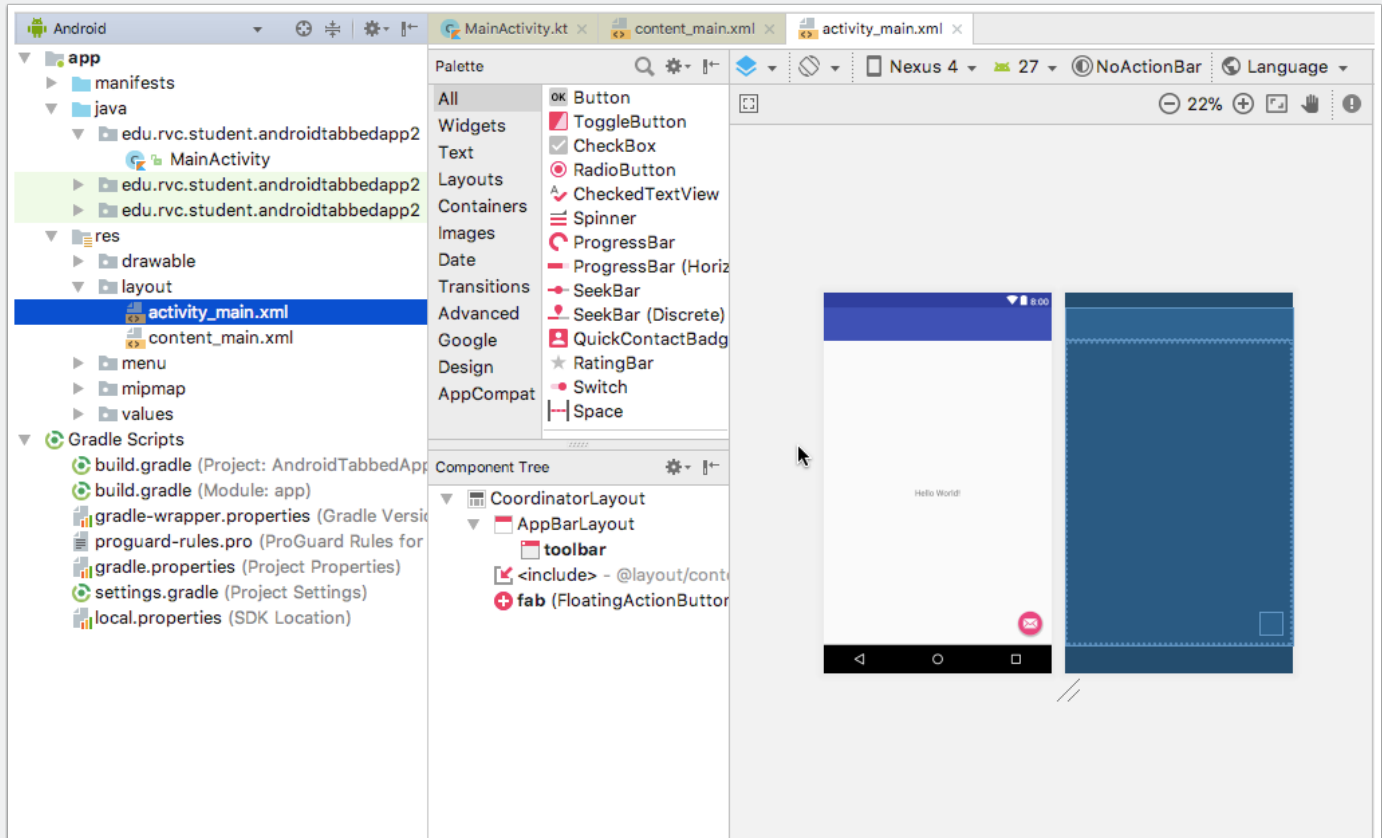


## activity\_main.xml -

1. Create New Project Named AndroidTabbedApp

# Tabbed App with Custom Icons

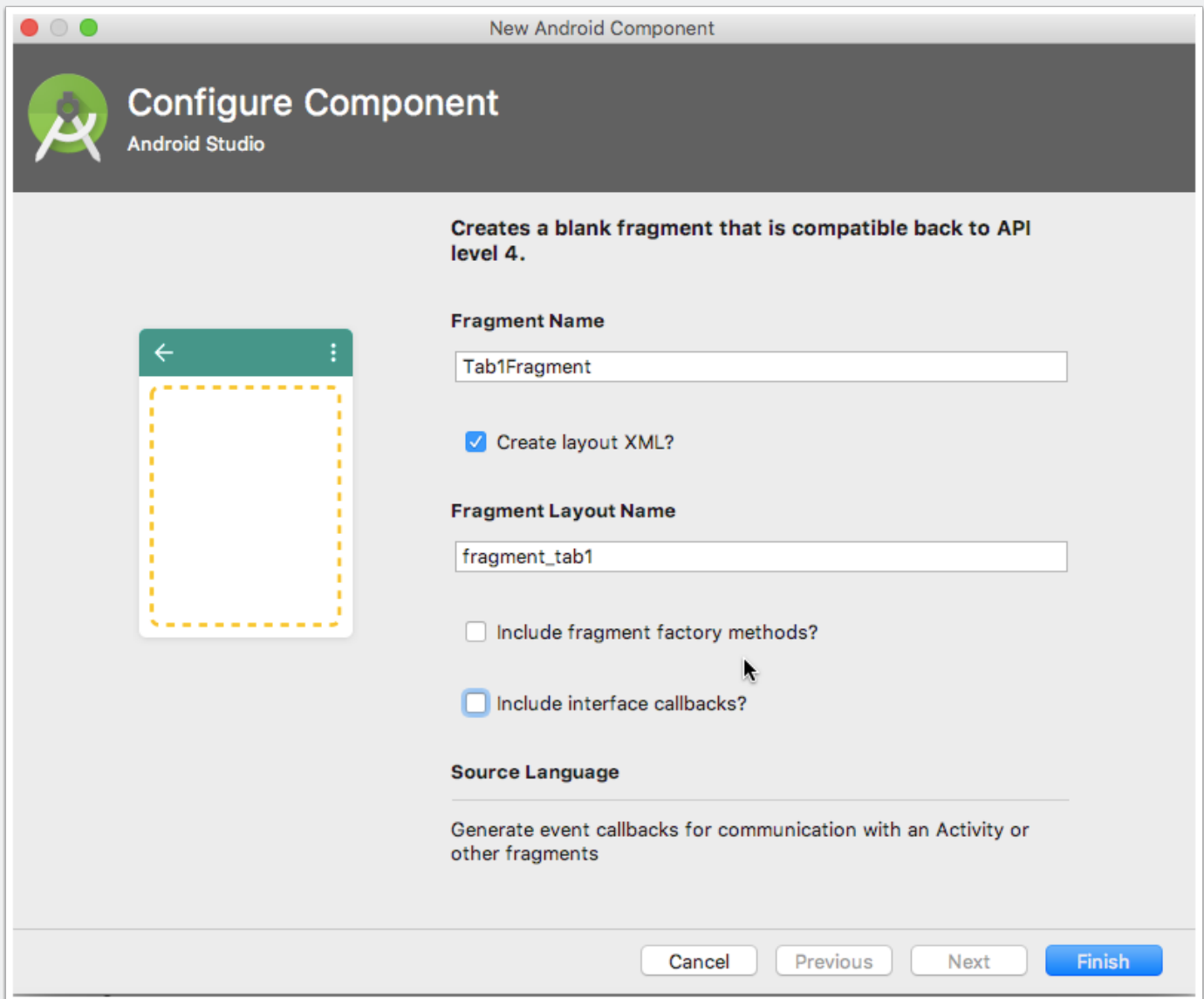
2. Choose Basic Activity
3. Do NOT choose fragment and Finish
4. Go to MainActivity
5. Delete TextView



# Tabbed App with Custom Icons

## Create First out of Four Fragments

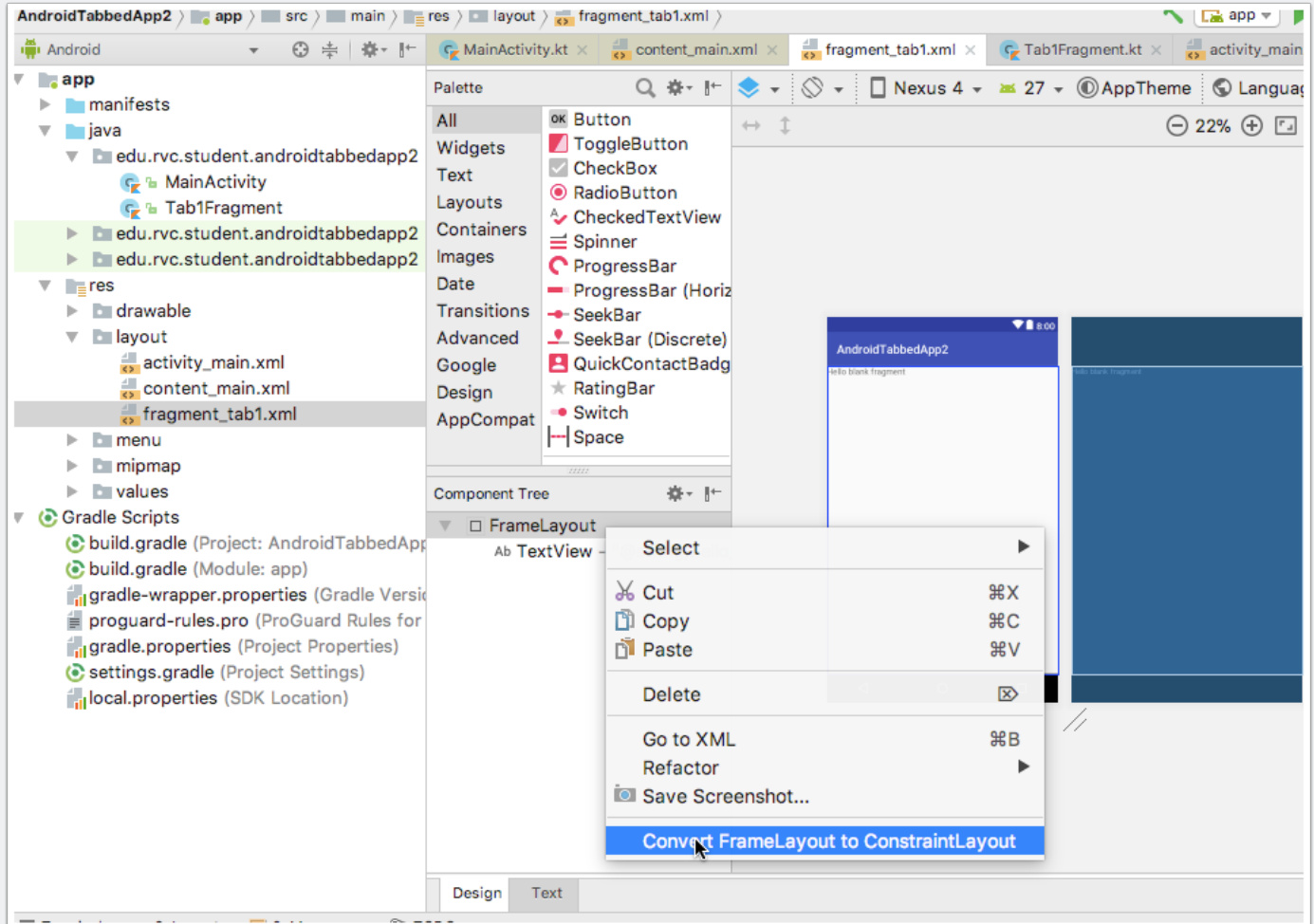
1. Right Click on Java Folder > New > Fragment > Blank
2. Rename to Tab1Fragment
3. Uncheck bottom two include checkboxes
4. Click Finish



# Tabbed App with Custom Icons

## Change Constraint

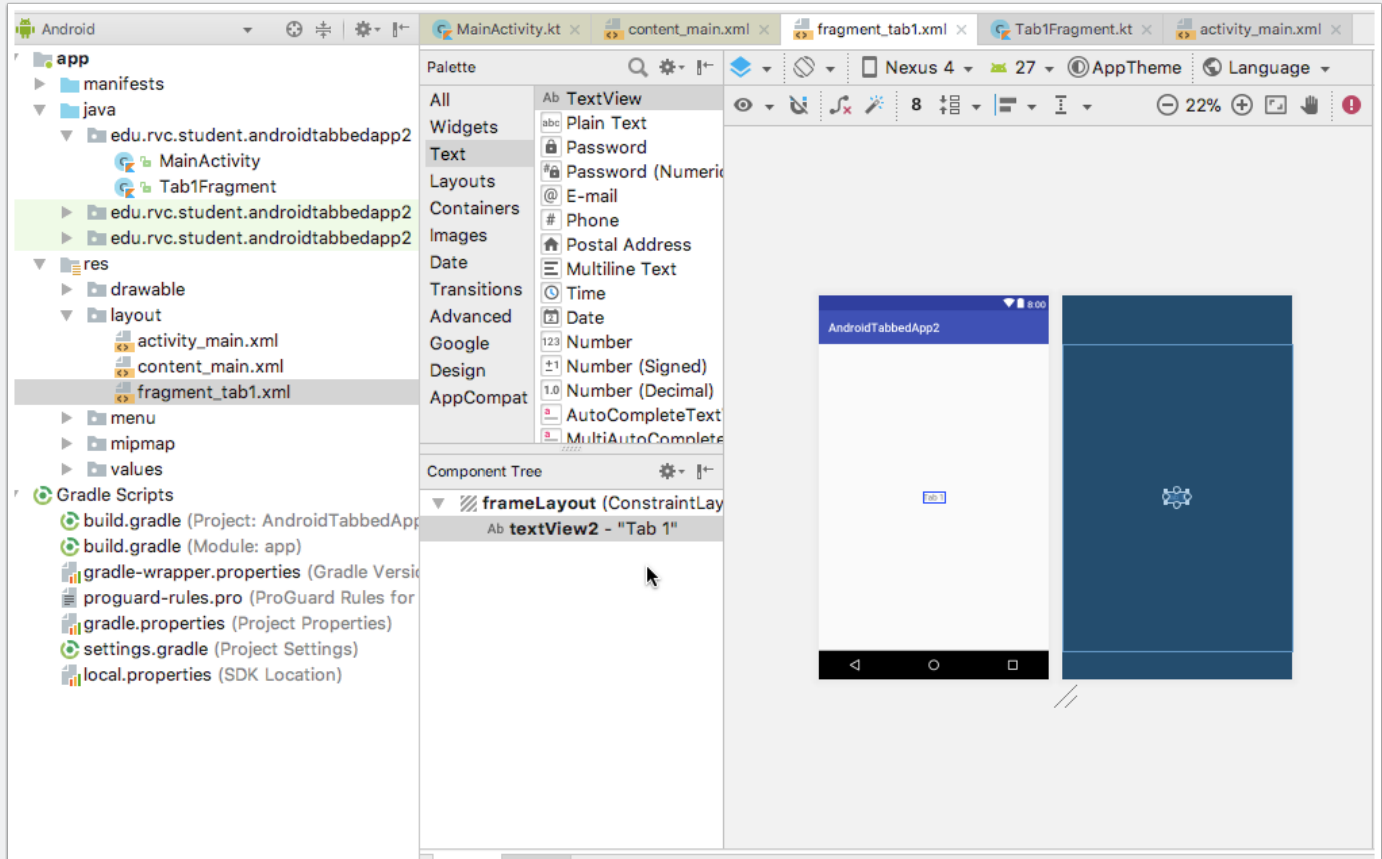
1. Right-click on FrameLayout and choose Convert FrameLayout to ConstraintLayout menu option



# Tabbed App with Custom Icons

## fragment\_tab1.xml -

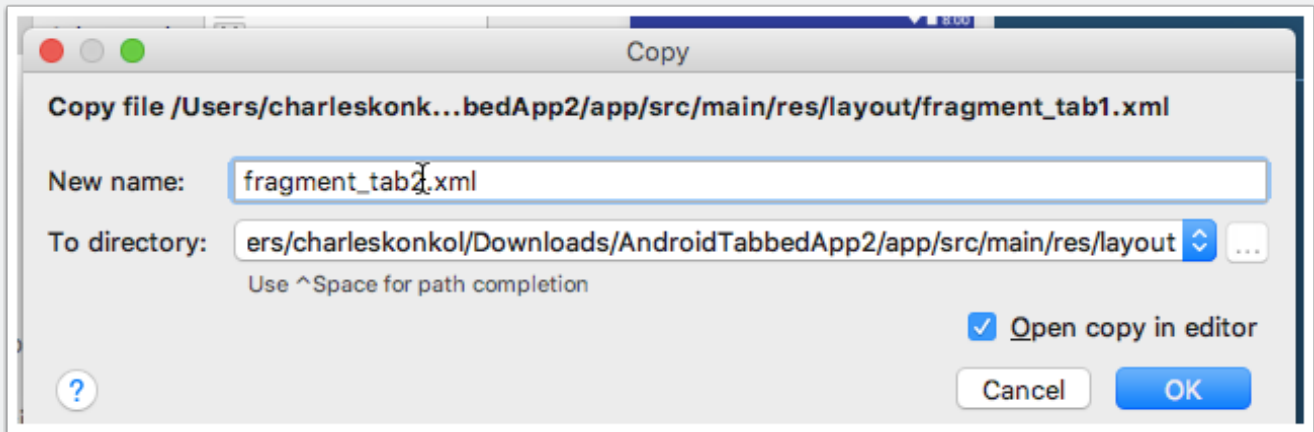
1. Delete TextView
2. Add New TextView and Name Tab 1 and Center in Page



# Tabbed App with Custom Icons

## Copy Fragment

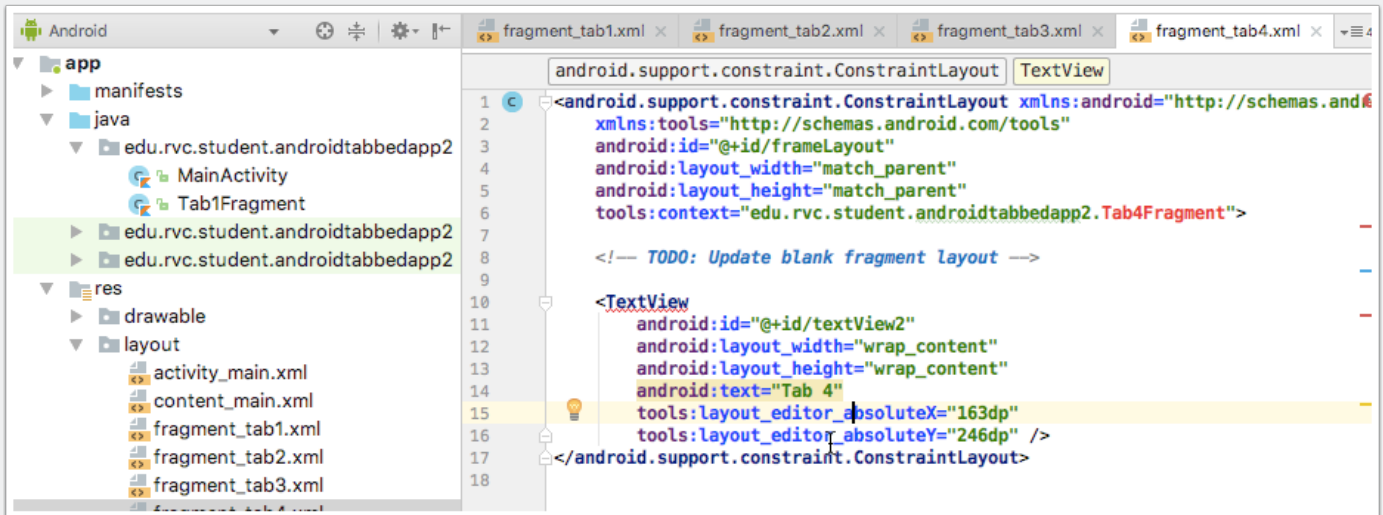
1. Right-click on fragment\_tab1.xml & select copy
2. Right-click on layout folder and select paste
3. Change number in name by adding 1 > select OK



# Tabbed App with Custom Icons

## fragment\_tab#.xml -

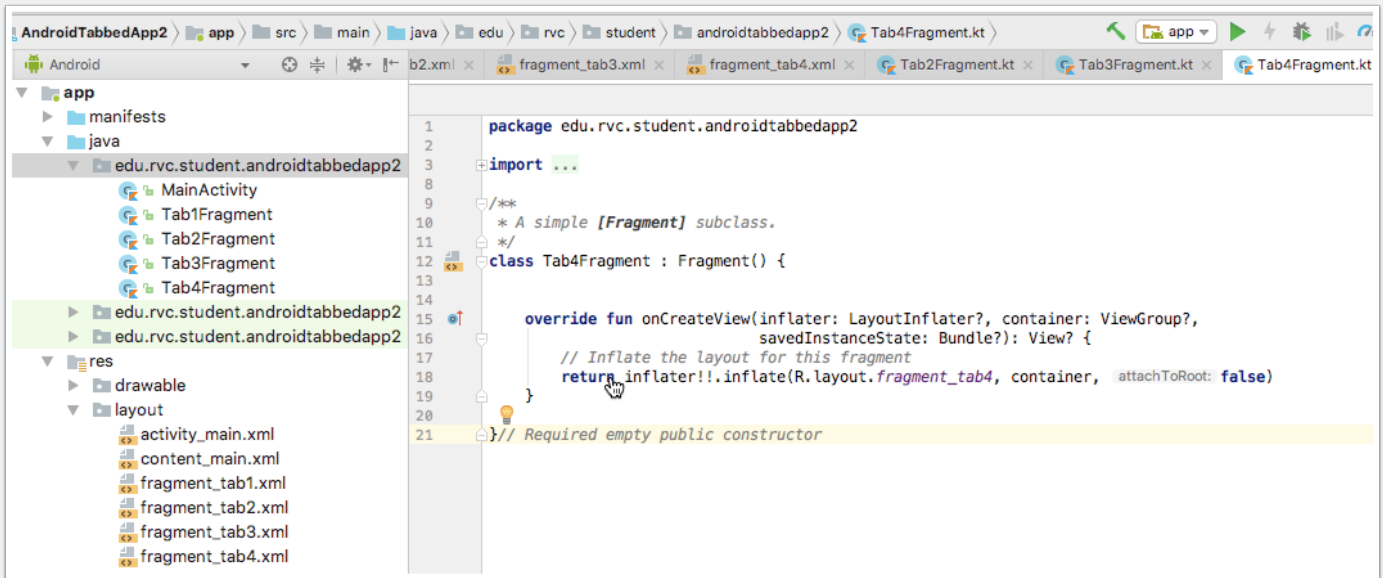
1. After file opens change number by adding 1 on Lines 6 & 14
2. Repeat above steps 1-4 2 more time for fragment\_tab3 & fragment\_tab4
3. You should now have, fragment\_tab3.xml, fragment\_tab4.xml



# Tabbed App with Custom Icons

## Tab#Fragment.kt -

1. Copy/Paste/Rename Tab1Fragment 3 times
2. Rename by adding 1 to number in name
3. Rename line 18 by adding 1 to number



## Adding Page Adapter

1. Right-Click on Java Folder and Create New Kotlin Class
2. Name: **TabPagerAdapter**
3. Paste Below Code in Adapter & Save

**import** android.support.v4.app.Fragment

**import** android.support.v4.app.FragmentManager

**import** android.support.v4.app.FragmentPagerAdapter



# Tabbed App with Custom Icons

```
/**
```

```
* Created by charleskonkol on 3/22/18.
```

```
*/
```

```
class TabPagerAdapter(fm: FragmentManager, private var tabCount: Int) :
```

```
    FragmentPagerAdapter(fm) {
```

```
    override fun getItem(position: Int): Fragment? {
```

```
        when (position) {
```

```
            0 -> return Tab1Fragment()
```

```
            1 -> return Tab2Fragment()
```

```
            2 -> return Tab3Fragment()
```

```
            3 -> return Tab4Fragment()
```

```
            else -> return null
```

```
        }
```

```
    }
```

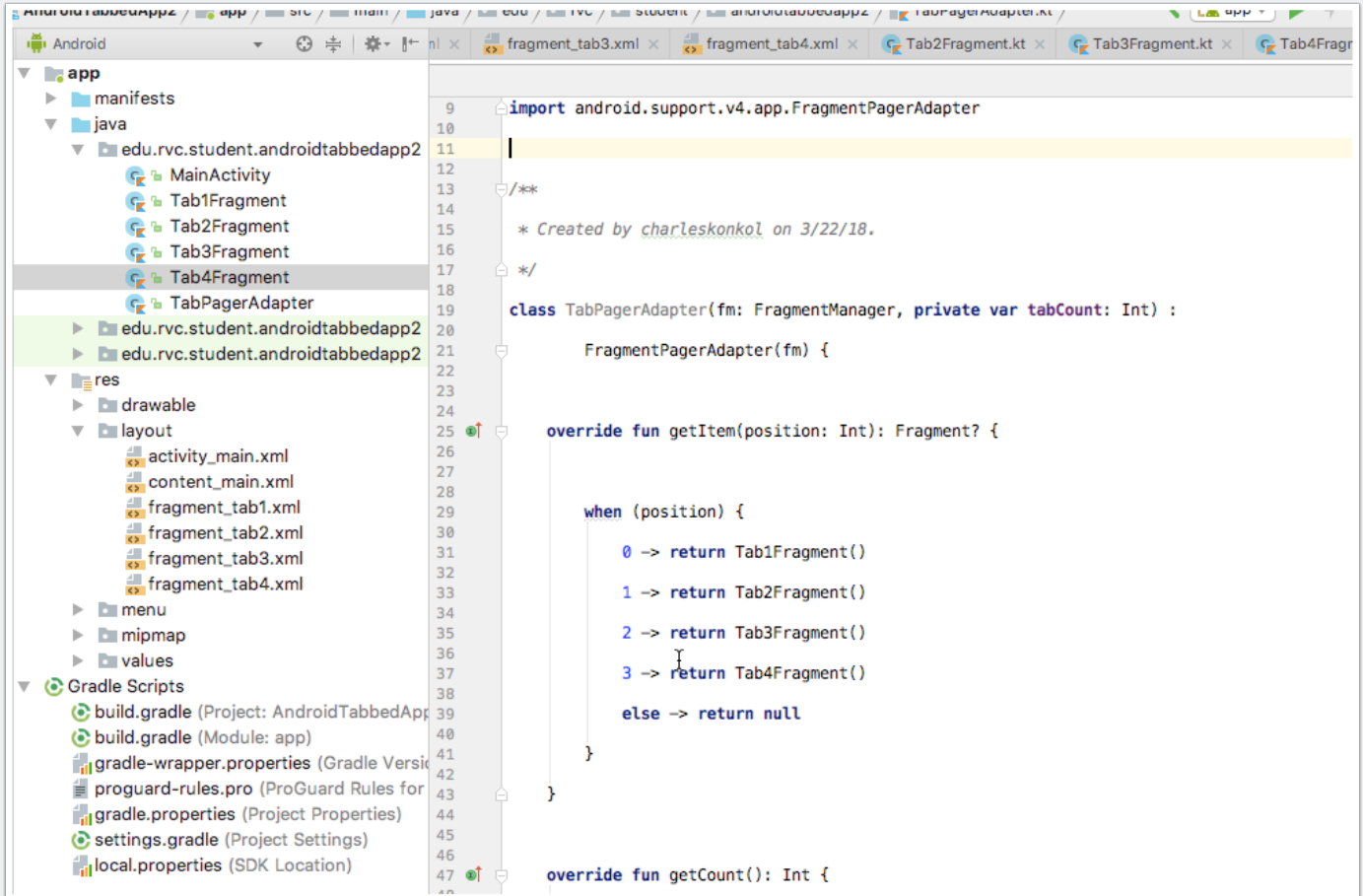
```
    override fun getCount(): Int {
```

```
        return tabCount
```

```
    }
```

# Tabbed App with Custom Icons

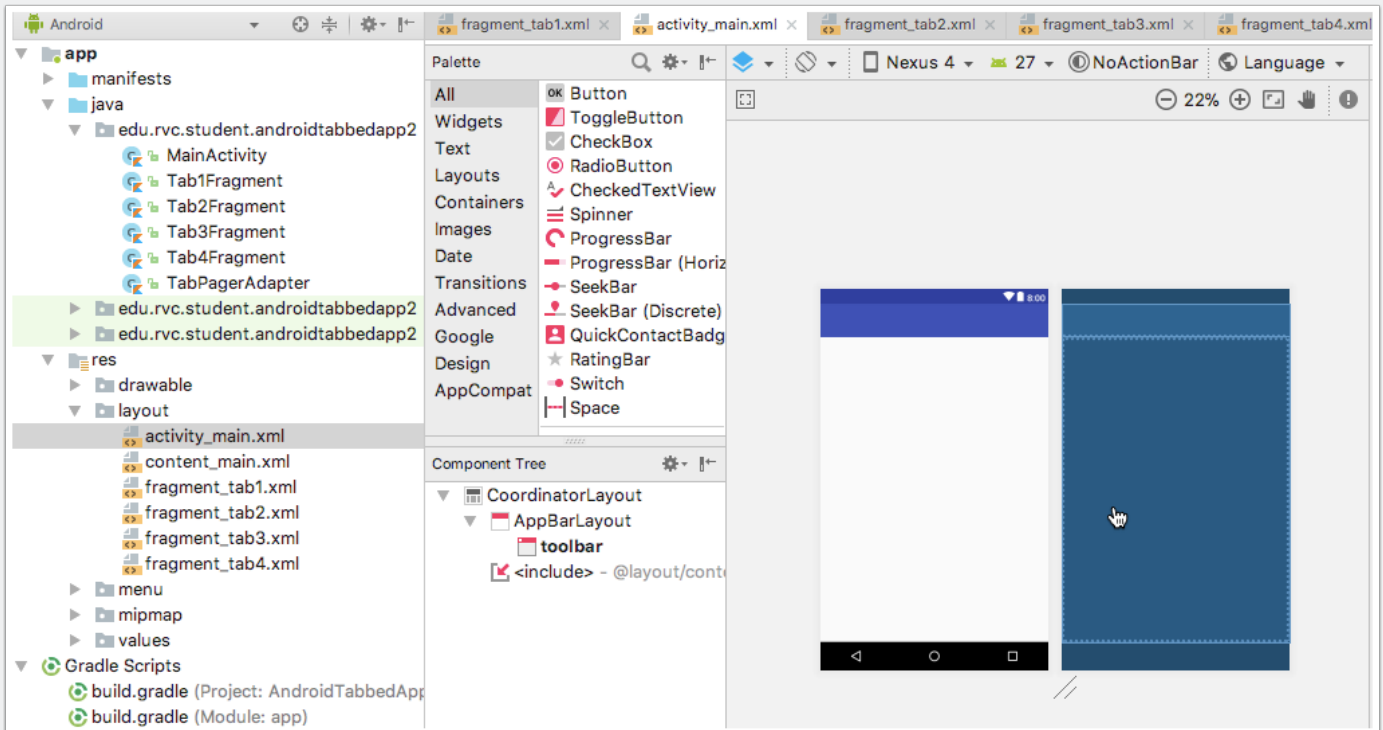
}



# Tabbed App with Custom Icons

## activity\_main.xml -

1. Go to activity\_main.xml and remove the floatingactionbutton
2. Go to MainActivity and remove code for fab.setOnClickListener {}
3. Delete **import xxx.R.id.fab** line



## activity\_main.xml - Add ViewPager and TabLayout

1. Place code right above the
1. `</android.support.design.widget.CoordinatorLayout>`
1. Copy/Paste Below Code for tabs placed on bottom

`<!-- add ViewPager -->`

# Tabbed App with Custom Icons

```
<android.support.v4.view.ViewPager  
  
    android:id="@+id/pager"  
  
    android:layout_width="match_parent"  
  
    android:layout_height="match_parent"  
  
    app:layout_behavior="@string/appbar_scrolling_view_behavior"  
  
/>
```

*<!-- add TabLayout at Botton -->*

```
<android.support.design.widget.TabLayout  
  
    android:id="@+id/tab_layout"  
  
    android:layout_width="match_parent"  
  
    android:layout_gravity="end|bottom"  
  
    android:layout_height="wrap_content"  
  
    app:tabGravity="fill"  
  
    app:tabMode="fixed" />
```

1. If you wanted tabs to be at top of pages place TabLayout code right above  
**</android.support.design.widget.AppBarLayout>**
2. Copy/Paste below *<!-- add TabLayout at Top >*

*<!-- add TabLayout at Top > -->*

# Tabbed App with Custom Icons

```
<android.support.design.widget.TabLayout
```

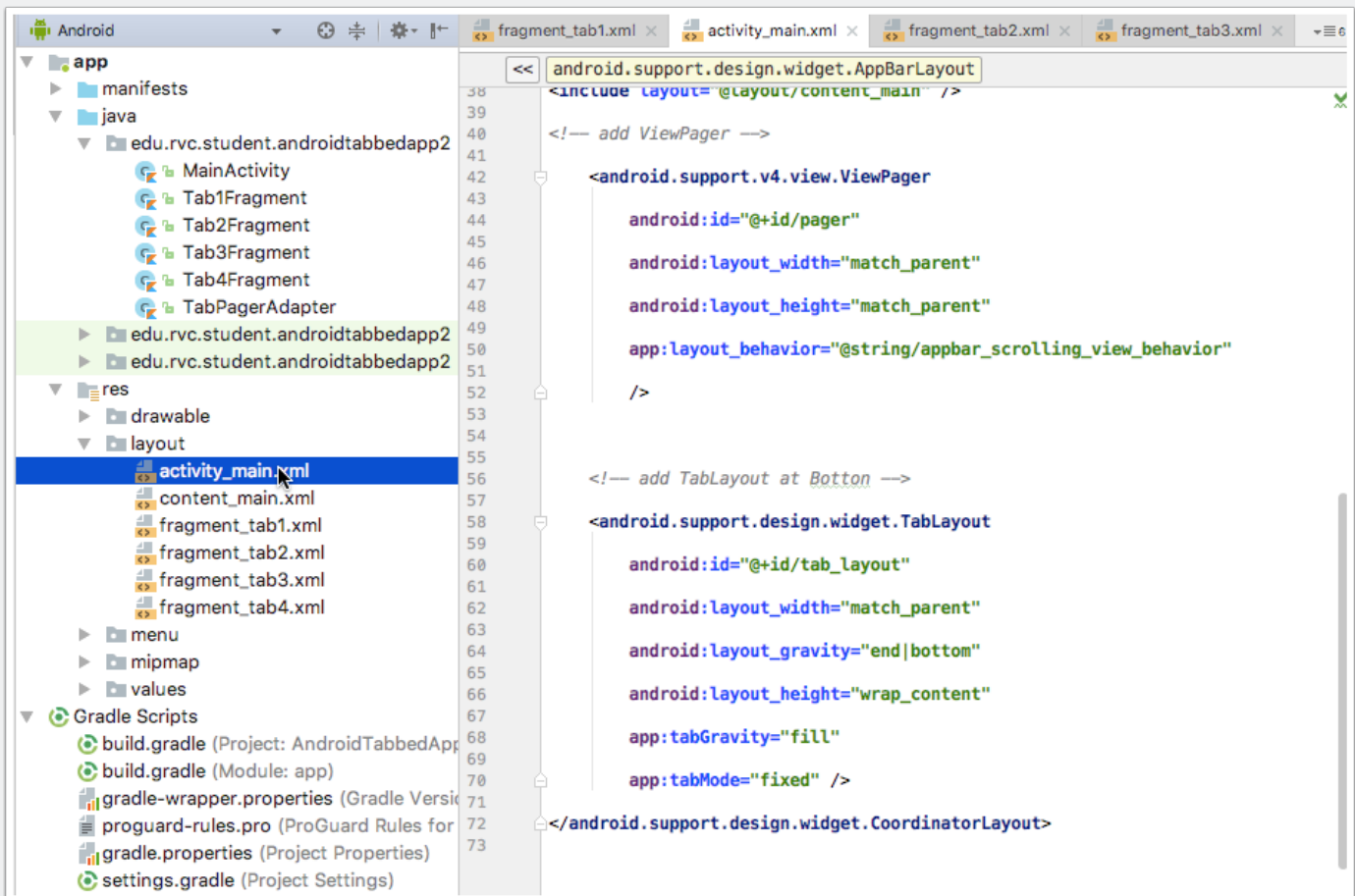
```
android:id="@+id/tab_layout"
```

```
android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"
```

```
app:tabGravity="fill"
```

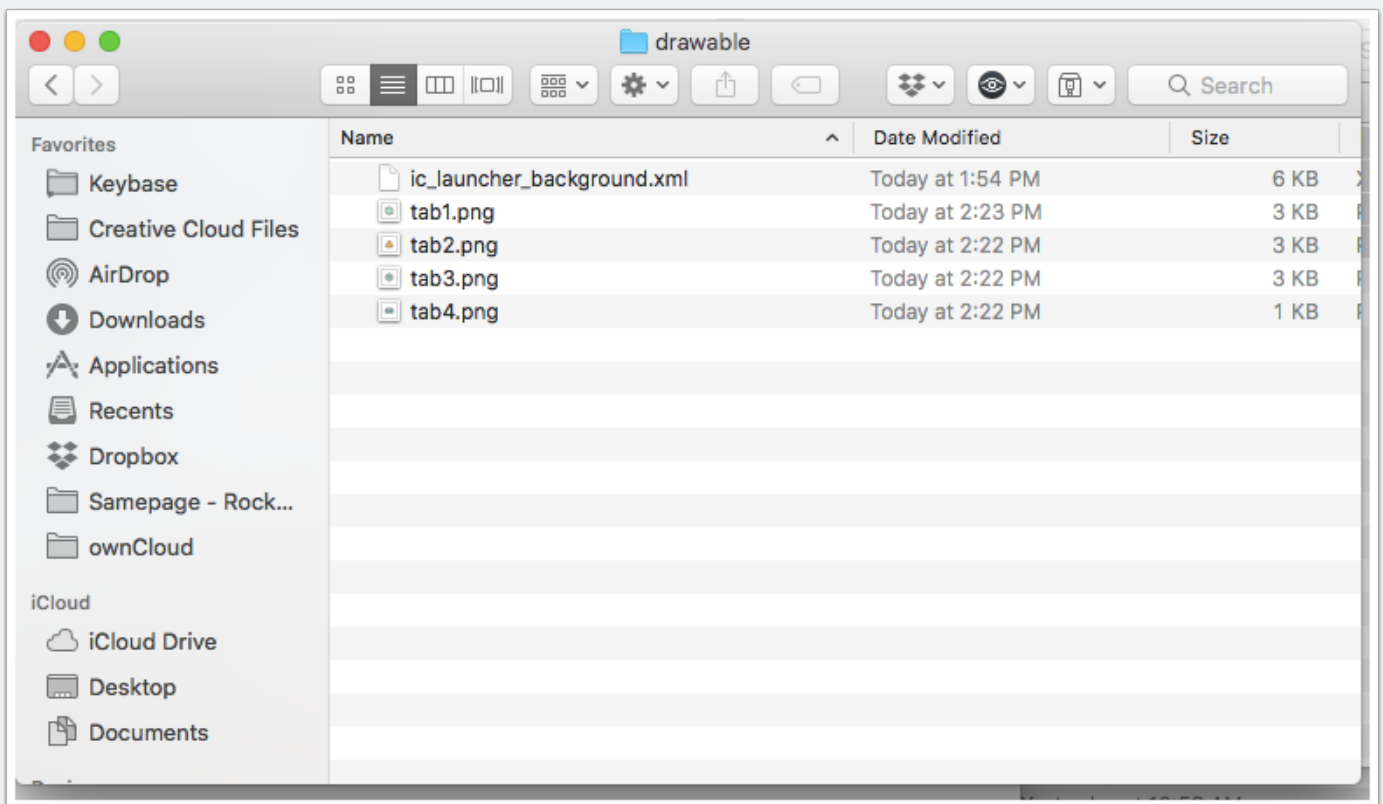
```
app:tabMode="fixed" />
```



# Tabbed App with Custom Icons

## Add icons & code to On\_Create

1. download 4 mobile png icons from <https://www.iconfinder.com/search/?q=menu&price=free>(Links to an external site.)Links to an external site.
2. rename icons to tab1.png, tab2.png, tab3.png, tab4.png
3. move icons to drawable folder
  1. Right-Click on drawable folder and open in folder
  2. Copy/Paste icons into drawable folder



## MainActivity.kt - Code for icons.

Add to newline below

# Tabbed App with Custom Icons

```
class MainActivity : AppCompatActivity() {
```

```
// icons for tabs
```

```
private val homeicons=intArrayOf(R.drawable.tab1, R.drawable.tab2,  
R.drawable.tab3, R.drawable.tab4)
```



# Tabbed App with Custom Icons

## MainActivity.kt - configureTabLayout()

1. Add configureTabLayout() function
2. Add call to configureTabLayout() right before end of **override fun onCreate {}**.
3. **Should look like below.**



```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
    setSupportActionBar(toolbar)

    configureTabLayout()
}

private fun configureTabLayout() {
    //see https://developer.android.com/reference/android/R.drawable.html
    tab_layout.addTab(tab_layout.newTab().setIcon(homeicons[0]))
    tab_layout.addTab(tab_layout.newTab().setIcon(homeicons[1]))
    tab_layout.addTab(tab_layout.newTab().setIcon(homeicons[2]))
    tab_layout.addTab(tab_layout.newTab().setIcon(homeicons[3]))
    val adapter = TabPagerAdapter(supportFragmentManager,
        tab_layout.tabCount)
    pager.adapter = adapter

    pager.addOnPageChangeListener(
        TabLayout.TabLayoutOnPageChangeListener(tab_layout))
    tab_layout.addOnTabSelectedListener(object :
        TabLayout.OnTabSelectedListener {
        override fun onTabSelected(tab: TabLayout.Tab) {
            pager.currentItem = tab.position
        }

        override fun onTabUnselected(tab: TabLayout.Tab) {
        }

        override fun onTabReselected(tab: TabLayout.Tab) {
        }
    })
}
```