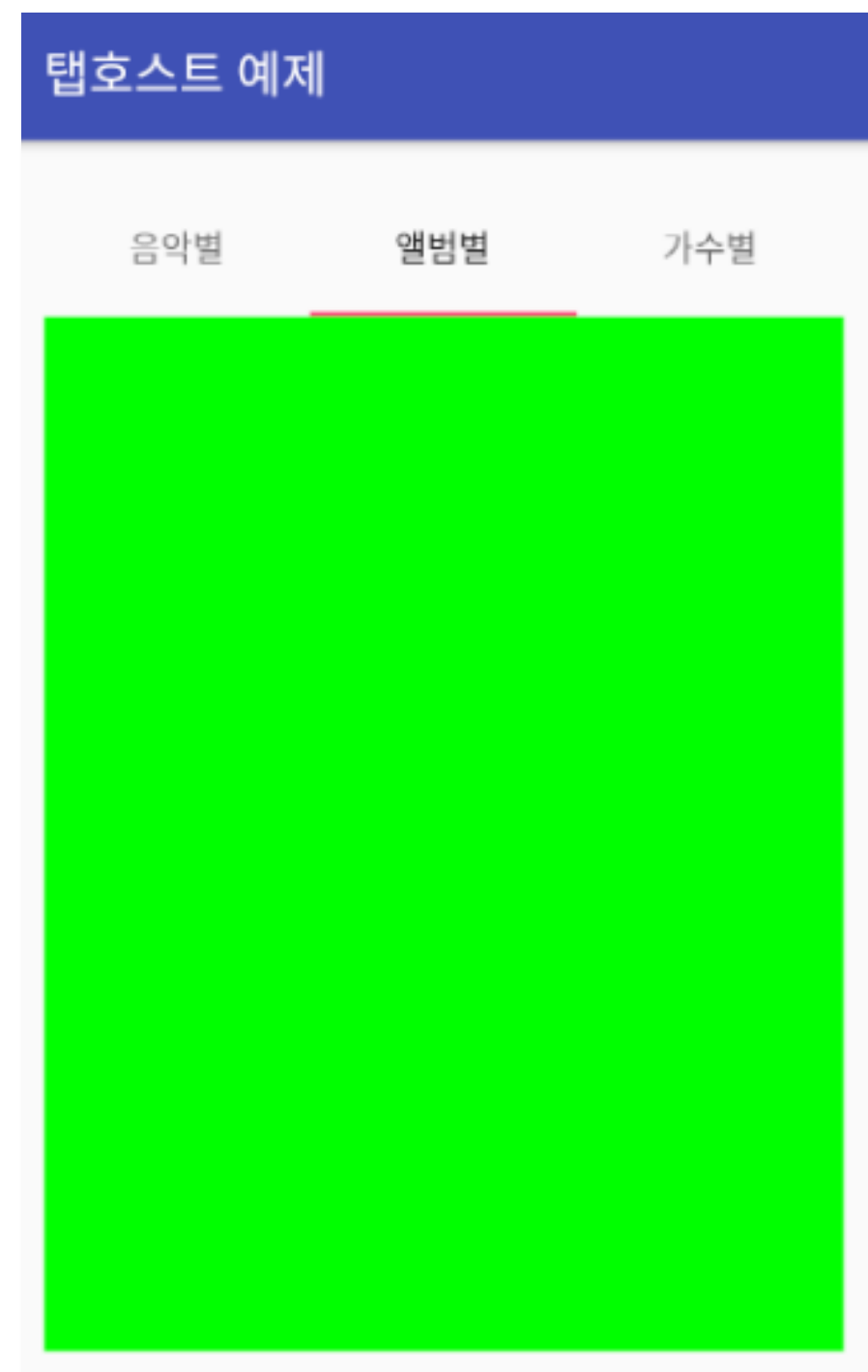
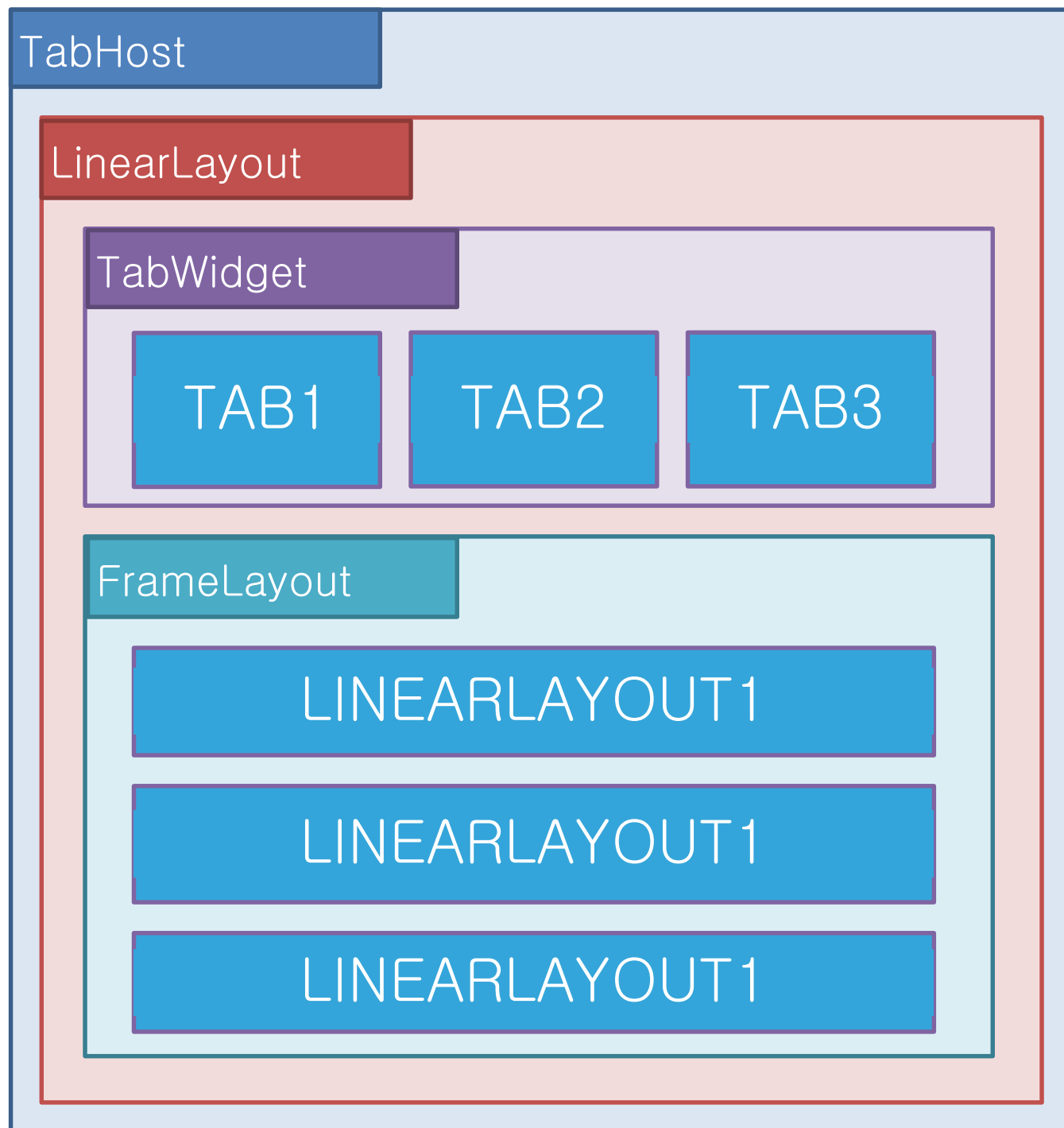


모바일 프로그래밍 입문

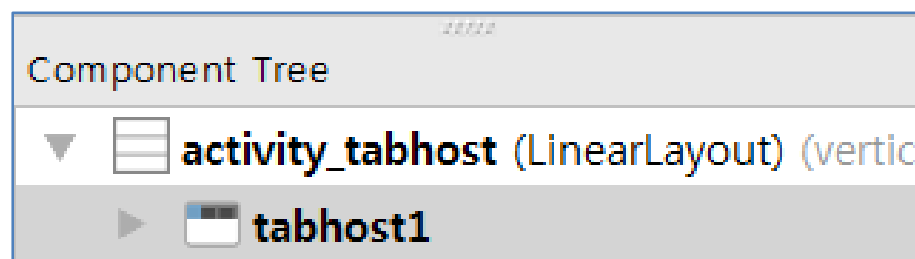
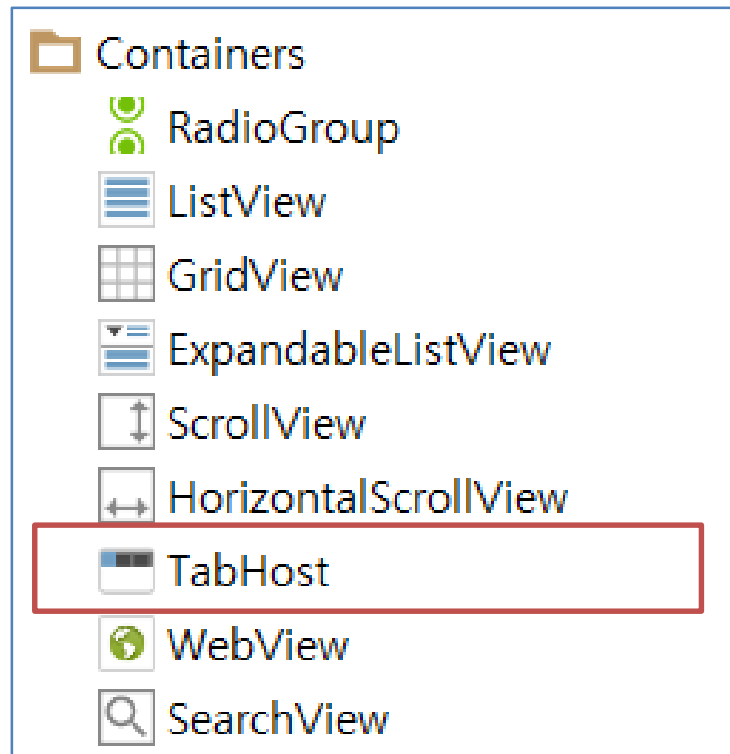
(2019년도 1학기)

탭호스트

뷰컨테이너 - 탭호스트



탭 호스트 작성 1 - LAYOUT



```
<FrameLayout
    android:id="@android:id/tabcontent"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <LinearLayout
        android:id="@+id/tab1"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

    </LinearLayout>

    <LinearLayout
        android:id="@+id/tab2"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

    </LinearLayout>

    <LinearLayout
        android:id="@+id/tab3"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

    </LinearLayout>

</FrameLayout>
```

탭호스트 작성 2 - 코딩

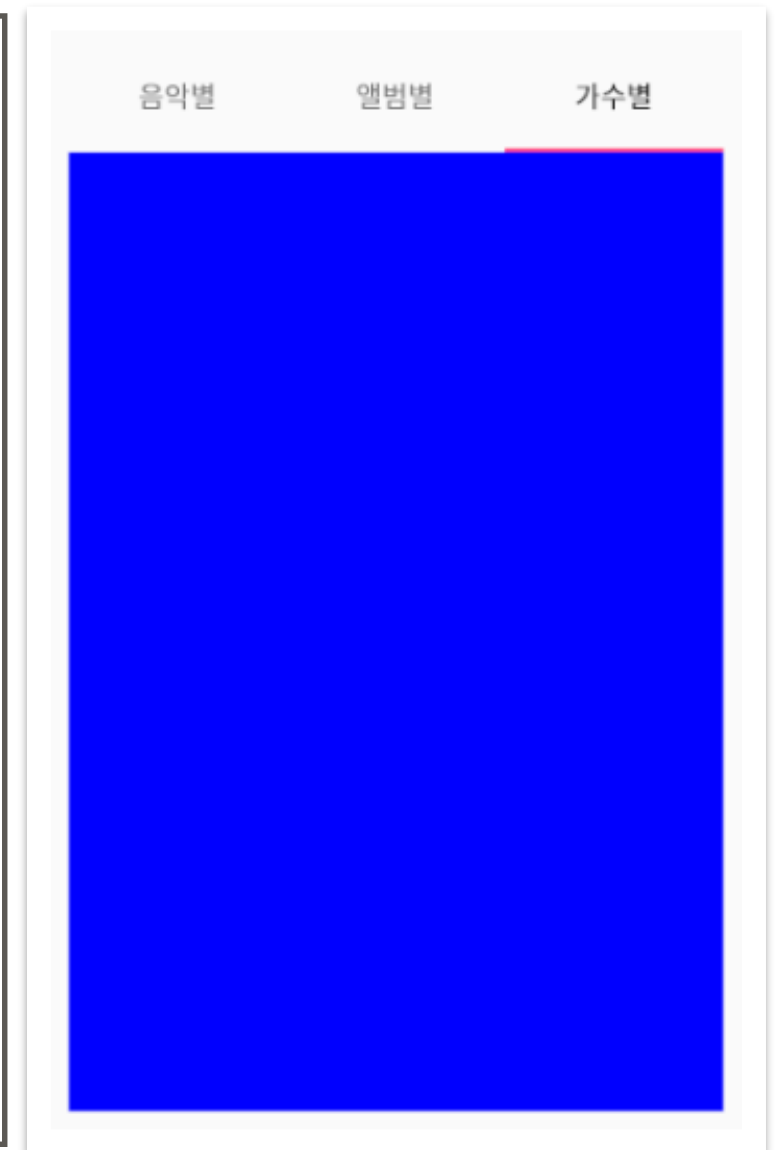
```
setTitle("탭호스트 예제");

tabHost = findViewById(R.id.tabhost1);

tabHost.setup();
TabHost.TabSpec tab1 = tabHost.newTabSpec("A");
TabHost.TabSpec tab2 = tabHost.newTabSpec("B");
TabHost.TabSpec tab3 = tabHost.newTabSpec("C");

tab1.setContent(R.id.tab1).setIndicator("음악별");
tab2.setContent(R.id.tab2).setIndicator("앨범별");
tab3.setContent(R.id.tab3).setIndicator("가수별");

tabHost.addTab(tab1);
tabHost.addTab(tab2);
tabHost.addTab(tab3);
```

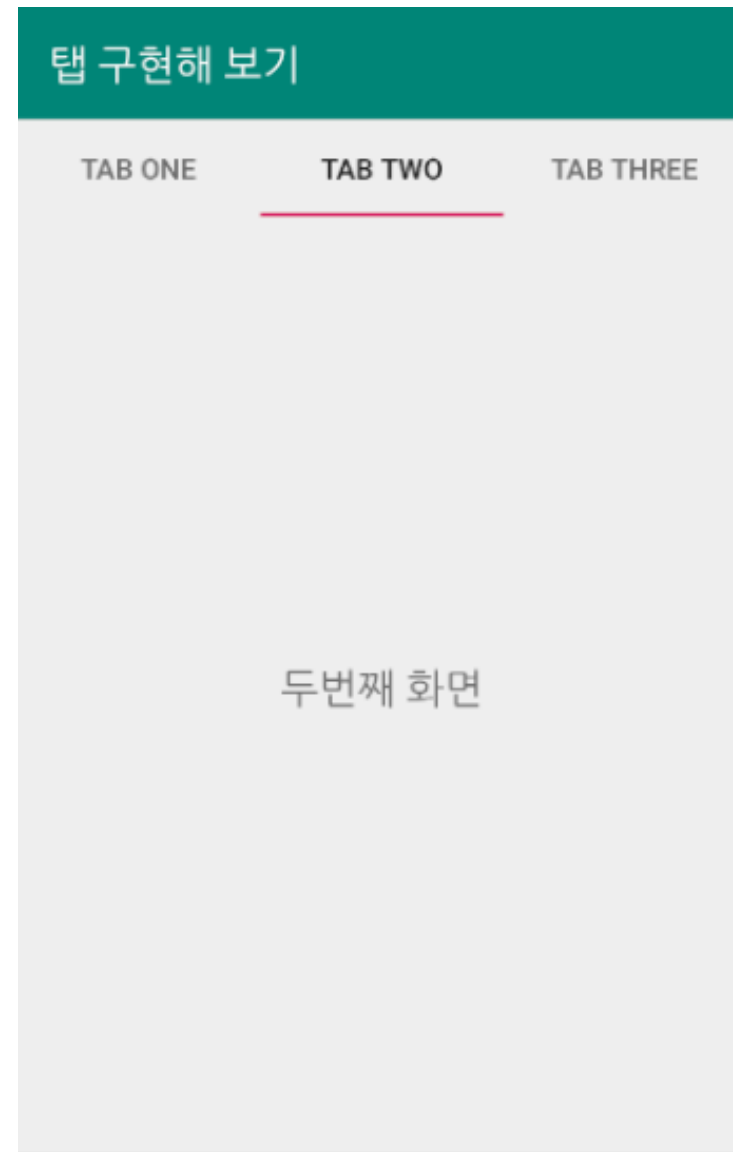
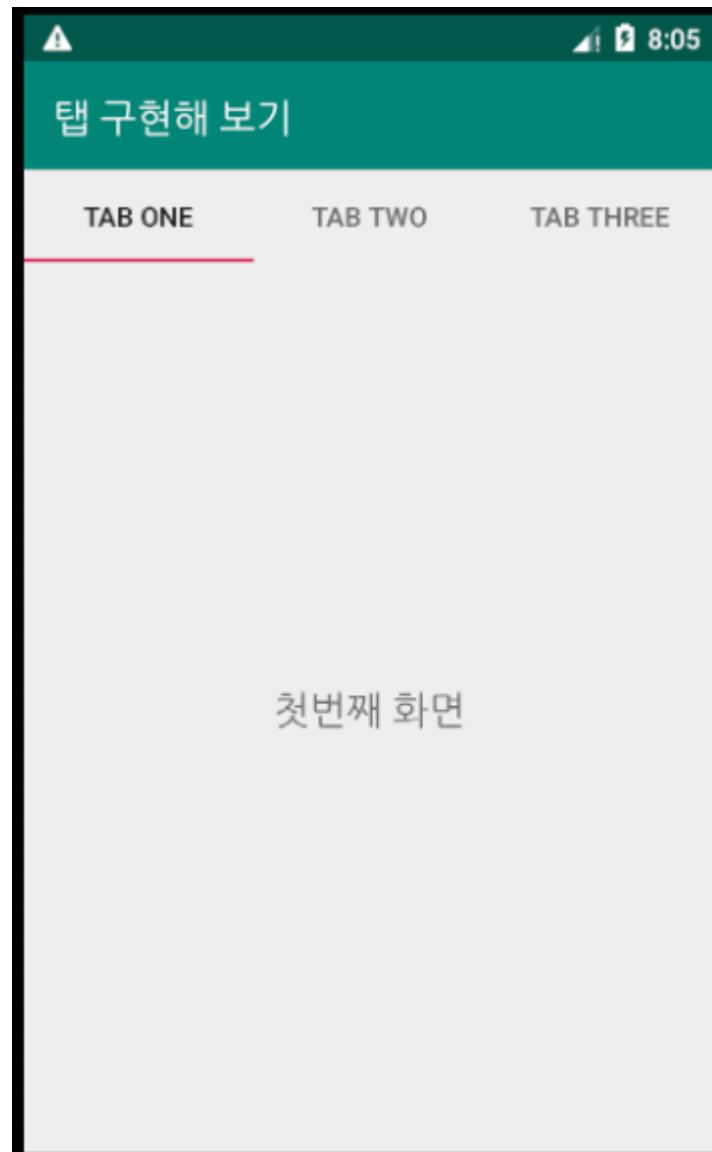


모바일 프로그래밍 입문

(2019년도 1학기)

탭 구현 해보기

▶ 화면 구성



▶ TabLayout 및 ViewPager를 사용하기 위한 셋팅

▶ Gradle Scripts

- ▶ Build.gradle (app)에 dependencies 추가 후 sync 클릭

```
dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support.design:28.0.0'
    implementation 'com.android.support.appcompat-v7:28.0.0'
    implementation 'com.android.support.constraint:constraint-layout:1.1.3'
    implementation 'com.android.support:support-v4:28.0.0'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
}
```

- ▶ values -> styles.xml 수정

<resources>

```
<!-- Base application theme. -->
<style name="AppTheme" parent="Theme.AppCompat.Light.NoActionBar">
    <!-- Customize your theme here. -->
    <item name="colorPrimary">@color/colorPrimary</item>
    <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
    <item name="colorAccent">@color/colorAccent</item>
</style>
```

</resources>

- ▶ 화면 구성

- ▶ Java 파일

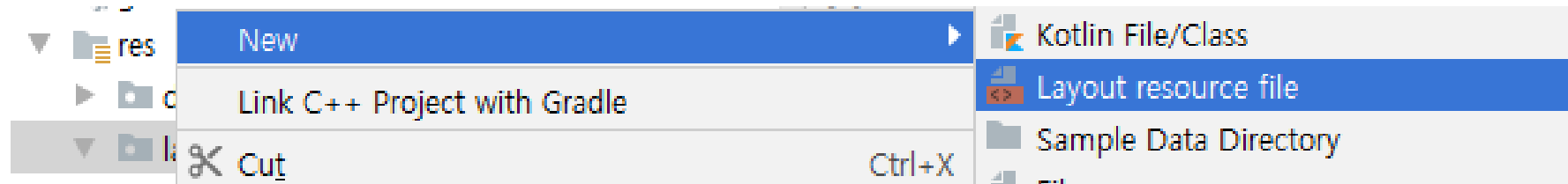
- ▶ MainActivity
 - ▶ TabFragment1
 - ▶ TabFragment2
 - ▶ TabFragment3
 - ▶ TabPagerAdapter

- ▶ layout xml 파일

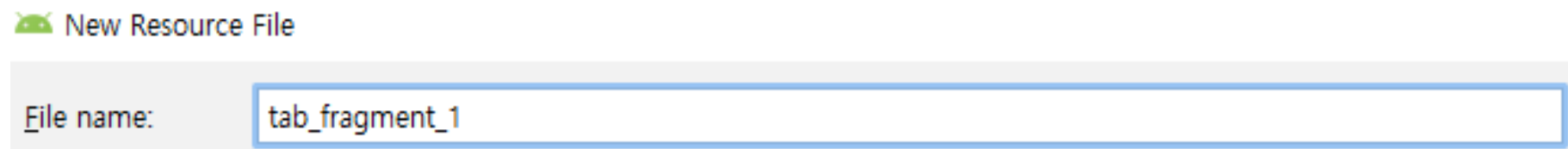
- ▶ activity_main.xml
 - ▶ tab_fragment_1.xml
 - ▶ tab_fragment_2.xml
 - ▶ tab_fragment_3.xml

▶ layout xml 파일 작성

- ▶ layout 선택 후 마우스 우클릭 => New => Layout resource file 클릭



- ▶ file name 에 tab_fragment_1 입력



- ▶ 위의 두 과정을 반복하여 tab_fragment_1, tab_fragment_2, tab_fragment_3 파일 생성

▶ tab_fragment_1.xml 파일 작성

▶ RelativeLayout 사용

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="첫번째 화면"
        android:textSize="20dp"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

▶ tab_fragment_2.xml 파일 작성

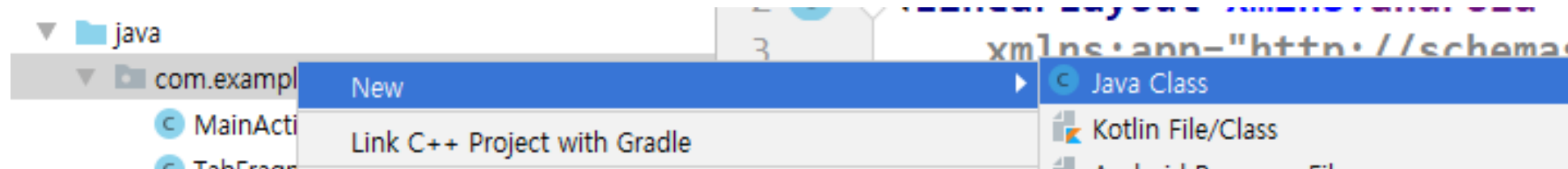
▶ tab_fragment_1.xml과 다른 내용 android:text="두번째 화면"

▶ tab_fragment_3.xml 파일 작성

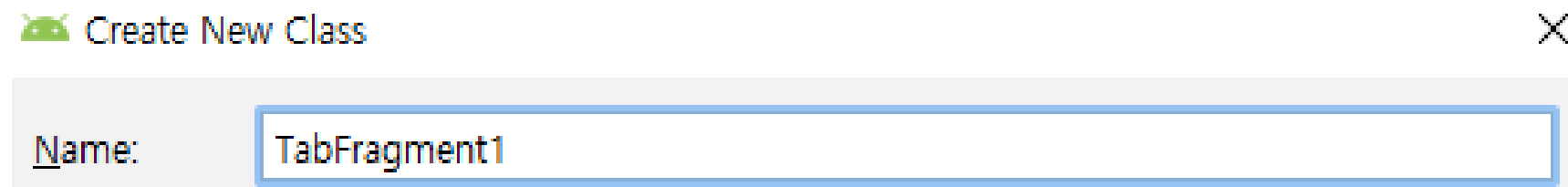
▶ tab_fragment_1.xml과 다른 내용 android:text="세번째 화면"

▶ Java 파일 생성

- ▶ Java 하위 폴더 선택 후 new => java class 선택



- ▶ Name에 TabFragment1 입력



- ▶ 위의 두 과정을 반복하여 TabFragment1, TabFragment2, TabFragment3, TabPagerAdapter 자바 파일 생성

▶ TabFragment1.Java 파일 작성

```
public class TabFragment1 extends Fragment {
    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @NonNull Bundle savedInstanceState) {
        return inflater.inflate(R.layout.tab_fragment_1, container, attachToRoot: false);
    }
}
```

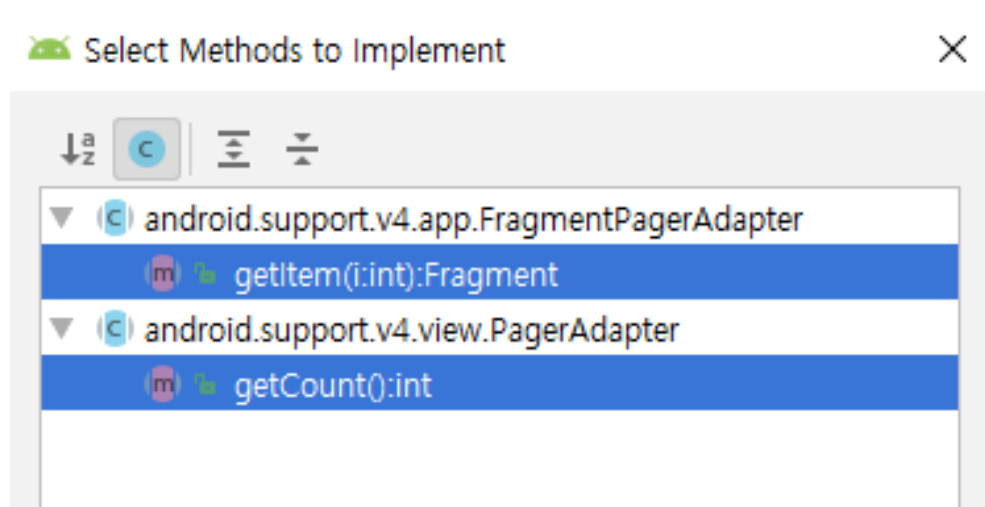
▶ TabFragment2.Java 파일 작성

- ▶ 위와 동일하며 tab_fragment_1을 tab_fragment_2로 변경

▶ TabFragment3.Java 파일 작성

- ▶ 위와 동일하며 tab_fragment_1을 tab_fragment_3로 변경

- ▶ TabPagerAdapter 자파 파일 작성 (1/3)
 - ▶ TabPagerAdapter 클래스 뒤에 extends FragmentStatePagerAdapter 작성 후 ALT + ENTER 클릭 후 Implement method 창이 뜨고 OK를 클릭



- ▶ 두 개의 메소드 생성

```
@Override
public Fragment getItem(int i) {
    return null;
}

@Override
public int getCount() {
    return 0;
}
```

▶ TabPagerAdapter 자파 파일 작성 (2/3)

- ▶ Tab을 카운트할 변수 선언 및 TabPagerAdapter 함수 생성

```
private int tabCount;
```

```
public TabPagerAdapter(FragmentManager fm, int tabCount){
    super(fm);
    this.tabCount = tabCount;
}
```

- ▶ getCount함수에서 return 0; 을 return tabCount()로 변경

```
public int getCount() {
    return tabCount;
}
```

▶ TabPagerAdapter 자파 파일 작성 (3/3)

- ▶ getItem함수에서 tab의 position에 따라 화면을 변환해 주기 위해 switch문 작성

```
@Override
public Fragment getItem(int position) {
    switch (position) {
        case 0 :
            TabFragment1 tabFragment1 = new TabFragment1();
            return tabFragment1;
        case 1:
            TabFragment2 tabFragment2 = new TabFragment2();
            return tabFragment2;
        case 2:
            TabFragment3 tabFragment3 = new TabFragment3();
            return tabFragment3;
        default:
            return null;
    }
}
```

▶ activity_main.xml 화면 구성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical">

    <android.support.v7.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
        android:background="?attr/colorPrimary"
        android:theme="@style/ThemeOverlay.AppCompat.Dark" />

    <android.support.design.widget.TabLayout
        android:id="@+id/tabLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/toolbar" />

    <android.support.v4.view.ViewPager
        android:id="@+id/viewPager"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@id/tabLayout" />

</LinearLayout>
```


▶ MainActivity 자바 파일 작성 (1/2)

▶ TabLayout, ViewPager 객체 생성

```
private TabLayout tabLayout;  
private ViewPager viewPager;
```

▶ Toolbar 지정 및 tabLayout 선언 (탭 추가)

```
Toolbar toolbar = (Toolbar)findViewById(R.id.toolbar);  
setSupportActionBar(toolbar);
```

```
tabLayout = (TabLayout)findViewById(R.id.tabLayout);  
tabLayout.addTab(tabLayout.newTab().setText("Tab One"));  
tabLayout.addTab(tabLayout.newTab().setText("Tab Two"));  
tabLayout.addTab(tabLayout.newTab().setText("Tab Three"));  
tabLayout.setTabGravity(TabLayout.GRAVITY_FILL);
```

▶ viewPager 선언

```
viewPager = (ViewPager)findViewById(R.id.viewPager);
```

▶ MainActivity 자바 파일 작성 (2/2)

▶ TabPagerAdapter 생성 및 viewPager에 pagerAdapter 연결

```
TabPagerAdapter pagerAdapter = new TabPagerAdapter(getSupportFragmentManager(), tabLayout.getTabCount());
viewPager.setAdapter(pagerAdapter);
```

▶ Page Change Listener 설정

```
viewPager.addOnPageChangeListener(new TabLayout.TabLayoutOnPageChangeListener(tabLayout));
```

▶ Tab Selected Listener 설정

```
tabLayout.addOnTabSelectedListener(new TabLayout.OnTabSelectedListener() {
    @Override
    public void onTabSelected(TabLayout.Tab tab) {
        viewPager.setCurrentItem(tab.getPosition());
    }

    @Override
    public void onTabUnselected(TabLayout.Tab tab) {
    }

    @Override
    public void onTabReselected(TabLayout.Tab tab) {
    }
});
```