

Android Studio の使い方

作成日: 2015/11/26

作成者: 堀口

更新履歴

更新日	更新概要	作業者
2015/11/26	・ 新規作成 (仮)	堀口
	・	
	・	
	・	
	・	

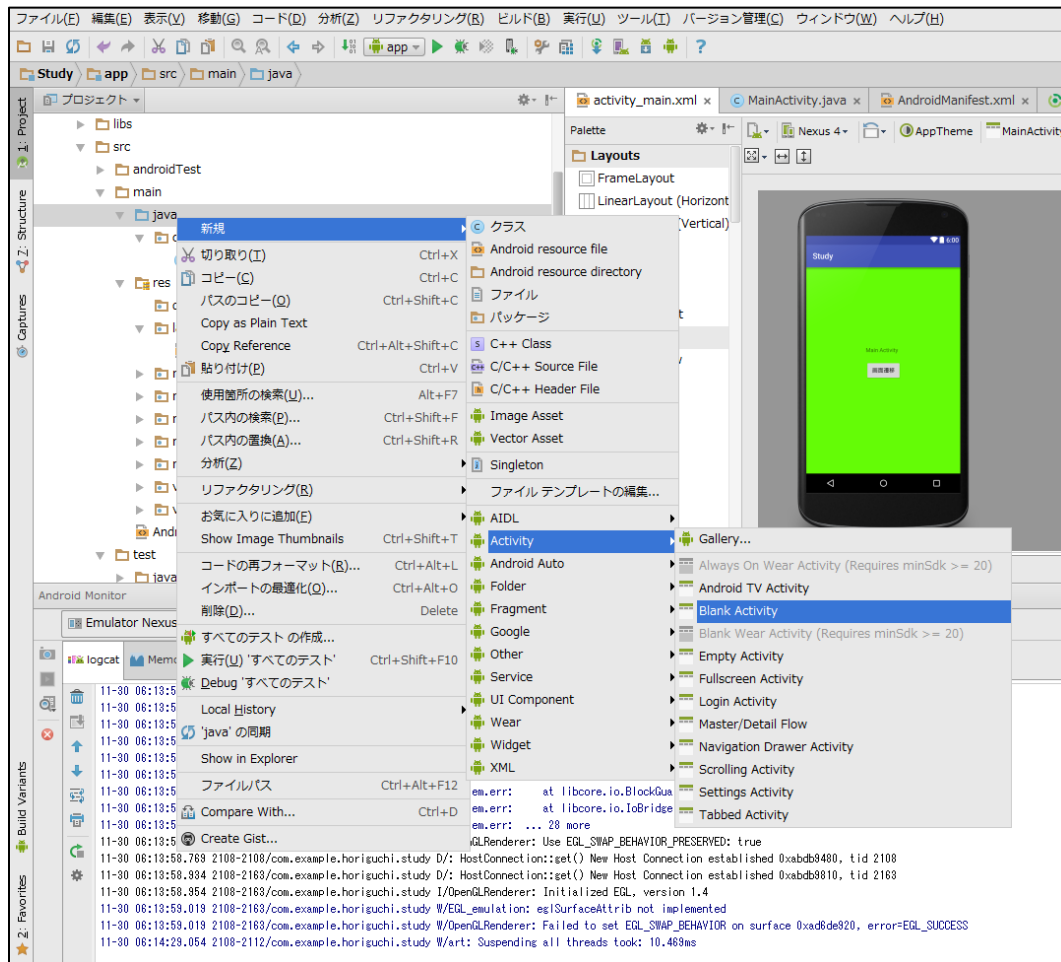
目次

更新履歴	1
画面遷移	3

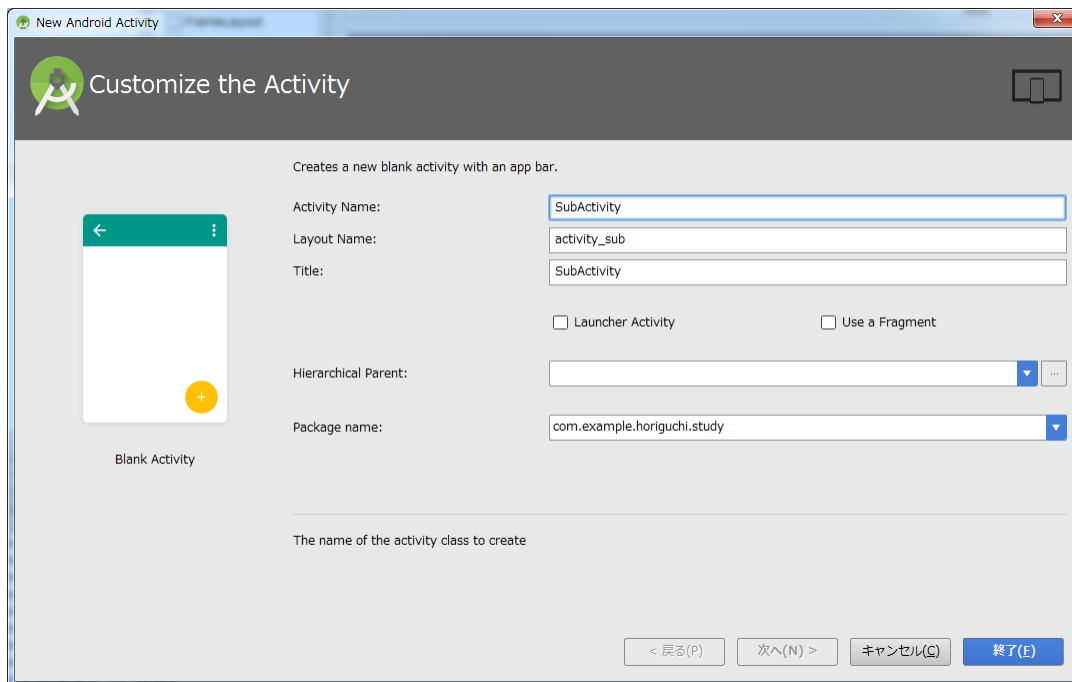
画面遷移

1.まずは遷移先の Activity を作成します。

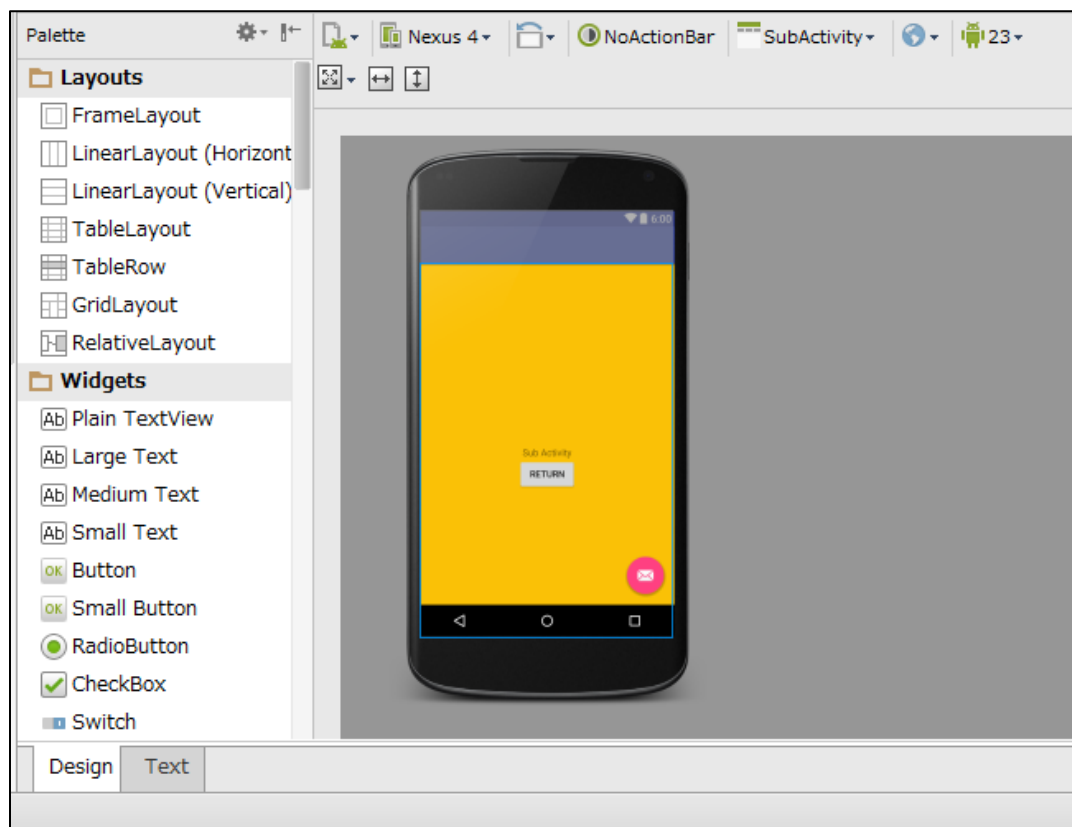
App<src<java<右クリック<新規<Activity<Blank Activity を選択します。



今回は画面名「SubActivity」とします。



自動的に layout フォルダに activity_sub.xml が作成されます。



2.AndroidManifesto.xmlを開き、作成した SubActivity を下記に追加します

※自分の時は Activity を作成した際に AndoidManifesto.xml に自動的に追加されました。

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.horiguchi.study" >

    <permission android:name="android.permission.INTERNET" />
    <permission android:name="android.permission.ACCESS_NETWORK_STATE" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="Study"
        android:supportRtl="true"
        android:theme="@style/AppTheme" >
        <activity android:name=".MainActivity" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".SubActivity"
            android:label="@string/title_activity_sub"
            android:theme="@style/AppTheme.NoActionBar" >
        </activity>
    </application>

</manifest>
```

3. MainActivity を下記の用に記述します。

レイアウトにボタンが置いてある前提です。

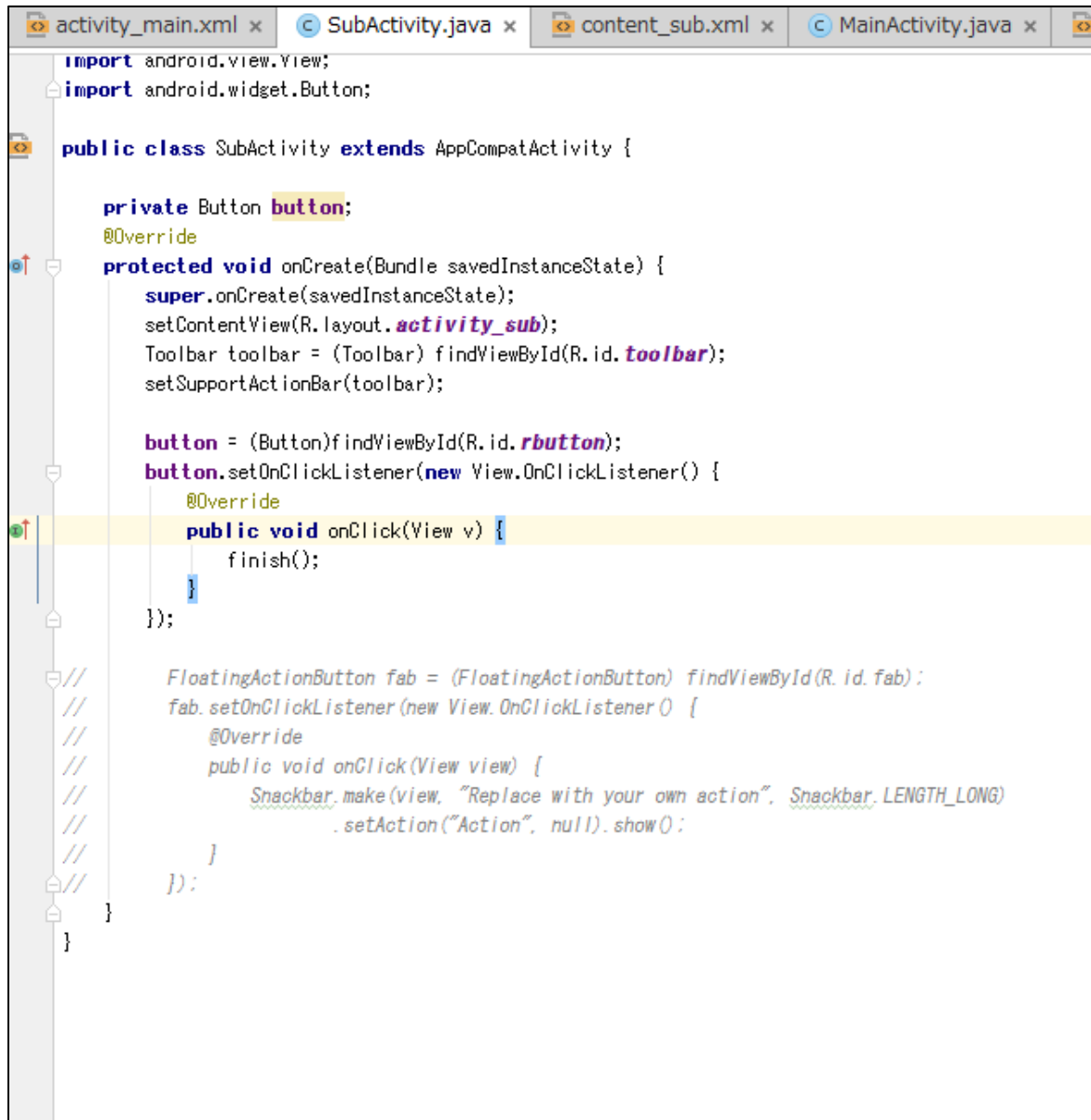
```
ty_main.xml x SubActivity.java x content_sub.xml x MainActivity.java x
public class MainActivity extends Activity {
    //private static final String url = "jdbc:mysql://10.0.2.2:3306/fujiofc";
    // private static final String url = "jdbc:mysql://192.168.1.126/fujiofc";
    // private static final String user = "root";
    // private static final String pass = "";
    private Button button;
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        button= (Button)findViewById(R.id.button);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(getApplicationContext(),SubActivity.class);
                startActivityForResult(intent,0);
            }
        });
        // testDB() :
    }

    /**
     public void testDB() {
         TextView tv = (TextView) this.findViewById(R.id.text_view);
         try {
```

4.SubActivity を下記の用に入力します。

レイアウトにボタンが置いてある前提です。



```
activity_main.xml x SubActivity.java x content_sub.xml x MainActivity.java x
import android.view.View;
import android.widget.Button;

public class SubActivity extends AppCompatActivity {

    private Button button;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sub);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        button = (Button) findViewById(R.id.rbutton);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                finish();
            }
        });

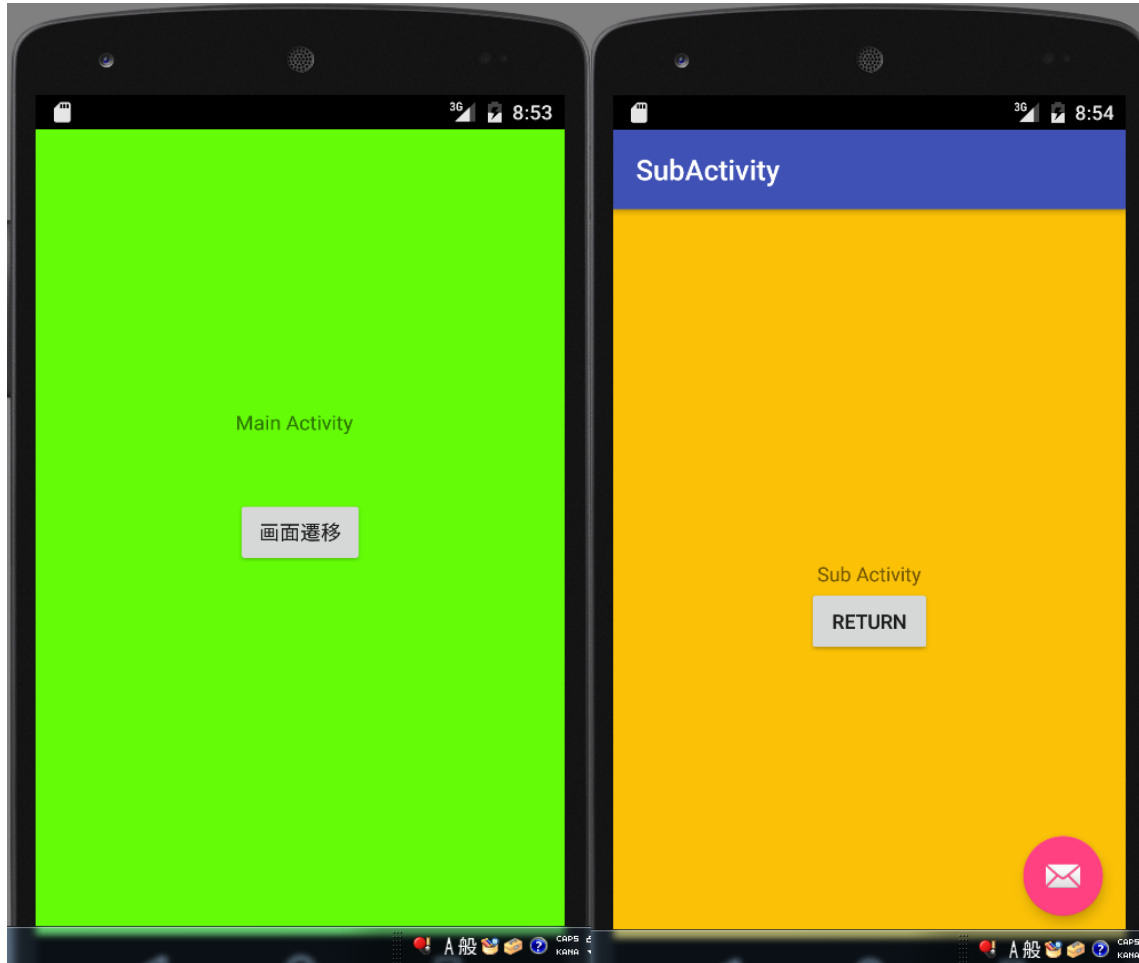
        // FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
        // fab.setOnClickListener(new View.OnClickListener() {
        //     @Override
        //     public void onClick(View view) {
        //         Snackbar.make(view, "Replace with your own action", Snackbar.LENGTH_LONG)
        //             .setAction("Action", null).show();
        //     }
        // });
    }
}
```

5.コードを入力したら画面上部ある下記の部分を押して実行します。



起動最初の画面

遷移先画面



→「画面遷移」をクリックすると SubActivity が呼び出されます。

→「RETURN」をクリックすると画面が終了します。