

移动互联网开发（4）：Fragment UI

Atidote 于 2023-12-09 19:53:22 发布



移动互联网开发 专栏收录该内容

0 订阅 7 篇文章

订阅专栏

一、实验内容

1. 实现CriminalIntent的 **Fragment** 的竖屏界面
2. 实现CriminalIntent的Fragment的横屏界面
3. 在Activity中加载一个Fragment的基本流程
4. 使用Fragment设计UI有哪些好处？

二、具体实现

strings.xml

```
<resources>
    <string name="app_name">工号+姓名</string>
     <string name="hello_blank_fragment">Hello blank fragment</string>
</resources>
```

CSDN @Atidote

activity_crime_list.xml

内容来源: csdn.net

作者昵称: Atidote

原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622

作者主页: https://blog.csdn.net/qq_56204872

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    android:id="@+id/list_fragment_container"
    tools:context=".CrimeListActivity">

</FrameLayout>
```

CSDN @Atidote

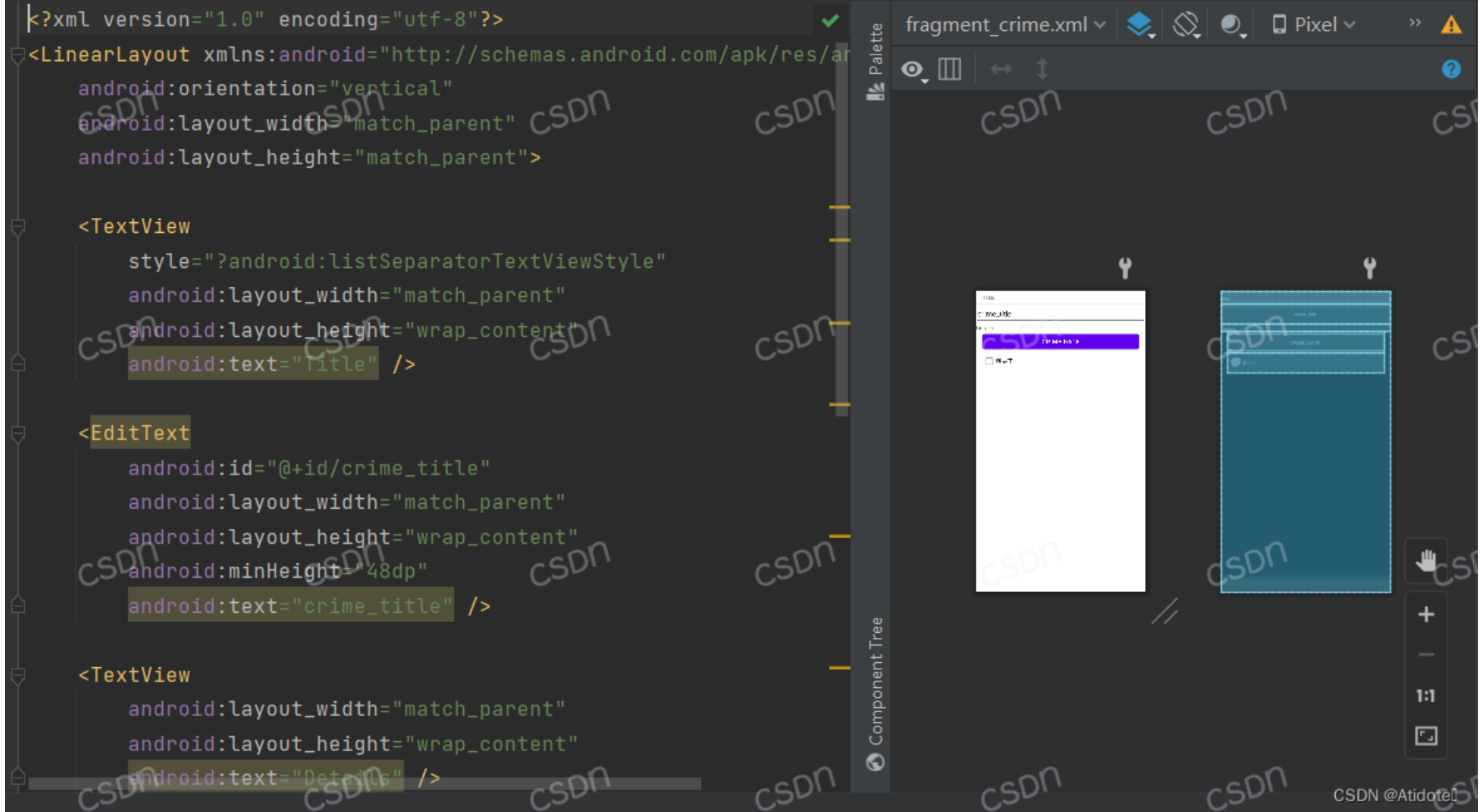
fragment_crime.xml

内容来源: csdn.net

作者昵称: Atidote

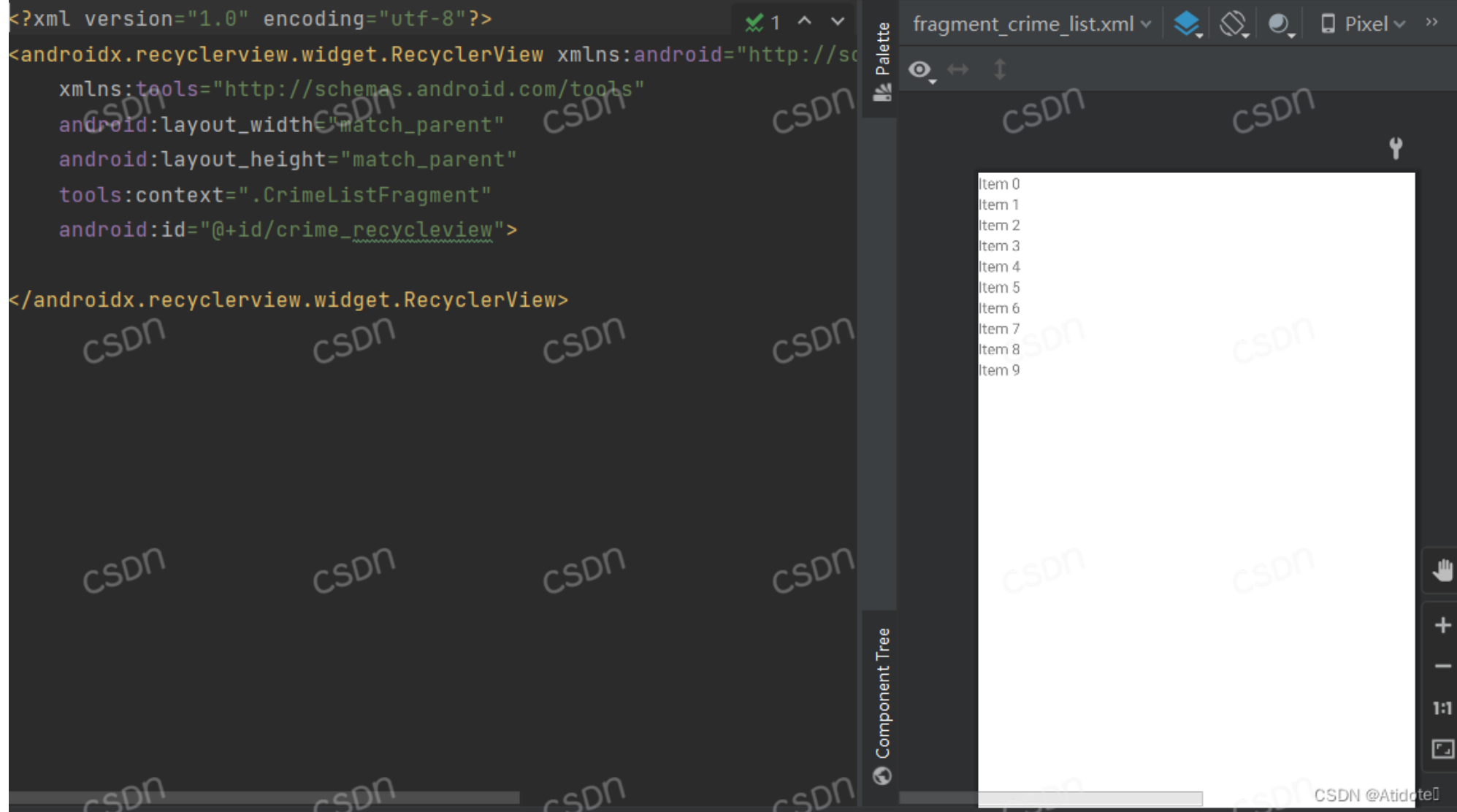
原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622

作者主页: https://blog.csdn.net/qq_56204872



fragment_crime_list.xml

内容来源: csdn.net
作者昵称: Atidote
原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622
作者主页: https://blog.csdn.net/qq_56204872



CrimeListActivity

内容来源: csdn.net

作者昵称: Atidote

原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622

作者主页: https://blog.csdn.net/qq_56204872

```

package com.usts2022.criminalintent;

import ...

public class CrimeListActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_crime_list);
        //获取FragmentManager
        FragmentManager fm = getSupportFragmentManager();
        Fragment fragment = fm.findFragmentById(R.id.list_fragment_container);
        if (fragment == null) {
            fragment = new CrimeListFragment();
            fm.beginTransaction()
                .add(R.id.list_fragment_container, fragment)
                .commit();
        }
    }
}

```

CSDN @Atidote

Crime

内容来源: csdn.net

作者昵称: Atidote

原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622

作者主页: https://blog.csdn.net/qq_56204872

```
package com.usts2022.criminalintent;
```

```
import ...
```

```
public class Crime implements Serializable {
```

```
    private UUID mId;
```

```
    private String mTitle;
```

```
    private Date mDate;
```

```
    private boolean mSolved;
```

```
    public Crime() {
```

```
        mId = UUID.randomUUID();
```

```
        mDate = new Date();
```

```
    }
```

```
    public Crime(UUID mId, String mTitle, Date mDate, boolean mSolved) {
```

```
        this.mId = mId;
```

```
        this.mTitle = mTitle;
```

```
        this.mDate = mDate;
```

```
        this.mSolved = mSolved;
```

```
    }
```

```
    @Override
```

```
    public String toString() {
```

```
        return "Crime{" +
```

CSDN @Atidote

CrimeFragment

内容来源: csdn.net

作者昵称: Atidote

原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622

作者主页: https://blog.csdn.net/qq_56204872

```

public class CrimeFragment extends Fragment {

    private Crime crime;
    private EditText crimeTitle;
    private Button crimeDate;
    private CheckBox crimeSolved;

    public CrimeFragment() {

    }

    @ public static CrimeFragment newInstance(Crime crime) {
        CrimeFragment fragment = new CrimeFragment();
        Bundle args = new Bundle();
        args.putSerializable("crime", crime);
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            this.crime = (Crime) getArguments().getSerializable(key: "crime");
        }
    }
}

```

CSDN @Atidote

CrimeListFragment

内容来源: csdn.net

作者昵称: Atidote

原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622

作者主页: https://blog.csdn.net/qq_56204872

```

package com.usts2022.criminalintent;

import ...

public class CrimeListFragment extends Fragment {

    //为CrimeListFragment配置视图
    private RecyclerView mRecyclerview;
    private CrimeAdapter mCrimeAdapter;

    public CrimeListFragment() {
        // 需要空的公共构造函数
    }

    @Override
    public void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        View v = inflater.inflate(R.layout.fragment_crime_list, container, attachToRoot: false);

        mRecyclerview = v.findViewById(R.id.crime_recycleview);
        mRecyclerview.setLayoutManager(new LinearLayoutManager(getActivity()));
    }
}

```

CSDN @Atidote

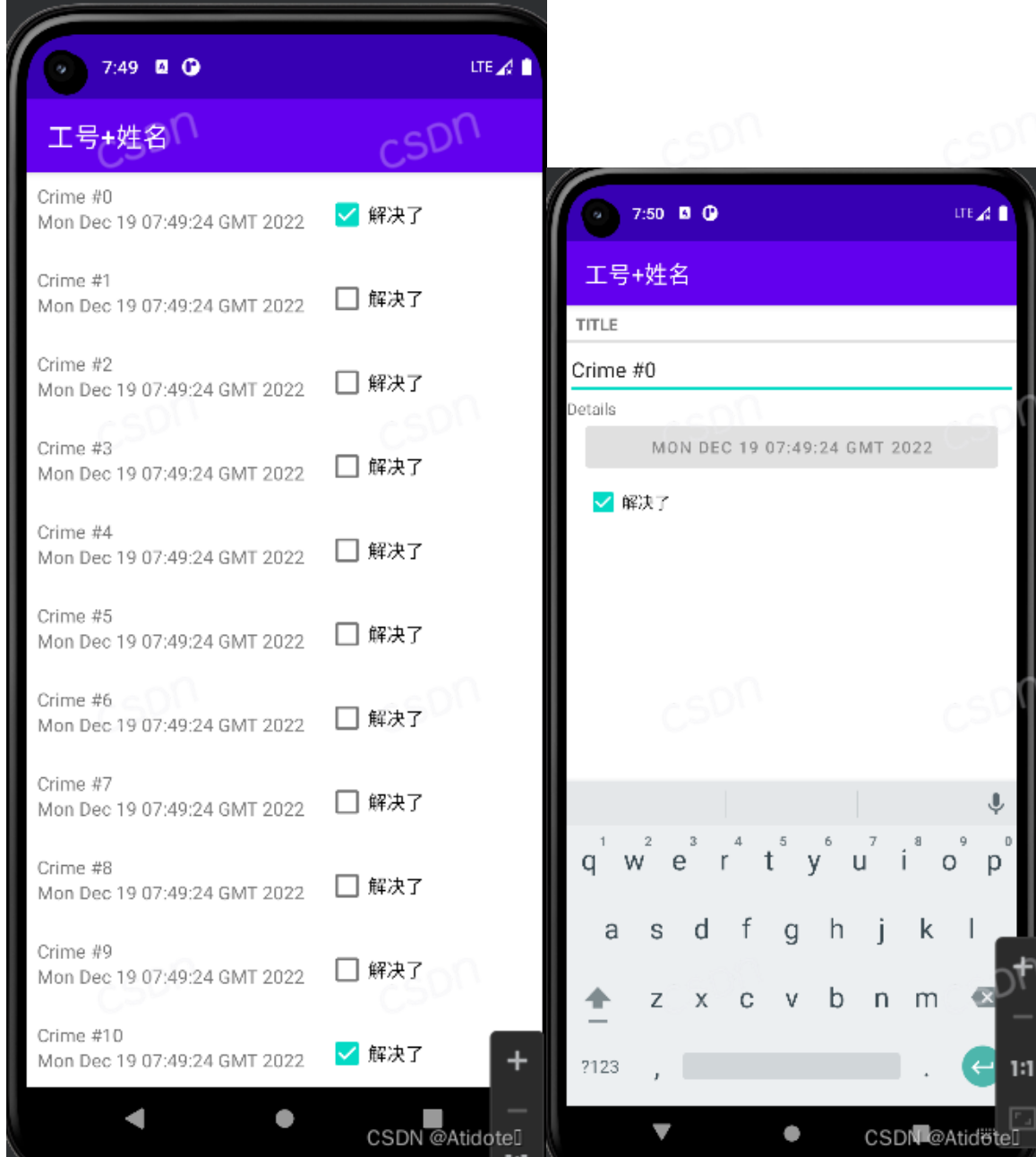
三、运行结果

内容来源: csdn.net

作者昵称: Atidote

原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622

作者主页: https://blog.csdn.net/qq_56204872



内容来源: csdn.net
作者昵称: Atidote
原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622
作者主页: https://blog.csdn.net/qq_56204872



四、问题解答

在Activity中加载一个Fragment的基本流程？

将Fragment加载到Activity当中有两种方式：

方式一：添加Fragment到Activity的布局文件当中

方式二：在Activity的代码中动态添加Fragment

第一种方式虽然简单但 **灵活性** 不够。添加Fragment到Activity的布局文件当中,就等同于将Fragment及其视图与activity的视图绑定在一起，且在activity的生命周期过程中，无法切换fragment视图。

第二种方式比较复杂，但也是唯一一种可以在运行时控制fragment的方式（加载、移除、替换）。

在Activity的布局文件中添加Fragment：（不推荐）

（1）新建文件fragment_***.xml和***Fragment.java：

（2）将Fragment添加到Activity的布局中，其中android:name属性填上你自己创建的fragment的完整类名。

使用Fragment设计UI有哪些好处？

1.Fragment可以使你能够将activity分离成多个可重用的组件，每个都有它自己的生命周期和UI。

2.Fragment可以轻松得创建动态灵活的UI设计，可以适应于不同的屏幕尺寸。从手机到平板电脑。

3.Fragment是一个独立的模块,紧紧地与activity绑定在一起。可以运行中动态地移除、加入、交换等。

4.Fragment提供一个新的方式让你在不同的安卓设备上统一你的UI。

内容来源: csdn.net

作者昵称: Atidote

原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622

作者主页: https://blog.csdn.net/qq_56204872

5.Fragment 切换流畅，轻量切换。

6.Fragment 替代TabActivity做导航，性能更好。

7.Fragment 在4.2.版本中新增嵌套fragment使用方法，能够生成更好的界面效果。

Fragment做局部内容更新更方便，原来为了到达这一点要把多个布局放到一个activity里面，现在可以用多Fragment来代替，只有在需要的时候才加载Fragment，提高性能。

可以从startActivityForResult中接收到返回结果,但是View不能。

内容来源: csdn.net

作者昵称: Atidote

原文链接: https://blog.csdn.net/qq_56204872/article/details/134899622

作者主页: https://blog.csdn.net/qq_56204872