



Fragment



目录

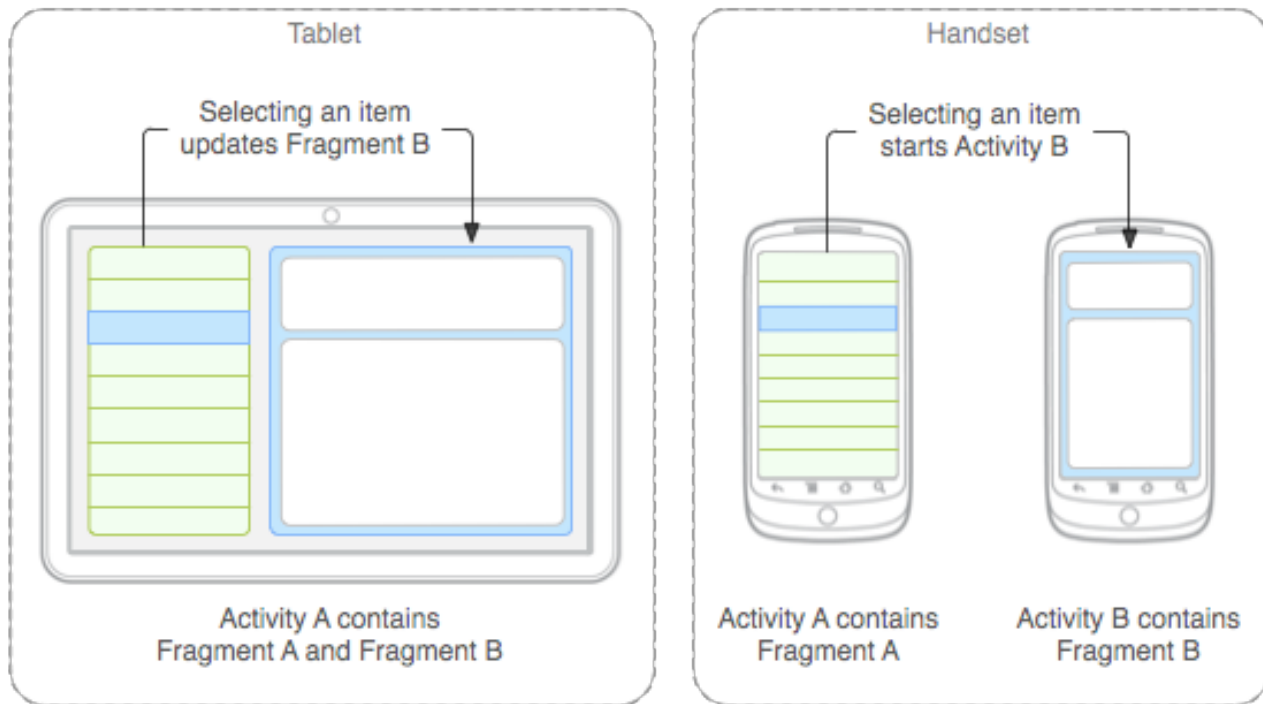
- ❑ 为什么
- ❑ 生命周期
- ❑ 如何动态添加
- ❑ 如何和 Fragment/Activity 通信



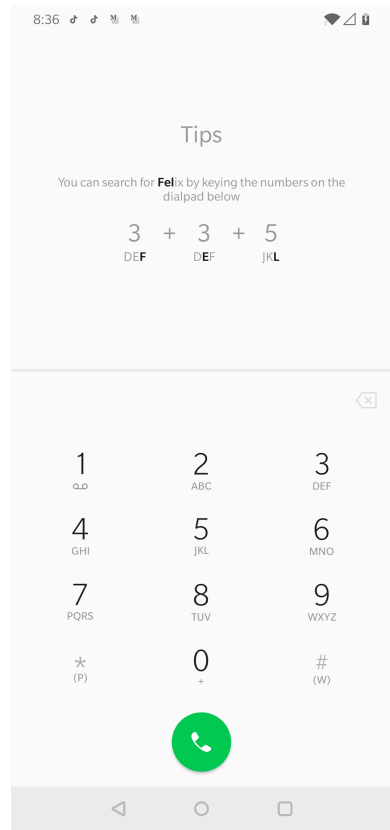
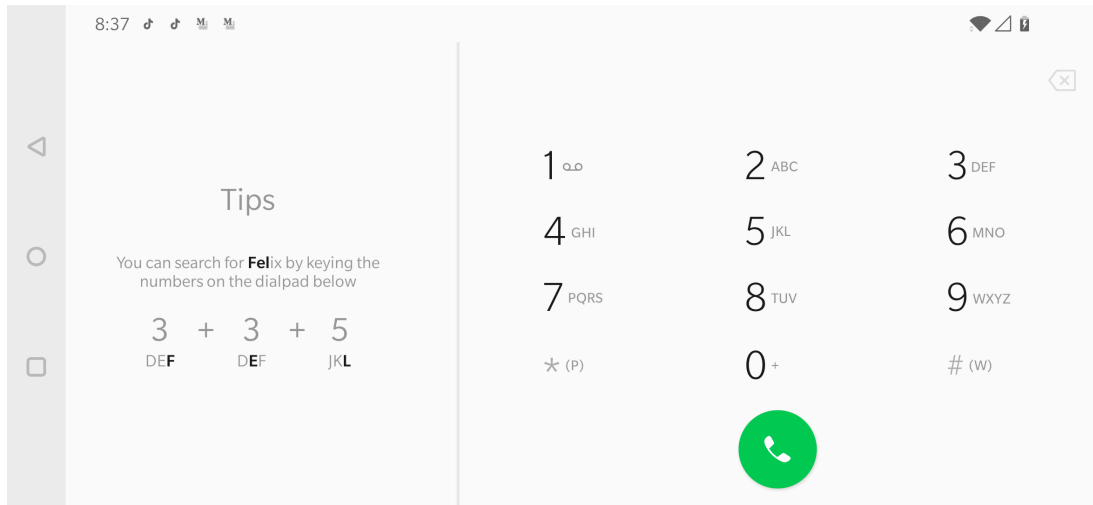
Fragment

- ❑ app components, mini-activity
- ❑ 可重用 UI 单元

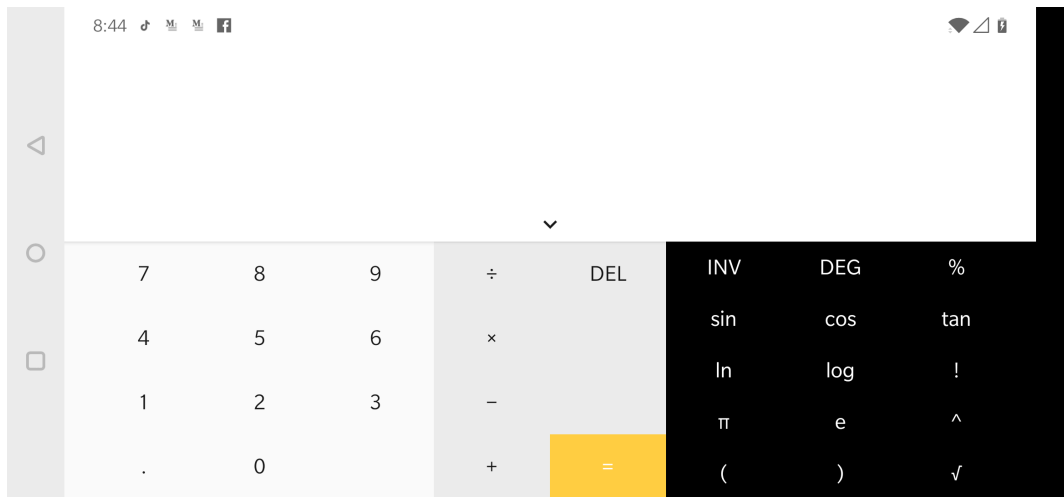
Fragment - UI 重用



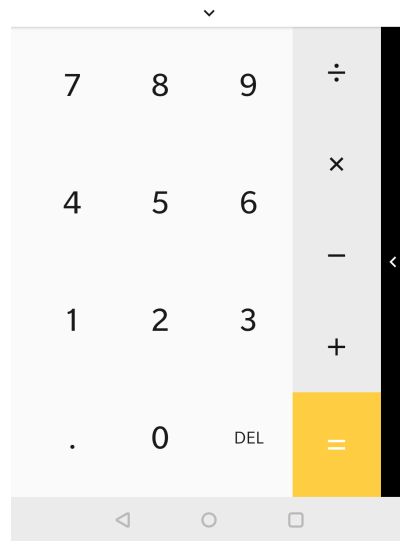
Fragment - Responsive Design



Fragment - Responsive Design



8:44 8 9 10 11 12





Why Fragment

- ❑ Activity 模块化
- ❑ 相比 View，带有生命周期管理
- ❑ 可重用，灵活
- ❑ 可以动态添加和删除

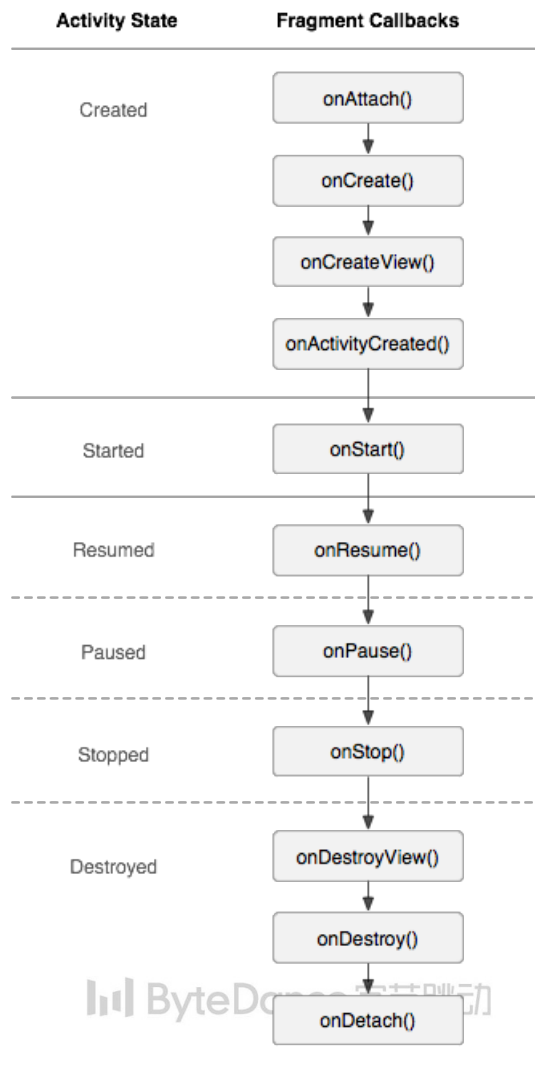


使用哪个？

- ❑ `android.support.v4.app.Fragment`
- ❑ `android.app.Fragment`
- ❑ 在各个版本的系统中行为都一致

生命周期

- ❑ onAttach/onDetach
- ❑ onCreate/onDestroy
- ❑ onCreateView/onDestroyView
- ❑ onActivityCreated
- ❑ onStart/onStop
- ❑ onResume/onPause





示例 - Lifecycle

- ❑ 定义 fragment 布局文件
- ❑ 定义 fragment 类
- ❑ 在 activity 布局文件中嵌入 fragment

示例 - Lifecycle - 1

❏ fragment_hello.xml

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="@string/hello_fragment"/>
</FrameLayout>
```

示例 - Lifecycle - 2

❏ HelloFragment.java

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

示例 - Lifecycle - 3

❏ activity_fragment_activity.xml

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

    <fragment android:name="com.example.fragment.f2.HelloFragment"
        android:id="@+id/hello_fragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
</FrameLayout>
```



动态添加/删除 Fragment

- ❑ FragmentManager
 - ❑ 动态添加/删除/替换 Fragment
 - ❑ FragmentTransaction
- ❑ Fragment 容器
 - ❑ 定义 Fragment 的位置和大小

示例 - Dynamically Add/Remove Fragment - 1

❑ 在 activity 布局文件中定义 fragment 容器

```
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">
```

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

```
</FrameLayout>
```

Demo - Dynamically Add/Remove Fragment - 2

❑ 使用 fragment transaction 添加 Fragment

```
getSupportFragmentManager().beginTransaction()  
    .add(R.id.fragment_container, new HelloFragment())  
    .commit();
```




Fragment/Activity 之间的通信

- ❑ 如何构造带参数的 Fragment
- ❑ 通过接口和回调

示例 - Communications - Arguments

```
public class StaticColorFragment extends Fragment {
    private static final String KEY_COLOR = "key_color";

    public static StaticColorFragment newFragment(int color) {
        StaticColorFragment fragment = new StaticColorFragment();
        Bundle args = new Bundle();
        args.putInt(KEY_COLOR, color);
        fragment.setArguments(args);
        return fragment;
    }

    public View onCreateView(...) {
        int color = Color.CYAN;
        Bundle args = getArguments();
        if (args != null) {
            color = args.getInt(KEY_COLOR, Color.CYAN);
        }
        View view = new View(inflater.getContext());
        view.setBackgroundColor(color);
        return view;
    }
}
```

示例 - Communications - Listeners & Setter

```
public class DynamicColorFragmentActivity extends AppCompatActivity implements OnColorSelectListener {
    ...

    @Override
    public void onAttachFragment(Fragment fragment) {
        super.onAttachFragment(fragment);
        if (fragment instanceof DynamicColorSelectorFragment) {
            ((DynamicColorSelectorFragment) fragment).setOnColorSelectListener(this);
        }
    }

    @Override
    public void onColorSelect(@ColorInt int color) {
        Fragment f = getSupportFragmentManager().findFragmentById(R.id.color_fragment_container);
        if (f instanceof DynamicColorFragment) {
            ((DynamicColorFragment) f).updateColor(color);
        }
    }
}
```



示例 - Master Detail

- ❑ Portrait
 - ❑ Master Activity: Item List
 - ❑ Detail Activity: Item Detail
- ❑ Landscape
 - ❑ One Activity: List & Detail



THANKS

liucheng@bytedance.com

