

Fragment

1、谈一谈 Fragment 的生命周期？

- 参考回答：
 - Fragment 从创建到销毁整个生命周期中涉及到的方法依次为：onAttach() → onCreate() → onCreateView() → onActivityCreated() → onStart() → onResume() → onPause() → onStop() → onDestroyView() → onDestroy() → detach()，其中和 Activity 有不少名称相同作用相似的方法，而不同的方法有：
 - **onAttach()**：当 Fragment 和 Activity 建立关联时调用；
 - **onCreateView()**：当 fragment 创建视图调用，在 onCreate 之后；
 - **onActivityCreated()**：当与 Fragment 相关联的 Activity 完成 onCreate() 之后调用；
 - **onDestroyView()**：在 Fragment 中的布局被移除时调用；
 - **detach()**：当 Fragment 和 Activity 解除关联时调用；

2、谈谈 Activity 和 Fragment 的区别？

- 参考回答：
 - 相似点：都可包含布局、可有自己的生命周期
 - 不同点：
 - Fragment 相比较于 Activity 多出 4 个回调周期，在控制操作上更灵活；
 - Fragment 可以在 XML 文件中直接进行写入，也可以在 Activity 中动态添加；
 - Fragment 可以使用 show()/hide() 或者 replace() 随时对 Fragment 进行切换，并且切换的时候不会出现明显的效果，用户体验会好；Activity 虽然也可以进行切换，但是 Activity 之间切换会有明显的翻页或者其他的效果，在小部分内容的切换上给用户的感觉不是很好；

3、Fragment 中 add 与 replace 的区别（Fragment 重叠）

- 参考回答：
 - add 不会重新初始化 fragment，replace 每次都会。所以如果在 fragment 生命周期内获取数据，使用 replace 会重复获取；

- 添加相同的 fragment 时，replace 不会有任何变化，add 会报 `IllegalStateException` 异常；
- replace 先 remove 掉相同 id 的所有 fragment，然后在 add 当前的这个 fragment，而 add 是覆盖前一个 fragment。所以如果使用 add 一般会伴随 `hide()` 和 `show()`，避免布局重叠；
- 使用 add，如果应用放在后台，或以其他方式被系统销毁，再打开时，`hide()` 中引用的 fragment 会销毁，所以依然会出现布局重叠 bug，可以使用 replace 或使用 add 时，添加一个 tag 参数；



```
//Fragment布局重叠解决方案，方法不唯一，仅供参考。
private int mCurrentPos = -1;
private List<Fragment> mFragments = new ArrayList<>();
private void switchFragmentIndex(int position) {
    FragmentTransaction transaction = getSupportFragmentManager().beginTransaction();
    if (mCurrentPos != -1) {
        transaction.hide(mFragments.get(mCurrentPos));
    }
    if (!mFragments.get(position).isAdded()) { //避免Fragment重复添加
        transaction.add(R.id.fl_content, mFragments.get(position));
    }
    transaction.show(mFragments.get(position)).commit();
    mCurrentPos = position; //记录当前Fragment下标
}
```

4、`getFragmentManager`、`getSupportFragmentManager`、`getChildFragmentManager` 之间的区别？

- 参考回答：
 - `getFragmentManager()` 所得到的是所在 fragment 的父容器的管理器，`getChildFragmentManager()` 所得到的是在 fragment 里面子容器的管理器，如果是 fragment 嵌套 fragment，那么就需要利用 `getChildFragmentManager()`；
 - 因为 Fragment 是 3.0 Android 系统 API 版本才出现的组件，所以 3.0 以上系统可以直接调用 `getFragmentManager()` 来获取 `FragmentManager()` 对象，而 3.0 以下则需要调用 `getSupportFragmentManager()` 来间接获取；

5、`FragmentPagerAdapter` 与 `FragmentStatePagerAdapter` 的区别与使用场景

- 参考回答：
 - 相同点：二者都继承 `PagerAdapter`
 - 不同点：**`FragmentPagerAdapter`** 的每个 Fragment 会持久的保存在 `FragmentManager` 中，只要用户可以返回到页面中，它都不会被销毁。因此适用于那些数据**相对静态**的页，Fragment **数量也比较少**的那种；

FragmentManager 只保留当前页面，当页面不可见时，该 Fragment 就会被消除，释放其资源。因此适用于那些数据动态性较大、占用内存较多，多 Fragment 的情况；