**1、什么是 Activity?**  
四大组件之一,一般的,一个用户交互界面对应一个 activity  
setContentView() ,// 要显示的布局  
button.setOnclickLinstener{  
}, activity 是 Context 的子类,同时实现了 window.callback 和  
keyevent.callback, 可以处理与窗体用户交互的事件.  
我开发常用的的有  
FragmentActivitiyListActivity ,PreferenceActivity ,TabAcitivty 等…

**2、请描述一下 Activity 生命周期**

Activity 从创建到销毁有多种状态，从一种状态到另一种状态时会激发相应  
的回调方法，这些回调方法包括：onCreate onStart onResume onPause  
onStop onDestroy  
其实这些方法都是两两对应的，onCreate 创建与 onDestroy 销毁；  
onStart 可见与 onStop 不可见；onResume 可编辑（即焦点）与 onPause；  
如果界面有共同的特点或者功能的时候,还会自己定义一个 BaseActivity.  
进度对话框的显示与销毁

**3、请描述一下 Activity 生命周期。**

生命周期描述的是一个类 从创建(new 出来)到死亡(垃圾回收)的过程中会执行的方法..  
在这个过程中一个单独的 Activity 的正常生命过程：oncreate（）、onstart （）、onpause  
（）、onStop（） 、 ondestroy（）。  
¬ onCreate: 在这里创建界面 ，做一些数据 的初始化工作  
¬ onStart: 到这一步变成用户可见不可交互的，一般在 OnCreate 后执行  
¬ onResume: 变成和用户可交互的，在 OnStart 后执行。（在 activity 栈系统通过栈  
的方式管理这些个 Activity 的最上面，运行完弹出栈，则回到上一个 Activity)  
¬ onPause: 窗体被交换到后台时，执行。到这一步是可见但不可交互 的，系统会停止  
动画等消耗 CPU 的事情从上文的描述已经知道，应该在这里保存你的一些数据,因为  
这个时候你的程序的优先级降低，有可能被系统收回。在这里保存的数据，应该在  
onResume 里读出来，注意：这个方法里做的事情时间要短，因为下一 个 activity 不  
会等到这个方法完成才启动  
¬ onstop: onPause 事件之后执行。如果一段时间内用户还没有重新查看该窗体，则该  
窗体的 onStop 事件将会被执行；或者用户直接按了 Back 键，将该窗体从当前 Task  
中移除，也会执行该窗体的 onStop 事件。  
¬ onDestroy: 这是 activity 被干掉前最后一个被调用方法了，可能是外面类调用 finish 方  
法或者是系统为了节省空间将它暂时性的干掉，可以用 isFinishing()来判断它，如果你  
有一个 Progress Dialog 在线程中转动，请在 onDestroy 里把他 cancel 掉，不然等线  
程结束的时候，调用

**4、如何保存 Activity 的状态？**  
Activity 的状态通常情况下系统会自动保存的，只有当我们需要保存额外的  
数据时才需要使用到这样的功能。  
一般来说, 调用 onPause()和 onStop()方法后的 activity 实例仍然存在于内  
存中, activity 的所有信息和状态数据不会消失, 当 activity 重新回到前台之后,  
所有的改变都会得到保留。  
但是当系统内存不足时, 调用 onPause()和 onStop()方法后的 activity 可能  
会被系统摧毁, 此时内存中就不会存有该 activity 的实例对象了。如果之后这个  
activity 重新回到前台, 之前所作的改变就会消失。 为了避免此种情况的发生, 我  
们可以覆写 onSaveInstanceState()方法。onSaveInstanceState()方法接受一  
个 Bundle 类型的参数, 开发者可以将状态数据存储到这个 Bundle 对象中, 这  
样 即 使 activity 被 系 统 摧 毁 , 当 用 户 重 新 启 动 这 个 activity 而 调 用 它 的  
传智播客武汉校区就业部出品 务实、创新、质量、分享、专注、责任  
onCreate()方法时, 上述的 Bundle 对象会作为实参传递给 onCreate()方法, 开  
发者可以从 Bundle 对象中取出保存的数据, 然后利用这些数据将 activity 恢复  
到被摧毁之前的状态。  
需要注意的是, onSaveInstanceState()方法并不是一定会被调用的, 因为有  
些场景是不需要保存状态数据的. 比如用户按下 BACK 键退出 activity 时, 用户  
显然想要关闭这个 activity, 此时是没有必要保存数据以供下次恢复的, 也就是  
onSaveInstanceState()方法不会被调用. 如果调用 onSaveInstanceState()方  
法, 调用将发生在 onPause()或 onStop()方法之前。  
@Override  
**protected void** onSaveInstanceState(Bundle outState) {  
// **TODO** Auto-generated method stub  
**super**.onSaveInstanceState(outState);  
}

**5、两个 Activity 之间跳转时必然会执行的是哪几个方  
法？**  
一般情况下比如说有两个activity,分别叫A,B,当在A里面激活B组件的时候,  
A 会调用 onPause()方法,然后 B 调用 onCreate() ,onStart(), onResume()。  
这个时候 B 覆盖了窗体, A 会调用 onStop()方法. 如果 B 是个透明的,或者  
是对话框的样式, 就不会调用 A 的 onStop()方法。

**6、横竖屏切换时 Activity 的生命周期**  
此时的生命周期跟清单文件里的配置有关系。  
1．不设置 Activity 的 android:configChanges 时，切屏会重新调用各个生  
传智播客武汉校区就业部出品 务实、创新、质量、分享、专注、责任  
命周期默认首先销毁当前 activity,然后重新加载。  
2．设置 Activity  
android:configChanges="orientation|keyboardHidden|screenSize"时，切  
屏不会重新调用各个生命周期，只会执行 onConfigurationChanged 方法。  
通常在游戏开发, 屏幕的朝向都是写死的。

**7、如何将一个 Activity 设置成窗口的样式**  
只需要给我们的 Activity 配置如下属性即可。  
android:theme="@android:style/Theme.Dialog"

**8、如 何 退 出 Activity ？ 如 何 安 全 退 出 已 调 用 多 个Activity 的 Application？**  
1、通常情况用户退出一个 Activity 只需按返回键，我们写代码想退出 activity  
直接调用 finish()方法就行。  
2、记录打开的 Activity：  
每打开一个 Activity，就记录下来。在需要退出时，关闭每一个 Activity 即可。  
//伪代码  
List<Activity> lists ;// 在 application 全局的变量里面  
lists = **new** ArrayList<Activity>();  
lists.add(**this**);  
**for**(Activity activity: lists)  
{  
activity.finish();  
}  
lists.remove(**this**);  
3、发送特定广播：  
传智播客武汉校区就业部出品 务实、创新、质量、分享、专注、责任  
在需要结束应用时，发送一个特定的广播，每个 Activity 收到广播后，关闭  
即可。  
//给某个 activity 注册接受接受广播的意图  
registerReceiver(receiver, filter)  
//如果过接受到的是 关闭 activity 的广播 就调用 finish()方法 把当前的  
activity finish()掉  
4、递归退出  
在打开新的 Activity 时使用 startActivityForResult，然后自己加标志，在  
onActivityResult 中处理，递归关闭。  
5、其实 也可以通过 intent 的 flag 来实现  
intent.setFlags(Intent.FLAG\_ACTIVITY\_CLEAR\_TOP)激活一个新的 activity。  
此时如果该任务栈中已经有该 Activity，那么系统会把这个 Activity 上面的所有  
Activity 干掉。其实相当于给 Activity 配置的启动模式为 SingleTop。

**9、Activity 的四种启动模式，singletop 和 singletask区别是什么？一般书签的使用模式是 singletop，那为什么不使用 singletask?**  
singleTop 跟 standard 模式比较类似。唯一的区别就是，当跳转的对象是位于栈顶的 activity（应该可以理解为用户眼前所 看到的 activity）时，程序将不会生成一个新的 activity 实例，而是直接跳到现存于栈顶的那个 activity 实例。拿上面的例子来说，当 Act1 为 singleTop 模式时，执行跳转后栈里面依旧只有

国信安犀浦校区就业部出品 务实、创新、质量、分享、专注、责任一个实例，如果现在按返回键程序将直接退出。

singleTask 模式和 singleInstance 模式都是只创建一个实例的。在这种模式下，无论跳转的对象是不是位于栈顶的 activity，程序都不会生成一个新的实例 （当然前提是栈里面已经有这个实例） 。 这种模式相当有用， 在以后的多 activity开发中，常会因为跳转的关系导致同个页面生成多个实例，这个在用户体验上始终有点不好，而如果你将对应的 activity 声明为 singleTask 模式，这种问题将不复存在。在主页的 Activity 很常用

**10、当打开 Activity 后，按 HOME 键时，当前 Activity 执行的生命周期是？**

HOME 键的执行顺序：onPause->onStop->onRestart->onStart->onResume

**11、 当打开 Activity 后，按下 Back 键，执行的生命周期是？**

BACK 键的顺序： onPause->onStop->onDestroy->onCreate->onStart->onResume

**12、Intent 传递数据时，可以传递哪些类型数据？**

Intent 可以传递的数据类型非常的丰富，java 的基本数据类型和 String  
以及他们的数组形式都可以，除此之外还可以传递实现了 Serializable 和  
Parcelable 接口的对象。

**13、Serializable 和 Parcelable 的区别**  
在使用内存的时候，Parcelable 类比 Serializable 性能高，所以推荐使用  
Parcelable 类。  
1．Serializable 在序列化的时候会产生大量的临时变量，从而引起频繁的  
GC。  
2．Parcelable 不能使用在要将数据存储在磁盘上的情况。 尽管 Serializable  
效率低点，但在这种情况下，还是建议你用 Serializable 。  
实现：  
1．Serializable 的实现，只需要继承 Serializable 即可。这只是给对象打  
了一个标记，系统会自动将其序列化。  
2．Parcelabel 的实现，需要在类中添加一个静态成员变量 CREATOR，这个变量需要继承 Parcelable.Creator 接口。

**14、请描述一下 Intent 和 IntentFilter**

Android 中通过 Intent 对象来表示一条消息，一个 Intent 对象不仅包含  
有这个消息的目的地，还可以包含消息的内容，这好比一封 Email，其中不仅应  
该包含收件地址，还可以包含具体的内容。对于一个 Intent 对象，消息“目的  
地”是必须的，而内容则是可选项。  
通过 Intent 可以实现各种系统组件的调用与激活.  
IntentFilter: 可以理解为邮局或者是一个信笺的分拣系统

这个分拣系统通过 3 个参数来识别  
Action: 动作 view  
Data: 数据 uri uri  
Category : 而外的附加信息  
Action 匹配  
Action 是一个用户定义的字符串，用于描述一个 Android 应用程序组件，  
一 个 IntentFilter 可 以 包 含 多 个 Action 。 在 AndroidManifest.xml 的  
Activity 定义时可以在其 <intent-filter >节点指定一个 Action 列表用于标  
示 Activity 所能接受的“动作”，例如：  
<intent-filter >  
<action android:name="android.intent.action.MAIN" />  
<action android:name="cn.itheima.action" />  
……  
</intent-filter>  
如果我们在启动一个 Activity 时使用这样的 Intent 对象：  
Intent intent =new Intent();  
intent.setAction("cn.itheima.action");  
那么所有的 Action 列表中包含了“cn.itheima”的 Activity 都将会匹配  
成功。  
Android 预定义了一系列的 Action 分别表示特定的系统动作。这些  
Action 通过常量的方式定义在 android.content. Intent 中，以“ACTION\_”  
开头。我们可以在 Android 提供的文档中找到它们的详细说明。

URI 数据匹配  
一个 Intent 可以通过 URI 携带外部数据给目标组件。在 <intent-filter >  
节点中，通过 <data/>节点匹配外部数据。  
mimeType 属性指定携带外部数据的数据类型，scheme 指定协议，host、  
port、path 指定数据的位置、端口、和路径。如下：  
<data android:mimeType="mimeType" android:scheme="scheme"  
android:host="host" android:port="port" android:path="path"/>  
电话的 uri tel: 12345  
http://www.baidu.com  
自己定义的 uri itcast://cn.itcast/person/10  
如果在 Intent Filter 中指定了这些属性，那么只有所有的属性都匹配成功  
时 URI 数据匹配才会成功。  
Category 类别匹配  
<intent-filter >节点中可以为组件定义一个 Category 类别列表，当  
Intent 中包含这个列表的所有项目时 Category 类别匹配才会成功。

**15、隐式 Intent 和显示 Intent 的区别**  
显示 Intent：明确指定了目标组件名称的 Intent。  
隐式 Intent：没有明确指出目标组件的名称，称为隐式 Intent，Android 通过 IntentFilter 来  
寻找与隐式 Intent 相关的对象。

**16、能说出几个系统内置的 Intent 吗？**打电话、发短信息、打开浏览器等其它系统定义的 Intent

**17、谈谈 android 为什么要设计意图？**Intent 就是这个这个意图 ，应用程序间 Intent 进行交流，打个电话啦，来个 电话啦都会发 Intent, 这个是 Android 架构的松耦合的精髓部分，大大提高了组件的复用性

**18、startActivityForResult()这个方法有什么用？**返回上一个 Activity 并且能将当前 Activity 的一些状态值和信息传递给上一 Activity