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, 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 ListActivity ,PreferenceActivity 等…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

2. 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 会针对不同的生命阶段会调用不同的方法
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 ondestroy onstop onstart onresume onpause
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；
这 6 个方法是相对应的,那么就只剩下一个 onRestart 方法了,这个方法在什么时候调用呢?
答案就是：在 Activity 被 onStop 后，但是没有被 onDestroy，在再次启动此 Activity 时就调用 onRestart（而不再调用 onCreate）方法；
如果被 onDestroy 了，则是调用 onCreate 方法。
最后讲自己项目中的经验,比如说豆瓣客户端每次进入某个界面的时候要刷新列表,这个刷新列表的操作 就放在 onStart()的方法里面.这样保证每次用户看到的数据都是最新的.
在读文档的时候 还发现 activity 还有两个方法 onPostResume() 和 OnPostCreate()这两个生命周期的方法,不过开发的时候没有用到过.

3. 两个 Activity 之间跳转时必然会执行的是哪几个方法。

一般情况比如说有两个 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 呢 是个透明的,或者是对话框的样式, 就不会调用 onStop()方法

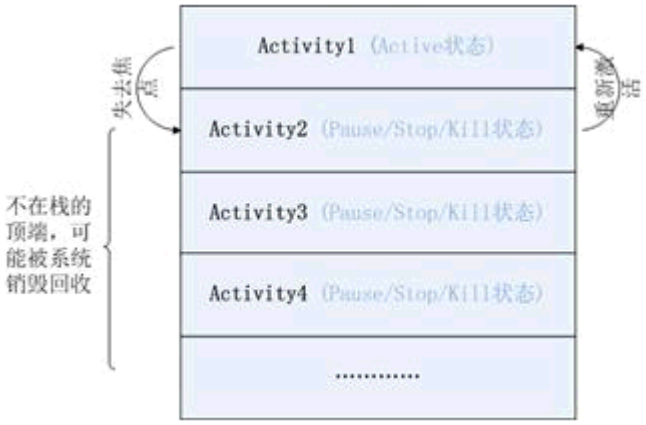
4. 横竖屏切换时候 Activity 的生命周期。

这个生命周期跟清单文件里的配置有关系
修改 AndroidManifest.xml，把该 Activity 添加 android:configChanges="xxx"，
1、不设置 Activity 的 android:configChanges 时，切屏会重新调用各个生命周期
2、设置 Activity 的 android:configChanges="orientation|keyboardHidden"时，切屏不会重新调用各个生命周期，只会执行 onConfigurationChanged 方法

5. 如何将一个 Activity 设置成窗口的样式。

可以自定义一个 activity 的样式,详细见手机卫士的程序详细信息
android:theme="@style/FloatActivity"
E:\day9\mobilesafe\res\values\style

6. 你后台的 Activity 被系统 回收怎么办？如果后台的 Activity 由于某原因被系统回收可了，如何在被系统回收之前保存当前状态？



除了在栈顶的 activity,其他的 activity 都有可能在内存不足的时候被系统回收,一个 activity 越处于栈底,被回收的可能性越大.

保存之前状态的方法：重写 activity 的 onSaveInstanceState(Bundle outState)方法，在里面进行一些保存状态的操作，然后跳转到该 activity 时，若是 activity 被回收，会重新调用它的 oncreate(Bundle bundle),然后根据 bundle 参数在 oncreate（）方法中写一些恢复状态的操作。

```
protected void onSaveInstanceState(Bundle outState) {
    super.onSaveInstanceState(outState);
    outState.putLong("id", 1234567890);
}
public void onCreate(Bundle savedInstanceState) {
//判断 savedInstanceState 是不是空.
//如果不为空就取出来
        super.onCreate(savedInstanceState);
}
```

7. 如何退出 Activity? 如何安全退出已调用多个 Activity 的 Application?

1、抛异常强制退出：

该方法通过抛异常，使程序 Force Close。

验证可以，但是，需要解决的问题是，如何使程序结束掉，而不弹出 Force Close 的窗口。

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。

3、发送特定广播：

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。

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。

上面是网上的一些做法.

其实 可以通过 intent 的 flag 来实现.. intent.setFlag(FLAG_ACTIVITY_CLEAR_TOP)激活一个新的 activity,然后在新的 activity 的 oncreate 方法里面 finish 掉.

讲一讲你对 activity 的理解

把上面的几点用自己的心得写出来

8. service 是否在 main thread 中执行, service 里面是否能执行耗时的操作?
默认情况,如果没有显示的指定 service 所运行的进程, Service 和 activity 是运行在当前 app 所在进程的 main thread(UI 主线程)里面

9. 两个 Activity 之间怎么传递数据?
基本数据类型可以通过. Intent 传递数据
Application 全局里面存放 对象 , 每个 activity 都可以取到
让对象实现 implements Serializable 接口把对象存放到文件上.

10. 怎么让在启动一个 Activity 是就启动一个 service?
在 activity 的 onCreate()方法里面 startService();

11. 同一个程序,但不同的 Activity 是否可以放在不同的 Task 任务栈中?
比方说在激活一个新的 activity 时候, 给 intent 设置 flag
Intent 的 flag 添加 FLAG_ACTIVITY_NEW_TASK
这个被激活的 activity 就会在新的 task 栈里面…

12. Activity 怎么和 service 绑定,怎么在 activity 中启动自己对应的 service?
bindService(). 让 activity 能够访问到 service 里面的方法
构建一个 intent 对象, 通过 bindService 的方法去启动一个服务,
ServiceConnection 对象(重写 onServiceConnected 和 OnServiceDisconnected 方法) 和 BIND_AUTO_CREATE.

13. 什么是 Service 以及描述下它的生命周期。Service 有哪些启动方法,有什么区别,怎样停用 Service?
在 Service 的生命周期中,被回调的方法比 Activity 少一些,只有 onCreate, onStart, onDestroy,onBind 和 onUnbind。
通常有两种方式启动一个 Service,他们对 Service 生命周期的影响是不一样的。
1) 通过 startService
Service 会经历 onCreate 到 onStart,然后处于运行状态,stopService 的时候调用 onDestroy 方法。
如果是调用者自己直接退出而没有调 用 stopService 的话,Service 会一直在后台运行。
2) 通过 bindService
Service 会运行 onCreate,然后是调用 onBind, 这个时候调用者和 Service 绑定在一起。
调用者退出了,Srevice 就会调用 onUnbind->onDestroyed 方法。
所谓绑定在一起就共存亡了。调用者也可以通过调用 unbindService 方法来停止服务,这时候 Srevice 就会调用 onUnbind->onDestroyed 方法。
需要注意的是如果这几个方法交织在一起的话,会出现什么情况呢?
一个原则是 Service 的 onCreate 的方法只会被调用一次,就是你无论多少次的 startService 又 bindService,Service 只被创建一次。
如果先是 bind 了,那么 start 的时候就直接运行 Service 的 onStart 方法,如果先是 start,那么 bind 的时候就直接运行 onBind 方法。
如果 service 运行期间调用了 bindService,这时候再调用 stopService 的话,service 是

不会调用 onDestroy 方法的，service 就 stop 不掉了，只能先 UnbindService，再 StopService。
如果一个 service 通过 startService 被 start 之后，多次调用 startService 的话，service 会多次调用 onStart 方法。多次调用 stopService 的话，service 只会调用一次 onDestroyed 方法。
如果一个 service 通过 bindService 被 start 之后，多次调用 bindService 的话，service 只会调用一次 onBind 方法。多次调用 unbindService 的话会出异常。

15. 不用 service，B 页面为音乐播放，从 A 跳转到 B，再返回，如何使音乐继续播放？
这个问题问的很山寨.默认不做任何处理,B 里面的音乐都能播放.
遇到问题，可以随机应变,灵活发挥,多考虑些细节,比如说这个题就可以这样说,说说你对 startActivityForResult 的理解()
A 开启 B 的时候,用 startActivityForResult()方法，B 返回的时候把播放的状态信息返回给 A ,A 继续播放音乐.

16. 什么是 IntentService？有何优点？
Sdk 给我们提供的方便的,带有异步处理的 service 类,
OnHandleIntent() 处理耗时的操作

17. 什么时候使用 Service？
拥有 service 的进程具有较高的优先级
官方文档告诉我们，Android 系统会尽量保持拥有 service 的进程运行，只要在该 service 已经被启动(start)或者客户端连接(bindService)到它。当内存不足时，需要保持，拥有 service 的进程具有较高的优先级。
1. 如果 service 正在调用 onCreate,onStartCommand 或者 onDestory 方法，那么用于当前 service 的进程则变为前台进程以避免被 killed。
2. 如果当前 service 已经被启动(start)，拥有它的进程则比那些用户可见的进程优先级低一些，但是比那些不可见的进程更重要，这就意味着 service 一般不会被 killed.
3. 如果客户端已经连接到 service (bindService),那么拥有 Service 的进程则拥有最高的优先级，可以认为 service 是可见的。
4. 如果 service 可以使用 startForeground(int, Notification)方法来将 service 设置为前台状态，那么系统就认为是对用户可见的，并不会在内存不足时 killed。
如果有其他的应用组件作为 Service,Activity 等运行在相同的进程中，那么将会增加该进程的重要性。
1.Service 的特点可以让他在后台一直运行,可以在 service 里面创建线程去完成耗时的操作.
2.Broadcast receiver 捕获到一个事件之后,可以起一个 service 来完成一个耗时的操作.
3.远程的 service 如果被启动起来,可以被多次 bind, 但不会重新 create. 索爱手机 X10i 的人脸识别的 service 可以被图库使用,可以被摄像机,照相机等程序使用.

18. 请描述一下 Intent 和 Intent Filter。
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.
Intent filter: 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
Data: 数据 uri
Category : 数据类型
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 Intent Filter 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.itcast.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.itcast");
```

那所有么的 Action 列表中包含了“cn.itcast”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
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。
默认是 DEFAULT

19. Intent 传递数据时，可以传递哪些类型数据？

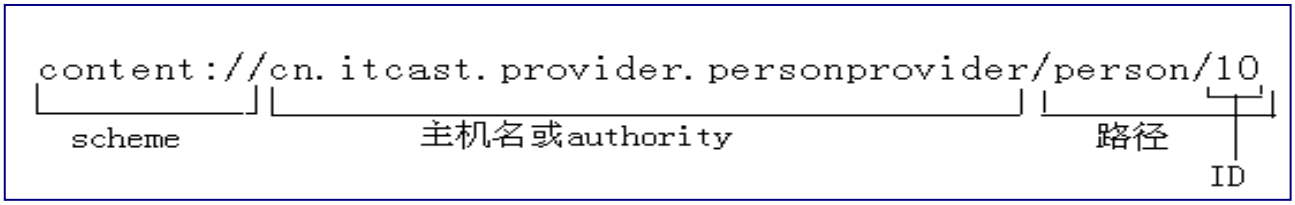
1.一般的基本数据类型 Intent .putExtra() intent.getExtra();
2.数据的 uri, intent.setData() intent.getData();

20. 说说 Activity，Intent，Service 是什么关系 。
麦当劳和麦当娜的关系是什么关系?
这种问题,就讲下 activity,讲一下 service,说一下 通过 intent 去激活组件,传递数据.
说自己项目中有这样一个网络更新的功能,显示界面就用的 activity, 后台有个 service 每隔半小时都去访问下服务器获取更新的数据…开启服务用的是 intent 来开启

21. 请描述一下 Broadcast Receiver。
用于接收系统的广播通知, 系统会有很多 sd 卡挂载,手机重启,广播通知,低电量,来电,来短信等….
手机卫士中自定义一个 broadcast receiver 来获取短信到来的广播, 根据黑名单来判断是否拦截该短信.
画画板生成图片后,发送一个 sd 挂载的通知,通知系统的 gallery 去获取到新的图片.

22. 在 manifest 和代码中如何注册和使 用 broadcast receiver 。
设置广播接收者的优先级,设置广播接受者的 action 名字 等…
详细见工程代码.

23. 请介绍下 ContentProvider 是如何实现数据共享的。
把自己的数据通过 uri 的形式共享出去



需要去实现一个类去继承 ContentProvider

```
public class PersonContentProvider extends ContentProvider{
    public boolean onCreate(){
        //..
    }
    query(Uri, String[], String, String[], String)
    insert(Uri, ContentValues)
    update(Uri, ContentValues, String, String[])
    delete(Uri, String, String[])
}
```

24. 请介绍下 Android 的数据存储方式。
文件
数据库
SharedPreference
网络

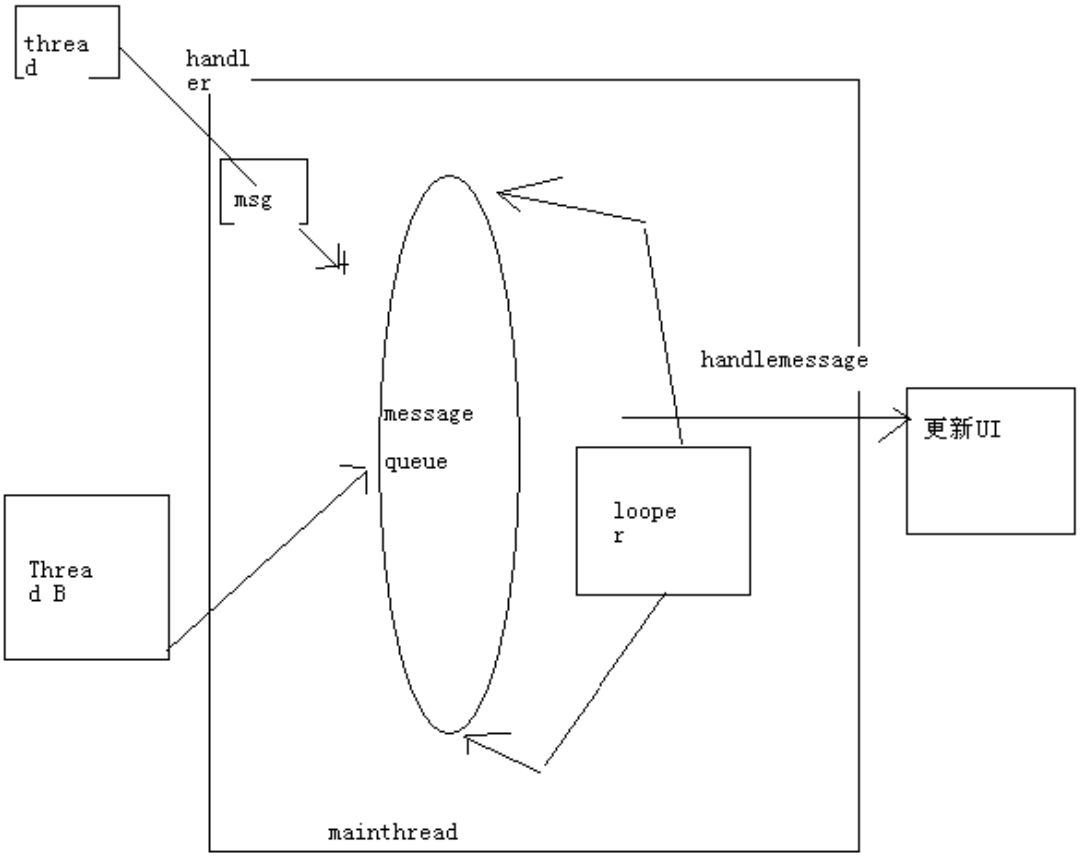
25. 为什么要用 ContentProvider？它和 sql 的实现上有什么差别？
屏蔽数据存储的细节,对用户透明,用户只需要关心操作数据的 uri 就可以了
不同 app 之间共享,操作数据
Sql 也有增删改查的方法.
但是 contentprovider 还可以去增删改查本地文件.

26. 请介绍下 Android 中常用的五种布局。
FrameLayout（框架布局），LinearLayout （线性布局），AbsoluteLayout（绝对布局），RelativeLayout（相对布局），TableLayout（表格布局）
FrameLayout
从屏幕的左上角开始布局,叠加显示, 实际应用 播放器的暂停按钮.
LinearLayout
线性布局，这个东西，从外框上可以理解为一个 div，他首先是一个一个从上往下罗列在屏幕上。每一个 LinearLayout 里面又可分为垂直布局（android:orientation="vertical"）和水平布局（android:orientation="horizontal"）。当垂直布局时，每一行就只有一个元素，多个元素依次垂直往下；水平布局时，只有一行，每一个元素依次向右排列。
AbsoluteLayout
绝对布局犹如 div 指定了 absolute 属性，用 X,Y 坐标来指定元素的位置
android:layout_x="20px"
android:layout_y="12px"
指定平板机型的游戏开发中经常用到绝对布局
RelativeLayout
相对布局可以理解为某一个元素为参照物，来定位的布局方式。主要属性有：相对于某一个元素
android:layout_below="@id/aaa" 该元素在 id 为 aaa 的下面
android:layout_toLeftOf="@id/bbb" 改元素的左边是 bbb
相对于父元素的地方
android:layout_alignParentLeft="true" 在父元素左对齐
android:layout_alignParentRight="true" 在父元素右对齐
TableLayout
表格布局类似 Html 里面的 Table。每一个 TableLayout 里面有表格行 TableRow，TableRow 里面可以具体定义每一个元素，设定他的对齐方式 android:gravity="" 。
每一个布局都有自己适合的方式，另外，这五个布局元素可以相互嵌套应用，做出美观的界面。

27. 谈谈 UI 中， Padding 和 Margin 有什么区别？
Padding 文字对边框, margin 是控件对父窗体.

28. widget 相对位置的完成在 antivity 的哪个生命周期阶段实现。
这个题没看懂… widget 可以理解成桌面小空间, 也可以理解成 某个 button, imageview 这样的控件…

29. 请解释下在单线程模型中 Message、Handler、Message Queue、Looper 之间的关系。



30. AIDL 的全称是什么？如何工作？

Android interface definition language (android 接口定义语言)，用来跨进程的访问方法,

访问远程的服务的方法. 如何工作 day7 queryStudent . 手机卫士 Itelephony 接口挂断电话.

31. 请解释下 Android 程序运行时权限与文件系统权限的区别。

Android 程序执行需要读取到安全敏感项必需在 androidmanifest.xml 中声明相关权限请求,打电话,访问网络,获取坐标,读写 sd 卡,读写联系人等..安装的时候会提示用户…

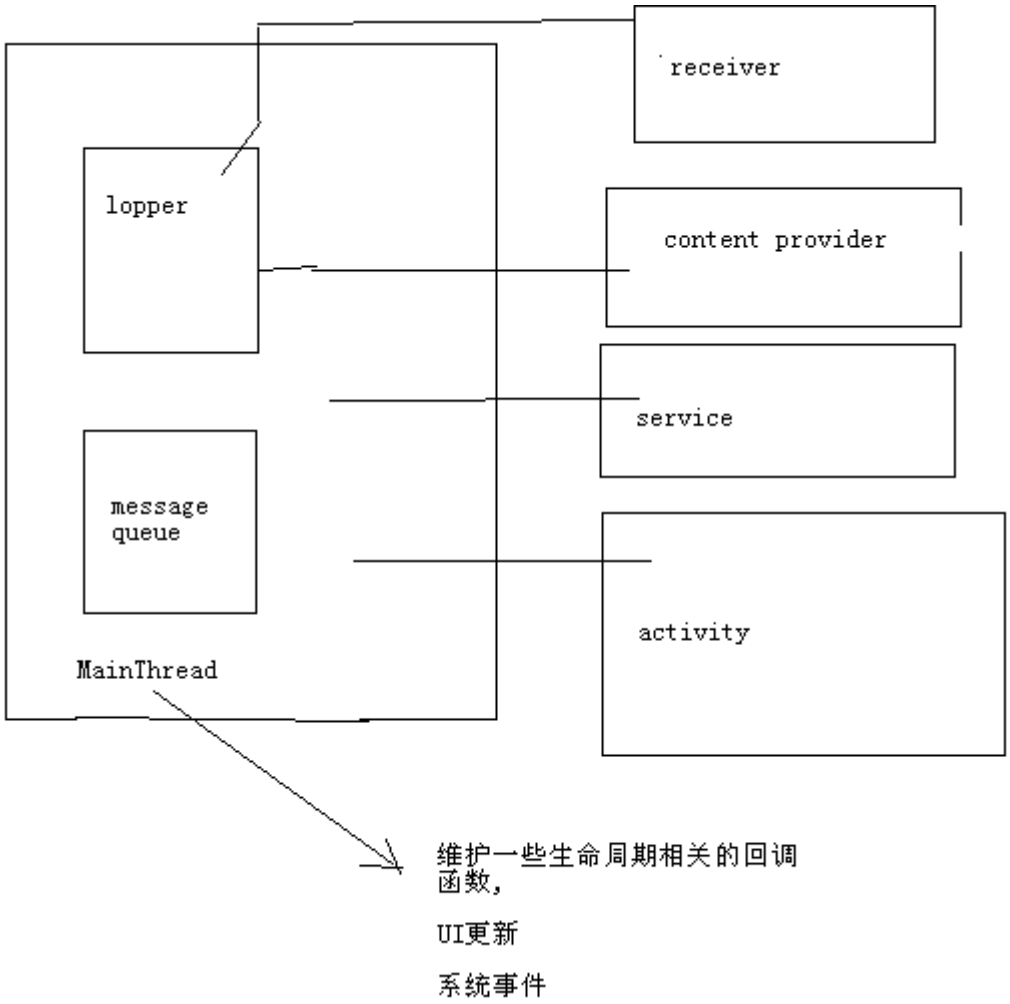
文件系统的权限是 linux 权限. 比如说 sharedpreference 里面的 Context.Mode.private Context.Mode.world_read_able Context.Mode_world_writeable

32. 系统上安装了多种浏览器，能否指定某浏览器访问指定页面？

找到对应的浏览器的意图,传递数据 URI ， 激活这个意图

33. 对 android 主线程的运用和理解。

主 ui 线程不能执行耗时的操作



34. 对 android 虚拟机的理解，包括内存管理机制垃圾回收机制。
虚拟机很小,空间很小,谈谈移动设备的虚拟机的大小限制 16M ， 谈谈加载图片的时候怎么处理大图片的，垃圾回收,没有引用的对象,在某个时刻会被系统 gc 掉.

35. Framework 工作方式及原理，Activity 是如何生成一个 view 的，机制是什么。
可以讲下 activity 的源码,比如说 每个 activity 里面都有 window.callback 和 keyevent.callback,一些回调的接口或者函数吧. 框架把 activity 创建出来就会调用里面的这些回调方法,会调用 activity 生命周期相关的方法.
Activity 创建一个 view 是通过 ondraw 画出来的, 画这个 view 之前呢,还会调用 onmeasure 方法来计算显示的大小.

36. android 本身的一些限制，比如 apk 包大小限制，读取大文件时的时间限。
这个问题问的有问题, apk 包大小限制不好说,极品飞车有 100M 还是能装到手机上,
读大文件的时间限制应该是 main 线程里面的时间限制吧.5 秒.

37. 如何加载的音乐信息，如何改善其效率。
Android 提供 mediascanner,mediaStore 等接口, 音乐文件的信息都会存放到系统的数据库表中,可以通过 content provider 获取,显示出来,改善效率,是个常见问题, 可以从以下几

个方面作答,分批加载数据， 延时加载数据， 合理使用缓存等...

38. ListView 如何提高其效率？
复用 convertview ， 异步加载数据,分页加载数据,使用静态的 view 对象 避免创建过多的 view.

39. 启动应用后，改变系统语言，应用的语言会改变么？
不会

40. 启动一个程序，可以主界面点击图标进入，也可以从一个程序中跳转过去，二者有什么区别？
区别是根据 activity 在 manifest 里面的配置,这个 activity 可能会放在不同的 task 栈里面

41. Android 程序与 Java 程序的区别？
Android 程序用 android sdk 开发,java 程序用 javasdk 开发.
Android SDK 引用了大部分的 Java SDK，少数部分被 Android SDK 抛弃，比如说界面部分，java.awt package 除了 java.awt.font 被引用外，其他都被抛弃，在 Android 平台开发中不能使用。将 Java 游戏或者 j2me 程序移植到 Android 平台的过程中,Android SDK 与 Java SDK 的区别是很需要注意的地方。

42. Android 中 Task 任务栈的分配。
首先我们来看下 Task 的定义，Google 是这样定义 Task 的：a task is what the user experiences as an "application." It's a group of related activities, arranged in a stack. A task is a stack of activities, not a class or an element in the manifest file. 这意思就是说 Task 实际上是一个 Activity 栈，通常用户感受的一个 Application 就是一个 Task。从这个定义来看,Task 跟 Service 或者其他 Components 是没有任何联系的，它只是针对 Activity 而言的。
Activity 有不同的启动模式， 可以影响到 task 的分配
Task，简单的说，就是一组以栈的模式聚集在一起的 Activity 组件集合。它们有潜在的前后驱关联，新加入的 Activity 组件，位于栈顶，并仅有在栈顶的 Activity，才会有机会与用户进行交互。而当栈顶的 Activity 完成使命退出的时候，Task 会将其退栈，并让下一个将跑到栈顶的 Activity 来于用户面对面，直至栈中再无更多 Activity，Task 结束。

事件	**Task 栈（粗体为栈顶组件）**
点开 Email 应用，进入收件箱（Activity A）	**A**
选中一封邮件，点击查看详情（Activity B）	**AB**
点击回复，开始写新邮件（Activity C）	**ABC**
写了几行字，点击选择联系人，进入选择联系人界面（Activity D）	**ABCD**
选择好了联系人，继续写邮件	**ABC**

写好邮件，发送完成，回到原始邮件	AB
点击返回，回到收件箱	A
退出 Email 程序	null

如上表所示，是一个实例。从用户从进入邮箱开始，到回复完成，退出应用整个过程的 Task 栈变化。这是一个标准的栈模式，对于大部分的状况，这样的 Task 模型，足以应付，但是，涉及到实际的性能、开销等问题，就会变得残酷许多。比如，启动一个浏览器，在 Android 中是一个比较沉重的过程，它需要做很多初始化的工作，并且会有不小的内存开销。但与此同时，用浏览器打开一些内容，又是一般应用都会有的一个需求。设想一下，如果同时有十个运行着的应用（就会对应着是多个 Task），都需要启动浏览器，这将是一个多么残酷的场面，十个 Task 栈都堆积着很雷同的浏览器 Activity，是多么华丽的一种浪费啊。于是你会有这样一种设想，浏览器 Activity，可不可以作为一个单独的 Task 而存在，不管是来自那个 Task 的请求，浏览器的 Task，都不会归并过去。这样，虽然浏览器 Activity 本身需要维系的状态更多了，但整体的开销将大大的减少，这种舍小家为大家的行为，还是很值得歌颂的
standard", "singleTop", "singleTask", "singleInstance"。

standard 模式， 是默认的也是标准的 Task 模式，在没有其他因素的影响下，使用此模式的 Activity，会构造一个 Activity 的实例，加入到调用者的 Task 栈中去，对于使用频度一般开销一般什么都一般的 Activity 而言，standard 模式无疑是最合适的，因为它逻辑简单条理清晰，所以是默认的选择。

而 singleTop 模式，基本上于 standard 一致，仅在请求的 Activity 正好位于栈顶时，有所区别。此时，配置成 singleTop 的 Activity，不再会构造新的实例加入到 Task 栈中，而是将新来的 Intent 发送到栈顶 Activity 中，栈顶的 Activity 可以通过重载 onNewIntent 来处理新的 Intent（当然，也可以无视...）。这个模式，降低了位于栈顶时的一些重复开销，更避免了一些奇异的行为（想象一下，如果在栈顶连续几个都是同样的 Activity，再一级级退出的时候，这是怎么样的用户体验...），很适合一些会有更新的列表 Activity 展示。一个活生生的实例是，在 Android 默认提供的应用中，浏览器（Browser）的书签 Activity（BrowserBookmarkPage），就用的是 singleTop。

singleTask，和 singleInstance，则都采取的另辟 Task 的蹊径。标志为 singleTask 的 Activity，最多仅有一个实例存在，并且，位于以它为根的 Task 中。所有对该 Activity 的请求，都会跳到该 Activity 的 Task 中展开进行。singleTask，很象概念中的单件模式，所有的修改都是基于一个实例，这通常用在构造成本很大，但切换成本较小的 Activity 中。最典型的例子，还是浏览器应用的主 Activity（名为 Browser...），它是展示当前 tab，当前页面内容的窗口。它的构造成本大，但页面的切换还是较快的，于 singleTask 相配，还是挺天作之合的。

singleInstance 显得更为极端一些。在大部分时候 singleInstance 与 singleTask 完全一致，唯一的不同在于，singleInstance 的 Activity，是它所在栈中仅有的一个 Activity，如果涉及到的其他 Activity，都移交到其他 Task 中进行。这使得 singleInstance 的 Activity，像一座孤岛，彻底的黑盒，它不关注请求来自何方，也不计较后续由谁执行。在

Android 默认的各个应用中，很少有这样的 Activity，在我个人的工程实践中，曾尝试在有道词典的快速取词 Activity 中采用过，是因为我觉得快速取词入口足够方便（从 notification 中点选进入），并且会在各个场合使用，应该做得完全独立。

43. 在 Android 中，怎么节省内存的使用，怎么主动回收内存？
回收已经使用的资源,合理设置变量的作用范围…
System.gc()

44. 不同工程中的方法是否可以相互调用？
可以,列举 aidl 访问远程服务的例子.

45. 在 Android 中是如何实现判断区分通话记录中的电话状态，去电，来电、未接来电？
Day8 showAddressService.java

46. dvm 的进程和 Linux 的进程, 应用程序的进程是否为同一个概念
Dvm 是 dalivk 虚拟机,每个 android 程序都运行在自己的进程里面,每个 android 程序系统都会给他分配一个单独的 uid, 每个 dvm 都是 linux 里面的一个进程.所以说这两个进程是一个进程.

47. sim 卡的 EF 文件有何作用
Sim 卡相关的东西,没接触过

48. 如何判断是否有 SD 卡？
配置文件中有 sd 卡的权限, 通过 environment 的静态方法,
if (Environment.getExternalStorageState().equals(Environment.MEDIA_MOUNTED)) {

49. 嵌入式操作系统内存管理有哪几种， 各有何特性。
页式，段式，段页，等…

50. 什么是嵌入式实时操作系统, Android 操作系统属于实时操作系统吗?
实时操作系统是指当外界事件或数据产生时，能够接受并以足够快的速度予以处理，其处理的结果又能在规定的时间之内来控制生产过程或对处理系统作出快速响应,并控制所有实时任务协调一致运行的嵌入式操作系统。主要用于工业控制、军事设备、航空航天等领域对系统的响应时间有 苛刻的要求，这就需要使用实时系统。又可分为软实时和硬实时两种，而 android 是基于 linux 内核的，因此属于软实时。

51. 一条最长的短信息约占多少 byte?
中文 70(包括标点)，英文 160，160 个字节 这个说法不准确,要跟手机制式运营商等信息有关.
做实验,看源码

```
ArrayList<String> msgs = sms.divideMessage(message);
        for (String msg : msgs) {
           sms.sendTextMessage(phoneNumber, null, msg, pi, null);
   }
```

52. Linux 中跨进程通信的几种方式 。

管道（pipe）：管道是一种半双工的通信方式，数据只能单向流动，而且只能在具有亲缘关系的进程间使用。进程的亲缘关系通常是指父子进程关系。

有名管道 (named pipe) ： 有名管道也是半双工的通信方式，但是它允许无亲缘关系进程间的通信。

信号量（ semophore ） ： 信号量是一个计数器，可以用来控制多个进程对共享资源的访问。它常作为一种锁机制，防止某进程正在访问共享资源时，其他进程也访问该资源。因此，主要作为进程间以及同一进程内不同线程之间的同步手段。

消息队列（ message queue ） ： 消息队列是由消息的链表，存放在内核中并由消息队列标识符标识。消息队列克服了信号传递信息少、管道只能承载无格式字节流以及缓冲区大小受限等缺点。

信号 （sinal） ： 信号是一种比较复杂的通信方式，用于通知接收进程某个事件已经发生。

共享内存（ shared memory ）：共享内存就是映射一段能被其他进程所访问的内存，这段共享内存由一个进程创建，但多个进程都可以访问。共享内存是最快的 IPC 方式，它是针对其他进程间通信方式运行效率低而专门设计的。它往往与其他通信机制，如信号两，配合使用，来实现进程间的同步和通信。

套接字（socket） ： 套解口也是一种进程间通信机制，与其他通信机制不同的是，它可用于不同及其间的进程通信。

53. 谈谈对 Android NDK 的理解。

1.什么时候用 ndk,为什么用 ndk,ndk 的优点,缺点,我们项目中那些地方用到了 ndk

54. 谈谈 Android 的优点和不足之处。

1、开放性,开源

2、挣脱运营商束缚

3、丰富的硬件选择

4、不受任何限制的开发商

5、无缝结合的 Google 应用

缺点也有 5 处：

1、安全问题、隐私问题

2、卖手机的不是最大运营商

3、运营商对 Android 手机仍然有影响

4、山寨化严重

5、过分依赖开发商，缺乏标准配置

55. Android 系统中 GC 什么情况下会出现内存泄露呢？

导致内存泄漏主要的原因是，先前申请了内存空间而忘记了释放。如果程序中存在对无用对象的引用，那么这些对象就会驻留内存，消耗内存，因为无法让垃圾回收器 GC 验证这些对象是否不再需要。如果存在对象的引用，这个对象就被定义为"有效的活动"，同时不会被释放。要确定对象所占内存将被回收，我们就要务必确认该对象不再会被使用。典型的做法就是把对象数据成员设为 null 或者从集合中移除该对象。但当局部变量不需要时，不需明显的设为 null，因为一个方法执行完毕时，这些引用会自动被清理。

Java 带垃圾回收的机制,为什么还会内存泄露呢?

```
Vector v = new Vector(10);
for (int i = 1; i < 100; i++) {
  Object o = new Object();
  v.add(o);
  o = null;
} // 此时，所有的 Object 对象都没有被释放，因为变量 v 引用这些对象。
```

Java 内存泄露的根本原因就是 保存了不可能再被访问的变量类型的引用

56. Android UI 中的 View 如何刷新。

在需要刷新的地方使用 handle.sendmessage 发送信息，然后在 handle 的 handlemessage 方法里面调用 invaliate()和 postinvaliate()方法刷新 view。

57. 简单描述下 Android 数字签名。

在 Android 系统中，所有安装到系统的应用程序都必有一个数字证书，此数字证书用于标识应用程序的作者和在应用程序之间建立信任关系

Android 系统要求每一个安装进系统的应用程序都是经过数字证书签名的，数字证书的私钥则保存在程序开发者的手中。Android 将数字证书用来标识应用程序的作者和在应用程序之间建立信任关系，不是用来决定最终用户可以安装哪些应用程序。这个数字证书并不需要权威的数字证书签名机构认证，它只是用来让应用程序包自我认证的。

同一个开发者的多个程序尽可能使用同一个数字证书，这可以带来以下好处。

(1)有利于程序升级，当新版程序和旧版程序的数字证书相同时，Android 系统才会认为这两个程序是同一个程序的不同版本。如果新版程序和旧版程序的数字证书不相同，则 Android 系统认为他们是不同的程序，并产生冲突，会要求新程序更改包名。

(2)有利于程序的模块化设计和开发。Android 系统允许拥有同一个数字签名的程序运行在一个进程中，Android 程序会将他们视为同一个程序。所以开发者可以将自己的程序分模块开发，而用户只需要在需要的时候下载适当的模块。

在签名时，需要考虑数字证书的有效期：

(1)数字证书的有效期要包含程序的预计生命周期，一旦数字证书失效，持有改数字证书的程序将不能正常升级。

(2)如果多个程序使用同一个数字证书，则该数字证书的有效期要包含所有程序的预计生命周期。

(3)Android Market 强制要求所有应用程序数字证书的有效期要持续到 2033 年 10 月 22 日以后。

Android 数字证书包含以下几个要点：

(1)所有的应用程序都必须有数字证书，Android 系统不会安装一个没有数字证书的应用程序

(2)Android 程序包使用的数字证书可以是自签名的，不需要一个权威的数字证书机构签名认证

(3)如果要正式发布一个 Android ，必须使用一个合适的私钥生成的数字证书来给程序签名，而不能使用 adt 插件或者 ant 工具生成的调试证书来发布。

(4)数字证书都是有有效期的，Android 只是在应用程序安装的时候才会检查证书的有效期。如果程序已经安装在系统中，即使证书过期也不会影响程序的正常功能。

58. 什么是 ANR 如何避免它？

在 Android 上，如果你的应用程序有一段时间响应不够灵敏，系统会向用户显示一个对话框，这个对话框称作应用程序无响应（ANR：Application Not Responding）对话框。

在 5 秒内没有响应输入的事件（例如，按键按下，屏幕触摸）和 BroadcastReceiver 在 10 秒内没有执行完毕这 2 种情况会导致出现 ANR，

避免的方法：运行在主线程里的任何方法都尽可能少做事情。一些耗时的操作，如网络或数据库操作，放在子线程里面执行。

59. android 中的动画有哪几类，它们的特点和区别是什么？

两种，一种是 Tween 动画、还有一种是 Frame 动画。

Tween 动画，这种实现方式可以使视图组件移动、放大、缩小以及产生透明度的变化；

1) 可以通过布局文件,可以通过代码控制 View 的动画

a) alpha(AlphaAnimation)

渐变透明

b) scale(ScaleAnimation)

渐变尺寸伸缩

c) translate(TranslateAnimation)

画面转换、位置移动

d) rotate(RotateAnimation)

画面转移，旋转动画

2) 控制一个 Layout 里面子 View 的动画效果

a) layoutAnimation(LayoutAnimationController)

b) gridAnimation(GridLayoutAnimationController)

另一种 Frame 动画，传统的动画方法，通过顺序的播放排列好的图片来实现，类似电影。

62. 说说 mvc 模式的原理，它在 android 中的运用。

MVC 英文即 Model-View-Controller，即把一个应用的输入、处理、输出流程按照 Model、View、Controller 的方式进行分离，这样一个应用被分成三个层——模型层、视图层、控制层。

Android 中界面部分也采用了当前比较流行的 MVC 框架，在 Android 中 M 就是应用程序中二进制的数据，V 就是用户的界面。Android 的界面直接采用 XML 文件保存的，界面开发变的很方便。在 Android 中 C 也是很简单的，一个 Activity 可以有多个界面，只需要将视图的 ID 传递到 setContentView()，就指定了以哪个视图模型显示数据。

在 Android SDK 中的数据绑定，也都是采用了与 MVC 框架类似的方法来显示数据。在控制层上将数据按照视图模型的要求（也就是 Android SDK 中的 Adapter）封装就可以直接在视图模型上显示了，从而实现了数据绑定。比如显示 Cursor 中所有数据的 ListActivity，其视图层就是一个ListView,将数据封装为ListAdapter,并传递给ListView,数据就在ListView中现实。

63. 通过点击一个网页上的 url 就可以完成程序的自动安装,描述下原理

Day11 AddJavascriptInterface

64. Service 和 Activity 在同一个线程吗

默认情况同一线程 main 主线程 ui 线程

65. java 中的 soft reference 是个什么东西
StrongReference 是 Java 的默认引用实现，它会尽可能长时间的存活于 JVM 内，当没有任何对象指向它时 GC 执行后将会被回收
SoftReference 会尽可能长的保留引用直到 JVM 内存不足时才会被回收(虚拟机保证)，这一特性使得 SoftReference 非常适合缓存
应用详细见豆瓣客户端图片的缓存

66. udp 连接和 TCP 的不同之处

67. android 开发中怎么去调试 bug
逻辑错误
1.断点 debug
2. logcat ，
界面布局,显示 hierarchyviewer.bat

68. service 里面可以弹土司么
可以

69. 写 10 个简单的 linux 命令

70. JNI 调用常用的两个参数

71. 书写出 android 工程的目录结构

72. ddms 和 traceview 的区别

总结:
关于项目
在就是你项目经验，一定要突出你遇到什么难点，然后是怎么解决的！把问题引导到你熟悉的领域,或者知识点上,尽量将每个技术点细节凸显出来。

心态:
什么样的面试官都有，去面试的时候要做好一切心理准备，不管是技术还是基础都得扎实。一个人的交谈能力也很重要，总之不是非常标准的普通话， 最起码你说的得让别人听得懂，而且得把面试官讲得非常彻底，这样你获得 offer 的机会更大，谈工资也有优势~~