Android工程师上岗考核试题

姓名:刘东远 岗位:Android工程师 工号:32601 得分：

1. 简述Activity的生命周期? (10分)

Activity从启动到结束会经历以下周期

onCreate()->onStart()->onResume()->onPause()->onStop()->onDestory()

2.如何进行屏幕适配的? (10分)

[今日头条屏幕适配方案 AndroidAutoSize](https://links.jianshu.com/go?to=https://juejin.im/post/5bce688e6fb9a05cf715d1c2" \t "https://www.jianshu.com/p/_blank)[——通过反射修正系统的 density值](https://links.jianshu.com/go?to=https://juejin.im/post/5b7a29736fb9a019d53e7ee2" \t "https://www.jianshu.com/p/_blank)

· dpi：屏幕像素密度，指的是在**系统软件上**指定的单位尺寸的像素数量，它往往是写在系统出厂配置文件的一个固定值；

· ppi：也是屏幕像素密度，但这个**是物理上**的概念，它是客观存在的不会改变。dpi是软件参考了物理像素密度后，人为指定的一个值，这样保证了**某一个区间内的物理像素密度**在软件上都使用同一个值；

· dp加上自适应布局和weight比例布局能解决**90%的适配问题**。因为并不是所有的1080P的手机dpi都是480，比如Google 的Pixel2（1920\*1080）的dpi是420；

· 宽高限定符适配：穷举市面上所有的Android手机的宽高像素值，设定一个基准的分辨率，其他分辨率都根据这个基准分辨率来计算，在不同的尺寸文件夹内部，根据该尺寸编写对应的dimens文件。

1. 简述多线程的用法 (10分)

涉及到多线程一般是使用线程池

新建一个线程与线程池的特点

1. 节省开销： 线程池中的线程可以重复利用。
2. 速度快：任务来了就能开始，省去创建线程的时间。
3. 线程可控：线程数量可空和任务可控。
4. 功能强大：可以定时和重复执行任务。

线程池的使用方法:

1.CachedThreadPool：闲置线程超时会释放，没有闲置线程的情况下，每次都会创建新的线程。

2.FixedThreadPool：线程池只能存放指定数量的线程池，线程不会释放，可重复利用。

3.SingleThreadExecutor：单线程的线程池。

4.ScheduledThreadPool：可定时和重复执行的线程池

1. 自定义控件的流程? (10分)
2. 自定义控件的属性，在res/values/ attrs.xml下生命属性
3. 在View的构造方法中获得我们自定义的属性
4. 重写onMesure
5. 如果是ViewGroup的话，需要重写onLayout
6. 重写onDraw
7. Service的启动方式和区别? (10分)

启动方式：

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，service 就会调用 onUnbind-> onDestroyd方法

StartService()

service会经历 onCreate到 onStart，然后处于运行状态，stopService 的时候调用 onDestroyd方法。

区别:

StartService:

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

bindService:

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 Service 就会调用 onUnbind -> onDestroyd方法。

1. 什么情况下会导致内存泄漏（10分）

动画、上下文对象、EventBus、AsycTask、Handler、单例Bitmap、mvp架构释放它们未释放的资源

1. 简述一下handler机制？(20分)

消息机制指Handler、Looper、MessageQueue、Message之间如何工作的。

1.handler是用来处理消息和接收消息的中间者，handler的创建会伴随着handler中产生looper和MessageQueue，handler依赖于looper，looper依赖于MessageQueue，所以在子线程中使用handler抛出异常是因为子线程中没有初始化looper对象，而主线程中looper是在ActivityThread中已经初始化过了，所以能直接在主线程中能拿到Handler。

2.Looper是用来轮询消息，说白了就是通过loop方法实现死循环，有消息的时候，通过MessageQueue.next方法取出message，没有消息的时候，线程处于阻塞的状态。在有消息的时候获取到消息，将消息交给了handler，handler会根据消息中有没有callback，如果有callback会直接callback，否则通过handleMessage处理。

3.MessaeQueue是一个单链表结构来存储Message，每次通过next方法取出Message消息后，取完之后将message.next给当前的message，再将message.next=null，实际上就是移除当前的message。但是在looper里面每次在next取出message后，放到了message的sPool里面，缓存起来方便使用。

4.Message就没什么好说的，主要存储平常经常用的obj和what信息，以及我们不用关心的target和callback等。

1. 如何进行android性能优化？(20分)
2. 布局优化，可以多使用include、merge、ViewStub、约束布局等，merge可以减少布局层次;ViewStub是使用的时候才去创建view;减少空间的占用;约束布局一来可以减少布局层次，二来可以提高开发效率。
3. 代码优化，static，Stringbuilder的使用，过多的创建对象，bitmap的释放等，减少内存资源的占用。
4. 内存的优化，释放掉该释放的资源，没有及时让GC处理器回收，通常主要表现是动画、上下文对象、EventBus、AsycTask、Handler、单例Bitmap都会影响，通常要做的是释放他们未终止的动作，释放锁定的上下文对象。