# 第一部分 Java

## Java基础

1. 面向过程、面向对象和面向切面编程的区别和特点。
2. 面向对象的三大特性。
3. 什么是JDK，什么是JRE，什么是JVM。
4. 字符串字面量和new String()有什么区别。
5. String、StringBuffer和StringBuilder的区别。
6. 在静态方法中能不能调用非静态的成员变量（能不能覆盖一个static方法）。
7. 抽象类和接口的区别。
8. equals方法的作用是什么，它和==有什么区别。
9. hashCode方法的作用是什么，它和equals方法有什么联系。
10. 可以直接根据hashcode判断两个对象相等吗。
11. final关键字的作用。
12. 什么是序列化，序列化前后对象有何区别。
13. 序列化Serializable和Parcelable的区别。
14. try catch finally，try里有return，finally还执行么。
15. Exception与Error类结构，Exception与Error的区别(Java异常分类)。
16. 使用for循环、foreach循环和迭代器遍历集合时能否进行删除操作，为什么？
17. 泛型的优缺点。
18. 解析XML的几种方式的原理与特点：DOM、SAX、PULL。
19. Java1.7、Java1.8和Java1.9的新特性。

## Java集合框架

1. Java集合框架的结构。
2. ArrayList实现原理。
3. LinkedList实现原理。
4. HashMap实现原理。
5. LinkedHashMap实现原理。
6. ConcurrentHashMap的原理。
7. Vector实现原理。
8. HashMap实现原理。
9. HashTable实现原理。
10. TreeMap实现原理。
11. TreeSet实现原理。
12. HashMap的原理，ConcurrentHashMap的原理。
13. SparseArray、ArrayMap和HashMap区别。
14. hashmap如何put数据（从hashmap源码角度讲解）。

## IO/NIO

## Java并发（Concurrent）

1. 什么是线程，什么是进程，能不能只用进程。
2. 多线程编程的好处是什么。
3. 一个线程包含了哪些状态。
4. wait()和sleep()的区别。
5. 有三个线程T1、T2、T3，怎么保证它们顺序执行。
6. 如何停止一个线程。
7. 一个线程发生异常会怎么样。
8. 什么是守护线程。
9. ThreadLocal的设计理念与作用，ThreadPool用法与优势（AsyncTask底层有使用）。
10. 什么是线程安全，Vector是线程安全的吗。
11. 如何保证线程安全（如何保证多线程读写文件的安全）。
12. 说一说自己对于synchronized的了解。
13. 说一说自己是怎么使用synhronized关键字的，在项目中用到了吗。
14. 讲一讲synchronized关键字的底层实现原理。
15. 讲讲JDK1.6之后的synchronized关键字底层做了哪些优化，可以详细介绍一下这些优化吗？
16. 谈谈synchronized和ReenTrantLock的区别。
17. 什么是线程死锁，线程死锁的四个条件。
18. 锁的等级：方法锁、对象锁、类锁。
19. Java锁的分类。
20. Java中volatile变量是什么。
21. 为什么要使用线程池。
22. Java默认提供几种线程池。
23. 什么是阻塞队列。
24. 如果提交任务时，线程池队列已满，会发生什么。
25. 生产者消费者模式的集中实现，阻塞队列，sync关键字，lock实现，ReentrantLock实现，写出生产者消费者模式简单代码实现。
26. 线程池的底层实现与工作原理。
27. 阻塞队列，ArrayBlockingQueue、CountDownLatch等。
28. 多线程场景遇见过没有。
29. Integer类是不是线程安全的，为什么。
30. 不使用同步锁如何保证线程安全。

## JVM

1. Java内存模型、堆区和栈区的区别，OOM、SOF。
2. JVM怎么判断那些对象是需要进行回收的（JVM怎么判断对象已死）。
3. Java的四种引用，强弱软虚，及其使用的场景。
4. 强引用置为null，会不会被回收。
5. Java类加载机制，类加载分为那几个阶段。
6. 垃圾回收机制和算法。
7. 动态代理的原理，代理对象在那个阶段生成的。
8. Java值传递。
9. 垃圾回收原理以及回收策略。
10. 类的加载机制，为什么用双亲委派模型，Java是否支持多态。

## 设计模式

1. 手动画出设计模式的UML类图。
2. 工具类使用单例和静态内部类有什么区别。
3. 什么是IOC，他和DI有什么区别，为什么使用IOC而不使用工厂模式。
4. 动态代理模式的原理。
5. 适配器模式，装饰者模式，外观模式的异同。

## 七、其他

# 第二部分 Android

## Android基础

1. Serializable和Parcalable的区别。
2. dpi、ppi、px、pt、dp、sp的区别。
3. 应用最多占多少内存。
4. 了解sp的apply和commit的区别吗，sp具体是怎么实现的，sp什么时候会将内存中的数据写入到文件中。
5. Android 5.0、6.0、7.0、8.0的新特性。
6. Android持久化方案。
7. 如何判断应用被强杀，应该如何解决。
8. 怎么退出终止App。
9. apk的目录结构。
10. AysncTask各个函数分别在什么线程回调的，以及底部实现原理。
11. Fragment之间传递数据的方式。
12. 广播分类。

## 四大组件相关

1. 手动画出Activity、Fragment的生命周期，他们是怎么关联的。
2. Activity常见情形下的生命周期，如按下home键、锁屏。
3. 异常情况下Activity的生命周期，如横竖屏切换、系统资源不足。
4. 什么是Activity任务栈。
5. Activity启动模式以及使用的场景，对应的Intent Flag。
6. IntentFilter匹配规则。
7. Service生命周期，与IntentService的区别。
8. startService和bindService混合使用的场景，这时候Service的生命周期是怎样的。
9. 广播的两种注册方式有什么区别。
10. 怎么增加广播的安全性。
11. 什么是Fragment回退栈。
12. Fragment commit和commitAllowingStateLossing的区别。
13. Fragment之间传递数据的方式。
14. FragmentPager、FragmentStatePagerAdapter区别。
15. 什么是Context，一个应用中有几个Context。
16. 能不能通过Application来启动Activity。
17. AppCompatActivity、FragmentActivity和Activity区别。

## View剖析

1. Touch事件的分发流程。
2. 点击事件被拦截，但是想传到下面的view，如何操作。
3. View的位置参数有哪些，left、x、translationX的含义以及三者的关系。
4. 什么是MeasureSpec。
5. View的绘制流程，onMeasure、onLayout的计算流程，MeasureSpec的几种模式。
6. 怎么获取View的宽高。
7. 自定义View的流程，自定义View需要注意的问题，例如自定义View是否需要重写onLayout，onMeasure。
8. Invalidate、postInvalidate、requestLayout的区别。
9. 动画的原理，底层如何给上层信号。
10. 插值器和估值器。
11. RecyclerView和ListView复用机制。
12. 处理滑动的几种方式，Scroller滑动的原理。
13. 简述工作线程更新UI的方法。
14. Android多点触控。
15. ViewPager、RecyclerView和SrcollView嵌套滑动冲突，显示不完整，滑动卡顿问题。
16. Native和js交互流程。
17. 滑动卡顿如何解决（不同原因及对应处理方式）。

## Framewrok相关

### Binder

1. Android进程的优先级。
2. Binder是什么？它是如何实现跨进程通信的？（详解Binder机制。）
3. 进程间的通信方式。

### 资源管理框架

1. Apk文件解析。
2. Android程序资源分类。
3. Android资源加载和打包的机制介绍。一个图片在app中调用R.id.调用后是如何找到的？
4. Android有多个资源文件夹，应用在不同分辨率下是如何查找对应文件夹下的资源的，描述整个过程。
5. Android系统的启动过程。
6. Launcher的启动过程。
7. 简述Activity的启动过程。
8. Service的启动过程。
9. ActivityThread相关。

### Android系统整理流程

1. Apk的安装流程。
2. Launcher的启动过程。
3. Android应用进程的启动过程。
4. Android应用程序的启动过程。

### other

1. Android消息循环机制。
2. 关于Handler，在任何地方new Handler都是在什么线程下。
3. HandlerThread的使用场景和原理。
4. Android PendingIntent内部机制。
5. Android ART机制，与dalvik的区别，JIT与AOT的区别是什么？ART GC有什么改善，还有什么缺点。
6. 一个apk从开始安装启动，系统都做了哪些事情？请从AMS，WMS，PMS的角度考虑下，以及进程是如何启动的？
7. AMS和WMS相关的数据结构和沟通的桥梁是什么？AMS的堆栈是如何管理的？WMS的堆栈是如何管理的？
8. Android内存管理机制以及LMK相关的机制，以及AMS中进程管理机制，请分别介绍下。
9. PMS相关的开机流程，apk安装的流程，adb install和pms scan的区别有哪些？
10. Broadcast的机制，分发的流程是什么？如何传递到每个app进程的？动态广播和静态广播的处理流程在哪里不一样？
11. 多用户最主要的机制以及创建一个新的用户系统需要做哪些事情。
12. Runtime permission，如何把一个预置的app默认给它权限，不要授权。
13. 如何实现预装的apk在卸载后，通过恢复出厂设置恢复过来，请介绍下方案。
14. Android Overlay的机制是什么？
15. Android权限管理的机制是什么。
16. 为何android.uid.system相关的进程不能访问sdcard。
17. 开机流程和关机流程请描述下。
18. Bootanimation是如何启动和退出的。
19. Binder相关的机制以及在Android平台的使用，Android还有什么IPC通信方式，各有什么优缺点。
20. 死机，重启等stability问题分析流程？watchdog reset如何分析？
21. Native Crash问题如何分析，以及crash在art相关的oat，odex文件如何分析。
22. ANR，OOM等问题的分析流程介绍。
23. Android++智能指针相关的使用介绍。
24. Android编译，优化，ART相关编译优化。
25. Input相关事件的分发机制，tp相关问题解决。
26. 按键事件和tp事件的处理有什么不同点和相同点。
27. 功耗相关问题的分析。
28. 性能相关问题的分析。
29. Android N与M的一些典型的改变有哪些？Multi-window机制介绍。
30. PowerManagerService主要做了哪些相关的操作？系统亮灭屏都有哪些流程。
31. Wakelock机制，Android如何和linux管理这些wakelock。
32. Alarm相关机制，doze相关的机制以及运行方式。
33. 性能优化如何分析systrace。
34. 简述IPC。

## 性能优化

### 内存优化

1. Java的四种引用，强弱软虚，及其适用的场景。
2. 内存泄漏如何产生，常见的Android内存泄漏的场景，怎么解决。
3. 怎么发现和分析内存泄漏。
4. OOM能不能用try catch捕获。
5. OOM遇到过哪些情况，如何解决的。
6. Bitmap使用的时候注意什么（Bitmap优化）。
7. Bitmap recycler相关（Bitmap复用）。
8. WebView的泄露如何解决。
9. 如何优化WebView。
10. LeakCanary原理，内存泄露的分析流程。

### 流程优化

1. 如何解决卡顿问题。
2. Android UI卡顿怎么进行监测。
3. 页面卡顿的原因，页面加载卡顿问题优化。
4. ANR原理（回答主线程阻塞5秒不算，要分析源码）。
5. Android启动冷白屏优化。
6. 如何处理App启动流程优化。
7. View的优化流程。

### View优化

1. 怎么解决RecyclerView刷新闪烁问题。
2. ListView的卡顿优化、资讯流懒加载如何实现。
3. ListView如何优化，ListView和RecyclerView的区别，二者的缓存逻辑。

### 其它

1. 常用的性能优化工具总结。
2. 如何进行屏幕适配。
3. 怎么减小APK大小。
4. 性能优化如何分析systrace。
5. 提高app安全性的方法。
6. 窗口泄露原因，及解决办法。
7. 如何对Android应用进行性能分析。

## 热门技术

1. 插件化开发流程，插件化优势，插件化开发中遇到的问题以及如何解决的。
2. 插件化优势，插件化加载Activity原理，加载资源原理。
3. 插件化怎么加载资源和绑定Activity生命周期。
4. 插件化技术原理及常用的插件化框架。
5. 热修复技术原理及常用的热修复框架。
6. 在线更新。
7. 增量更新。
8. 多渠道打包。
9. 插件化开发流程，插件化优势，插件化开发中遇到的问题以及如何解决的。
10. 插件化Activity生命周期的管理，资源的访问原理。
11. 组件化开发流程，ARouter路由协议的原理分析。
12. 插件化首次启动耗时长，如何优化。
13. 什么是gradle task。
14. 插件化优势，插件化加载Activity原理，加载资源原理。

## 常用框架

1. Glide传入不同Context的区别，如何绑定生命周期的。
2. Glide使用什么缓存，如何设置缓存大小。
3. Glide优势是什么。
4. OkHttp采用的是哪种线程池，这样设计的好处是什么。
5. OkHttp缓存策略有什么缺点。
6. 什么是响应式编程，它有什么优点。
7. RxJava切换线程的原理。
8. RxJava的线程池与自己实现的任务管理框架有什么区别。
9. 为什么使用Retrofit，Retrofit与其它网络库有什么优势。

## 架构设计

1. MCV、MVP、MVVM比较。
2. MVP模式的优缺点，怎么改进。
3. 怎么实现MVP Presenter的复用，怎么实现在不固定位置进行网络请求。
4. MVC到MVP的架构重构流程。

## 经典案例实现

1. RecyclerView图片加载错位和闪烁问题。
2. 加载1000张图片，怎么进行优化。
3. 怎么管理线程任务，暂停和提前中断任务。
4. 怎么加载大图，怎么进行优化。
5. 实现一个库，完成日志的实时上报和延时上报两种功能，该从哪些方面考虑。
6. 如果想要频繁的读写一个大文件，提供实现方案。
7. 给你一个list数量长达1000，如何实现高效的插入数据库，使用事务，事务的效率为什么高，多线程并发插入。
8. 如何实现一个监控ANR的库，有哪几种方案。
9. fragment实现懒加载，与ViewPager搭配时。

## 其他

1. 进程保活的实现方案和优缺点。
2. 检测前后台切换。
3. Android长连接心跳机制。
4. 自定义gradle插件，gradle加载的几个步骤，gradle语法。
5. json、xml数据格式有什么优势和劣势。
6. JNI用过么。
7. sqlite升级，增加字段的语句。
8. Sqlite 怎么增加一个字段。
9. Sqlite 怎么删除一个字段。
10. 多进程场景遇见过么。

# 第三部分 计算机网络

1. TCP三次握手过程，为什么需要三次握手。
2. TCP四次挥手过程，为什么需要四次挥手。
3. TCP四次挥手为什么需要等待2MSL。
4. 什么是DNS。
5. Http 的session和cookie的区别。
6. Http 和 Https的区别。
7. Https工作原理。
8. HTTP协议请求响应过程和HTTPS工作原理。
9. TCP、UDP区别，拥塞控制原理，视频传输为什么用UDP，UDP丢包会产生什么问题，如何处理数据包的发送速度和带宽的关系。
10. TCP、IP各自所在的网络层次，IP报文中的内容。
11. Https中的对称加密和非对称加密。

# 第四部分 操作系统

1. 对象和类之间的相互关系。
2. 单例模式。

# 第五部分 数据结构与算法分析