答案：

1. Android为什么不允许工作线程更新UI？

答：

因为Android是单线程模型，那么为什么Android是单线程模型？

因为多线程并发访问UI可能会导致UI控件处于不可预期的状态。如果加锁，虽然能解决，但是缺点也很明显：

1)锁机制让UI访问逻辑变得复杂；

2)加锁导致效率低下。

1. 如何在工作线程中更新UI？

答：

使用Handler机制传递消息到主线程（UI线程）

1. Looper.loop()是无限循环，是否会导致线程阻塞？

答：主线程中通过Looper遍历消息队列，有消息则处理，没有消息则休眠。

1. 以下三种方式执行的优先顺序是？

1)post方法发送Runnable

2)创建Handler子类中并覆写handleMessage方法

3)创建Handler对象时，在构造方法中覆写Callback接口的handleMessage方法

答：通过查看Handler.dispatchMessage方法的源码，发现以上三种方式的优先顺序是：

1、3、2．

1. 一个线程只能绑定一个Handler。Handler对象也只能对应一个线程，对吗？

答：

可以有多个Handler绑定到同一线程，但Handler不能同时绑定多个线程。因此线程与Handler是一对多的关系。

1. 一个Looper只能对应一个线程，如何证明？

答：通过查看Looper 的prepare(boolean quitAllowed)方法源码，可以发现每个线程的Looper对象都保存在sThreadLocal变量的Map集合中，只能有一份，多余一份则会抛出异常。

1. Android的线程结束时，其内部的Looper都可以通过Looper.quit结束消息循环吗？

答：主线程不能。

1. 简述Handler机制。

答：Handler机制就是线程间通信的机制。

1. 的
2. 简述主线程

答：主线程：也叫UI线程，或称ActivityThread，用于运行四大组件和处理他们用户的交互。 ActivityThread管理应用进程的主线程的执行(相当于普通Java程序的main入口函数)，在Android系统中，在默认情况下，一个应用程序内的各个组件(如Activity、BroadcastReceiver、Service)都会在同一个进程(Process)里执行，且由此进程的主线程负责执行。

ActivityThread既要处理Activity组件的UI事件，又要处理Service后台服务工作，通常会忙不过来。为了解决此问题，主线程可以创建多个子线程来处理后台服务工作，而本身专心处理UI画面的事件。

1. 简述Android主线程。

答：

主线程：也叫UI线程，或称ActivityThread，用于运行四大组件和处理他们用户的交互。 ActivityThread管理应用进程的主线程的执行(相当于普通Java程序的main入口函数)，在Android系统中，在默认情况下，一个应用程序内的各个组件(如Activity、BroadcastReceiver、Service)都会在同一个进程(Process)里执行，且由此进程的主线程负责执行。

ActivityThread既要处理Activity组件的UI事件，又要处理Service后台服务工作，通常会忙不过来。为了解决此问题，主线程可以创建多个子线程来处理后台服务工作，而本身专心处理UI画面的事件。

1. 简述MessageQueue。

答：MessageQueue用于存放Message，用链表实现队列的先进先出的数据结构。

MessageQueue在Looper对象创建时被创建，每个线程只有一个消息队列。

MessageQueue用next方法遍历链表，取出首节点的Message。

MessageQueue用enqueueMessage方法遍历链表，将队列尾部插入消息。