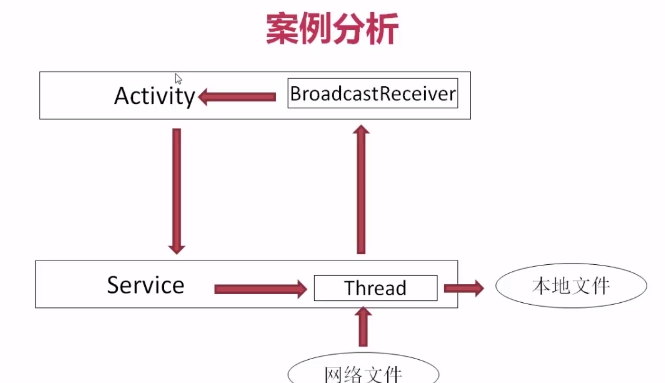
1. 基本UI定义
2. 数据库的操作
3. Service的启动
4. Activity给Service传递参数
5. 使用广播回传数据到Activity
6. 线程和Handler
7. 网络操作



网络下载的关键点

1. 获得网络文件的长度
2. 在本地创建一个文件，设置其长度
3. 从数据库中获得上次下载的进度（继续下载）
4. 从上次下载的设置下载数据，同时保持进度到数据库
5. 将下载进度回传Activity（进度显示）
6. 现在完成后删除下载信息

注：在Android4.0以后的版本中，主要涉及到网络操作，就必须使用多线程处理，不然会报错！！

1. 主界面（Activity）

设置下载url和名字，实例化文件信息类，传递给Service

二、实体类

1．文件信息类：id、url、fileName、length、finished。实现serializable接口，序列化。

2. 线程类：id、url、start、end、finished

三、Service（接受Activity传递的数据）

1.初始化子线程：获得下载文件（连接网络（url），获取文件长度，在本地创建文件）的长度，并返回。

从服务器下载文件，选择GET方式来获取数据。除了下载操作以外，还有POST。

RandomAcessFile：随机访问文件

2.Handler:消息回传，把文件长度返回

四、数据库帮助类

1.创建数据库：保存线程的url，开始位置，结束位置，完成进度。

2.删除表：当下载完成之后就删除

3.更新表：先删除，再创建

五、数据访问接口（用于数据库管理线程的增删查）

六、实现数据访问接口：连接数据库帮助类和数据访问

七、数据下载任务类

根据文件信息下载对应的数据。

1. 实现文件下载线程

（1）向数据库中插入线程信息

（2）设置该文件的下载位置：setRequestProperty（”Range”,”bytes”）

（3）设置文件写入的位置：RandomAccessFile，seek()方法：在读写的湿乎乎跳过设置好的字节数，从下一个字节数开始读写。

（4）开始下载🡪读取数据、写入文件、下载暂停（保存下载进度）、把下载进度发送到Activity(广播方式)。

（5）如果现在完成，删除数据库中的线程信息

八、添加BroadcastReceiver广播接收器更新UI。