**Service与BroadcastReceiver**

1. **Service**
   1. 服务（Service，android四大组件之一）是Android实现程序后台运行的解决方案，适合用于去执行那些不需要和用户交互而且还要求长期运行的任务。
   2. 服务的运行不依赖于任何用户界面，而是依赖于创建服务时所在的应用程序进程。
   3. 服务运行在主线程中，一般在服务内部创建子线程来执行具体任务，防止主线程被阻塞。
   4. Android中的异步消息处理主要由四个部分组成：

Message：用于在线程之间传递的消息。

Handler：用于发送和处理消息。

MessageQueue：消息队列，用于存放消息。每个线程中只会有一个MessageQueue对象。

Looper：每个线程中的MessageQueue的管家，每个线程中也只会有一个Looper对象。

* 1. Service和activity都是从context派生出来的，都需要在AndroidManifest.xml文件中进行注册。
  2. 服务中的四个方法：
  3. onCreate()方法会在服务创建的时候调用；
  4. onStartCommand()方法会在每次服务启动的时候调用；
  5. onBind()用bindService()来获取一个服务的持久连接
  6. onDestroy()方法会在服务销毁的时候调用。
  7. 服务启动的两种方法：
  8. 调用者跟service之间没有关联，调用stopService()后service仍然运行，两者之间无法通信、数据交换

startService()-> onCreate()-> onStartCommand()

stopService()或stopSelf()-> onDestroy()

* 1. 调用者跟service相关联，两者共存亡，调用unbindService()后service终止，两者之间可通信、交换数据

bindService()->onCreate()->onBind()

unbindService()-->onUnbind()-->onDestroy()

**注**：若activity调用bindService()绑定到一个**已启动**的Service时，并不会将该service的生命周期与activity绑定。

1. Service不会专门开启一个进程，与应用程序位于同一个进程；

Service也不会开启一个新线程，因此service处理耗时任务时，需要新建一个线程。

1. IntentService使用队列来管理请求Intent，其特征如下：
2. IntentService会创建单独的worker线程处理所有Intent请求和onHandleIntent()方法实现的代码，只需重写onHandleIntent()方法，无需开发者处理多线程问题；
3. 当所有请求处理完成后，IntentServic会自动停止，无需调用stopSelf()来停止该Service；
4. 为service的onBind()和onStartCommand()提供默认实现。
5. **BroadcastReceiver**
6. Android中的广播主要可以分为两种类型，标准广播（完全异步执行的广播）和有序广播（同步执行的广播）。
7. 注册广播的两种方式：
8. 代码中注册（动态注册）：可以自由地控制注册与注销。
9. AndroidManifest.xml中注册（静态注册）：实现开机启动。
10. BroadcastReceiver像一个全局的事件监听器，只是它用于监听系统发出的Broadcast，可实现不同应用程序之间的通信。
11. BroadcastReceiver的onRecerve( )方法不能再10秒内执行完成，Android就认为该程序无响应。所以不能再onReceive( )中执行耗时操作。由于BroadcastReceiver的生命周期很短，因此也不能在onRecerve( )中使用子线程，否则将出现子线程还没结束，BroadcastReceiver已经退出。对于耗时操作可以通过Intent启动一个service来完成该耗时任务。