**Android 异步任务 AsyncTask**

1. AsyncTask<Params, Progress, Result>抽象类
2. Params：启动任务执行的输入参数类型；
3. Progress：后台任务完成的进度值类型；
4. Result：后台执行任务完成后返回结果的类型。
5. AsyncTask<Params, Progress, Result>抽象类的方法
6. excute()：触发异步线程任务
7. onPreExecute()：后台执行耗时操作时被调用，用于完成一些初始化准备工作。
8. doInBackground(Params…)：后台线程将要完成的任务，可以调用publish Progress(Progress… values)方法更新任务的执行进度；
9. onProgressUpdate(Progress… values)：在doInBackground(Params…)方法中调用publishProgress(Progress… values)方法后触发该方法；
10. onPostExecute(Result result)：当doInBackground(Params…)完成后，调用postResult()方法，postResult()通过构造函数中创建的handler处理doInBackground(Params…)获取的结果（默认handler实现会调用AsyncTask#finish()，AsyncTask#finish()再根据取消状态调用onCancelled()或者onPostExecute()）。
11. 在UI线程创建AsyncTask实例，并在UI线程调用AsyncTask的execute(Params… params)方法。
12. AsyncTask执行流程：
13. 构造函数中定义了mHandler、WorkerRunnable和FutureTask，FutureTask保存了WorkerRunnable实例;
14. 调用excute方法，先执行onPreExecute()，然后通过线程池的excute执行FutureTask中的任务；
15. 线程池调用FutureTask#run()，FutureTask#run()调用WorkerRunnable#call()；
16. WorkerRunnable#call()先调用AsyncTask#doInBackground()得到执行的结果，然后调用AsyncTask#postResult();
17. AsyncTask#postResult()通过AsyncTask#mHandler发送消息到主线程；
18. mHandler#handlerMesssage中调用AsyncTask#finish()；
19. AsyncTask#finish()判断是否取消，若取消则执行AsyncTask#onCancelled()方法，否则执行AsyncTask#onPostExecute()
20. FutureTask：一个可取消的异步任务，解决通过继承Thread类或者实现Runnable接口创建多线程不能在执行完成后获取执行的结果的问题

参考：<https://blog.csdn.net/qq_39654841/article/details/90631795>

<https://www.cnblogs.com/linghu-java/p/8991824.html>