# 线程池

ThreadPoolExecutor构造函数参数解析

Int corePoolSize

Int maximumPoolSize

Long keepAliveTime

TimueUnit unit

BlockingQueue<Runnable> workQueue

ThreadFactory threadFactory

RejectedExecutionHandle handle

corePoolSize: 核心线程数，当工作队列不满时，同一时间最多可以同时运行的线程数量。

maximumPoolSize：最大可以运行的线程数量，当需要运行的线程数大于核心线程数并且工作队列已满时，会创建额外的工作线程，但是创建的线程数量必须小于maximumPoolSize。

keepAliveTime：当线程池中的线程数大于核心线程数时，超出核心线程数的线程在执行完毕之后，会根据此值来判断是否要退出。

workQueue：阻塞队列，当线程池中的线程数大于核心线程数时，需要将多余的线程添加到这个队列中去。

threadFactory: 创建线程时需要用到的线程工厂。

Handle：拒绝策略，如果队列已满并且线程池中的线程数量大于最大线程数时，会执行。

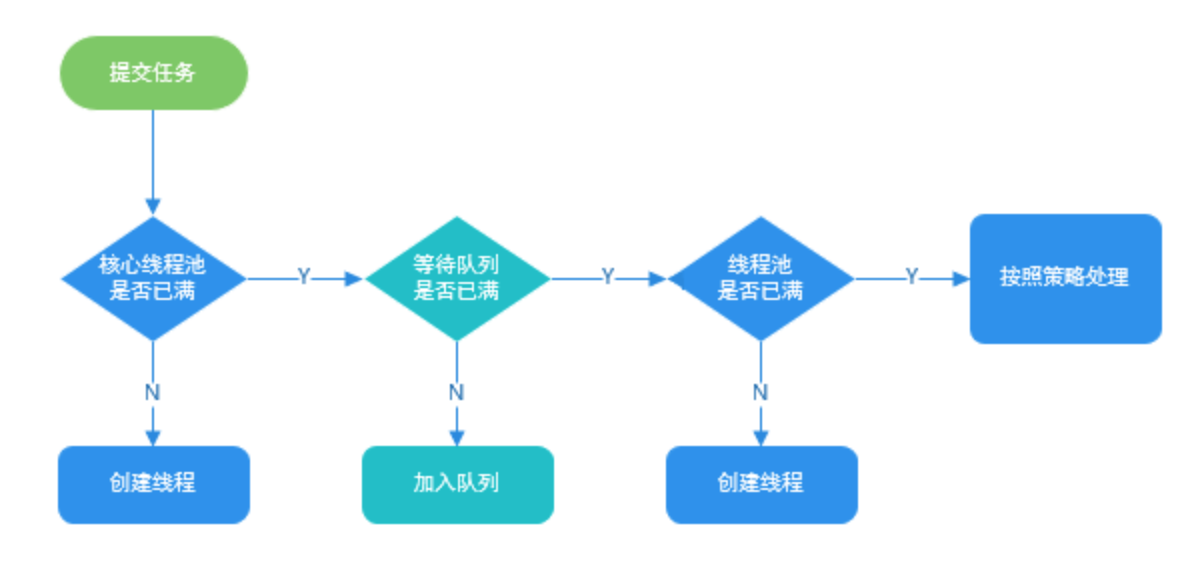
ThreadPoolExecutor.AbortPolicy，会直接报错异常拒绝新任务的处理。

ThreadPoolExecutor.callerRunsPolicy： 调用执行自己的线程运行任务

ThreadPoolExecutor.DiscardPolicy：不处理新任务，直接丢弃

ThreadPoolExecutor.DiscardOldestPolicy：此策略将丢弃最早的未处理的任务请求。

execute逻辑：



## Runnable与Callable

Runnable不会返回结果或抛出检查异常，但是callable接口可以返回。

## Execute()与submit()

1. execute() 方法用于提交不需要返回值的任务，所以无法判断任务是否被线程池执行成功与否。
2. Submit()方法用于提交需要返回值的任务。线程池会返回一个Future类型的对象，通过这个Future对象可以判断任务是否执行成功。

## Shutdown()与shutdownNow()

Shutdown: 关闭线程池，线程的状态为SHUTDOWN。线程池不再接受新的任务了，但是队列中的任务要执行完毕。

Shutdownnow：关闭线程池，线程状态为stop。线程池会终止当前正在运行的任务，并停止处理排队的任务并返回正在等待执行的list。