## 多线程介绍、列举线程池和业务场景：

多线程基本介绍：http://www.cnblogs.com/yxt9322yxt/p/4804026.html

1）.什么是多线程

1个进程中可以开启多条线程，每条线程可以并行（同时）执行不同的任务

进程 ->车间，线程->车间工人

多线程技术可以提高程序的执行效率

比如同时开启3条线程分别下载3个文件

2）.多线程的原理

同一时间，CPU只能处理1条线程，只有1条线程在工作（执行）

多线程并发（同时）执行，其实是CPU快速地在多条线程之间调度（切换）

如果CPU调度线程的时间足够快，就造成了多线程并发执行的假象

思考：如果线程非常非常多，会发生什么情况？

CPU会在N多线程之间调度，CPU会累死，消耗大量的CPU资源

每条线程被调度执行的频次会降低（线程的执行效率降低）

3）线程池：

是指在初始化一个多线程应用程序过程中创建一个线程集合，然后在需要执行新的任务时重用这些线程而不是新建一个线程。线程池中线程的数量通常完全取决于可用内存数量和应用程序的需求。然而，增加可用线程数量是可能的。线程池中的每个线程都有被分配一个任务，一旦任务已经完成了，线程回到池子中并等待下一次分配任务。

2）Java提供的四种线程池的好处在于：

a. 重用存在的线程，减少对象创建、消亡的开销，性能佳。

b. 可有效控制最大并发线程数，提高系统资源的使用率，同时避免过多资源竞争，避免堵塞。

c. 提供定时执行、定期执行、单线程、并发数控制等功能

3）Java通过Executors提供四种线程池，分别为：

newCachedThreadPool

创建一个可缓存线程池，如果线程池长度超过处理需要，可灵活回收空闲线程，若无可回收，则新建线程。

newFixedThreadPool

创建一个定长线程池，可控制线程最大并发数，超出的线程会在队列中等待。

newScheduledThreadPool

创建一个定长线程池，支持定时及周期性任务执行。

newSingleThreadExecutor

创建一个单线程化的线程池，它只会用唯一的工作线程来执行任务，保证所有任务按照指定顺序(FIFO, LIFO, 优先级)执行。