# 线程池

1.为什么要创建线程池？

答：创建线程和消耗线程需要消耗大量的系统资源，再加上业务处理时长，会大大的降低工作效率，使用线程池可将创建好一定数量的线程放入线程池中，在需要时从线程池中取出。当线程任务执行完毕则放回线程池。

同时方便对线程进行管理，如果当前请求数量超过已有线程执行范围，比如说程序启动时创建了100个线程，现在有101个请求进来，每个请求分配一个线程去执行，那么线程就会不够用，这时线程池可以管理成多出来的一个请求就会进入排队等候，避免无休止的创建线程导致系统崩溃。

2.线程池的种类

1.newSingleThreadExecutor

创建一个单线化的线程池，他只会用唯一的工作线程来执行任务，保证所有的任务按照指定的顺序来执行。

2，newFixedThreadEXecutor

创建指定数量级线程的线程池，可控制线程的最大并发数，超出的线程会进入排队等候。

3.newCachedThreadExecutor

创建一个可缓存的线程池

如果线程池线程数量超出处理需要，则会回收空闲线程，若无可回收的，则新创建线程池。

4.newScheduleThreadExecutor

创建定时线程池，支持定时任务周期性的执行

定时任务使用ThreadPoolExecutor创建而不是Executor创建

3.扩展。。。

## Mysql和Oracle的区别

Oracle是完全支持事务的而Mysql支持innodb储存引擎行级锁情况下支持事务

# 缓存击穿 缓存穿透 缓存雪崩区别

缓存穿透：指和查询一个一定不存在的数据，缓存和数据库都没有，可能是黑户恶意发送大量请求导致DB的承受压力过大

解决：通常我们会在接口层加校验，不合法的参数进行校验，用户鉴权，不合法的参数直接return，或者在数据库查询不到的结果时，给缓存加null值

缓存雪崩：指设置缓存失效时间相同，导致大量请求直接打入DB,导致DB崩溃。

解决：在设置缓存失效时间上加上随机事件，比如1-5分钟。

缓存击穿：对于一些热点key会有大量的请求涌入，如果这时key值失效，那么请求会全部打入DB。

解决：加入分布式锁，热点key永远不会失效，或者对线程请求加锁，保证获取到锁的线程才能操作db。

# 单例模式

什么是单例模式？

单一的类只负责创建自己的对象，同时确保单个的对象被创建，该类提供了唯一访问对象的方式，可以直接访问，不需要实例化

特点：单例类只有一个实例

必须创建自己的对象

给其他对象提供这一实例

# HashMap的实现原理

hashMap的底层数据结构是一个链表加数组的一个数组结构，首先根据key值得hashCode重新计算Hash值，根据hash值得到数组中元素的下标位置，如果数组已存在元素，则将新添加的元素放入链头，存在的元素在链尾。

jdk1.8之后对HashMap的是先做出了优化，当链表中的加点数据超过8个，该链表转为红黑树来提高查询效率。

#### Cookie和session的区别

session是存在于服务端的保存用户登录数据对象，在第一次请求时服务器自动生成一个session，当第二次请求时。客户端将发送cookie（文本信息）携带sessionId去请求服务，判断当前sessionId是否和保存的sessionIdz一致，找到对应用户的session

# 如何防止SQL注入

前端可能通过参数传递字符串在sql执行时满足条件完成请求，这时需要使用占位符#{}使其成为参数而不是通过字符串拼接去执行sql