# Redis知识点

## 1、Redis好处

1. 速度快，存于内存中。
2. 数据类型丰富，String、List、Set、Hash。
3. 支持事务，操作是原子性。
4. 功能强，缓存数据库、消息队列，按Key设置过期时间。

## 2、Redis持久化

1. RDB，每隔一定时间通过子线程写入临时rdb文件，完毕后替换老的二进制文件dump.rdb。缺点是中间这段时间会丢失。
2. AOF，将命令写入文本文件中。为了解决AOF日志文件过大的问题，可以采用AOF重写方式。

## 3、Redis事务

Redis只能保证一个client发起的事务中的命令可以连续的执行，而中间不会插入其他client的命令。

由于Redis是单线程来处理所有client的请求的所以做到这点是很容易的。一般情况下Redis在接受到一个client发来的命令后会立即处理并返回处理结果，但是当一个client在一个连接中发出multi命令有，这个连接会进入一个事务上下文，该连接后续的命令并不是立即执行，而是先放到一个队列中。当从此连接受到exec命令后，redis会顺序的执行队列中的所有命令。

并将所有命令的运行结果打包到一起返回给client.然后此连接就 结束事务上下文。

## 4、Redis乐观锁

乐观锁介绍：

watch指令在redis事物中提供了CAS（check-and-set）的行为。为了检测被watch的keys在是否有多个clients同时改变引起冲突，这些keys将会被监控。如果至少有一个被监控的key在执行exec命令前被修改，整个事物将会回滚，不执行任何动作，从而保证原子性操作，并且执行exec会得到null的回复。

乐观锁工作机制：

watch 命令会监视给定的每一个key，当exec时如果监视的任一个key自从调用watch后发生过变化，则整个事务会回滚，不执行任何动作。注意watch的key是对整个连接有效的，事务也一样。如果连接断开，监视和事务都会被自动清除。当然exec，discard，unwatch命令，及客户端连接关闭都会清除连接中的所有监视。还有，如果watch一个不稳定(有生命周期)的key并且此key自然过期，exec仍然会执行事务队列的指令。

## 数据结构

String、Hash、List、Set、有序SortedSet、队列Push/Pop

## 分布式锁

利用setnx争抢锁，然后用expire给锁加上过期时间。

这两个操作不是原子性的，所以会出现setnx之后，expire之前程序崩溃的局面。

2.6之后的Redis，set涵盖了setnx的功能，还能设置过期时间。

## 查找Key

有N多个Key，如何找出固定开头的Key？

使用keys指令查找。但这个有问题，因为Redis是单线程，使用keys指令会阻塞线程直到完成，这在生产环境下是不合适的。

可以用scan指令，无阻塞提取指定key列表，但有重复的可能。需要程序再去重。