# Redis笔记

## Redis的分片技术

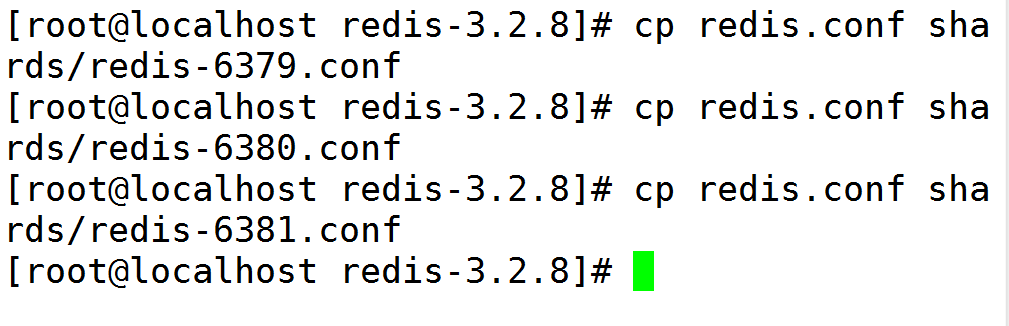
为什么要引入分片技术，因为单台的Redis不能分配过大的内存空间，内存空间过大就会导致寻址过慢，降低效率

定义：多台Redis为用户提供一个超大的内存空间，并且每个Redis中只保存一部分的数据

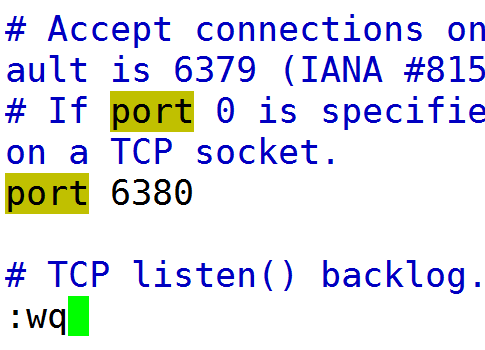
### 如何实现Redis的分片

1. 首先复制Redis.conf的文件，一份配置文件代表一台Redis ，配置文件不

能同名



1. 改Port端口



1. 启动Redis：redis-server Redis.conf（Redis相应的配置文件）

[root@localhost shards]# redis-server redis-6379.conf

[root@localhost shards]# redis-server redis-6380.conf

[root@localhost shards]# redis-server redis-6381.conf

1. 然后用命令 : ps -ef|grep redis 查看进程是否已启动

### Spring 整合Redis的分片

* 采用ShardedJedis 这个类进行构造，其构造函数中的参数可传入单个或一个Set集合，因为Set集合可以去重

### Redis分片的缺点

* Redis本身没有实现高可用
* 如果有一台Redis宕机了就会影响整个分片系统的运行
* 一旦运行了就不能关机
* 可能会造成数据的丢失

## Redis 哈希的特性

Redis如何实现数据的动态划分

就是根据Hash一致性的算法进行划分

1. 一致性，根据RedisIp+端口进行哈希运算，得到的结果唯一，根据key值进行哈希运算，确定唯一位置，由Key值位置按顺时针的顺序找到最近的节点node，然后进行储存数据，取值也是一样，确定位置然后顺时针找到最近节点Node,进行取值
2. 均衡性，如果哈希后得到的Node节点范围不均也就是负载不均，Redis就是调用虚拟的Node节点进行均衡操作，然后将数据转储
3. 单调性，Redis中分片的节点只能单调递增，因为当分片节点减少时，相当于内存没了，会导致系统奔溃
4. 负载：在分布式系统中用户使用内存空间时，不能获得全部的内存空间，导致不同的key可能会存到同一个位置
5. 离散性：在分布式系统中用户使用内存空间时，不能获取全部的内存空间，导致同一个key可能会存到不同的地方

## Redis的主从搭建（实现Redis高可用的前提条件）

* 首先也是要复制Redis.conf这个配置文件 例如配置：1主两从的搭建
* 改端口
* 启动服务

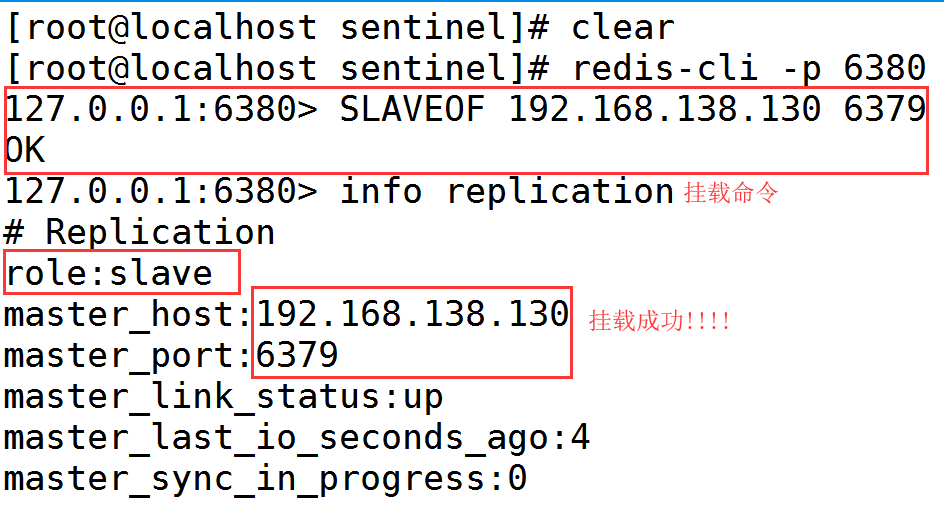
[root@localhost sentinel]# redis-server redis-6379.conf

[root@localhost sentinel]# redis-server redis-6380.conf

[root@localhost sentinel]# redis-server redis-6381.conf

* 进入从机的客户端界面 redis-cli -p 6380
* 执行主从挂载的命令，也就是在从机的客户端操作界面挂载主机

SLAVEOF 127.0.0.1(IP) 6379(主机的端口)



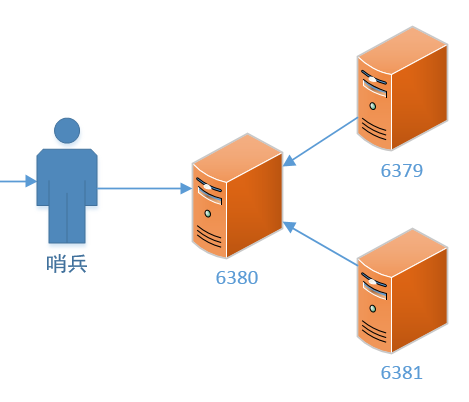
* 查看是否挂载成功的命令： info replication

上述操作完成后,将6380挂载到6379上.

被挂载后的从机只能读，不能写

## 哨兵sentinel

为什么要引入哨兵？因为主从挂载主机可能会挂掉，然后要有人从从机中选出一台当主机，继续实现主从挂载



* 哨兵可以监听只主机的生命状态(心跳检测，利用PING-PONG实现
* 如果发起了三次PING 主机都没有响应PONG,则认为主机宕机了
* 并且修改其他机器的配置文件，实现主从挂载，切换主机操作