Redis学习记录

# 安装

* 在 /opt 下载Redis，下载完毕后，进行解压缩

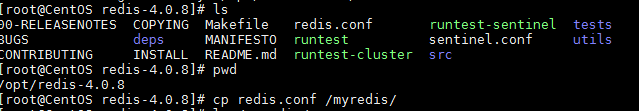
下载命令： wget http://download.redis.io/releases/redis-4.0.8.tar.gz

解压命令： tar zxvf redis-4.0.8.tar.gz

进入目录： # cd redis-4.0.8

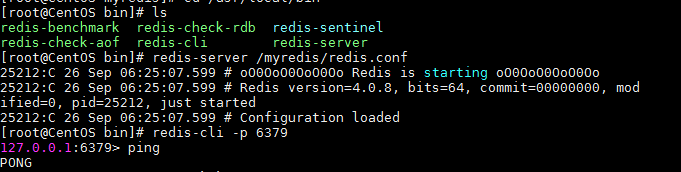


* 解压完成后在 redis-4.0.8 目录下执行make 命令（需要gcc），再执行make install
* 安装完成后，把redis.conf复制一份到根目录下的myredis文件夹，这样保证了不操作原来的配置文件

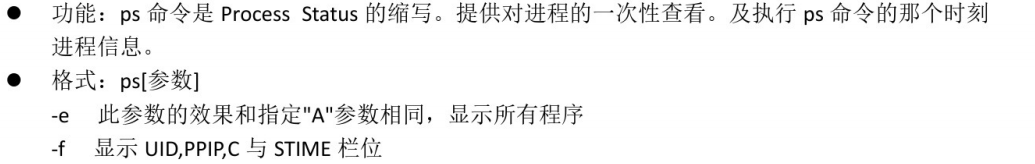


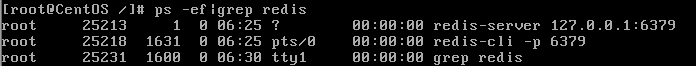
# 启动与关闭

首先进到 /usr/local/bin 下



查看经常状态的命令：ps -ef |grep redis





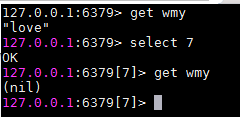
关闭



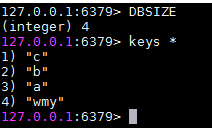


## Redis杂项知识讲解

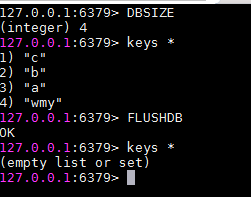
1. redis默认有16个库，角标是从0开始的



1. Dbsize 查看当前数据库key的个数（tab键提示，会自动补全，并变成大写）



1. Flushdb命令清空当前数据库的键；flushall是清空所有的库



1. Redis索引都是从0开始的；默认端口是6379

# Reidis的五大数据类型

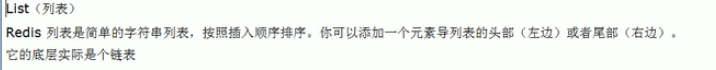
* String



* Hash



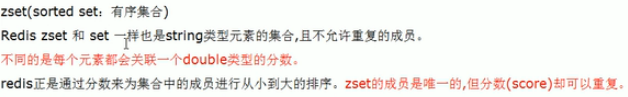
* List（列表）



* Set（集合）

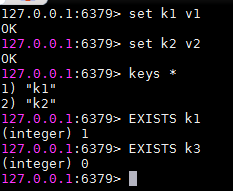


* Zset（有序集合）

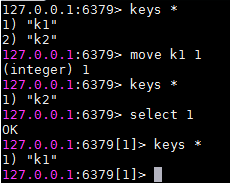


## Redis键(key)

* keys \* 列出所有的key
* exists key的名字，判断某个key是否存在

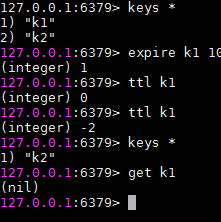


* move key db 当前库就没有了，被移除到了指定的库



* expire key 秒数：为给定的key设置过期时间

ttl key 查看还有多少秒过期，-1表示永不过期，-2表示已过期



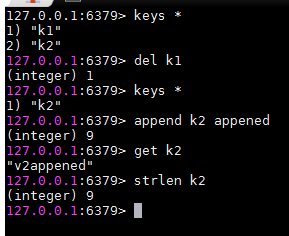
get 是获得不到过期数据的；并且keys \* 里面也没有改key了

过期后是彻底消失

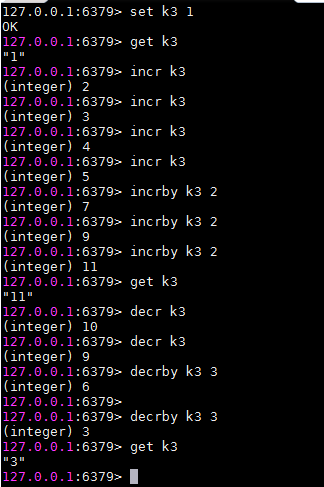
ttl(time to live)

## Redis 字符串（string）

* set/get/del/append/strlen

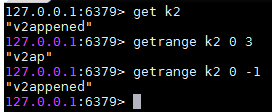


* incr/decr/incrby/decrby，一定要是数字才能进行加减

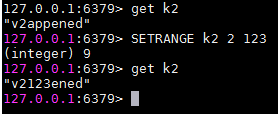


* getrange:获取指定范围内的值，类似between...and的关系

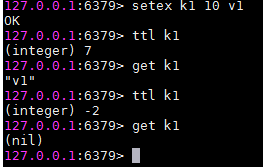
从0到-1表示全部

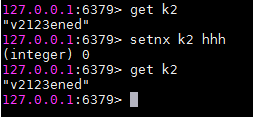


* setrange设置指定区间范围内的值，格式是setrange key index 具体值



* setex(set with expire)/setnx(set if not exist)

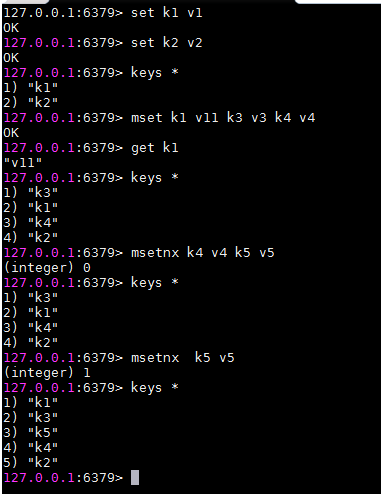




* mset/mget/msetnx

对于mset可以批量设置键值，对于已经存在的会进行覆盖；

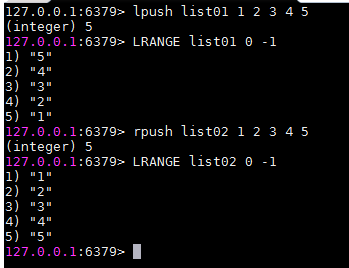
对于msetnx 可以进行批量设置，如果设置已经存在的，则该条命令不会成功。



## Redis列表（list）

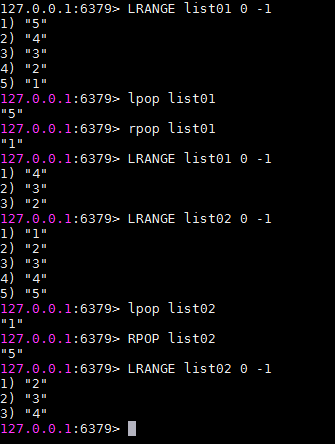
* Lpush/rpush/lrange

Lpush 可模拟栈；rpush可模拟队列

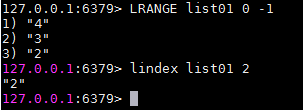


* Lpop/rpop

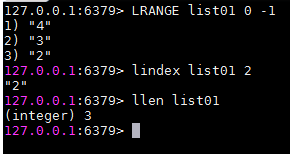
其实就是双向链表，左边可出、右边也可出



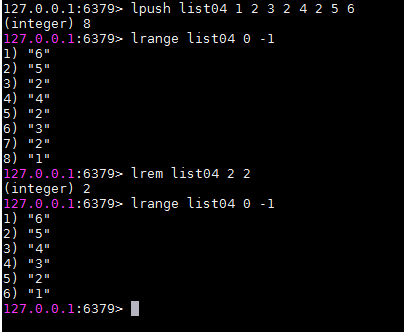
* Lindex，按照索引下标获得元素（从左到右）



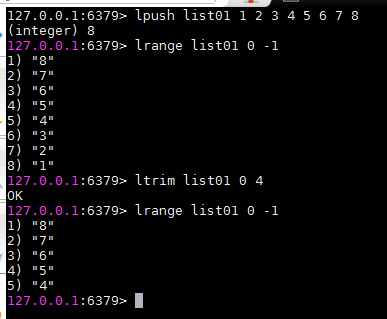
* 获得list长度



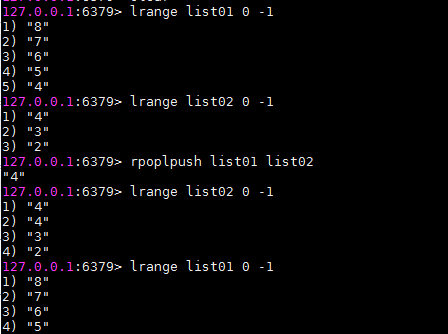
* Lrem key num value ：删除num个value；如图，删除list04中的2个2 ，先遍历到的先删除

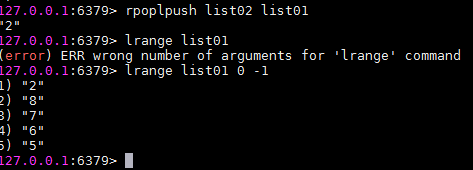


* Ltrim key 开始index 结束index，截取指定范围内的值后再赋值给key

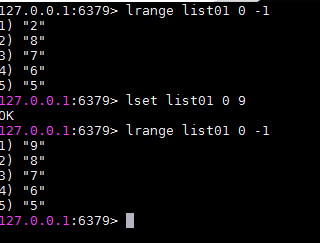


* Rpoplpush 源列表 目的列表：从源列表右边移出放入目的列表左边

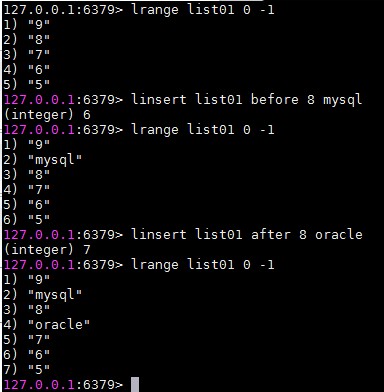




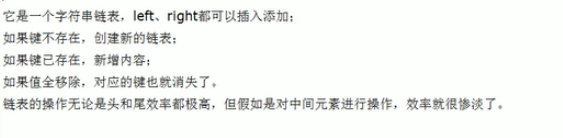
* Lset key index value：设置key中某个索引位置的值



* Linsert key before/after 值1 值2

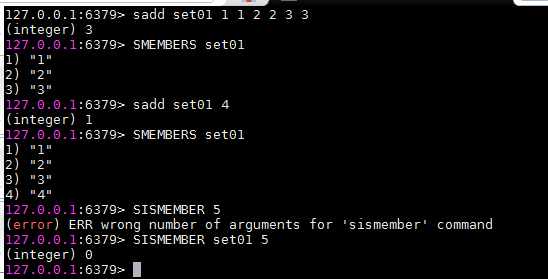


* 性能总结

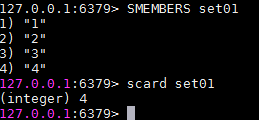


## Redis集合（set）

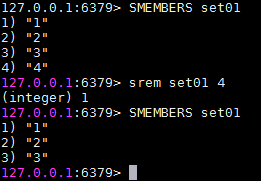
* Sadd/smembers/sismember



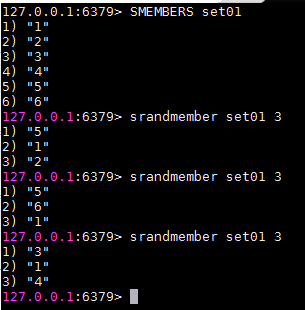
* Scard：获取集合里面的元素个数



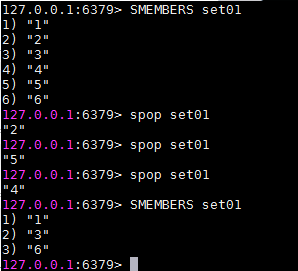
* Srem key value：删除集合中元素



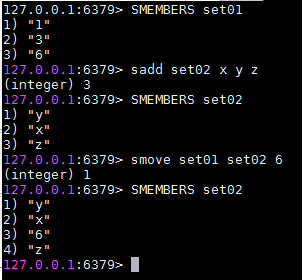
* Srandmember key 某个整数：随机取出几个数



* Spop key :随机出栈



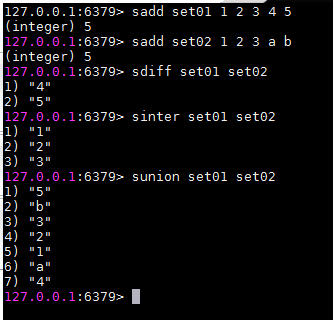
* Smove key1 key2 key1里某个值：将key1里的某个值赋给key2



* 差集：sdiff（在一个集合中而不再另一个集合中）

交集：sinter

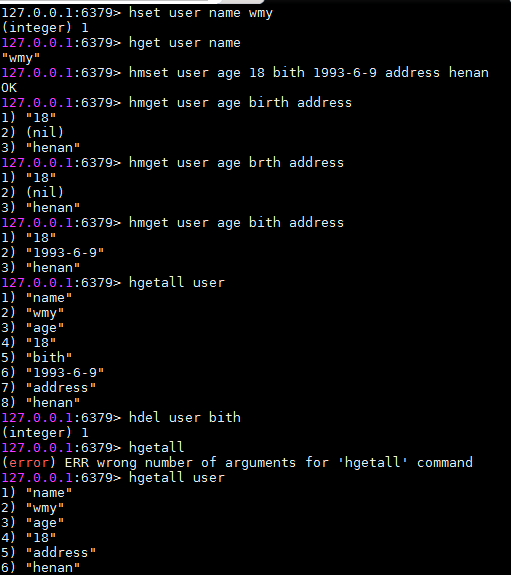
并集：sunion



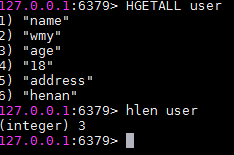
## Redis哈希（hash）

KV 模式不变，但是V是键值的集合

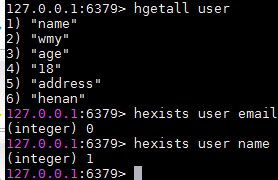
* Hset/hget/hmset/hmget/hgetall/hdel



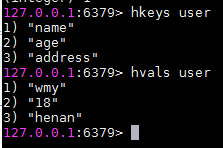
* Hlen ：获得field 的个数



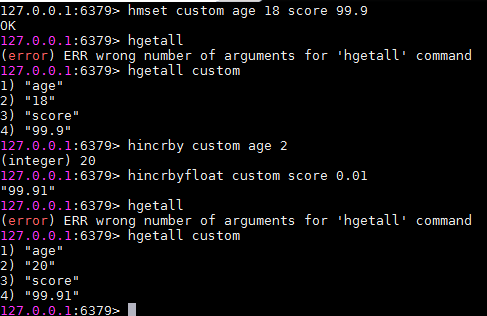
* Hexists key field ：在key对应的value中是否有field字段



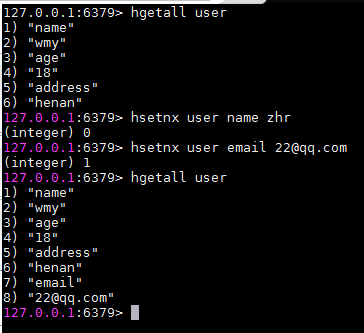
* Hkeys/hvals：获得所有的field；获得所有field对应的value



* Hincrby/hincrbyfloat key field increment：给某个字段增加相应的整数或小数



* Hsetnx key field value :如果key对应的value中不存在field字段，则增加该字段

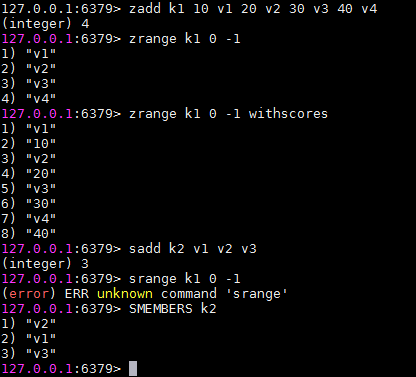


## Zset(sorted set)

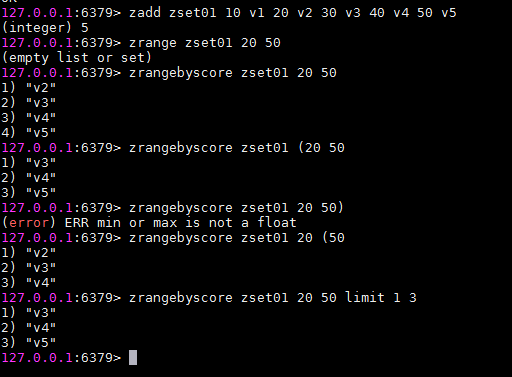
该类型是在set的基础上，加上了一个score值

比如set对应k1 v1 v2 v3；zset对应k1 score1 v1 score v2

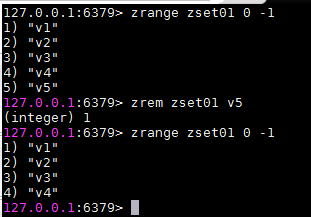
* Zadd/zrange



* Zrangebyscore key 开始score 结束score
  + Withscores
  + (
  + Limit:返回限制，相当于分页（limit 开始下标步 多少步）



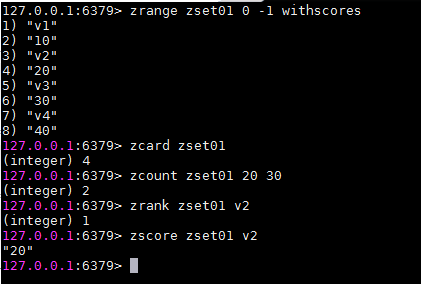
* Zrem key 某score 下对应的value值，作用是删除元素



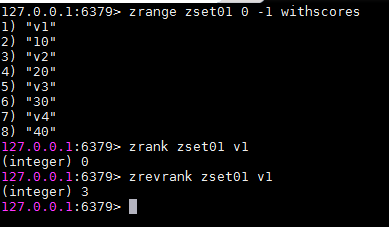
* Zcard/zcount key score区间：统计个数，不带score区间是统计所有；

Zrank key values值，作用是获得下标值；

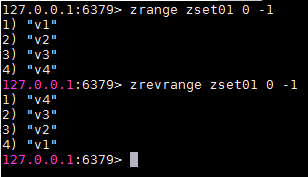
Zscore key 对应值，获得分数。



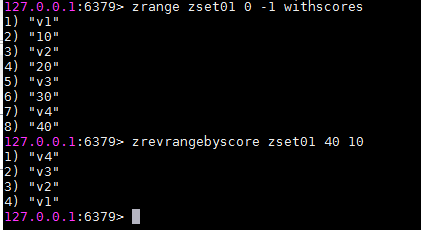
* Zrevrank key values：逆序获得下标值



* Zrevrange key start end



* Zrevrangebyscore key max min

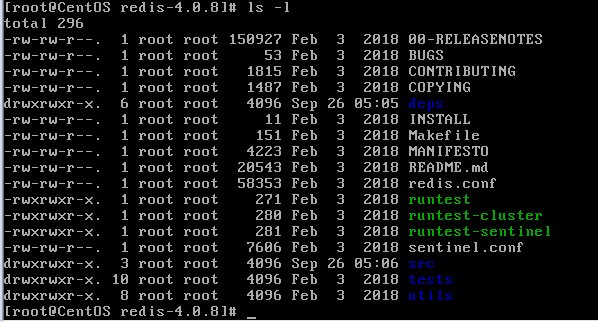


# Redis配置文件介绍

## 它在哪

Ls-l :列出目录内容，-l表示使用详细格式列表

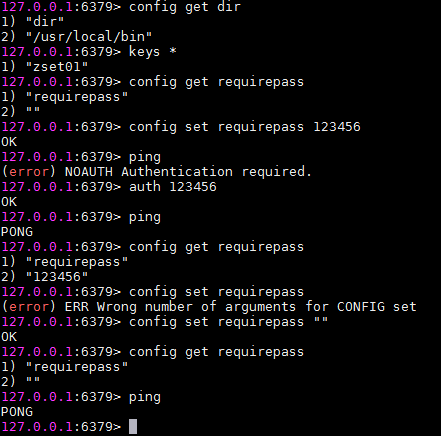
此命令可以简写成ll



## Security 安全配置

Config get dir：得到当前启动redis 的目录

Config get requirepass：查看访问redis的密码；默认没有密码



## Limits限制



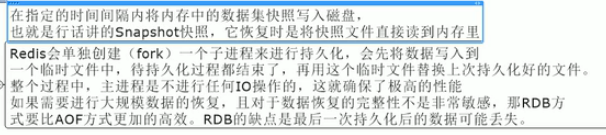
## 常见配置redis.conf介绍





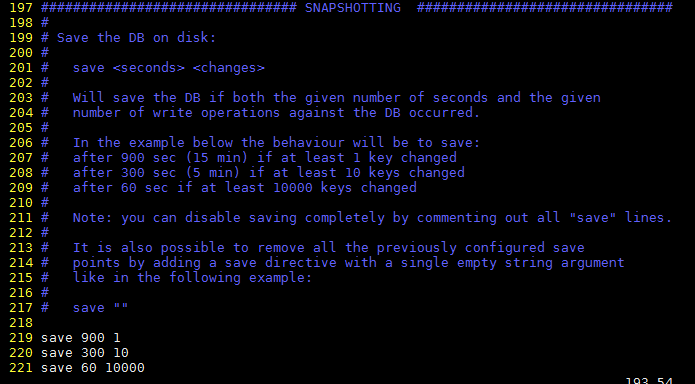
# Redis持久化

## RDB(redis datebase)



Rdb 保存的是dump.rdb文件

Save seconds changes：在多少秒内有多少个key被改变就自动触发保存。



如果想要立刻马上进行保存，可以使用save命令

AOF(append only file)