## 1.1五大数据类型简介

1. String（字符串）

string是redis最基本的类型，可以理解成与Memcacheed一模一样的类型，一个key对应一个value。

string类型是二进制安全的。意思是redis的string可以包含任何数据。比如jpg图片或者序列化的对象。

string类型是redis最基本的数据类型，一个redis中字符串value最多可以是512M。

1. Hash（哈希，类似java里的Map）

hash是一个键值对集合。

是一个string类型的field和value的映射表，hash特别适合用于存储对象。类似java里面的Map<String,Object>

1. List（列表）

list是简单的字符串列表，按照插入顺序排序，可以添加一个元素到列表的头部（左边）或者尾部（右边）。它的底层实际是个链表。

1. Set（集合）

set是string类型的无序集合。它是通过HashTable实现的。

1. Zset（sorted set：有序集合）

zset和set一样也是string类型元素的集合，且不允许重复的成员。不同的是每个元素都会关联一个double类型的分数。

Redis正是通过分数来为集合中的成员进行从小到大的排序。zset的成员是唯一的，但分数（score）却是可以重复的。

## 1.2Redis常见数据类型操作命令

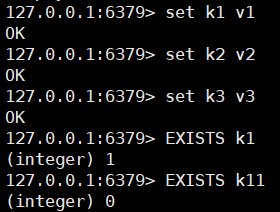
<http://redisdoc.com>



## 1.3键（key）

1. EXISTS key（检查给定key是否存在）

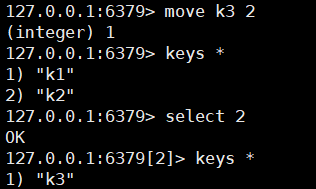
若key存在，返回1，否则返回0。



1. MOVE key db（将当前数据库的key移动到给定的数据库db当中）

如果当前数据库（源数据库）和给定数据库（目标数据库）有相同名字的给定key，或者key不存在于当前数据库，那么MOVE没有任何效果。

移动成功返回1，失败则返回0。



1. TTL key（以秒为单位，返回给定key的剩余生存时间（time to live））

当key不存在时，返回-2；

当key存在但没有设置剩余时间时，返回-1；

否则，以秒为单位，返回key的剩余生存时间。

1. EXPIRE key seconds（为给定key设置生存时间，当key过期时（生存时间为0），它会被自动删除）

设置成功返回1；

当key不存在或者不能为key设置生存时间时，返回0。

1. TYPE key（返回key所存储的值的类型）

none（key不存在）

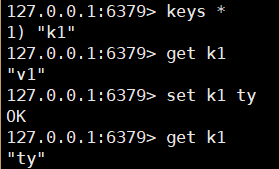
string（字符串）

list（列表）

set（集合）

zset（有序集）

hash（哈希表）



## 1.4字符串（String）

单值单value

1. set/get/del/append（如果key已经存在并且是一个字符串，此命令将value追加到key原来的值的末尾，如果key不存在就和set效果一样）/strlen（返回key所储存的字符串值的长度，当key储存的不是字符串值时，返回一个错误）
2. incr（将key中储存的数字值增一）/decr（将key中储存的数字值减一）/incrby（将所储存的值加上增量increment）/decrby（将所储存的值减去减量decrement），一定要是数字才能进行加减。
3. setx key seconds value（set with expire，将值value关联到key，并将key的生存时间设为seconds）键秒值/setnx key value（set if not exist，将key的值设为value，当且仅当key不存在）
4. mset key value [key value ...]（同时设置一个或多个key-value对）/mget key [key ...]（返回所有给定key的值）/msetnx key value [key value]（同时设置一个或多个key-value对，当且仅当所给定key都不存在）
5. getset key value（先get再set。将给定key的值设为value，并返回key的旧值）

## 1.5列表（List）

单值多value

1. lpush（将一个或多个值value插入到列表key的表头）/rpush（将一个或多个值value插入到列表key的表尾，最右边）/lrange（返回列表key中指定区间内的元素，区间以偏移量start和stop指定）
2. lpop（移除并返回列表key的头元素）/rpop（移除并返回列表key的尾元素）
3. lindex（返回列表key中，下标为index的元素）
4. llen（返回列表key的长度）
5. lrem key count value（根据参数count的值，移除列表中与参数value相等的元素）
6. ltrim key start stop（对一个列表进行修剪trim，即让列表只保留指定区间内的元素，不在指定区间之内的元素都将被删除）
7. rpoplpush source destination（在一个原子时间内，执行以下两个动作：）

将列表source中的最后一个元素（尾元素）弹出，并返回给客户端；

将source弹出的元素插入到列表destination，作为destination列表的头元素。

1. lset key index value（将列表key下标为index的元素的值设置为value）
2. linsert key before/after pivot value（将值value插入到列表key当中，位于值pivot之前或之后）
3. 性能总结

它是一个字符串链表，left、right都可以插入添加；

如果键不存在，创建新的链表；

如果键已存在，新增内容；

如果值全移除，对应的键也就消失了。

链表的操作无论是头和尾效率都极高，但假如是对中间元素进行操作，效率就很惨淡了。

## 1.6集合（Set）

单值多value

1. sadd（将一个或多个member元素加入到集合key当中，已经存在于集合的member元素将被忽略）/smembers（返回集合key中的所有成员）/sismember（判断member元素是否集合key的成员）
2. scard key（返回集合key的基数（集合中元素的数量））
3. srem key member [member ...]（移除集合key中的一个或多个member元素，不存在的member元素会被忽略）
4. srandmember key [count]（只提供key参数，那么返回集合中的一个随机元素）
5. spop key（移除并返回集合中的一个随机元素）
6. smove source destination member（将member元素从source集合移动到destination集合）
7. 数学集合类
8. 差集：sdiff key [key ...]（返回一个集合的全部成员，该集合是所有给定集合之间的差集）
9. 交集：sinter key [key ...]（返回一个集合的全部成员，该集合是所有给定集合的交集）
10. 并集：sunion key [key ...]（返回一个集合的全部成员，该集合是所有给定集合的并集）

## 1.7哈希（Hash）

KV模式不变，但V是一个键值对。

1. hset（将哈希表key中的域field的值设为value）/hget（返回哈希表中国给定域field的值）/hmset（同时将多个field-value（域-值）对设置到哈希表key中）/hgetall（返回哈希表key中，所有的域和值）/hdel（删除哈希表key中的一个或多个指定域，不存在的域将被忽略）
2. hlen key（返回哈希表key中域的数量）
3. hexists key field（查看哈希表key中，给定域field是否存在）
4. hkeys（返回哈希表key中的所有域）/hvals（返回哈希表key中所有域的值）
5. hincrby（为哈希表key中的域field的值增加上增量increment）/hincrbyfloat（为哈希表key中的域field加上浮点数增量increment）
6. hsetnx key field value（将哈希表key中的域field的值设置为value，当且仅当域field不存在）

## 1.8有序集合Zset（sorted set）

在set基础上，加一个score值。之前set是k1 v1 v2 v3，现在zset是k1 score v1 score2 v2

1. zadd（将一个或多个member元素及其score加入到有序集key当中）/zrange（返回有续集key中，指定区间内的成员）
2. zrangebyscore key min max [withscores] [limit offset count]（返回有序集key中，所有score值介于min和max之间（包括等于min或max）的成员。有序集成员按score值递增（从小到大）次序排列）
3. zrem key member [member ...]（移除有序集key中的一个或多个成员，不存在的成员将被忽略）
4. zcard（返回有序集key的基数）/zcount key min max（返回有序集key中，score值在min和max之间（默认包括score值等于min或max）的成员的数量）/zrank key member（返回有序集key中成员member的排名。其中有序集成员按score值递增（从小到大）顺序排列。排名以0为底，也就是说，score值最小的成员排名为0.）
5. zrevrange key start stop [withscores]（返回有序集key中，指定区间内的成员）
6. zrevrank key member（返回有序集key中成员member的排名。其中有序集成员按score值递减（从大到小）排序。排名以0为底，也就是说，score值最大的成员排名为0）
7. zrevrangebyscore key max min [withscores] [limit offset count]（返回有序集key中，score值介于max和min之间（默认包括等于max或min）的所有的成员。有序集成员按score值递减（从大到小）的次序排列）