

# 命令

## 一、字符串

INCR key 将可以对应的数字值加一  
DECR key 将 key 中储存的数字值减一  
set get

## 二、列表

lpop key  
rpop key  
blpop key  
brpop key  
lpush key  
rpush key

1	<a href="#">BLPOP key1 [key2 ] timeout</a> 移出并获取列表的第一个元素， 如果列表没有元素会阻塞列表直到等待超时或发现可弹出元素为止。
2	<a href="#">BRPOP key1 [key2 ] timeout</a> 移出并获取列表的最后一个元素， 如果列表没有元素会阻塞列表直到等待超时或发现可弹出元素为止。
3	<a href="#">BRPOPLPUSH source destination timeout</a> 从列表中弹出一个值，将弹出的元素插入到另外一个列表中并返回它； 如果列表没有元素会阻塞列表直到等待超时或发现可弹出元素为止。
4	<a href="#">LINDEX key index</a> 通过索引获取列表中的元素
5	<a href="#">LINSERT key BEFORE/AFTER pivot value</a> 在列表的元素前或者后插入元素
6	<a href="#">LLEN key</a> 获取列表长度
7	<a href="#">LPOP key</a> 移出并获取列表的第一个元素
8	<a href="#">LPUSH key value1 [value2]</a> 将一个或多个值插入到列表头部
9	<a href="#">LPUSHX key value</a> 将一个值插入到已存在的列表头部
10	<a href="#">LRANGE key start stop</a> 获取列表指定范围内的元素
11	<a href="#">LREM key count value</a> 移除列表元素
12	<a href="#">LSET key index value</a> 通过索引设置列表元素的值
13	<a href="#">LTRIM key start stop</a> 对一个列表进行修剪(trim)，就是说，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。
14	<a href="#">RPOP key</a> 移除并获取列表最后一个元素
15	<a href="#">RPOPLPUSH source destination</a> 移除列表的最后一个元素，并将该元素添加到另一个列表并返回
16	<a href="#">RPUSH key value1 [value2]</a> 在列表中添加一个或多个值
17	<a href="#">RPUSHX key value</a> 为已存在的列表添加值

## 三、集合

1	<a href="#">SADD key member1 [member2]</a> 向集合添加一个或多个成员
2	<a href="#">SCARD key</a> 获取集合的成员数
3	<a href="#">SDIFF key1 [key2]</a> 返回给定所有集合的差集
4	<a href="#">SDIFFSTORE destination key1 [key2]</a> 返回给定所有集合的差集并存储在 destination 中
5	<a href="#">SINTER key1 [key2]</a> 返回给定所有集合的交集
6	<a href="#">SINTERSTORE destination key1 [key2]</a> 返回给定所有集合的交集并存储在 destination 中
7	<a href="#">SISMEMBER key member</a> 判断 member 元素是否是集合 key 的成员
8	<a href="#">SMEMBERS key</a> 返回集合中的所有成员
9	<a href="#">SMOVE source destination member</a> 将 member 元素从 source 集合移动到 destination 集合
10	<a href="#">SPOP key</a> 移除并返回集合中的一个随机元素
11	<a href="#">SRANDMEMBER key [count]</a> 返回集合中一个或多个随机数
12	<a href="#">SREM key member1 [member2]</a> 移除集合中一个或多个成员
13	<a href="#">SUNION key1 [key2]</a> 返回所有给定集合的并集
14	<a href="#">SUNIONSTORE destination key1 [key2]</a> 所有给定集合的并集存储在 destination 集合中
15	<a href="#">SSCAN key cursor [MATCH pattern] [COUNT count]</a> 迭代集合中的元素

#### 哈希

1	<a href="#">HDEL key field1 [field2]</a> 删除一个或多个哈希表字段
2	<a href="#">HEXISTS key field</a> 查看哈希表 key 中，指定的字段是否存在。
3	<a href="#">HGET key field</a> 获取存储在哈希表中指定字段的值。
4	<a href="#">HGETALL key</a> 获取在哈希表中指定 key 的所有字段和值
5	<a href="#">HINCRBY key field increment</a> 为哈希表 key 中的指定字段的整数值加上增量 increment 。
6	<a href="#">HINCRBYFLOAT key field increment</a> 为哈希表 key 中的指定字段的浮点数值加上增量 increment 。
7	<a href="#">HKEYS key</a> 获取所有哈希表中的字段
8	<a href="#">HLEN key</a> 获取哈希表中字段的数量
9	<a href="#">HMGET key field1 [field2]</a> 获取所有给定字段的值
10	<a href="#">HMSET key field1 value1 [field2 value2 ]</a> 同时将多个 field-value (域-值)对设置到哈希表 key 中。
11	<a href="#">HSET key field value</a> 将哈希表 key 中的字段 field 的值设为 value 。
12	<a href="#">HSETNX key field value</a> 只有在字段 field 不存在时，设置哈希表字段的值。
13	<a href="#">HVALS key</a> 获取哈希表中所有值
14	<a href="#">HSCAN key cursor [MATCH pattern] [COUNT count]</a> 迭代哈希表中的键值对。

#### 有序集合

1	<a href="#">ZADD key score1 member1 [score2 member2]</a> 向有序集合添加一个或多个成员，或者更新已存在成员的分值
2	<a href="#">ZCARD key</a> 获取有序集合的成员数
3	<a href="#">ZCOUNT key min max</a> 计算在有序集合中指定区间分值的成员数
4	<a href="#">ZINCRBY key increment member</a> 有序集合中对指定成员的分数加上增量 increment
5	<a href="#">ZINTERSTORE destination numkeys key [key ...]</a> 计算给定的一个或多个有序集的交集并将结果集存储在新的有序集合 key 中
6	<a href="#">ZLEXCOUNT key min max</a> 在有序集合中计算指定字典区间内成员数量
7	<a href="#">ZRANGE key start stop [WITHSCORES]</a> 通过索引区间返回有序集合指定区间内的成员
8	<a href="#">ZRANGEBYLEX key min max [LIMIT offset count]</a> 通过字典区间返回有序集合的成员
9	<a href="#">ZRANGEBYSCORE key min max [WITHSCORES] [LIMIT]</a> 通过分数返回有序集合指定区间内的成员
10	<a href="#">ZRANK key member</a> 返回有序集合中指定成员的索引
11	<a href="#">ZREM key member [member ...]</a> 移除有序集合中的一个或多个成员
12	<a href="#">ZREMRANGEBYLEX key min max</a> 移除有序集合中给定的字典区间的所有成员
13	<a href="#">ZREMRANGEBYRANK key start stop</a> 移除有序集合中给定的排名区间的所有成员
14	<a href="#">ZREMRANGEBYSCORE key min max</a> 移除有序集合中给定的分数区间的所有成员
15	<a href="#">ZREVRANGE key start stop [WITHSCORES]</a> 返回有序集中指定区间内的成员，通过索引，分数从高到底
16	<a href="#">ZREVRANGEBYSCORE key max min [WITHSCORES]</a> 返回有序集中指定分数区间内的成员，分数从高到低排序
17	<a href="#">ZREVRANK key member</a> 返回有序集合中指定成员的排名，有序集成员按分数值递减(从大到小)排序
18	<a href="#">ZSCORE key member</a> 返回有序集中，成员的分数值
19	<a href="#">ZUNIONSTORE destination numkeys key [key ...]</a> 计算给定的一个或多个有序集的并集，并存储在新的 key 中
20	<a href="#">ZSCAN key cursor [MATCH pattern] [COUNT count]</a> 迭代有序集合中的元素（包括元素成员和元素分值）