

redis五种数据结构:

1. String

可以是字符串, 整数或者浮点数, 对整个字符串或者字符串中的一部分执行操作, 对整个整数或者浮点数执行自增或者自减操作。

get: 获取存储在指定键中的值。

set: 设置在存储在指定键中的值。

del: 删除存储在指定键中的值。

2. list

一个链表, 链表上的每个节点都包含了就一个字符串, 从链表的两端推入或者弹出元素, 根据偏移量对链表进行修剪, 读取单个或者多个元素, 根据值查找或者移除元素。

rpush: 将给定值推入列表的右端。

lrange: 获取列表在指定范围上的所有值。

lindex: 获取列表在指定范围上的单个元素。

lpop: 从列表的左端弹出一个值, 并返回被弹出的值。

3. set

包含字符串的无序收集器, 并且被包含的每个字符串都是独一无二的。添加、获取、移除单个元素, 检查一个元素是否存在于集合中, 计算交集、并集、差集, 从集合里面随机获取元素。

sadd: 将给定元素添加到集合

smembers: 返回集合包含的所有元素。

sismember: 检查指定元素是否存在于集合中。

srem: 检查指定元素是否存在集合中, 那么移除这个元素。

4. hash

包含键值对无序散列表, 添加、获取、移除当前键值对, 获取所有键值对。

hset: 在散列里面关联起指定的键值对

hget: 获取指定散列键的值

hgetall: 获取散列包含的所有键值对

hdel: 如果给定键存在于散列里面, 那么移除这个键。

5. zset

字符串成员于浮点数分值之间的有序映射, 元素的排列顺序由分值的大小决定。添加、获取、删除单个元素, 根据分值范围或者成员来获取元素。

`zadd`: 将一个带有给定分值的成员添加到有序集合里面

`zrange`: 根据元素在有序排列中所处的位置，从有序集合里面获取多个元素。

`zrangebyscore`: 获取有序集合在给定分值范围内的所有元素。

`zrem`: 如果指定成员存在于有序集合中，那么移除这个成员。