

Redis支持五种数据类型：string（字符串），hash（哈希），list（列表），set（集合）及 zset(sorted set：有序集合)。

String（字符串）

string是redis最基本的类型，你可以理解成与Memcached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，一个键最大能存储512MB。

实例

```
redis 127.0.0.1:6379> SET name "runoob"
OK
redis 127.0.0.1:6379> GET name
"runoob"
```

在以上实例中我们使用了 Redis 的 SET 和 GET 命令。键为 name，对应的值为 runoob。
注意：一个键最大能存储512MB。

Hash（哈希）

Redis hash 是一个键值(key=>value)对集合。

Redis hash是一个string类型的field和value的映射表，hash特别适合于存储对象。

实例

```
redis> HMSET myhash field1 "Hello" field2 "World"
"OK"
redis> HGET myhash field1
"Hello"
redis> HGET myhash field2
"World"
```

以上实例中 hash 数据类型存储了包含用户脚本信息的用户对象。实例中我们使用了 Redis HMSET, HGETALL 命令，user:1为键值。

每个 hash 可以存储 2³² -1 键值对（40多亿）。

List（列表）

Redis 列表是简单的字符串列表，按照插入顺序排序。你可以添加一个元素到列表的头部（左边）或者尾部（右边）。

实例

```
redis 127.0.0.1:6379> lpush runoob redis
(integer) 1
redis 127.0.0.1:6379> lpush runoob mongodb
(integer) 2
redis 127.0.0.1:6379> lpush runoob rabbitmq
(integer) 3
redis 127.0.0.1:6379> lrange runoob 0 10
1) "rabbitmq"
2) "mongodb"
3) "redis"
redis 127.0.0.1:6379>
```

列表最多可存储 $2^{32} - 1$ 元素 (4294967295, 每个列表可存储40多亿)。

Set（集合）

Redis的Set是string类型的无序集合。

集合是通过哈希表实现的，所以添加，删除，查找的复杂度都是 $O(1)$ 。

sadd 命令

添加一个 string 元素到 key 对应的 set 集合中，成功返回1，如果元素已经在集合中返回 0，如果 key 对应的 set 不存在则返回错误。

sadd key member

实例

```
redis 127.0.0.1:6379> sadd runoob redis
(integer) 1
redis 127.0.0.1:6379> sadd runoob mongodb
(integer) 1
redis 127.0.0.1:6379> sadd runoob rabbitmq
(integer) 1
redis 127.0.0.1:6379> sadd runoob rabbitmq
(integer) 0
redis 127.0.0.1:6379> smembers runoob
```

```
1) "redis"
2) "rabbitmq"
3) "mongodb"
```

注意：以上实例中 rabbitmq 添加了两次，但根据集合内元素的唯一性，第二次插入的元素将被忽略。

集合中最大的成员数为 $2^{32} - 1$ (4294967295, 每个集合可存储40多亿个成员)。

zset(sorted set: 有序集合)

Redis `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)却可以重复。

zadd 命令

添加元素到集合,元素在集合中存在则更新对应score

```
zadd key score member
```

实例

```
redis 127.0.0.1:6379> zadd runoob 0 redis
(integer) 1
redis 127.0.0.1:6379> zadd runoob 0 mongodb
(integer) 1
redis 127.0.0.1:6379> zadd runoob 0 rabbitmq
(integer) 1
redis 127.0.0.1:6379> zadd runoob 0 rabbitmq
(integer) 0
redis 127.0.0.1:6379> > ZRANGEBYSCORE runoob 0 1000
1) "mongodb"
2) "rabbitmq"
3) "redis"
```