Redis Basic 基本概念-02课

88322511@qq.com
bigdatalyn
2018/03/28

1.key string list hash结构中，每个至少完成5个命令，包含插入 修改 删除 查询，list 和hash还需要增加遍历的操作命令，把执行结果截图贴出来

A1.

++

```
127.0.0.1:6380> set str01 1
OK
127.0.0.1:6380> get str01
"1"
127.0.0.1:6380> set str01 2
OK
127.0.0.1:6380> get str01
"2"
127.0.0.1:6380> set str01 3 NX
(nil)
127.0.0.1:6380> get str01
"2"
127.0.0.1:6380> set str02 1
OK
127.0.0.1:6380> mget str01 str02
1) "2"
2) "1"
127.0.0.1:6380> append str01 2
(integer) 2
127.0.0.1:6380> get str01
"22"
127.0.0.1:6380> append str01 apend
(integer) 7
127.0.0.1:6380> get str01
"22apend"
127.0.0.1:6380> getset str01 2
"22apend"
127.0.0.1:6380> get str01
"2"
127.0.0.1:6380>
```

++

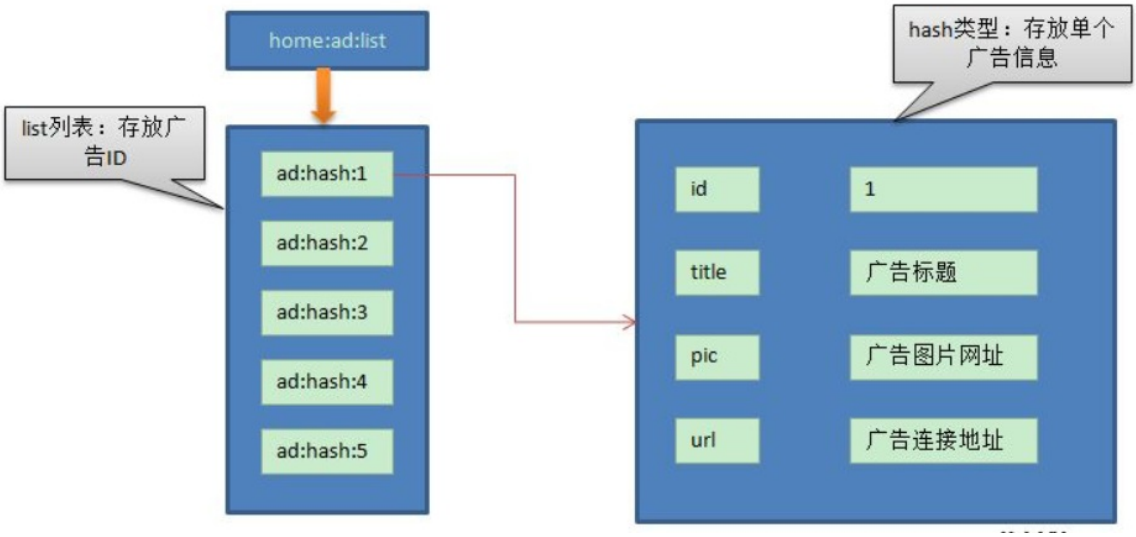
```
127.0.0.1:6380> lpush llist01 11 22 33 44 55
(integer) 5
127.0.0.1:6380> lrange llist01 0 6
1) "55"
2) "44"
3) "33"
4) "22"
5) "11"
127.0.0.1:6380>
127.0.0.1:6380> lpush llist01 66
(integer) 6
127.0.0.1:6380> lrange llist01 0 6
1) "66"
2) "55"
3) "44"
4) "33"
5) "22"
```

```
6) "11"
127.0.0.1:6380> lrem llist01 1 11
(integer) 1
127.0.0.1:6380> lrange llist01 0 6
1) "66"
2) "55"
3) "44"
4) "33"
5) "22"
127.0.0.1:6380>
127.0.0.1:6380> lset llist01 2 22222
OK
127.0.0.1:6380> lrange llist01 0 6
1) "66"
2) "55"
3) "22222"
4) "33"
5) "22"
127.0.0.1:6380> lpop llist01
"66"
127.0.0.1:6380>  lrange llist01 0 6
1) "55"
2) "22222"
3) "33"
4) "22"
127.0.0.1:6380>
127.0.0.1:6380> llen llist01
(integer) 4
127.0.0.1:6380> rpop llist01
"22"
127.0.0.1:6380>  lrange llist01 0 6
1) "55"
2) "22222"
3) "33"
127.0.0.1:6380> linsert llist01 before 33 44
(integer) 4
127.0.0.1:6380>  lrange llist01 0 6
1) "55"
2) "22222"
3) "44"
4) "33"
127.0.0.1:6380>
127.0.0.1:6380> lrem llist01 2 22222
(integer) 1
127.0.0.1:6380>  lrange llist01 0 6
1) "55"
2) "44"
3) "33"
127.0.0.1:6380>
++++++++++++++++++++++++++++++++++++++++++
127.0.0.1:6380> hset hasht01 name x1
(integer) 1
127.0.0.1:6380> hlen hasht01
(integer) 1
127.0.0.1:6380> hkeys hasht01
1) "name"
127.0.0.1:6380>
127.0.0.1:6380> hdel hasht01 name
```

```
(integer) 1
127.0.0.1:6380> hvals hasht01
(empty list or set)
127.0.0.1:6380>
127.0.0.1:6380> hset hasht01 name xxxx
(integer) 1
127.0.0.1:6380> hvals hasht01
1) "xxxx"
127.0.0.1:6380>
127.0.0.1:6380> hexists hastt01 name
(integer) 1
127.0.0.1:6380> hgetall hastt01
1) "name"
2) "x3"
127.0.0.1:6380>
127.0.0.1:6380> hmset h01 f1 v1 f2 v2
OK
127.0.0.1:6380> hkeys h01
1) "f1"
2) "f2"
127.0.0.1:6380> hlen h01
(integer) 2
127.0.0.1:6380>
127.0.0.1:6380> hscan h01 0 count 1
1) "0"
2) 1) "f1"
   2) "v1"
   3) "f2"
   4) "v2"
127.0.0.1:6380>
++++++++++++++++++++++++++++++++++++++++++++
127.0.0.1:6380> del llist01
(integer) 1
127.0.0.1:6380>  lrange llist01 0 6
(empty list or set)
127.0.0.1:6380>
127.0.0.1:6380> set key01 test01
OK
127.0.0.1:6380> get key01
"test01"
127.0.0.1:6380> expire key01 3
(integer) 1
127.0.0.1:6380> get key01
"test01"
127.0.0.1:6380> get key01
"test01"
127.0.0.1:6380> get key01
(nil)
127.0.0.1:6380>
127.0.0.1:6380> set key01 111
OK
127.0.0.1:6380> keys k*
1) "key01"
127.0.0.1:6380> get key01
"111"
127.0.0.1:6380>
++++++++++++++++++++++++++++++++++++++++++++
```

2.举例说明list和hash的应用场景，每个至少一个场景，比如：通过list实现秒杀的请求排队

list 和 hash一起使用的应用场景：



秒杀请求：

使用Redis，基本上用以下命令：

RPUSH key value

插入秒杀请求

当插入的秒杀请求数达到上限时，停止所有后续插入。

后台启动多个工作线程，使用

LPOP key

读取秒杀成功者的用户id，进行后续处理。

或者使用LRANGE key start end命令读取秒杀成功者的用户id，进行后续处理。

每完成一条秒杀记录的处理，就执行INCR key_num。一旦所有库存处理完毕，就结束该商品的本次秒杀，关闭工作线程，也不再接收秒杀请求。