**一、引言**  
  
    今天正式开始了Redis的学习，如果要想学好Redis，必须先学好Redis的数据类型。Redis为什么会比以前的Memchaed等内存缓存软件使用的更频繁，适用范围更广呢？就是因为Redis使用起来更方便，之所以方便，是因为Redis支持的数据类型比以前的Memchaed缓存支持数据类型的更多了。Redis有五种基本数据类型，String(字符串)，Hash(哈希),List(链表),Set(集合),ZSet(有序集合)，在这五种基本的数据类型中，String类型是最基础的。为什么说String类型是最基础的，就拿List为例来说，它是以列表的形式组织字符串数据，Set类型是以集合类型来组织字符串数据的。今天就让我们比较全面的来认识一下redis的基本数据类型吧。  
  
**二、NoSQL的介绍**  
  
     NoSQL(NoSQL = Not Only SQL )，意即“不仅仅是SQL”，意为反Sql运动，提倡运用非关系型的数据存储，随着Web2.0网站的兴起，传统的关系型数据库在应付web2.0网站，特别是超大规模和高并发的SNS类型的web2.0纯动态网站已经力不从心，暴露了很多难以克服的问题，而非关系型数据库则由于其本身的特点得到了迅速发展。  
  
    NoSQL是以key-value形式存储，和传统的关系型数据库不一样，不一定遵循传统数据库的一些基本要求，比如：遵循Sql标准，ACID属性、表结构特性等等。分关系型数据库的特点如下：非关系型的、分布式的、开源的、水平可以扩展的。  
  
    **NoSQL数据库的四大分类**  
  
    1、键值(Key-Value)存储数据库：该类数据库主要会使用到一个哈希表，这个表中有一个特定的键和一个指针指向特定的数据。例如：Tokyo Cabinet/Tyrant, Redis, Voldemort, Oracle BDB.  
  
    2、列存储数据库：该库通常是用来应对分布式存储的海量数据。键仍然存在，但是它们的特点是指向了多个列。如：Cassandra, HBase, Riak.  
  
   3、文档型数据库：该类型的数据模型是版本化的文档，半结构化的文档以特定的格式存储，比如JSON。文档型数据库可以看作是键值数据库的升级版，允许之间嵌套键值，而且文档型数据库比键值数据库的查询效率更高。如：CouchDB, MongoDb. 国内（SequoiaDB，已经开源）。  
  
   4、图形(Graph)数据库：图形结构的数据库同其他行列以及刚性结构的SQL数据库不同，它是使用灵活的图形模型，并且能够扩展到多个服务器上。  
  
    **NoSQL使用场景：**  
  
**1、对数据高并发的读写，可以使用NoSQL类型数据库。**  
  
**2、对海量数据的高效率的存储和访问，可以使用NoSQL类型数据库。**  
  
**3、对数据的高可扩展性和高可用性要求比较高的也可以使用NoSQL类型数据库。**  
  
**三、Redis的介绍**  
  
      redis是一个key-value存储系统。它支持存储的value类型相对更多，包括string(字符串)、list(链表)、set(集合)、zset(sorted set --有序集合)和hash（哈希类型）。这些数据类型都支持push/pop、add/remove及取交集并集和差集及更丰富的操作，而且这些操作都是原子性的。在此基础上，redis支持各种不同方式的排序。为了保证效率，数据都是缓存在内存中。为了可靠性，redis会周期性的把更新的数据写入磁盘或者把修改操作写入追加的记录文件，并且在此基础上实现了master-slave(主从)同步。  
  
    redis的优点如下：  
  
    (1) 速度快，因为数据存在内存中，类似于HashMap，HashMap的优势就是查找和操作的时间复杂度都是O(1)  
  
    (2) 支持丰富数据类型，支持string，list，set，sorted set，hash  
  
    (3) 支持事务，操作都是原子性，所谓的原子性就是对数据的更改要么全部执行，要么全部不执行  
  
    (4) 丰富的特性：可用于缓存，消息，按key设置过期时间，过期后将会自动删除  
  
    有了这些优点，才会有这么多的人使用它。  
     
**四、Redis的数据类型**  
  
      redis的数据类型是学好redis的基础，所以要想很好的掌握redis，必须把它的数据类型掌握好，并且熟练记忆各个类型的基本操作。这里面针对每种数据类型的操作不是全部都涉及到，只是会涉及到一些常用的命令，如果大家想观看全部的命令，可以去官网上查看，地址如下：https://redis.io/commands。还有一点需要说明一下，有些例子，可能几个命令都是相关联的，比如说第一个命令执行了增加操作，我在介绍第二个命令的时候有时候会使用第一个命令的结果。  
  
**1、String字符串类型使用**  
  
         1.1、redis一共分为五种基本数据类型：**String（字符串），**Hash（哈希）、List（链表），Set（集合），ZSet（有序集合）  
  
         1.2、String类型是包含很多种类型的特殊类型，并且是二进制安全的。比如：序列化的对象可以以String类型进行存储，也可以把一张图片的二进制数据以String类型进行存储，当然也可以存储简单的字符串数据，也可以以String类型存储数字。字符串最大的容量是512M，也就是2的32次方-1。  
  
         1.3、常用命令如下：  
  
            1.3.1】、**语法：set key value [EX seconds] [PX milliseconds] [NX|XX]**   设置成功OK,如果针对同一个key设置，会覆盖原来的值。  
  
               1】、192.168.127.128:6379>set name HuangFeiHong 可以针对同一个key重复设置值  
                       OK  
  
               2】、**EX:（expired过期时间，单位秒）,设置成功返回OK**。查看过期时间ttl key,组合命令：setex

192.168.127.128:6379>set age 18 ex 10

OK

192.168.127.128:6379>ttl age 返回0过期，-2已经删除该标志

(integer)8

            3】、**PX:（过期时间，单位毫秒），设置成功返回OK，查看过期时间pttl**

[IMG_256](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>set sex nan px 5000

OK

192.168.127.128:6379>pttl sex 返回0为过期，-2已经删除标志

(integer)1125

192.168.127.128:6379>pttl sex

(integer)-2

[IMG_257](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

            4】、**NX(not exists)：如果指定的Key不存在，就设置该值,组合命令：sexnx**

[IMG_258](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>set name FangShiYu nx

(nil)

192.168.127.128:6379>set name1 FangShiYu nx

OK

设置失败，因为为name 的key 已经存在了，设置成功返回OK，设置失败返回nil

[IMG_259](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

            5】、**XX:如果key存在就设置该值**

[IMG_260](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>set address hebeisheng xx(该值不存在)

(nil)

192.168.127.128:6379>set address hebeisheng xx(该值存在在)

OK

设置失败，因为key不存在，返回nil，存在才可以设置，成功返回OK。

[IMG_261](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

        1.3.2】、**语法：get key**，获取指定key的值，有值返回值，无值返回nil。

192.168.127.128:6379>get name

"HuangFeiHong"

        1.3.3】、**语法：del key [key...]，**删除多个key的值，删除成功返回删除的个数，删除失败返回0

192.168.127.128:6379>del name

（integer）1

        1.3.4】、**语法：keys Pattern，**查看有什么key

192.168.127.128:6379>keys \*

1)"name"

2)"name1"

        1.3.5】、**语法：dbsize**，可以查看key的数量

192.168.127.128:6379>dbsize

(integer) 1

        1.3.6】、**语法：setrange key offset value，**从第几位开始替换成我们想要得值，第几位就是offset的值。替换字符串，最后返回整个字符串的长度。

192.168.127.128:6379>set email 1234567890@com

OK

192.168.127.128:6379>setrange email 10 #

(integer)13

        1.3.7】、**语法：mset key value [key value ...]**，设置成功返回OK，重复设置会覆盖

192.168.127.128:6379>mset name zhangsan sex nan address hebei

OK

        1.3.8】、**语法：mget key [key ...]，**有值得返回值，无值的返回nil

192.168.127.128:6379>mget name sex address

"ZhangSan"

(nil)

        1.3.9】、**语法：msetnx key value [key value ...]，**设置成功会返回成功的个数，失败返回0

192.168.127.128:6379>msetnx school daxue postnum 1234567

(integer) 1 或者 （integer） 0

       1.3.10】、**语法：getset key value，**支持并发，是原子操作的，先返回该key的原始值，然后用新值替换原来的值。

[IMG_262](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>get name

"zhangsan"

192.168.127.128:6379>getset name lisi

"zhangsan"

192.168.127.128:6379>get name

"lisi"

[IMG_263](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       1.3.11】、**语法：incr key**，该key的值必须是数值类型字符串表示，该函数对key的值加一

[IMG_264](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>set age 18

OK

192.168.127.128:6379>get age

"18"

192.168.127.128:6379>incr age

(integer)19

[IMG_265](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       1.3.12】、**语法：decr key**，该key的值必须是数值型字符串表示，该函数对key的值减一

[IMG_266](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>set age 18

OK

192.168.127.128:6379>get age

"18"

192.168.127.128:6379>decr age

(integer)17

[IMG_267](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       1.3.13】、**语法：incrby key increment**，该key的值必须是数值型字符串表示，该函数对key的值增加指定的数值（正数情况）或者减少指定数值（负数情况）

[IMG_268](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>set age 18

OK

192.168.127.128:6379>get age

"18"

192.168.127.128:6379>incrby age 3(增加3)-3（减少3）

(integer)21

[IMG_269](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       1.3.14】、**语法：decrby key decrement**，该key的值必须是数值型字符串表示，该函数对key的值增加指定的数值（负数情况）或者减少指定数值（正数情况）

[IMG_270](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>set age 18

OK

192.168.127.128:6379>get age

"18"

192.168.127.128:6379>decrby age 3(减少3)-3（增加3）

(integer)15

[IMG_271](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       1.3.15】、**语法：incrbyfloat key increment**，该key的值必须是数值型字符串表示，如果想要给key的值增加或者减少有小数的情况，可以使用该命令，正小数（增加），负小数（减少）

[IMG_272](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>set age 18

OK

192.168.127.128:6379>get age

"18"

192.168.127.128:6379>decrby age 3.5(减少3)-3.5（增加3）

"21.5"

[IMG_273](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       1.3.16】、**语法：append key value**，向指定的key追加内容，不会覆盖原始的内容。返回追加后字符串的总长度

192.168.127.128:6379>append name zhangsan（该key不存在会有set的效果，如果存在就追加）

（integer）3

       1.3.17】、**语法：strlen key**，获取指定key值的长度

192.168.127.128:6379>strlen name

(integer)10

       1.3.18】、**语法：type key**，返回当前key的值的类型。

192.168.127.128:6379>type age

string

        以上就是Redis的String类型的基本操作，但是这里没有包含所有的命令，只是一些经常用到的命令，string类型是redis基本数据类型中最基本，最重要的类型，其他的类型都是基于此的，所以该命令大家要熟练使用。  
    
**2、Hash哈希类型的使用**  
  
       2.1、redis一共分为五种基本数据类型：String（字符串）,**Hash（哈希）**、List（链表），Set（集合），ZSet（有序集合）  
  
      2.2、Hash类型是String类型的field和value的映射表，或者说是一个string 集合，它特别适合存储对象，相比较而言，将一个对象类型存储在Hash类型里比存储在string类型占用的内存空间更少，并方便对象的存取。Hash类型的field字段必须是字符串类型，不能是其他数据类型，换句话说Hash类型不能嵌套其他数据类型。一个Hash类型可以包含的key（其实是field）是2的32次方减去1个，也就是说40多亿个key（这里的key也是指field）都可以被支持，所以说Hash可以存储很大的数据量。  
  
      2.3、常用命令如下：  
  
        2.3.1】、**语法：hset key field value**，向键值为key的里面增加field和value的映射数据。增加成功返回1，如果重复增加，会覆盖，但是返回值是0。

[IMG_274](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>hset key1 name zhangsan

(integer)

192.168.127.128:6379>hset key1 age 18

(integer)1

192.168.127.128:6379>hset key1 age 18

(integer)0

[IMG_275](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

        2.3.2】、**语法：hget key field**，根据指定的key和field获取其对应的值。如果有值就返回其值，如果没有返回值为nil。

192.168.127.128:6379>hget key1 name

"zhangsan"

192.168.127.128:6379>hget key1 address (没有该field值)

（nil）

        2.3.3】、**语法：hmset key field value [field value ...]**，可以一次设置多个值。设置成功返回OK。如果重复设置会覆盖原来的值。

192.168.127.128:6379>hmset key name zhangsan sex nan age 18

OK

        2.3.4】、**语法：hmget key field [field ...]**，可以一次性获取指定key的多个field的值，有值的field会返回其值，没有值的返回nil。

192.168.127.128:6379>hmget key name sex age address(地址属性没定义，会返回nil)

"zhangsan"

"nan"

"18"

(nil)

        2.3.5】、**语法：hsetnx key field value**，如果指定key的field的值不存在，则增加该值。增加成功返回1，增加失败返回0.

192.168.127.128:6379>hsetnx key name li

(integer)0

192.168.127.128:6379>hsetnx key name1 li(该key的field的值不存在，则增加成功)

(integer)1

        2.3.6】、**语法：hincrby key field increment**，该value的值必须是数字类型的字符串，否则有错。让指定key的指定field的值增加（正值）或者减少（负值）多少。(ERR hash value is not an integer)

192.168.127.128:6379>hincrby key age 5

(integer)25

192.168.127.128:6379>hincrby key age -5

(integer)20

        2.3.7】、**语法：hincrbyfloat key field increment，**该value的值必须是数字类型的字符串，否则有错。让指定key的指定field的值增加（正值）或者减少（负值）多少小数。(ERR hash value is not an integer)

192.168.127.128:6379>hincrbyfloat key age 0.54

"20.54"

        2.3.8】、**语法：hexists key field**，判断指定key的field字段是否有值，有值返回1，无值返回0.

192.168.127.128:6379>hexists key age

(integer)1

192.168.127.128:6379>hexists key age1

(integer)0

        2.3.9】、**语法：hlen key**，判断指定key下有几个field。有值返回就是几个，没有纸就返回0。

192.168.127.128:6379>hlen key

(integer)4

192.168.127.128:6379>hlen key1

(integer)0

       2.3.10】、**语法：hdel key field**，删除指定key的field键值对，删除成功返回1，删除失败返回0.

192.168.127.128:6379>hdel key name1

(integer)1

192.168.127.128:6379>hdel key name1

(integer)0

       2.3.11】、**语法：hkeys key**，列出指定key的field字段值。

192.168.127.128:6379>hkeys key

"name"

"sex"

"age"

       2.3.12】、**语法：hvals key**，列出指定key的field的所对应的值，也就是value。

192.168.127.128:6379>hvals key

"zhangsan"

"nan"

"20.54"

       2.3.13】、**语法：hgetall key**，列出指定key的field和value的映射值。

[IMG_276](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>hgetall key

"name"

"zhangsan"

"sex"

"nan"

"age"

"20.54"

[IMG_277](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

**3、List列表类型的使用**  
  
     3.1、redis一共分为五种基本数据类型：String（字符串）,Hash（哈希）、**List（链表）**，Set（集合），ZSet（有序集合）  
  
     3.2、List类型是一个链表结构的集合，并且是有序的集合，其值是可以重复的，其主要功能操作有push，pop、获取元素等。更详细的说，List类型是一个双端链表结构，我们可以通过相关操作在集合的头部和尾部增加或者删除元素。List类型设计的非常巧妙，既可以做【栈-先进后出】使用，也可以做【队列-先进先出】使用，能满足绝大多数的需求。List类型有点类似数组的概念，所以具有下标，可以针对指定下标进行操作，比如替换。  
  
     3.3、常用命令如下：  
  
        3.3.1】、**语法：lpush key value [value ...]** ，向指定列表key从左面增加元素。增加成功，返回当前列表元素的个数。这个显示的方式是堆栈的形式，先进后出。

[IMG_278](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>lpush list1 hello

(integer)1

192.168.127.128:6379>lpush list1 world

(integer)2

192.168.127.128:6379>lpush list1 world（可以重复增加）

(integer)3

[IMG_279](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

        3.3.2】、**语法：rpush key value [value ...]**，向指定的列表key从右面增加元素。增加成功，返回当前列表元素的个数。这个显示的方式是队列的形式，先进先出。

[IMG_280](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>rpush list2 hello

(integer)1

192.168.127.128:6379>rpush list2 world

(integer)2

192.168.127.128:6379>rpush list2 !!!

(integer)3

先进先出的显示方式：

192.168.127.128:6379>lrange list1 0 -1

"hello"

"world"

"!!!"

[IMG_281](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

        3.3.3】、**语法：linsert key BEFORE|AFTER pivot value**，向指定列表key在指定元素【前|后】插入元素，成功返回列表当前元素的个数。

[IMG_282](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>linsert list2 before world my

(integer)4

192.168.127.128:6379>lrange list2 0 -1

"hello"

"my"

"world"

"!!!"

[IMG_283](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

        3.3.4】、**语法：lrange key start stop**，将制定列表key的元素按指定范围显示出来。 因为list是双端列表，如果计数是从左到右开始，那就是从0开始，依次是1,2,3等，如果计数是从右到左开始，那就是从-1开始，依次为-2，-3等。

192.168.127.128:6379>lrange list1 0 -1 这个意思就是从左边第一位到右面最后一位，全部显示

"world"

"world"

"hello"

        3.3.5】、**语法：lset key index（列表下标） value**，将指定下标的元素替换掉，替换成功返回OK,可以重复操作，会替换以前的值。

[IMG_284](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>lset list2 3 !!，将hello my world !!!里面的三个！替换成2个！

OK

192.168.127.128:6379>lrange list2 0 -1

"hello"

"my"

"world"

"!!"

[IMG_285](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

        3.3.6】、**语法：lrem key count value**，删除和指定元素想相等的元素并返回删除的个数，删除正确返回个数，删除错误返回0。count，删除个数，如果count是正数，从列表表头开始搜索，如果count是负数，从列表表尾开始搜索，删除的个数就是count的绝对值。

[IMG_286](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

当前值：

192.168.127.128:6379>lrange list2 0 -1

"hello"

"my"

"world"

"!!"

192.168.127.128:6379>lrem list2 1 my

(integer)1

192.168.127.128:6379>lrange list2 0 -1

"hello"

"world"

"!!"

192.168.127.128:6379>lrem list2 1 my（已经没有my这个值了）

(integer)0

[IMG_287](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

        3.3.7】、**语法：ltrim key start(开始值) stop（结束值）**，保留指定key的索引值范围内的数据，其他数据删除，开始从0开始。修剪成功返回OK,可以重复执行。

[IMG_288](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

当前值：

192.168.127.128:6379>lrange list2 0 -1

"hello"

"world"

"!!"

192.168.127.128:6379>ltrim list2 0 1

(integer)1

192.168.127.128:6379>lrange list2 0 -1

"hello"

"world"

[IMG_289](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

        3.3.8】、**语法：lpop key**，从list的头部删除元素，并返回删除元素

192.168.127.128:6379>rpush list3 a b c d e f

(integer)6

192.168.127.128:6379>lpop list3

"a"

        3.3.9】、**语法：rpop key**，从list的尾部删除元素，并返回删除元素。

192.168.127.128:6379>rpop list3

"f"

       3.3.10】、**语法：rpoplpush source destination**，第一步从指定list列表的尾部删除元素，第二步，从指定list列表头部加入元素。

192.168.127.128:6379>rpoplpush list3 list4

"e"

192.168.127.128:6379>lrange list4 0 -1

1）"e"

       3.3.11】、**语法：lindex key index**，返回指定list列表中index为止的元素。

192.168.127.128:6379>lindex list2 0

"hello"

       3.3.12】、**语法：llen key**，返回指定key列表的元素的个数。

192.168.127.128:6379>llen list2

(integer)2

        这里，我要说明一下，这里面的所有代码，上下是有关系的，这个大家要知道一下，否则不知道为什么会有这样的结果。  
  
**4、Set集合类型的使用**  
  
     4.1、redis一共分为五种基本数据类型：String（字符串）,Hash（哈希）、List（链表），**Set（集合）**，ZSet（有序集合）  
  
     4.2、set集合是string类型的无需集合，集合里面的元素不能重复，或者说必须是唯一的。set内部是通过Hashtable来实现的，并且这个Hashtable的value是没有值的，既然是集合，因此，我们可以对集合实施交集（Inter--InterStore）、并集（Union--UnionStore）和差集（Diff--DiffStore）的操作。操作复杂度都是：O(1)  
  
     4.3、常用命令如下：  
  
       4.3.1】、**语法：sadd key member [member ...]，**可以向名称为key的set集合中添加元素，一次可以添加多个，也可以添加一个，以空格为分隔符，并且增加的元素不能有重复。成功返回添加的元素的个数。

192.168.127.128:6379>sadd set1 aaa bbb ccc ddd eee

(integer)5

192.168.127.128:6379>sadd set1 eee

(integer)1

       4.3.2】、**语法：smembers key，**我们可以通过该方法查看名称为key集合的元素

[IMG_290](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>smembers set1

1)"bbb"

2)"ddd"

3)"ccc"

4)"aaa"

5)"fff"

6)"eee"

[IMG_291](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       4.3.3】、**语法：srem key member [member ...]，**删除set集合元素，可以删除一个，也可以删除多个。成功删除返回删除个数，失败返回0。

192.168.127.128:6379>srem set1 aaa bbb ccc ddd eee fff

(integer)6

       4.3.4】、**语法：spop key [count]，**随机返回删除的key，可以返回【count】个，返回的元素会被删除。

原始值：aaa bbb ccc ddd eee fff

192.168.127.128:6379>spop set1 3

1）"aaa"

2）"ddd"

3）"ccc"

       4.3.5】、**语法：sinter key [key ...]**，取两个或者多个集合的交集。

[IMG_292](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 aaa bbb ccc ddd

set2 bbb ccc ddd eee

192.168.127.128:6379>sinter set1 set2

1）"bbb"

2）"ddd"

3）"ccc"

[IMG_293](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       4.3.6】、**语法：sinterstore destination key [key ...]**，取两个或者多个集合的交集并存到指定集合里。

[IMG_294](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 aaa bbb ccc ddd

set2 bbb ccc ddd eee

192.168.127.128:6379>sinterstore setinter set1 set2

（integer）3

192.168.127.128:6379>smembers setinter

1)"bbb"

1)"ccc"

1)"ddd"

[IMG_295](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       4.3.7】、**语法：sunion key [key ...]**，取两个或者多个集合的并集。如果有重复元素，会去掉重复元素然后在显示。

[IMG_296](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 aaa bbb ccc ddd

set2 bbb ccc ddd eee

192.168.127.128:6379>sunion set1 set2

1）"aaa"

2）"bbb"

3）"eee"

4）"ddd"

5）"ccc"

[IMG_297](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       4.3.8】、**语法：sunionstore destination key [key ...]，**取两个或者多个集合的并集并存储到指定集合里。

[IMG_298](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 aaa bbb ccc ddd

set2 bbb ccc ddd eee

192.168.127.128:6379>sunionstore setunion set1 set2

（integer）5

192.168.127.128:6379>smembers setunion

1）"aaa"

2）"bbb"

3）"eee"

4）"ddd"

5）"ccc"

[IMG_299](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       4.3.9】、**语法：sdiff key [key ...]**，取两个或者多个集合不同的元素，第一个参数集合为参考对象，由于参考对象不同，结果可能也不同。

[IMG_300](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 aaa bbb ccc ddd

set2 bbb ccc ddd eee

192.168.127.128:6379>sdiff set1 set2

1）"aaa"

[IMG_301](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       4.3.10】、**语法：sdiffstore destination key [key ...]**，取两个或者多个集合不同的元素，并把结果存储到指定集合里。

[IMG_302](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 aaa bbb ccc ddd

set2 bbb ccc ddd eee

192.168.127.128:6379>sdiffstore setdiff set1 set2

(integer)1

192.168.127.128:6379>smembers setdiff

1)"aaa"

[IMG_303](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       4.3.11】、**语法：smove source destination member**，把一个元素member从一个set集合移动到另一个set集合。

[IMG_304](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 aaa bbb ccc ddd

set2 bbb ccc ddd eee

192.168.127.128:6379>smove set1 set2 aaa

(integer)1

192.168.127.128:6379>smembers set2

1)"eee"

2)"bbb"

3)"ddd"

4)"ccc"

5)"aaa" "移动过来的"

[IMG_305](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       4.3.12】、**语法：scard key，**查看集合里元素的个数。

原始值：

set1 aaa bbb ccc ddd

192.168.127.128:6379>scard set1

(integer)4

       4.3.13】、**语法：sismember key member**，判断某个元素是否在指定的集合里面，存在返回1，不存在返回0

原始值：

set1 aaa bbb ccc ddd

192.168.127.128:6379>sismember set1 aaa

(integer)1

       4.3.14】、**语法：srandmember key [count]**，随机返回一个或者几个元素，不会删除元素，默认返回一个元素。每次执行都会返回不同的结果。

[IMG_306](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 aaa bbb ccc ddd

192.168.127.128:6379>srandmember set1 3

1）"bbb"

1）"ddd"

1）"ccc"

[IMG_307](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

**5、ZSet有序集合类型的使用**  
  
     5.1、redis一共分为五种基本数据类型：String（字符串），Hash（哈希）、List（链表），Set（集合），**ZSet（有序集合）**  
  
     5.2、ZSet数据类型，是有序的集合类型，集合里面的元素必须保证唯一，不会出现重复的元素。无论以什么方式访问有序集合都很快，因为它内部是通过哈希表和跳跃表来实现的。默认排序是由小到大。  
  
     5.3、常用命令如下：  
  
       5.3.1】、**语法：zadd key [NX|XX] [CH] [INCR] score member [score member ...]，**向有序集合中添加一个h或者多个元素，该元素如果存在，则更新顺序。如果在重复插入的时候，会根据顺序属性更新。

[IMG_308](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

192.168.127.128:6379>zadd set1 7 aaa 5 bbb 10 ccc

（integer）3

如果想要修改某元素的分数，例如把aaa的分数改成10，可以如下：

zadd set1 10 aaa

当前返回值为0，但是分数已经成功修改。默认返回值是增加到有序集合里面元素的个数。

nx参数：如果该元素不存在则增加该元素

xx参数：如果该元素存在则修改该元素

ch参数:可以让方法返回值是更改分数的元素的个数

incr参数:可以对指定元素的分数进行增加或者减少指定数值，正数增加，负数减少

zadd set1 [nx|xx|ch|incr] 10 aaa

[IMG_309](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       5.3.2】、**语法：zrange key start stop [WITHSCORES]，**查看有序集合的元素，分数可以是选项，可以显示，也可以不显示。

[IMG_310](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zrange set1 0 -1 [withscores]

1）"bbb"

2）"5"

3）"aaa"

4）"7"

5）"ccc"

6）"10"

[IMG_311](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       5.3.3】、**语法：zrevrange key start stop [WITHSCORES]**，按分数从大到小的排序集合元素，默认是小到大排序。

[IMG_312](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zrevrange set1 0 -1 [withscores]

1）"ccc"

2）"10"

3）"aaa"

4）"7"

5）"bbb"

6）"5"

[IMG_313](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       5.3.4】、**语法：zrem key member [member ...]**，删除名称为key的zset中的元素member，可以删除一个，也可以删除多个。成功返回删除个数，失败返回0。

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zrem set1 aaa bbb

(integer)2

       5.3.5】、**语法：zincrby key increment member**，以指定值去自动递增或者递减少指定元素的分数，用法和之前的incryby类似

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zincrby set1 3 ccc

“13”

       5.3.6】、**语法：zrangebyscore key min max [WITHSCORES]**，查找分数在指定区间范围的元素，返回找到的个数。

[IMG_314](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zrangebyscore set1 7（分数最小值） 10（分数最大值） withscores

1）"aaa"

1）"7"

1）"ccc"

1）"10"

[IMG_315](http://www.cnblogs.com/PatrickLiu/p/javascript:void(0);)

       5.3.7】、**语法：zremrangebyrank key start stop**，根据索引值删除指定的元素，返回删除的个数，没有数据返回0.

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zremrangebyrank set1 0(索引值) 1(索引值)

(integer)2

       5.3.8】、**语法：zremrangebyscore key min max**根据指定的分数删除元素，返回删除的个数，没有数据返回0.

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zremrangebysocre set1 5(分数最小值) 6（分数最大值）

(integer)1

       5.3.9】、**语法：zrank key member**，（从小到大）返回排序索引从小到大排序的某有序集合里面某元素的索引，无值返回nil，索引从0开始计数。

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zrank set1 aaa

（integer）1

       5.3.9】、**语法：zrevrank key member**，返回排序索引从大到小排序的索引值，也就是说最大的分数的索引是0（降序排序之后再找索引）

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zrevrank set1 ccc

（integer）0

       5.3.10】、**语法：zcard key**，返回集合里所有元素的个数。

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zcard set1

(integer)3

       5.3.11】、**语法：zcount key min max**，返回集合中score在给定区间中的数量

原始值：

set1 7 aaa 5 bbb 10 ccc

192.168.127.128:6379>zcount set1 7(分数最小值) 10（分数最大值）

(integer)2

**五、总结**  
  
      redis的数据类型今天就写到这里了，学了更多的东西在更新吧。每一步走的都挺艰难的，革命还没成功，我辈还需努力啊。现在对redis的认识和以前不一样了，整体轮廓有了。下一步要掌握集群的高可用性和可扩展方面的知识点了，继续上路吧