Redis数据库的数据类型

1. **Redis中主要是5种数据类型：**

**string 、hash、list、set、zset。**

1. **String类型及操作：**

**string最简单的类型，键值对中，一个key对应一个value，string类型是二进制安全的。Redis中的string可以包含任何数据，比如jpg图片、序列化的对象等等。**

（1）设置key对应的值为**string类型的value：**

添加一个 key1=zhaohong的键值对：利用set。

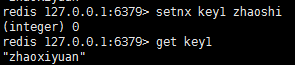


修改此key1对应的value：（**一个键只能对应一个value**）



（2） **setnx： nx 表示not exist，不存在。即只有key不存在时，才会成功。**

**如果key已经存在，则返回0，且修改不成功，如果不存在，设置成功，返回1。**



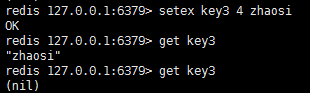


（3）如果没有这个键，直接get，则会返回nil：



（4） **setex ： 设置键值对的生存时间。**

**使用方法 setex key time value**

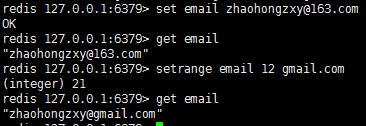


**3秒后，就消失了。**

（5） **setrange** ： 设置指定key的value 的子字符串，相当于修改value值。

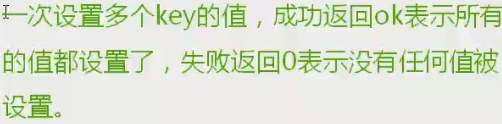
**setrange key index subValue #index从0开始**

**返回的是修改后value的字符个数。**

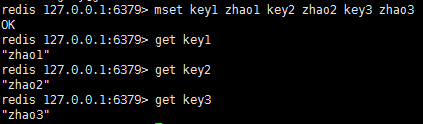


**注意**：如果替代的字符串较短，不能把原来的字符串覆盖掉。如：

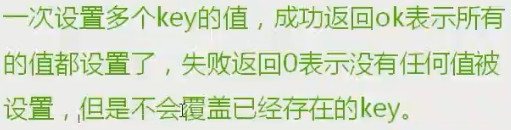
**（6）mset： 一次设置多个值。**

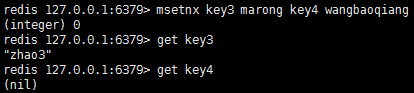


**mset key1 value1 key2 value2 key3 value3 ….**



**msetnx : 返回0，则都不成功。**



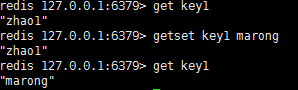


1. （1）**get**： 获取key对应的value值，如果key不存在，则返回nil。

get key1

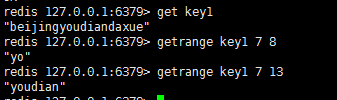
（2）**getset** ： 设置key的值，并返回key的旧值。获取旧值，并设置新值。

**getset key newvalue**



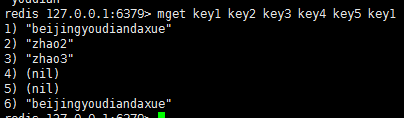
（3）getrange: 获取key对应的value值得子字符串。

**getrange key fromindex endindex**



（4） **mget** ：一次获取多个key的值，如果对应的key不存在，则返回nil。

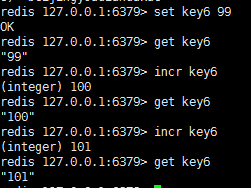
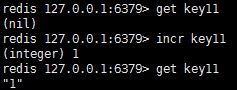
mget key1 key2 key3 ….



1. **incr** : 对**key的值**做加加操作，并返回新的值。

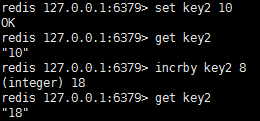
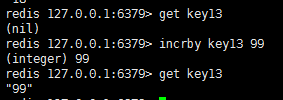
incr key

如果key本身不存在，则默认为原来的值 为0，并子增1.

**incrby** ： 同incr类似，加指定值num，key不存在时，会设置key，并默认为原来的值为0，在0基础上加num。

incry key num #num可以为负值。

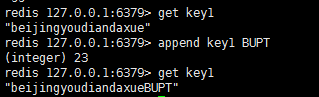
1. 类似的，有decr和decrby。

decr key： 对key对应的值做减操作。

decrby key num：减去设定的值num。

1. append ：给key对应的字符串添加子字符串，返回新字符串的长度。

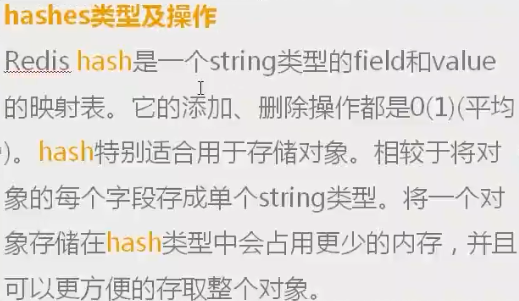
append key subvalue

****

1. **strlen ： 查看key对应的value的长度。 strlen key**



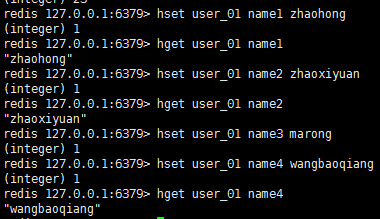
1. 第二种数据类型：hash：



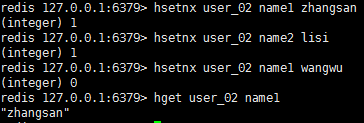
1. **关于set的方法：**

（1）hset : 设置hash field 为指定值，如果key不存在，则先创建。设置成功则返回1，否则返回0。

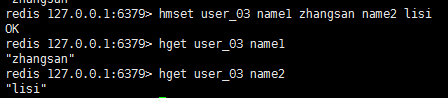
hset hashtablename field1 value



（2）**hsetnx ： 设置成功返回1，否则返回0.**



（3）hmset ： 批处理，同时设置多个。



注意没有 hmsetnx这个命令。

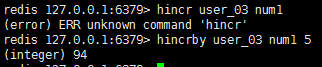
1. 关于get的方法：
2. hget ： 获取key对应的值。



1. hmget ： 获取多个key对应的值。

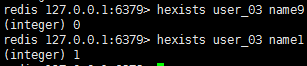


1. hincrby ：增量。对hash上某个field对应的value增加一定值。返回增量后的value值。



没有hincr方法。

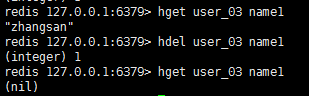
1. hexists： 判断hash中某个field是否存在。存在则返回1，否则返回0.



1. hlen ： 获取hash中几个field，返回个数。

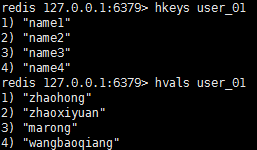


1. hdel ： 删除hash中指定的field。删除成功返回1，否则返回0.

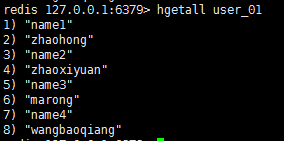


1. hkeys：返回hash中所有的key。

hvals：返回hash中所有的value。



1. hgetall：获取某个hash中全部的field和value。



1. 注意：hash中没有 hmsetnx 、hsetex、hsetrange、hincr等方法。