

牛客大学高薪加成系列课

Redis对象的底层实现



1. Redis数据库中的键值对由对象来表示，其中键是一个对象，值是另一个对象；
2. Redis设计了redisObject结构来表示一个对象，该结构的源代码如下：

```
// redis.h, server.h
typedef struct redisObject {
    unsigned type:4;           // 对象类型
    unsigned encoding:4;       // 对象编码
    unsigned lru:LRU_BITS;     // 访问时间
    int refcount;               // 引用计数
    void *ptr;                  // 指向底层数据结构
} robj;
```

类型常量	类型取值	类型含义	TYPE命令的输出
OBJ_STRING	0	字符串	string
OBJ_LIST	1	列表	list
OBJ_SET	2	集合	set
OBJ_ZSET	3	有序集合	zset
OBJ_HASH	4	哈希	hash

编码常量	编码取值	数据结构	OBJECT ENCODING命令的输出
OBJ_ENCODING_RAW	0	SDS	raw
OBJ_ENCODING_INT	1	long类型整数	int
OBJ_ENCODING_HT	2	字典	hashtable
OBJ_ENCODING_ZIPMAP	3		
OBJ_ENCODING_LINKEDLIST	4	双端链表	linkedlist
OBJ_ENCODING_ZIPLIST	5	压缩列表	ziplist
OBJ_ENCODING_INTSET	6	整数集合	intset
OBJ_ENCODING_SKIPLIST	7	跳跃表	skiplist
OBJ_ENCODING_EMBSTR	8	SDS (embstr编码)	embstr
OBJ_ENCODING_QUICKLIST	9	快速列表	quicklist

类型	编码	说明
OBJ_STRING	OBJ_ENCODING_INT	
	OBJ_ENCODING_EMBSTR	
	OBJ_ENCODING_RAW	
OBJ_LIST	OBJ_ENCODING_ZIPLIST	< 3.2
	OBJ_ENCODING_LINKEDLIST	< 3.2
	OBJ_ENCODING_QUICKLIST	>= 3.2
OBJ_SET	OBJ_ENCODING_INTSET	
	OBJ_ENCODING_HT	
OBJ_ZSET	OBJ_ENCODING_ZIPLIST	
	OBJ_ENCODING_SKIPLIST	
OBJ_HASH	OBJ_ENCODING_ZIPLIST	
	OBJ_ENCODING_HT	

1. lru属性用于记录对象最后一次被程序访问的时间，可用于实现缓存淘汰策略；
2. OBJECT IDLETIME命令可以打印出某个键的空闲时间，该时间是由lru计算而来；

OBJECT IDLETIME 命令的实现是特殊的，
通过该命令访问键时，不会修改其lru属性！

`refcount`属性用于记录对象的引用次数：

1. 在创建一个新对象时，引用计数的值会被初始化为1；
2. 当对象被一个新程序使用时，它的引用计数值会加1；
3. 当对象不再被某程序使用时，他的引用计数值会减1；
4. 当对象的引用计数值变为0时，它所占用的内存空间将会被释放；

对象的引用计数，可用于实现对象的内存回收，以及对象共享的功能。

Redis会在初始化服务器时，创建一万个字符串对象，这些对象包含了从0到9999的所有整数值。当服务器需要用到值为0到9999的字符串对象时，就会使用这些共享对象，而不是创建新的对象！



牛客大学

- 专业求职辅导 -

THANKS



关注【牛客大学】公众号
回复“牛客大学”获取更多求职资料