**Redis学习笔记**

**V1.0.0**

**Author: clearuo**

Date: 2016-04-06

## 一 介绍

### 1.1 可视化管理工具

Redis Desktop Manager

### 1.2 简单介绍

Redis是一个开源的，先进的 key-value 存储可用于构建高性能，可扩展的 Web 应用程序的解决方案。Redis官方网网站是：<http://www.redis.io/>

### 1.3 Redis特点

Redis 有三个主要使其有别于其它很多竞争对手的特点：

* Redis是完全在内存中保存数据的数据库，使用磁盘只是为了持久性目的；
* Redis相比许多键值数据存储系统有相对丰富的数据类型；
* Redis可以将数据复制到任意数量的从服务器中；

### 1.4 Redis优点

* **异常快速** : Redis是非常快的，每秒可以执行大约110000设置操作，81000个/每秒的读取操作。
* **支持丰富的数据类型** : Redis支持最大多数开发人员已经知道如列表，集合，可排序集合，哈希等数据类型。这使得在应用中很容易解决的各种问题，因为我们知道哪些问题处理使用哪种数据类型更好解决。
* **操作都是原子的** : 所有 Redis 的操作都是原子，从而确保当两个客户同时访问 Redis 服务器得到的是更新后的值（最新值）。
* **MultiUtility工具**：Redis是一个多功能实用工具，可以在很多如：缓存，消息传递队列中使用（Redis原生支持发布/订阅），在应用程序中，如：Web应用程序会话，网站页面点击数等任何短暂的数据；

### 1.5 Redis环境

要在 Ubuntu 上安装 Redis，打开终端，然后输入以下命令：

|  |
| --- |
| $sudo apt-get update $sudo apt-get install redis-server |

这将在您的计算机上安装Redis

启动 Redis

|  |
| --- |
| $redis-server |

查看 redis 是否还在运行

这将打开一个 Redis 提示符，如下图所示：



在上面的提示信息中：127.0.0.1 是本机的IP地址，6379是 Redis 服务器运行的端口。现在输入 PING 命令，如下图所示：



这说明现在你已经成功地在计算机上安装了 Redis。

### 1.6 Redis配置

在Redis有配置文件(redis.conf)可在Redis的根目录下找到。可以通过Redis的CONFIG命令设置所有Redis的配置。

* 语法

Redis的CONFIG命令的基本语法如下所示：

|  |
| --- |
| redis 127.0.0.1:6379> CONFIG GET CONFIG\_SETTING\_NAME |

* 例如

|  |
| --- |
| redis 127.0.0.1:6379> CONFIG GET loglevel 1) "loglevel" 2) "notice" |

* 编辑配置

要更新配置，可以直接编辑redis.conf文件或更新配置，通过CONFIG set命令

* 语法

CONFIG SET命令的基本语法如下所示：

|  |
| --- |
| redis 127.0.0.1:6379> CONFIG SET CONFIG\_SETTING\_NAME NEW\_CONFIG\_VALUE |

* 例子

|  |
| --- |
| redis 127.0.0.1:6379> CONFIG SET loglevel "notice" OK redis 127.0.0.1:6379> CONFIG GET loglevel 1) "loglevel" 2) "notice" |

## 二 Redis数据类型

Redis支持5种数据类型,它们描述如下:

### 2.1 Strings-字符串

Redis的字符串是字节序列。在Redis中字符串是二进制安全的，这意味着他们有一个已知的长度，是没有任何特殊字符终止决定的，所以可以存储任何东西，最大长度可达512兆。

例子:

|  |
| --- |
| redis 127.0.0.1:6379> SET name "yiibai" OK redis 127.0.0.1:6379> GET name "yiibai" |

在上面的例子使用Redis命令set和get，Redis的名称是yiibai的键存储在Redis的字符串值。

**注**：字符串值可以存储最大512兆字节的长度。

### 2.2 Hashes-哈希值

Redis的哈希键值对的集合。 Redis的哈希值是字符串字段和字符串值之间的映射，所以它们被用来表示对象

例子:

|  |
| --- |
| redis 127.0.0.1:6379> HMSET user:1 username yiibai password yiibai points 200 OK redis 127.0.0.1:6379> HGETALL user:1 1) "username" 2) "yiibai" 3) "password" 4) "yiibai" 5) "points" 6) "200" |

在上面的例子中的哈希数据类型，用于存储包含用户的基本信息用户的对象。这里HMSET，HEXTALL对于Redis 命令 user:1 是键。

每个哈希可存储多达232 - 1个 字段 - 值对(超过四十亿)。

### 2.3 Lists-列表

Redis的列表是简单的字符串列表，排序插入顺序。可以添加元素到Redis列表的头部或尾部。

例子:

|  |
| --- |
| redis 127.0.0.1:6379> lpush tutoriallist redis (integer) 1 redis 127.0.0.1:6379> lpush tutoriallist mongodb (integer) 2 redis 127.0.0.1:6379> lpush tutoriallist rabitmq (integer) 3 redis 127.0.0.1:6379> lrange tutoriallist 0 10 1) "rabitmq" 2) "mongodb" 3) "redis" |

列表中最大长度为232 - 1(4294967295，集合成员超过四十亿)。

### 2.4 Sets-集合

Redis集合是字符串的无序集合。在Redis中可以添加，删除和测试文件是否存在在O(1)的时间复杂度的成员。

例子

|  |
| --- |
| redis 127.0.0.1:6379> sadd tutoriallist redis (integer) 1 redis 127.0.0.1:6379> sadd tutoriallist mongodb (integer) 1 redis 127.0.0.1:6379> sadd tutoriallist rabitmq (integer) 1 redis 127.0.0.1:6379> sadd tutoriallist rabitmq (integer) 0 redis 127.0.0.1:6379> smembers tutoriallist 1) "rabitmq" 2) "mongodb" 3) "redis" |

注意：在上面的例子中rabitmq设置属性加两次，但由于唯一性只加一次。

成员中集最大数量为232 - 1(4294967295，集合成员超过4十亿)。

### 2.5 集合排序

Redis的集合排序类似于Redis集合，字符串不重复的集合。不同的是，一个有序集合的每个成员关联分数，用于以便采取有序set命令，从最小的到最大的分数有关。虽然成员都是独一无二的，分数可能会重复。

|  |
| --- |
| redis 127.0.0.1:6379> zadd tutoriallist 0 redis (integer) 1 redis 127.0.0.1:6379> zadd tutoriallist 0 mongodb (integer) 1 redis 127.0.0.1:6379> zadd tutoriallist 0 rabitmq (integer) 1 redis 127.0.0.1:6379> zadd tutoriallist 0 rabitmq (integer) 0 redis 127.0.0.1:6379> ZRANGEBYSCORE tutoriallist 0 1000  1) "redis" 2) "mongodb" 3) "rabitmq" |