# Redis介绍

## 版本

本文档基于redis 6.0版本

## 官方主页

<https://redis.io/>

<https://github.com/redis/redis>

## Redis是什么

基于内存亦可持久化的日志型、Key-Value数据库，并提供多种语言的API。

redis是一个key-value存储系统。它支持存储的value类型：string(字符串)、list(链表)、set(集合)、zset(sorted set 有序集合)和hash（哈希类型）。

## 为什么要用redis

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)。

支持丰富数据类型，支持string，list，set，sorted set，hash。

支持事务，操作都是原子性，所谓原子性就是对数据的更改要么全部执行，要么全部不执行。

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。

# 安装

## Windows下安装redis

需要下载redis源码，还要安装gcc，make编译环境，自行编译。

这里提供我编译好的redis-6.0.6版本下载地址：

<https://github.com/phh2008/redis-6.0.6-windows-download>

## Centos下安装redis

直接yum 安装的redis 不是最新版本

yum install redis

如果要安装最新的redis，需要安装Remi的软件源，官网地址：http://rpms.famillecollet.com/

yum install -y http://rpms.famillecollet.com/enterprise/remi-release-7.rpm

然后可以使用下面的命令安装最新版本的redis：

yum --enablerepo=remi install redis

安装完毕后，即可使用下面的命令启动redis服务

systemctl start redis

redis安装完毕后，我们来查看下redis安装时创建的相关文件，如下：

rpm -qa |grep redis

# 配置

## 常用配置：

daemonize no #是否以后台进程启动

requirepass foobared # 需要密码

slowlog-log-slower-than #决定要对执行时间大于多少微秒的查询进行记录

slowlog-max-len #它决定 slow log 最多能保存多少条日志

save #在N秒内有M条数据被改变时将被保存到硬盘

appendonly #是否启用aof持久化方式

aof-use-rdb-preamble #混合模式持久化

## 更多配置说明：

可以打开此文件查看更多配置说明

# 数据类型与应用场景

## String（字符串）

string 是 redis 最基本的类型，string 类型是二进制安全的，redis 的 string 可以包含任何数据。比如jpg图片或者序列化的对象。string 类型的值最大能存储 512MB。

常用命令：set、get、decr、incr、mget等。

应用场景：应于类似Memcached key/value 缓存

## Hash（哈希）

Redis hash 是一个键值(key=>value)对集合；是一个 string 类型的 field 和 value 的映射表，hash 特别适合用于存储对象。

每个 hash 可以存储 232 -1 键值对（40多亿）。

常用命令：hget、hset、hgetall等。

应用场景：存储一些结构化的数据，比如用户的昵称、年龄、性别、积分等，存储一个用户信息对象数据

## List（列表）

Redis 列表是简单的字符串列表，按照插入顺序排序。你可以添加一个元素到列表的头部（左边）或者尾部（右边）。

list类型经常会被用于消息队列的服务，以完成多程序之间的消息交换。

常用命令：lpush、rpush、lpop、rpop、lrange等。

列表最多可存储 232 - 1 元素 (4294967295, 每个列表可存储40多亿)。

应用场景：Redis链表经常会被用于消息队列的服务，以完成多程序之间的消息交换。

## Set（集合）

Redis的Set是string类型的无序集合。和列表一样，在执行插入和删除和判断是否存在某元素时，效率是很高的。集合最大的优势在于可以进行交集并集差集操作。Set可包含的最大元素数量是4294967295。

集合是通过哈希表实现的，所以添加，删除，查找的复杂度都是O(1)。

常用命令：sadd、spop、smembers、sunion等。

应用场景：

1、利用交集求共同好友。

2、利用唯一性，可以统计访问网站的所有独立IP。

3、好友推荐的时候根据tag求交集，大于某个threshold（临界值的）就可以推荐。

## zset(sorted set：有序集合)

Redis zset 和 set 一样也是string类型元素的集合,且不允许重复的成员。

不同的是每个元素都会关联一个double类型的分数。redis正是通过分数来为集合中的成员进行从小到大的排序。

zset的成员是唯一的,但分数(score)却可以重复。

sorted set是插入有序的，即自动排序。

常用命令：zadd、zrange、zrem、zcard等。

当你需要一个有序的并且不重复的集合列表时，那么可以选择sorted set数据结构。

应用场景：

（1）例如存储全班同学的成绩，其集合value可以是同学的学号，而score可以是成绩。

（2）排行榜应用，根据得分列出topN的用户等。

## 类型与命令使用

更多使用帮助与示例，请查看此文件：



# Java客户

Redis现有java客户端程序主要有：Jedis，Redisson，Lettuce.

推荐使用Redisson、Lettuce、或Spring-data-redis

## Jedis

Jedis中的方法调用是比较底层的暴露的Redis的API。

Jedis：使用阻塞的I/O，且其方法调用都是同步的，程序流需要等到sockets处理完I/O才能执行，不支持异步。Jedis客户端实例不是线程安全的，所以需要通过连接池来使用Jedis

## Redisson

Redisson实现了分布式和可扩展的Java数据结构，提供很多分布式相关操作服务，例如，分布式锁，分布式集合，可通过Redis支持延迟队列。

Redisson：基于Netty框架的事件驱动的通信层，其方法调用是异步的。Redisson的API是线程安全的，所以可以操作单个Redisson连接来完成各种操作。

Redisson不仅提供了一系列的分布式Java常用对象，基本可以与Java的基本数据结构通用，还提供了许多分布式服务，其中包括（BitSet, Set, Multimap, SortedSet, Map, List, Queue, BlockingQueue, Deque, BlockingDeque, Semaphore, Lock, AtomicLong, CountDownLatch, Publish / Subscribe, Bloom filter, Remote service, Spring cache, Executor service, Live Object service, Scheduler service）。

## Lettuce

Lettuce:高级Redis客户端，用于线程安全同步，异步和响应使用，支持集群，Sentinel，管道和编码器。目前springboot默认使用的客户端。

Lettuce：基于Netty框架的事件驱动的通信层，其方法调用是异步的。Lettuce的API是线程安全的，所以可以操作单个Lettuce连接来完成各种操作。

## Spring-data-redis

在Springboot项目中也可使用Spring-data-redis，底层默认使用Lettuce，对一些操作进行封装。整合Spring-data-redis需要完成以下几步：

添加依赖：

<dependency>

<groupId>**org.springframework.boot**</groupId>

<artifactId>**spring-boot-starter-data-redis**</artifactId>

</dependency>

配置application.yml:

**spring:**

**redis**:

**host**: localhost

**port**: 6379

**password**:

**database**: 2

在代码中注入：RedisTemplate<String, Object> redisTemplate 即可使用。

# Redis6.0特性

## ACL用户权限控制

## RESP3：新的redis通信协议

## Cluster管理工具

## SSL支持

## IO多线程支持

## 客户端缓存