# 初识Redis

Redis是远程内存数据库。提供了5种基本的数据类型。

## 1.1 Redis简介

Redis是一个非关系数据库，与memcache的主要区别是，能持久化存储，支持更多的数据结构，redis实现了主从复制策略。

## 1.2 Redis数据结构简介

Redis的值可以是5种不同的数据结构：String、List，Set、HashMap、有序的HashSet；命令有的通用，有的数据结构独有；含义与程序语言中的集合类型都差不多。

String：get、set、del；

List：[l|r]push、[l|r]pop、[l|r]range、[l|r]index；

Set：sadd、smembers、sismember、srem；

HashMap：hset、hget、hgetall、hdel；

HashSet：zadd、zrange、zrangebyscore、zrem；

## 1.3 你好redis

示例问题的解决：文章排名的功能。

# 使用Redis构建Web应用

将传统的关系数据库的数据处理任务与存储任务放到Redis中，可以提高网页载入速度与系统性能。

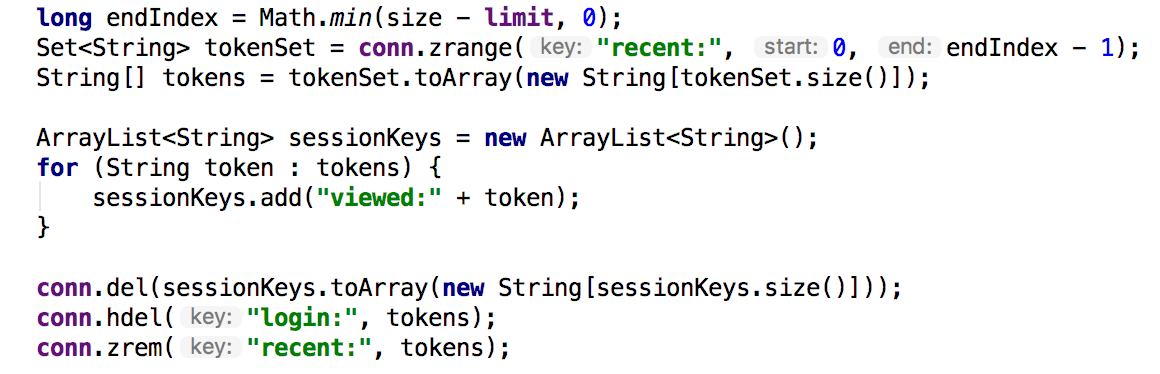
## 2.1 登陆与COOKIE缓存

Cookie记录身份信息，存储在浏览器中，记录登陆信息的方式：签名cookie，令牌cookie；签名cookie额外包含签名数据；令牌cookie存储随机字符串；关系数据库记录令牌与用户浏览的商品与浏览时间等信息，会有频繁的数据库操作，用户量大时，关系数据库很难很好的处理；此时可以使用Redis；登陆cookie的功能由Redis实现更好；解决方法：

1. 创建很多关于用户Token与浏览商品等信息的映射：{login：令牌与用户之间的映射Recent: 记录令牌最后一次出现的时间；Viewed:+token：记录这个Token浏览过的商品与浏览的时间，并且只保留最近浏览过的25个商品}
2. 定期清理Redis中会话的数据。
3. 登陆的代码如下：



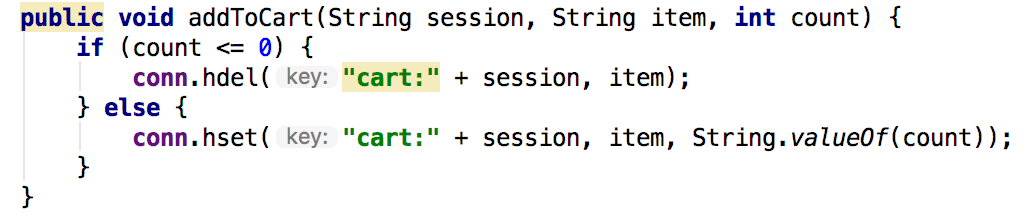
1. 清理的代码如下：



清理函数可以定期执行或者作为守护线程执行，而且可以设置redis数据的超时时间。

## 2.2 使用Redis实现购物车

Cookie购物车的优点：避免数据库频繁读写，缺点，每次请求都要附带并验证cookie信息；购物车存储商品的散列，商品ID与购买数量。

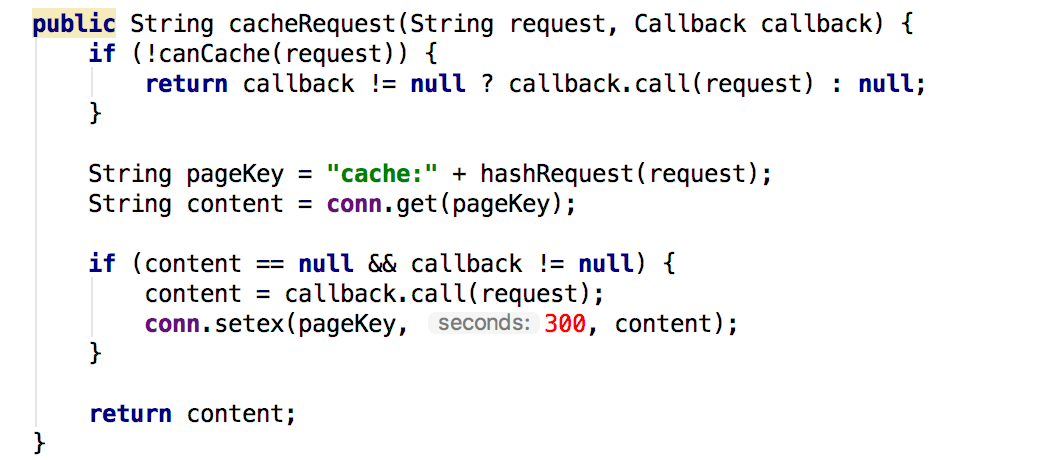


## 2.3 网页缓存

代码如下：提取URL连接的信息，生成URL的key，当访问这个连接时，检测是否能缓存，能缓存，去缓存里面去取，取不到，则重新生成，然后写到Redis。

## 2.4 数据行缓存

针对不能缓存的页面，可以缓存它需要的数据库数据，具体的做法是开启一个守护进程，守护进程负责维护数据行缓存，缓存函数使用JSON格式存储数据库的一行数据，使用2个集合来维护缓存：调度有序集合与延时有序集合；Redis不支持嵌套结构。



## 2.5 网页分析

# 第三章 Redis命令

## 3.1 字符串

Redis的字符串存储字符串、整数与浮点数；整数与浮点数的加减法操作（incr与decr）；字符串的一些操作；

## 3.2 列表

Push/pop/index/range/trim/bpop等阻塞操作；

## 3.3 集合

## 3.7 其他命令

可以操作多种数据类型的sort命令，相当于order by字句。

Redis的事务性命令：WATCH、MULTI、EXEC、UNWATCH、DISCARD；他们都是独占redis服务端的命令，处理完他们包围的操作，才会处理其他命令，键入MULTI命令后，每次添加的命令都会被加入到客户端的队列中，返回QUEUED标志；键入EXEC后一起送入到服务端执行；

Redis里面的数据可以设置生存时间或者过期时间，当数据过期时，Redis服务端会自动删除这个数据；Redis只会为整体的数据设置过期时间，集合里面的单个元素不能设置过期时间；[p]expire[at] 添加过期时间；[p]ttl 查看过期时间；persist 移除过期时间。

# 数据安全与性能保障

## 4.1持久化选项

2种方法持久化：1.记录数据；2.记录命令。

Redis.conf文件中用于配置记录数据的持久化的参数是

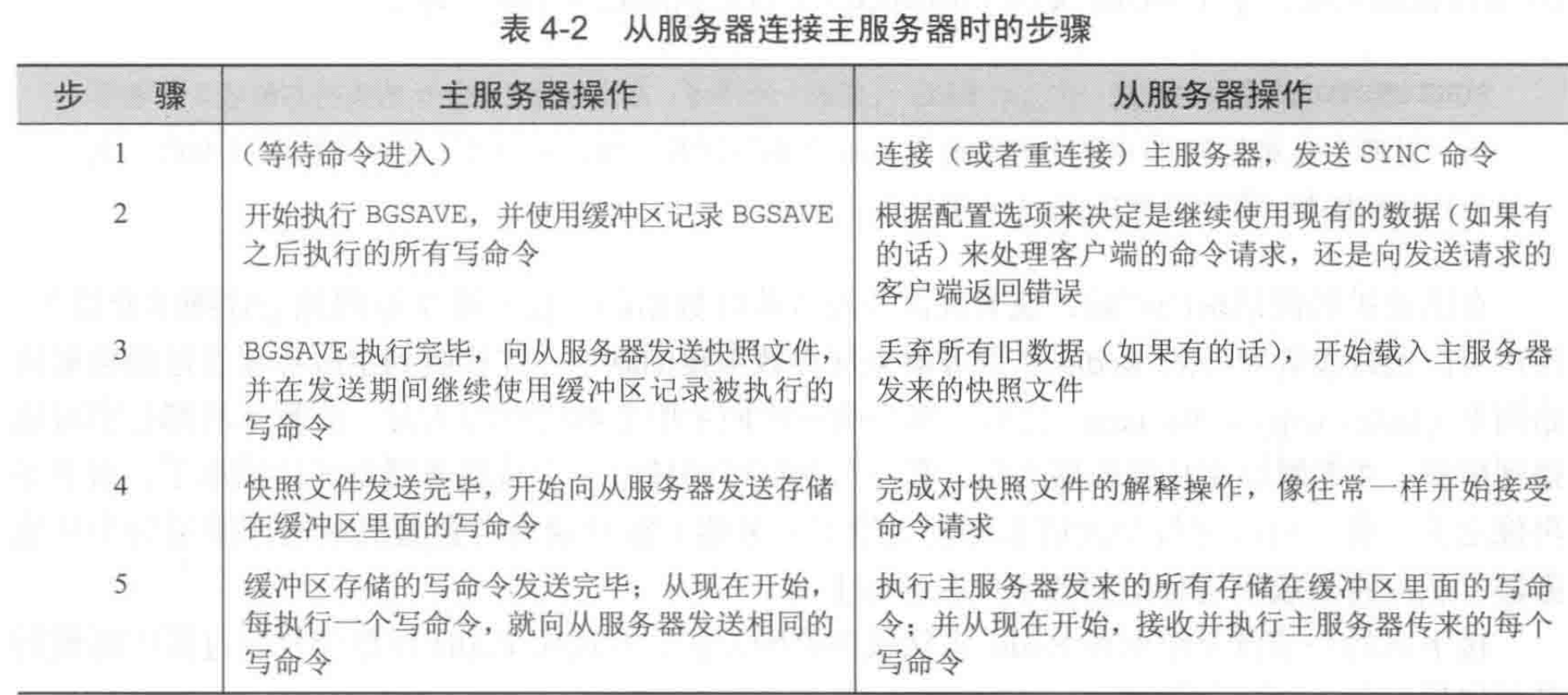
save \* \*;stop-write-on-bgsave-error;rdbcompression;dbfilename；bgsave（子进程负责持久化）命令执行持久化数据，save（不是后台执行），配置了save选项，save条件满足，自动触发bgsave操作；shutdown时，会自动save；2台服务器同步的时候出发bgsave；只适合可以忍受部分数据丢失的场景，因为系统奔溃时，没有持久化成功的数据将会丢失。

Redis占有内存越多，bgsave产生子进程造成的redis停顿时间越多；大数据量时，停顿时间过长会丢失数据，同时bgsave比save要花更长的时间存储数据；

Aof记录命令的配置：appendonly,appendfsync;随着写入，AOF文件会变得特别大。可以通过重写来减小它的体积。

## 4.2 复制

Redis实现了读写分离机制。分担主服务器的压力。如果要实现Redis的主从分离机制，只需要在conf配置文件中配置slaveof参数。



Redis的从服务器也可以有从服务器，形成主从链；主从链可以用来分担主服务器的压力，当从服务器过多时，可以根据主从链的特性建立一个树形结构，中间节点作为复制的中介；叶子结点是所有的从服务器。

## 4.3 处理系统故障

Redis-check-aof/redis-check-rdb。

## 4.4 redis事务