**Redis笔记**

1. 什么是redis？

Redis是王权开源免费的，遵守BSD协议，是一个高性能(NOSQL)的key-value数据库。Redis是一个开源的使用ANSI C语言编写、支持网络、可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。

BSD：伯克利软件发行版。开源协议是一个给予使用者很大自由的协议

NOSQL：非关系型数据库，只要不是rdbms的都是非关系型数据库。NOSQL数据库的产生就是为了解决大规模数据集合多重数据种类带来的挑战，尤其是大数据应用难题

1. 关系型数据库和非关系型数据库的区别？
2. 关系型数据库：表与表之间建立关联关系
3. 非关系型数据库：数据与数据之间没有关联关系
4. NOSQL数据库的四大分类:
5. 键值(key/value)存储数据库：会使用到一个哈希表，这个表中有一个特定的键和一个指针指向特定的数据。如：redis
6. 列存储数据库：通常是用来应对分布式存储的海量数据。如：HBase
7. 文档型数据库：跟键值存储相似，该类型的数据模型是版本化的文档，半结构化的文档以特定的格式存储，比如json， 可以看做是键值数据库的升级版，运行嵌套键值。并且文档型数据库比键值数据库的查询效率高。如：Mong oDb
8. 图形(Graph)数据库：图形结构数据库同其他结构的SQL数据库不同，他是使用灵活的图形模型，并且能扩展到多个 服务器上。NoSQL数据库没有标准的查询语言(SQL)，因此进行数据库查询需要定制数据模型。许多的NoSQL数据 库都是rest式的数据接口或者查询API。如Neo4J。
9. NoSQL数据库在以下几种情况下比较适用：
10. 数据模型比较简单
11. 需要灵活性更强的IT系统
12. 对数据库性能要求较高
13. 不需要高度的数据一致性
14. 对于给定key，比较容易映射复杂值的环境
15. redis与其他key-value缓存产品有一下三个特点：
16. redis支持数据的持久性，可以将内存中的数据保存在磁盘中，重启的时候可以再次加载进行使用。
17. redis不仅仅支持简单的key-value类型的数据，同时还提供list，set，zset，hash等数据结构的存储。
18. redis支持数据的备份，集群等高可用功能。（高可用：就是一直可以用）
19. redis的特点：
20. 性能极高：能读的速度是110000次/s，写的速度是81000次/s。使用C语言编写，C语言是面向机器的语言。
21. 丰富的数据库类型：redis支持的类型string，list，hash，set及ordered set数据类型操作。
22. 原子性：redis的所有操作都是原子性的，意思就是要么成功执行要么失败王权不执行。单个操作是原子性的。多个 操作也支持事物，即原子性，通过MULTI和EXEC指令包起来。
23. 丰富的特性：redis还支持publish/subscribe，通知，key过期等等特性。

------------------------------------------------【一】redis简介总结---------------------------------------------------

1. redis简介总结：
2. redis是一个简单的，分布式的，基于内存的缓存工具。
3. 架设号服务器后，通过网络连接(类似数据库)，提供key-value式缓存服务。
4. 简单，是redis突出的特点。
5. 简单可以保证核心功能的稳定和优异。
6. Redis单个key存入512M大小
7. Redis支持多种类型的数据结构(string,list,hash,set,zset)。
8. Redis是单线程，支持原子性
9. Redis可以持久化，因为使用了RDB(数据文件)和AOF(日志文件)机制
10. RDB：周期性的将内存中的数据备份到磁盘
11. AOF：借助于日志文件记录每次操作
12. Redis支持集群，而且redis支持(0-15)16个库
13. Redis还可以做消息列表，比如聊天室等等
14. 企业级开发中：可以用作数据库、缓存(热点数据(进程会被查询；但是不经常被修改或者删除的数据))和消息中间件等大部分功能
15. redis的优点：
16. 丰富的数据结构
17. 高速读写，redis使用自己实现的分离器，代码量很短，没有使用lock(MySql)，因此效率非常高。
18. redis的缺点：
19. 持久化。Redis直接将数据存储到内存中，要将数据保存在磁盘上，redis可以使用两种方式实现持久化过程。
20. 第一种：定时快照(snapshot)：每隔一段事件将整个数据库写在磁盘上，每次均是写全部数据，代价高。
21. 第二种：基于语句追加(aof)：只追踪变化的数据，但是追加的log可能过大，同时所有操作需重新执行一遍， 回复速度慢。
22. 耗内存，占用内存过高。
23. redis的配置文件

redis的配置文件位于redis安装目录下，文件名为redis.conf(window名为redis.windows.conf)

十一、redis.conf配置文件详解

1. redis默认不是以守护进程的方式运行，可以通过该配置项修改，使用yes启用守护进程

daemonize no

1. 当redis以守护进程方式运行时，redis默认会吧pid写入/var/run/redis.pid文件，可以通过pidfile指定

Pidfile /var/run/redis.pid

1. 指定redis监听端口，默认端口为6379

Port 6379

1. 绑定的主机地址

Bind 127.0.01

1. 当客户端闲置多长时间后关闭连接，如果指定为0，表示关闭该功能

Timeout 300

1. 指定日志记录级别，redis总共支持四个级别：debug、verbose、notice、warning，默认为verbose

Loglevel verbose

1. 日志记录方式，默认为标准输出，如果配置redis为守护进程方式运行，而这里又配置为日志记录方式为标准输出，日志将会发送给/dev/null。

Logfile stdout

1. 设置数据库的数量，默认数据库为0，可以使用select <dbid>命令在连接上指定数据库id

Databases 16

1. 指定在多长时间内，有多少次更新操作，就将数据同步到数据文件，可以多个条件配合

Save <seconds><changes>

Redis默认配置文件中提供了三个条件：

Save 900 1

Save 300 10

Save 60 10000

分别表示900秒(15分钟)内有1个更改，300秒(5)分钟内有10个更改以及60秒内有10000个更改。

1. 指定存储至本地数据库时是否压缩数据，默认为yes，redis采用LZF压缩，如果为了节省CPU时间，可以关闭该选项，但会导致数据库文件变的巨大。

Rdbcompression yes

1. 指定本地数据库文件名，默认值为dump.rdb

Dbfilename dump.rdb

1. 指定本地数据库存放目录

Dir ./

1. 设置当本机为slave服务时，设置master服务的IP地址及端口，在redis启动时，它会自动从master进行数据同步

Slaveof <masterip><masterport>

1. 当master服务设置了密码保护时，slave服务连接master的密码

Masterauth <master-password>

1. 设置redis连接密码，如果配置了连接密码，客户端在连接redis时需要通过AUTH<password>命令提供密码，默认关闭。

Requirepass foobared

1. 设置同一时间最大客户端连接数，默认无限制。Redis可以同时打开的客户端连接数为redis进程可以打开的最大文件描述符数，如果设置maxclients 0，表示不作限制。当客户端连接数到达限制时，redis会关闭新的连接并向客户端返回max number of clients reached错误信息。

Maxclients 128

1. 指定redis最大内存限制，redis在启动时会把数据加载到内存中，达到最大内存后，redis会先尝试清除已到期或即将到期的key，当此方法处理后，仍然到达最大内存设置，将无法再进行写入操作，但仍然可以进行读取操作。Redis新的vm机制，会把key存放内存，value会存放在swap区

Maxmemory <bytes>

1. 指定是否在每次更新操作后进行日志记录，redis在默认情况下是一部的把数据写入磁盘，如果不开启，可能会在断点时导致一段时间内的数据丢失。因为redis本身同步数据文件是按上面save条件来同步的，所以有的数据会在一段时间内只存在于内存中。默认为no。

Appendonly no

1. 指定更新日志文件名，默认为appendonly.aof

Appendfilename appendonly.aof

1. 指定更新日志条件，共有3个可选值：
2. No：表示等操作系统进行数据缓存同步到磁盘(快)
3. Always：表示每次更新操作后手动调用fsync()将数据写到磁盘(慢，安全)
4. Everysec：表示每秒同步一次(折中，默认值)

Appendfsync everysec

1. 指定是否启用虚拟内存机制，默认值为no，简单介绍一下，VM机制将数据分页存放，由redis将访问量较少的页即冷数据swap到磁盘上，访问多的页面由磁盘自动换出到内存中。

Vm-enabled no

1. 虚拟内存文件路径，默认值为/tmp/redis swap，不可多个redis实例共享

Vm-swap-file /tmp/redis.swap

1. 将所有大于vm-max-memory的数据存入虚拟内存，无论vm-max-memory设置多小，所有索引数据都是内存存储的(redis的索引数据 就是keys)也就是说，当vm-max-memory设置为0的时候，其实是所有value都存在于磁盘。默认值为0

Vm-max-memory 0

1. redis swap文件分成了很多的page，一个对象可以保存在多个page上面，但一个page上不能被多个对象共享，vm-page-size是要更具存储的数据大小来设定的，如果存储很多小对象，page大小最好设置为32或者64bytes;如果存储很大大对象，则可以使用更大的page，如果不确定，就使用默认值。

Vm-page-size 32

1. 设置swap文件中的page数量，由于页表(一种表示页面空闲或使用的bitmap)是在放的内存中的，在磁盘上每8个pages将消耗1byte的内存。

Vm-pages 134217728

1. 设置访问swap文件的线程数，最好不要超过及其的核数，如果设置为0，那么所有对swap文件的操作都是串行的，可能会造成比较长时间的延迟。默认值为4。

Vm-max-threads 4

1. 设置在向客户端应答是，是否把较小的包含并为一个包发送，默认为开启

Glueoutputbuf yes

1. 指定在超过一定的数量或者最大的元素超过某一临界值时，采用一种特殊的哈希算法

Hash-max-zipmap-entries 64

Hash-max-zipmap-value 512

1. 指定是否激活重置哈希，默认为开启

Activerehashing yes

1. 指定包含其他的配置文件，可以在同一主机上多个redis实例之间使用同一份配置文件，而同时各个实例又拥有自己的特定的配置文件

Include /path/to/lacal.conf

1. redis作为优秀的中间缓存件，时常会存储大量的数据，及时采取了集群部署来动态扩容，也应该及时的整理内存，维护系统性能(如果数据一直新增，内存很快就会占满).

十二、对于内存占满了redis中的两种解决方案：

1. 为数据设置超时时间：设定内存空间，建议内存空间不要超过1G 256-512M
2. 采用LRU算法动态将不用的数据删除。
3. Volatile-lru：设定超时时间的数据中，删除最不常使用的数据
4. Allkeys-lru：查询所有的key中最近最不常使用的数据进行删除，这是应用最广泛的策略
5. Volatile-random：在已经设定了超时的数据中随即删除
6. Allkeys-random：查询所有的key之后随即删除
7. Volatile-ttl：查询全部设定超时时间的数据，之后排序，将马上将要过期的数据进行删除操作。
8. Noeviction：如果设置为该属性，则不会进行删除操作，如果内存溢出则报错返回。(默认)
9. Volatile-lfu：从所有配置了过期时间的键中驱逐使用频率最少的键
10. Allkeys-lfu：从所有键中驱逐使用频率最少的键。

注意：后面两个是4.0版本之后新加的

十三、自定义配置redis

1. 进入对应的安装目录/usr/local/redis

修改redis.conf配置文件 vim redis.conf(进入命令模式 通过/内容 查找相应字符串)

1. redis配置默认必须修改：
2. daemonize no修改为daemonize yes
3. bind 127.0.01 注释掉(在前面加个#号)
4. requirepass 设置密码

十四、远程服务上执行命令

语法：redis-cli –h host –p port –a password

Redis-cli –h Ip地址 –p 端口 –a 密码

十五、redis关闭

1. 第一种关闭方式：(断点、非正常关闭，容易数据丢失)

查询PID ps-ef | grep –I redis

Kill -9 PID

1. 第二种关闭方式(章程关闭、数据保存)

./bin/redis-cli shutdown 关闭redis服务，通过客户端进行shutdown如果redis设置了密码，需要先在客户端通过密码登录，再信息shutdown即可关闭服务端。

十六、redis命令描述

Redis命令用于在redis服务上执行操作。要在redis服务上执行命令需要一个redis客户端。Redis客户端在我们之前下载的redis安装包中。

十七、redis支持的五种数据类型：

1. string(字符串
2. hash(哈希)
3. list(列表)
4. set(集合)
5. zset(sorted set;有序集合)等等

十八、redis中的内存维护策略

Redis作为优秀的中间缓存件，市场会存储大量的数据，即使采取了集群部署来动态扩容，也应该即使的整理内存，维持系统性能。

**十九、redis中的两大解决方案：**

1. 为数据设置超时时间
2. 采用LRU算法动态将不用的数据删除。内存管制的一种页面置换算法，对于在内存中但又不用的数据块(内存块)叫做LRU，操作系统会根据哪些数据属于LRU而将其移除内存而腾出空间来加载另外的数据。
3. Volatile-lru：设定超时时间的数据中，删除最不常使用的数据
4. Allkeys-lru：查询所有的key中最近最不常使用的数据进行删除，这是应用最广泛的策略。
5. Volatile-random：在已经设定了超时的数据中随机删除
6. Allkeys-random：查询所有的key，之后随即删除
7. Volatile-ttl：查询全部设定超时时间的数据，之后排序，将马上要过期的数据进行删除操作。

二十、应用场景

命令：Expire key seconds

1. 显示的优惠活动信息
2. 网站数据缓存(如积分排行榜)
3. 手机验证码
4. 限制网站访客评率

二十一、key的命令建议

1. key不要太长，尽量不要超过1024字节，这不仅消耗内存，而且会降低查找的效率
2. key也不要太短，太短的话，key的可读性会降低
3. 在一个项目中，key最好使用统一的命名模式，例如：user：123，password；