1、redis是一款开源的nosql，非关系型数据库，以key--value的形式进行存储数据

2、redis的特点

2-1：redis是单线程单进程的

2-1：reids读写速度非常快，应为redis是将数据库的的数据存储到内存，然后再进行操作，所以读写速度特别快

3、redis数据结构

redis有五种数据结构

String

list

hash

set

sort set

3-2、设置一个主键的存活的时间

如设置key为wukong的键值对存活的时间为60s ，60s后自动删除

expire wukong 60

查看一个主键还能生存多久 ，当时间过去的时候变为负数

ttl wukong

直接取消过期时间并且让他删除

presist wukong

4、redis持久化

由于所有数据保存到内存上，所以对数据异步更新到磁盘上，redis提供了一些策略

数据库快照（snapshotting rdb）：根据时间进行更新，将修改过的数据保存下来，每隔一段时间将数据更新磁盘

aof：将数据库所有修改操作记录下来，当redis重启时，会通过重新执行文件中保存的写命令在内存

中重建整个数据库的内容。

aof优势在于:就算出问题了,最多丢失1秒内的更新数据

aof的劣势:

aof文件的体积可能会很大(可能比快照文件还大),另一方面,系统重启的时候回从aof里读命令,

如果aof文件太大,读命令也就要还很久

5、redis集群

调配器，实现负载均衡（jdiesCluster）

redis采用主从集群方式进行工作

主

从

从

从

从

主

主

主

1. 一个主机和从机的数据一样，主机负责修改操作，从机只读，主从相互更新，达到数据一致（分片存储）
2. 每一个主机和从机为一个节点，各个节点互不相关
3. 每个节点都能做相同的事，公共完成一件一个台主机完成不了的事，这就是集群
4. 当主节点down机后从节点自动升级为主节点，原来的主节点重启后变为从节点
5. 调配器将数据均衡分给各个节点处理
6. 当访量过大，可以添加节点实现扩充（最多）

6window下启动redis

将redis解压后放到指定的位置

用cd命令打开redis目录

执行redis “reids-sever redis.Windows.conf”运行redis

再次进入redis目录，用redis-cli开始运行redis

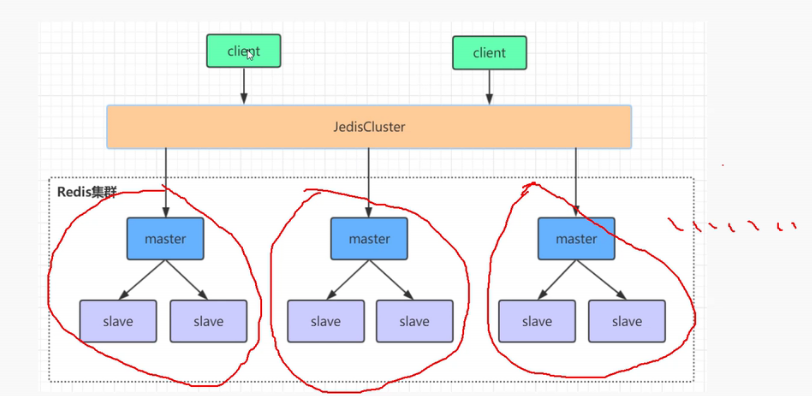
7往redis放入一条数据

Set aa　”123”

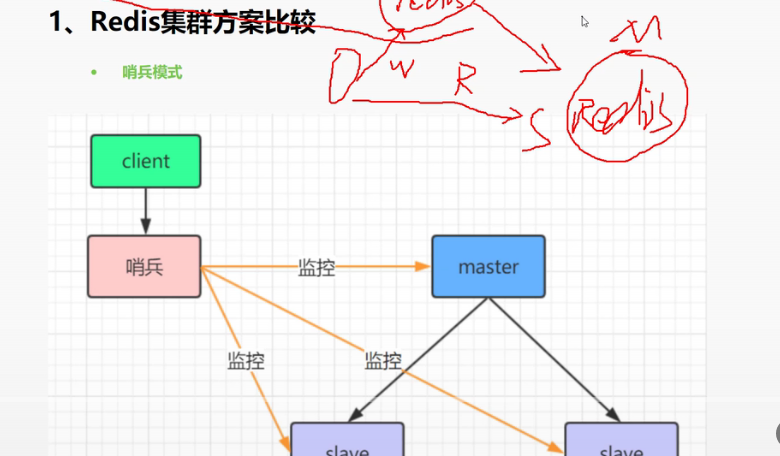
取出一条数据

get aa;

高可用集群，master为读，slave为写，每个集群独立提供服务。比起主从哨兵模式更加高可用



哨兵模式



为什么redis是单线程的

因为redis的主要功能是用于做缓存，他的主要的操作就是数据的读写，那么也就是io密集型，而多线程的主要的目的就是充分利用cpu的计算能力，io并不会太消耗cpu，多线程份儿会带来线程的切换，浪费时间，所以redis就做成了单线程的

Reids 的SetNx命令

**格式：setnx key value**

将 key 的值设为 value ，当且仅当 key 不存在。

若给定的 key 已经存在，则 SETNX 不做任何动作。

SETNX 是『SET if Not eXists』(如果不存在，则 SET)的简写。

用于分布式锁

Redis的incr

**格式：incr key**

将 key 中储存的数字值增一。

如果 key 不存在，那么 key 的值会先被初始化为 0 ，然后再执行 INCR 操作。

如果值包含错误的类型，或字符串类型的值不能表示为数字，那么返回一个错误。

本操作的值限制在 64 位(bit)有符号数字表示之内。

注：这是一个针对字符串的操作，因为 Redis 没有专用的整数类型，所以 key 内储存

的字符串被解释为十进制 64 位有符号整数来执行 INCR 操作

作用

**计数器：计算某个页面被点击了多少次，或者多少个赞**

可以通过组合使用 INCR 和 EXPIRE ，来达到只在规定的生存时间内进行计数

(counting)的目的。

**限流器**

Redis的sortSet可以实现排行榜功能

# 添加单个元素

redis> ZADD page\_rank 10 google.com

(integer) 1

# 添加多个元素

redis> ZADD page\_rank 9 baidu.com 8 bing.com

(integer) 2

redis> ZRANGE page\_rank 0 -1 WITHSCORES

1) "bing.com"

2) "8"

3) "baidu.com"

4) "9"

5) "google.com"

6) "10"

因为每个元素都带有权重，这个权重刚好可以作为分数，而且sortset可以轻松实现排序