

# Ubuntu 安装redis

Apt-get install redis-server

Redis-cli

默认配置文件在etc/redis/redis.conf 中

Bind 绑定 需要哪台机器就绑定哪台

Requirepass 中修改密码

Database 修改数据库个数

# 多键值对存取

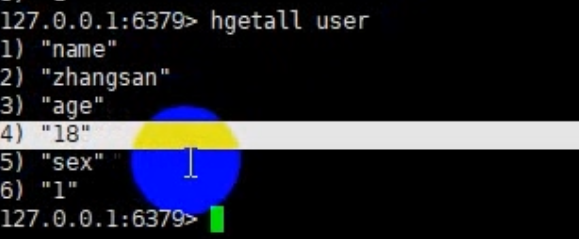


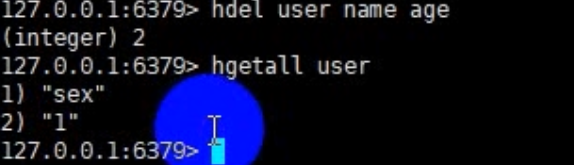
# Hash set

## User是redis key name 是hashkey

# 

取出所有的hash 键值对





删除一个键值对

# List

# 

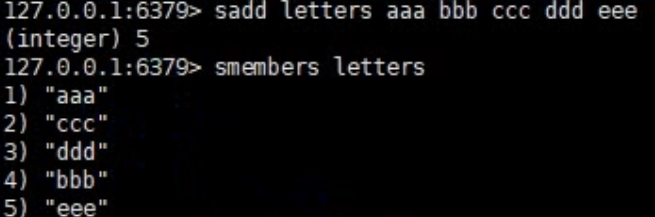
## 左添加 和右添加

## 

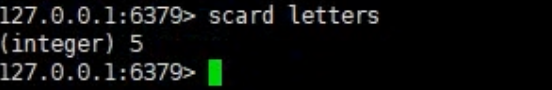
## 删除list元素（从左往右删除）

## 

# Set（无序）



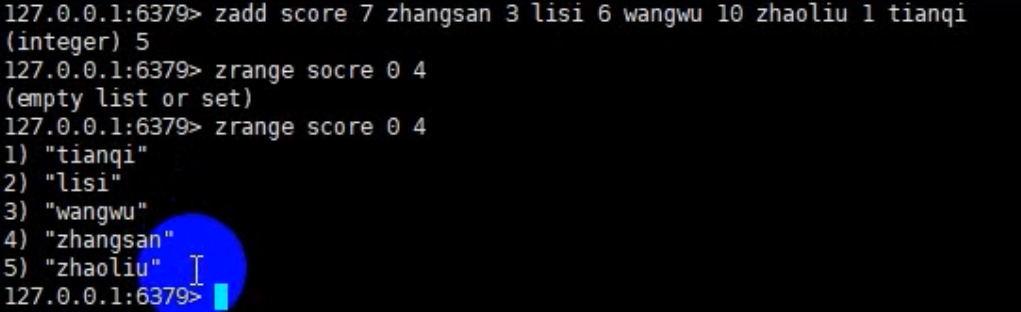
## 查询



## 删除



# Sorted set

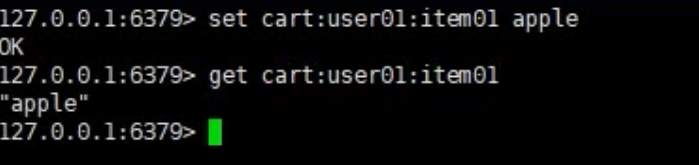


## 删除

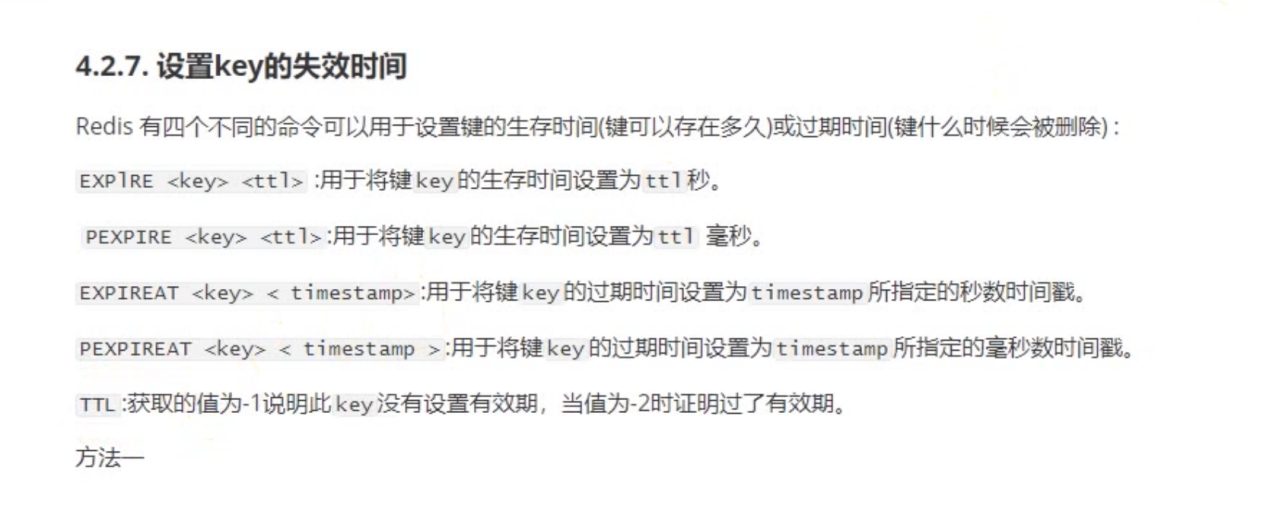
## 

# 层级数据

# 



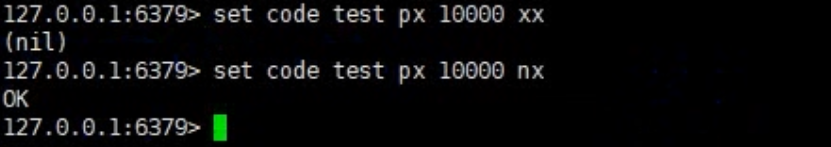
# 失效时间



## 设置10s 失效

## 

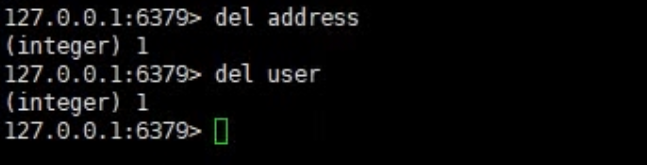
Nil 表示空



Px表示毫秒数 nx 表示值不存在才能设置成功

xx表示值存在才能设置成功

# 通用删除命令del

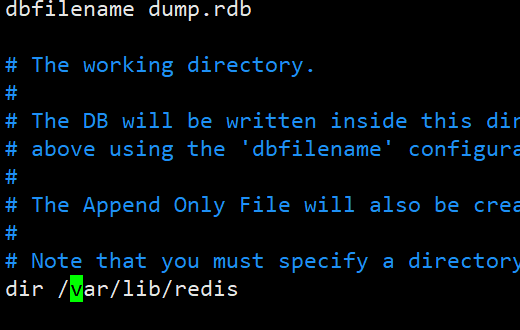


# Redis 磁盘持久化

1. 
2. Background save磁盘持久

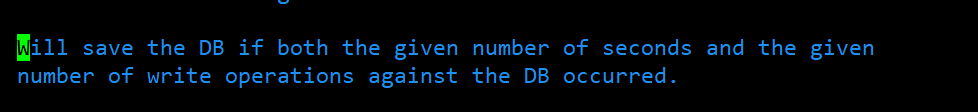
## 默认开启的持久化方案

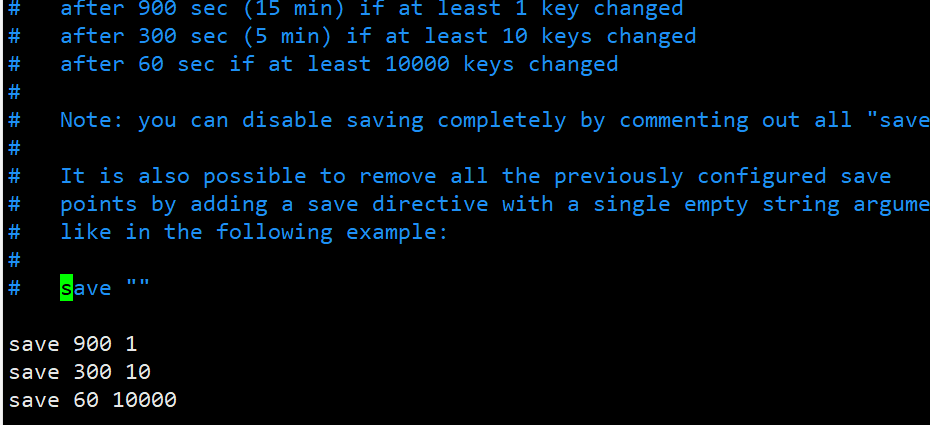
### 持久化数据的路径



## 持久化机制

900s 改变一次 则持久化



、



可以更改配置

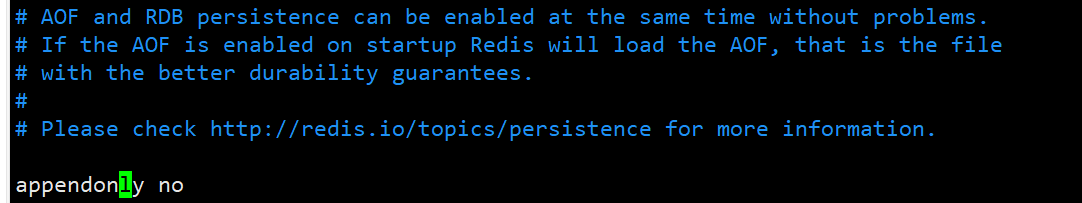
# 持久化缺点：9999s时60 次写入 宕机了

# Aof方案

****AOF通过将redis每次执行的修改命令记录到单独的日志文件（appendonly.aof），Redis回复数据的时候通过再次执行一遍日志中的命令来恢复数据。****

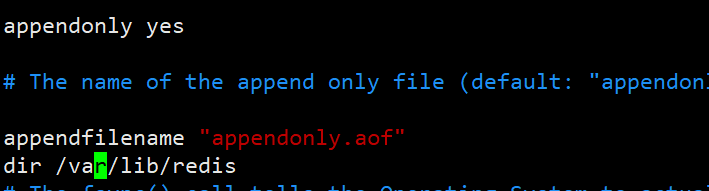
****持久化：****

****缺点：随着时间积累 日志文件越来越大 也越来越慢****



改为yes 开启aof

设置aof 路径



# 主从集群

多窗口卖票

挂掉的服务器大于等于一半 就都不可用了

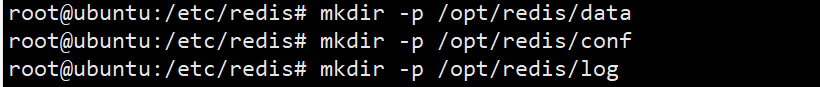
所以集群服务器都是奇数

Redis 主从

发布 订阅机制

# 主从集群搭建

1. 建立文件目录



2.复制配置文件



1. 更改redis-comm 文件 comm作为一个静态导入 注释ip 使得所有ip都可以访问 保护模式关掉 端口号注释掉 daemonize 后台启动yes
2. 进程文件注释掉
3. 

日志文件注释掉



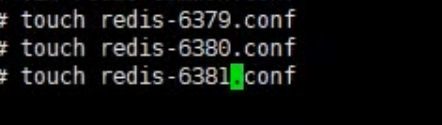
Dbfilename 注释掉

Dir 目录改为/opt/redis/data

从服务器访问密码 改为root

从服务器默认只读

# 

5**新建各个服务器的conf**文件

# Redis 默认位置是/etc/init.d

# 在Linux上，如果开了redis的守护进程，kill -9和redis-cli shutdown 命令是无法杀掉 redis 进程的 ，

# 杀掉就会重新启动一个新的进程

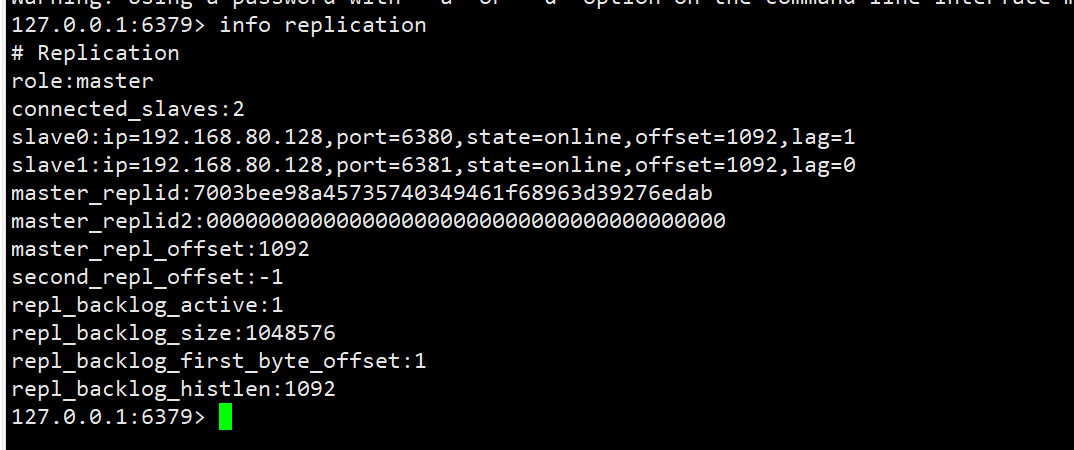
# 最后在网上找到这个命令： /etc/init.d/redis-server stop

# 主从集群启动命令

Centos 下载安装 etc/init.d/redis-server start redis.conf

Ubuntu apt-get安装 redis-server 所在路径下的config

查看主从信息



从服务器只能读，主从复用 读写分离

# 哨兵功能

哨兵是用来监视和选举服务器的

如果主服务器宕机 那么从服务器提供的是国企的数据

哨兵功能解决这个问题

当主服务器挂掉后

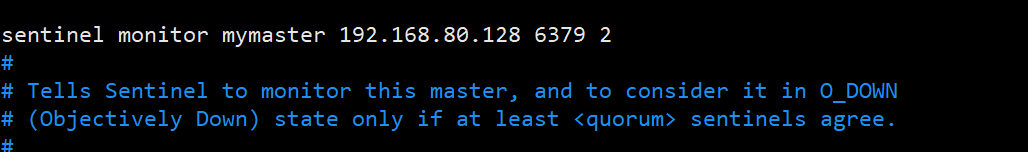
重新选取一个从服务器作为一个新的主服务器

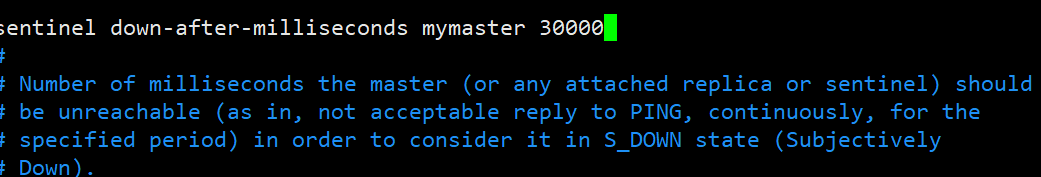
哨兵选举出新的主服务器

哨兵安装 sudo apt-get install redis-[sentinel](https://so.csdn.net/so/search?q=sentinel" \t "https://blog.csdn.net/weixin_44158401/article/details/_blank)

配置哨兵 sentinel.conf 同样位于 /etc/redis 下

配置sentinel-common.conf



# How many replicas we can reconfigure to point to the new replica simultaneously

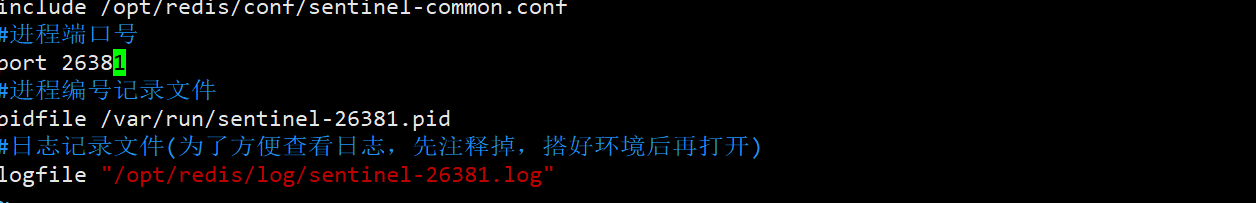
# during the failover. Use a low number if you use the replicas to serve query

# to avoid that all the replicas will be unreachable at about the same

# time while performing the synchronization with the master.



各个哨兵配置文件



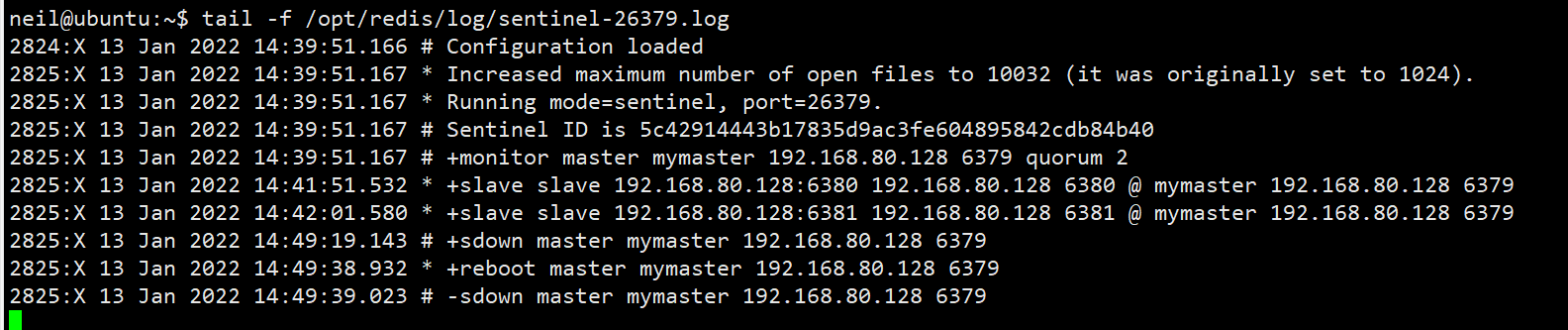
# Boot sentinel

Redis-sentinel <conf path>

杀死哨兵进程

/etc/init.d/redis-sentinel stop

查看各哨兵日志文件



而我自己部署的时候，粗心的把后面的--sentinel竟然给忘了。为了避免类似的事情发生，建议大家使用redis-sentinel sentinel.conf代替redis-server命令

# Myid sentinel

## 哨兵主从切换

如下是在Redis主从复制的基础上，依次启用三个哨兵节点的后，sentinel.cnf的变化情况

可以发现，当启用了三个哨兵节点之后，sentinel.cnf配置文件会被自动重写，主要有一下几点：

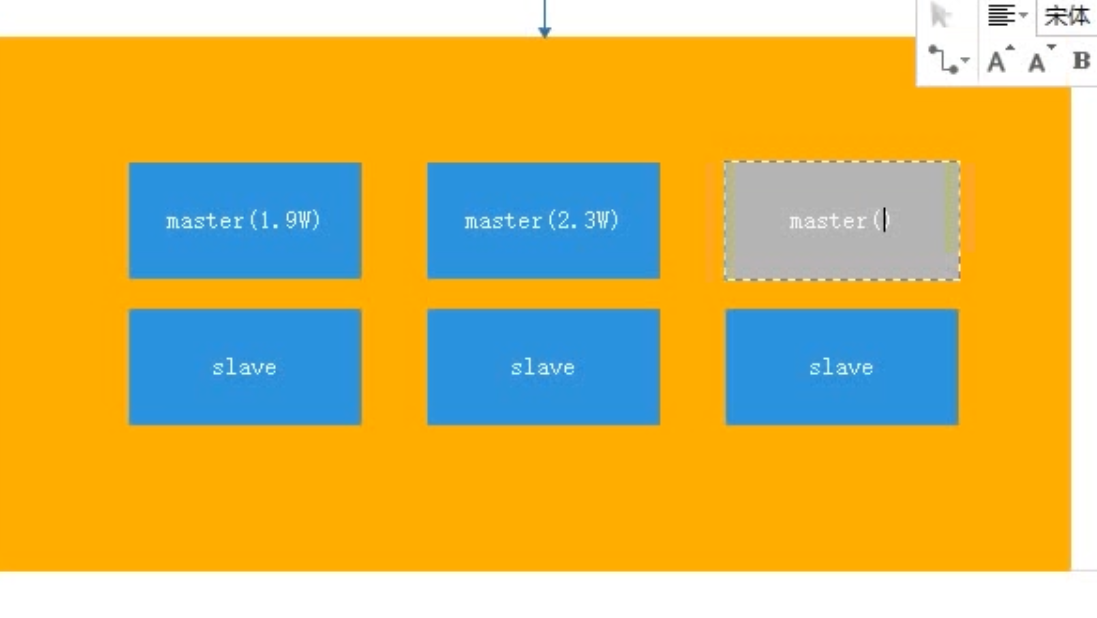
增加了一个sentinel myid （标识哨兵节点的唯一性）

自动追加哨兵节点本身的信息（这样哨兵节点之间就会相互自动发现），以及redis数据服务的slave的信息

自动移除主节点的密码

dir 的相对路径被修改为绝对路径

# Redis 集群



Hash 取值计算 16348个槽 按序取值 10w以内 主备swith

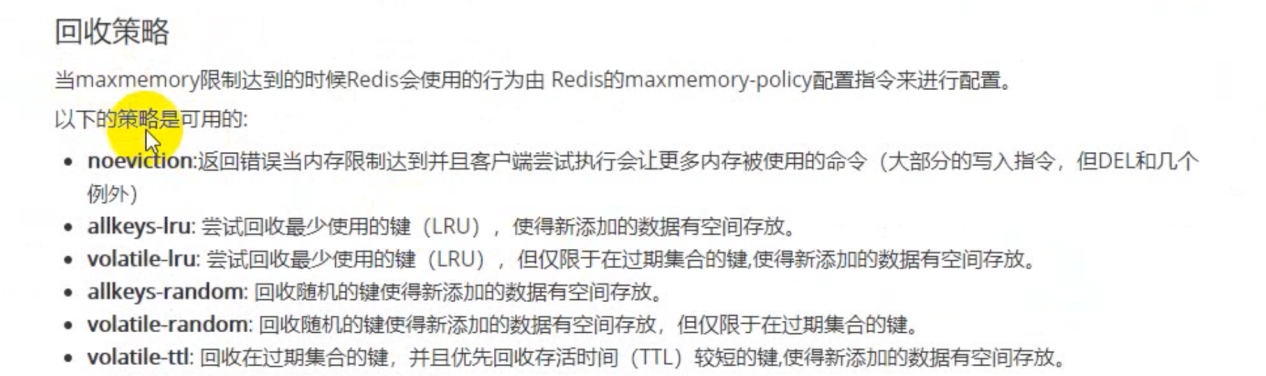
几百万 几千万 集群

如果是几万几千 单节点就够了

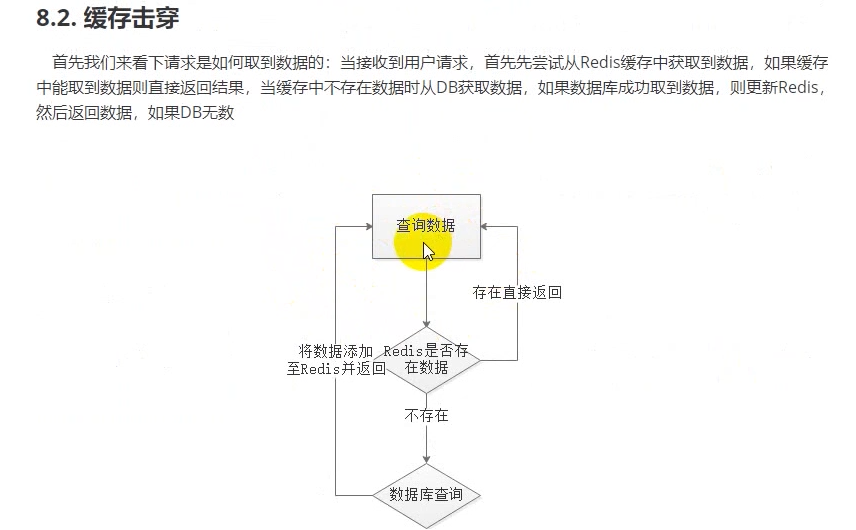
根据实际情况

# 内存淘汰机制

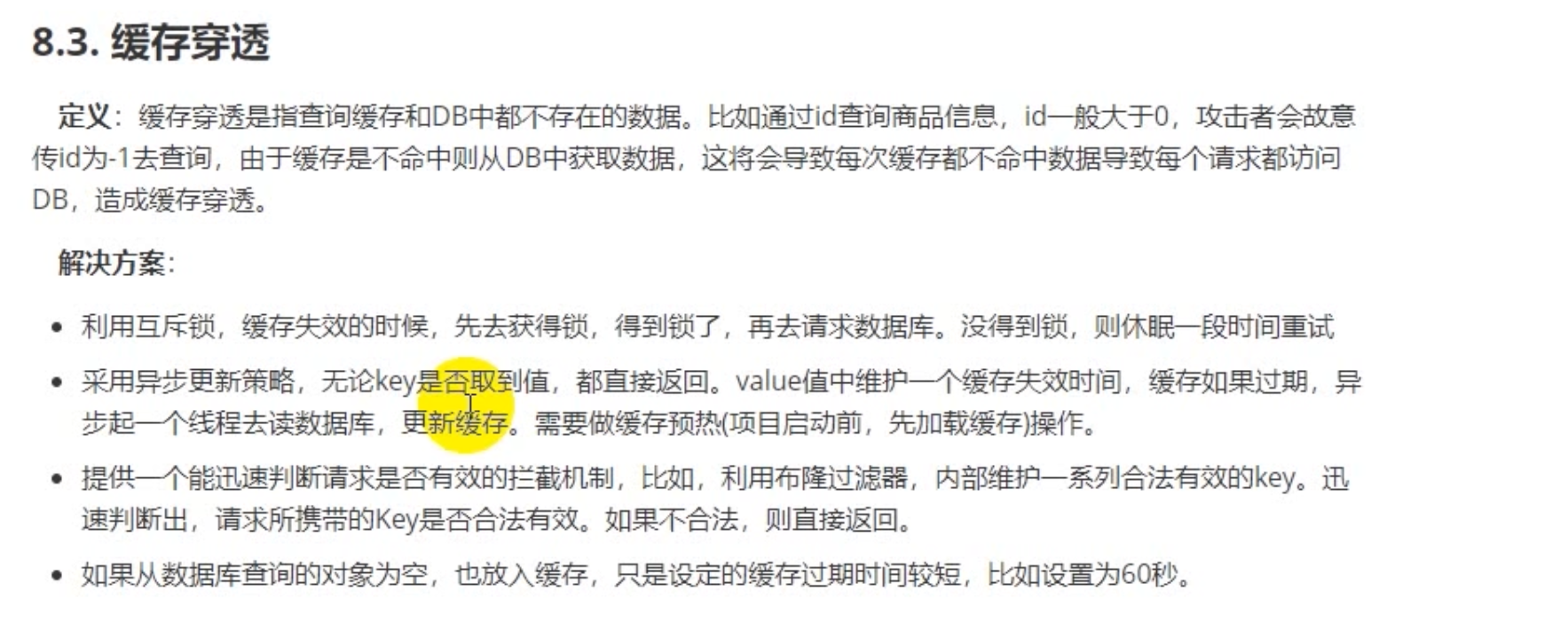




# 缓存击穿



秒杀iphone11 iphone11 这个key 突然失效

缓存雪崩

