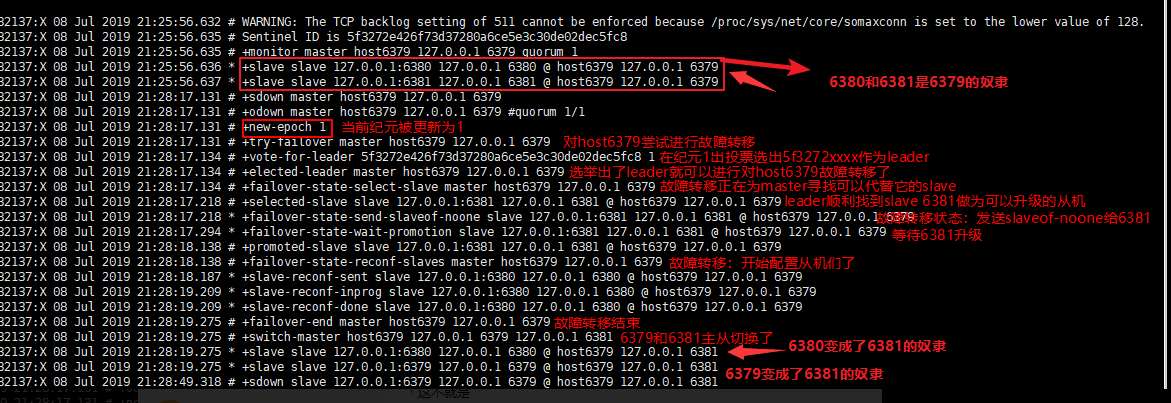


解释一下哨兵日志文件。



<https://yq.aliyun.com/articles/707938>

<https://www.jb51.net/article/109465.htm>

Q:什么是Redis?它和MongoDB的区别呢？

A:一种基于内存的非关系型数据库，性能很高。它也是一种分布式数据库，同时满足高可用的特点。

Redis是一种非关系型数据库，这是因为它的存储模式是K-V键值对模式，这种模式建立在简单动态字符串的基础上（也保证了二进制安全）。因为SDS相对于普通字符串拿空间换时间保证了Redis的高性能。另外Redis同样是一种基于内存的数据库，所以这也保证了Redis的高性能。高性能同时让我们有了更大的空间来换取高并发和高可用（都是为了安全），Redis同样支持事务来获得高并发；Redis是一种分布式数据库，它的主从复制的特点尤其是哨兵模式很好地支持了高可用，它的读写分离机制也很好地支持了高并发。再者Redis的持久化机制也很好地保证了数据的安全性。-----很好！

Q:RDBMS和NoSQL区别是什么？

Q:说一下字符串的五种数据类型？

A:String,List,Set,Hash,Zset.

String是一个SDS，是一个字符数组。它和普通的原生C字符串的区别在于：

安全---体现在两方面，一方面是它的长度属性很好地杜绝了缓冲区溢出，另外一方面在SDS中它并不是只把“\0”作为字符串的末尾，这使得SDS存储的数据不局限与文本，还可以存储图片、视频、音频、压缩文件等二进制数据，也就是二进制安全。

性能---也体现在两方面，len属性使得它获取长度的复杂度是O(1)---在查询方面占有优势；而，free属性实现了在增加字符串长度时的空间预分配策略和删除字符串长度时的惰性删除，减少了内存重分配的次数-------这也是一种拿空间换时间的策略。

开发效率高----因为“\0”在redis中也是字符串末尾的标志，所以兼容c函数库中的<string.h>，实现了code reuse,大大提升了开发效率。

Zset的实现是跳跃表

Hash的实现是字典，数据库也是用字典的数据结构来实现的，可以将Hash视为字典中的字典。

Q:什么Redis的主从复制？它的优点是什么，应用场景是什么？

A:

Q:那说一下Redis的哨兵模式。

A: <https://www.jianshu.com/p/06ab9daf921d>

Q: Redis的持久化机制是什么？

A：RDB+AOF

Q:说一下Redis的数据类型？

A:Redis基于Key-Value,键的话就是String类型，值的话一般常用的有5种：String、list、set、zset、hash（K-V），别的

<https://www.cnblogs.com/dijia478/p/8058775.html>

Q:打印一个redis.conf来解释一下其中的含义。

Q:Redis事务和Mysql事务的区别？

事务特性不同：ACID和ACI(D)

回滚机制：Redis不支持----因为这违背了开发简单有效的原则，而且Redis事务执行错误一般都是开发时编程错误产生，很少会在实际生产环境中出现。

Q:什么是一致性？-----原子性和隔离性很好地保证了这一点。

A: 如果数据库中的数据全部符合现实世界中的约束（all defined rules），我们说这些数据就是一致的，或者说符合一致性的。  
如何保证数据库中数据的一致性（就是符合所有现实世界的约束）呢？这其实靠两方面的努力：  
  
数据库本身能为我们保证一部分一致性需求（就是数据库自身可以保证一部分现实世界的约束永远有效）。  
更多的一致性需求需要靠写业务代码的程序员自己保证。----《掘金小册》