# 学习地址

https://blog.csdn.net/qq\_36236890/article/details/83964399

# Redis的key规范

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；

1. 在一个项目中，key最好使用统一的命名模式，例如user:10000:passwd。

# 简介

是分布式的，遵循CAP

可以用作：数据库，缓存，和消息中间件

Reids是基于内存的单进程单线程模型的KV数据库。

# 支持的数据类型

String字符串：

格式: set key value

string类型是二进制安全的。意思是redis的string可以包含任何数据。比如jpg图片或者序列化的对象 。

string类型是Redis最基本的数据类型，一个键最大能存储512MB。

Hash（哈希）

格式: hmset name  key1 value1 key2 value2

Redis hash 是一个键值(key=>value)对集合。

Redis hash是一个string类型的field和value的映射表，hash特别适合用于存储对象。

List（列表）

Redis 列表是简单的字符串列表，按照插入顺序排序。你可以添加一个元素到列表的头部（左边）或者尾部（右边）

格式: lpush  name  value

在 key 对应 list 的头部添加字符串元素

格式: rpush  name  value

在 key 对应 list 的尾部添加字符串元素

格式: lrem name  index

key 对应 list 中删除 count 个和 value 相同的元素

格式: llen name

返回 key 对应 list 的长度

Set（集合）

格式: sadd  name  value

Redis的Set是string类型的无序集合。

集合是通过哈希表实现的，所以添加，删除，查找的复杂度都是O(1)。

zset(sorted set：有序集合)

格式: zadd  name score value

Redis zset 和 set 一样也是string类型元素的集合,且不允许重复的成员。

不同的是每个元素都会关联一个double类型的分数。redis正是通过分数来为集合中的成员进行从小到大的排序。

zset的成员是唯一的,但分数(score)却可以重复。

# Redis持久化

## RDB

RDB是定期去数据快照，所以说可能会造成部分数据丢失。其次RDB每次fork子进程生成快照文件的时候，如果数据文件特别大，可能导致对客户端提供服务暂停数毫秒，甚至数秒。

## AOF

AOF可以更好的避免数据丢失问题，默认每个1s就会通过后台线程执行一次fsync操作，最多丢失1s的数据。写入性能比较高，而且文件不易损坏，AOF日志文件可读性比较高，可以追踪操作日志。

这两者应该一起使用，AOF作为数据恢复的第一选择，利用RDB做数据的冷备。当AOF丢失或者不可用的时候来使用RDB快照快速回复。