

章节

问答

笔记

WIKI

第1章 Redis急速入门与复习【有Redis基础的可以跳过】

▶ 1-1 分布式架构概述 (15:13)

▶ 1-2 为何引入Redis ? (06:53)

▶ 1-3 什么是NoSql ? (05:41)

▶ 1-4 什么是分布式缓存, 什么是Redis ? (03:32)

▶ 1-5 分布式缓存方案与技术选型 : Redis VS Memcache VS Ehcache (05:17)

▶ 1-6 安装与配置Redis (17:12)

≡ 1-7 附 : 安装与配置 Redis

▶ 1-8 Redis命令行客户端基本使用 (04:46)

≡ 1-9 附 : Redis的命令行客户端

▶ 1-10 Redis的数据类型 - string (15:03)

≡ 1-11 附 : Redis的数据类型 - string

▶ 1-12 Redis的数据类型 - hash (06:52)

≡ 1-13 附 : Redis的数据类型 - hash

▶ 1-14 Redis的数据类型 - list (12:05)

≡ 1-15 附 : Redis的数据类型 - list

▶ 1-16 Redis的数据类型 - set (08:23)

▶ 1-17 Redis的数据类型 - zset (10:58)

≡ 1-18 附 : Redis的数据类型 - zset

第2章 SpringBoot整合Redis实战

▶ 2-0 聊一聊多路复用器, 阻塞和非阻塞 (07:16)

▶ 2-1 Redis 架构单线程模型原理解析 (11:20)

▶ 2-2 SpringBoot整合Redis实战 (13:56)

课程介绍

本周首先会对Redis的架构与原理进行讲解, 然后介绍Redis哨兵, 搭建主从复制高可用Redis集群等, 最后带大家解决一些实际应用中遇到的问题。

- 2-3 附：SpringBoot 整合 Redis
 - 2-4 Redis 操作工具类讲解（08:31）
- 2-5 基于Redis优化首页轮播图查询（09:16）
- 2-6 基于Redis优化购物车 - 添加商品（09:25）
- 2-7 Redis 购物车 - 删除商品与更新购买数量（14:04）
- 2-8 Redis 购物车 - 清理已结算商品（09:14）
- 2-9 Redis 购物车 - 同步购物车（1）（17:12）
- 2-10 Redis 购物车 - 同步购物车（2）（14:01）

第3章 Redis进阶提升与主从复制

- 3-1 Redis 的发布（pub）与订阅（sub）（10:14）
- 3-2 Redis的持久化机制 - RDB（20:03）
- 3-3 附：Redis的持久化机制 - RDB
- 3-4 Redis的持久化机制 - AOF（23:24）
- 3-5 附：Redis的持久化机制 - AOF
- 3-6 Redis 主从复制原理解析（12:23）
- 3-7 多虚拟机克隆方案（09:14）
- 3-8 搭建Redis主从复制（读写分离）（17:40）
- 3-9 Redis无磁盘化复制原理解析（10:35）
- 3-10 Redis 缓存过期处理与内存淘汰机制（11:00）
- 3-11 附：Redis 缓存过期处理与内存淘汰机制

第4章 Redis 哨兵机制与实现

- 4-1 Redis 的哨兵模式
- 4-2 Redis 哨兵机制与实现 - 1（14:23）
- 4-3 Redis 哨兵机制与实现 - 2（16:42）
- 4-4 解决原Master恢复后不同步问题
- 4-5 图解哨兵（18:03）
- 4-6 附：哨兵信息检查
- 4-7 SpringBoot 集成Redis哨兵（05:11）
- 4-8 附：SpringBoot 集成Redis哨兵-配置

第5章 Redis集群



▶ 5-1 Redis-Cluster 集群 (06:43)



≡ 5-2 附：Redis 集群与环境准备



▶ 5-3 搭建Redis的三主三从集群模式 (16:06)



≡ 5-4 附：构建Redis集群

最近学习 ●

▶ 5-5 什么是slot槽节点 (12:36)



▶ 5-6 Springboot集成Redis集群 (05:35)



[网站首页](#) [企业服务](#) [人才招聘](#) [联系我们](#) [讲师招募](#) [帮助中心](#) [意见反馈](#) [慕课大学](#) [代码托管](#) [友情链接](#)

Copyright © 2020 imooc.com All Rights Reserved | 京ICP备 12003892号-11  京公网安备11010802030151号

