# 前端知识

## html

## css上

## js

## vue

## react

## AngularJS

## bootstarp

## heigchar

# java基础知识

## 算法和数据结构

### 数据结构

### 算法

## 设计模式 多态

### 23种设计模式 上

### 设计原则

## 集合结构

### Iterator

## java8新特性

### lambda

### map底层改变

## java基础知识

### java压缩和解压 减少带宽

### java 编码 解决中文乱码

## Servlet/listener/filter应用大全

# jvm

## JAVA 内存模型（JMM）

### 线程通信

### 消息传递

## jvm加载过程

## 动态加载

# 基础框架

## springMvc

## spring --->spring5新特性

### Beans

### Context

### Transaction

### MVC

### Spring5新特性

## struts2

## Mybatis-->ORM框架

### mybatis 38

### 代码生成器：Generator 38

### MyBatis下1对多，多对多嵌套结果，嵌套查询 38

### 一级缓存，二级缓存使用场景及选择策略 38

### Mybatis与Spring集成Spring-mybatis.jar分析 38

### Sring集成下的SqlSession与Mapper 38

### Mybatis的事物 38

### 分析Mybatis的动态代理真正实现 38

### 手写MyBatis 2.0 38

## hibernate

## redis

## netty

### BIO-NIO-AIO的原理

### netty 数据压缩 二进制流推送

## shiro 权限

## Quartz定时器

# 微服务

## dubbo

## springboot

### 工具类

### Annotation讲解

### 集成Mybatis实现多数据源路由实战

### 集成Dubbo实战

### 集成Redis缓存实战

### 集成Swagger2 API管理及测试体系

### 实现多环境配置动态解析

## springcloud

### 1-注册中心

### 2-Fegin声明式服务调用

### 3-Ribbon 16

### 4-Hystrix服务熔断降级方式

### 5-Zuul实现微服务网关 17

### 6-Config分布式统一配置中心

### 7-Sleuth和Zipkin进行分布式链路跟踪

### 8-Bus消息总线

### 基于Hystrix实现接口降级实战

### SpringBoot集成Spring Cloud实现统一整合方案

## Docker虚拟化

### docker 安装方式 离线安装 在线安装

### Docker常用命令

### Docker的镜像，仓库，容器

### DockerFile构建Lnmp环境部署个人微博Wordpress

### Docker Compose构建LNMP环境部署个人博客wordpress

### Docker网络组成，路由互联，openvswitch

### 基于swarm构建Docker集群实战

### Kubernetes简介

## 浅谈微服务架构

### SOA架构和微服务架构之间的区别和联系

### SpringBoot流行因素及能够解决什么问题

### 寄去全局分析Spring Cloud各个组件所解决问题

### 微服务支撑

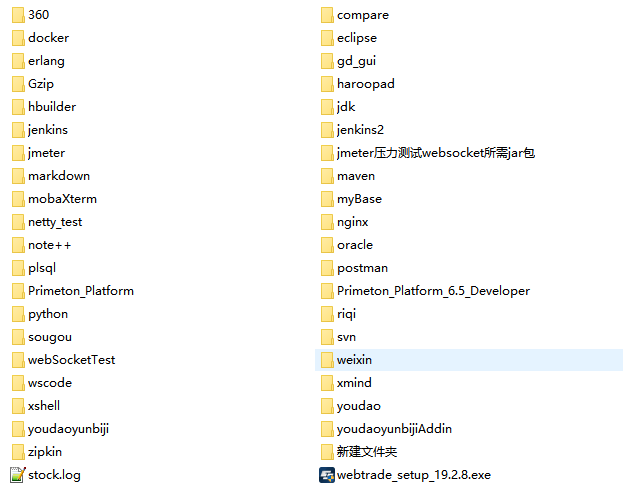
### 集成测试

### 云龙流水线

### 服务器支撑

# 编程工具

常用工具:



## git

## svn

## eclipse

## maven

### Maven打包三种方法和常用命令大全

## ea软件

### 数据库建模

### 接口设计

### 流程图

### 时序图

### 类关系图

## linux应用

### Linux基本命令大全

### linux系统ssh免密钥登录配置 超级简单

# 数据库

## mysql

## oracle

### 重要sql

### oracle设计

## redis

## 数据库设计

### 多对多

# 并发编程

## 内存模型

### 重排序

### 顺序一致性

### happens-before

### as-if-serial

## synchronized

### 同步，重量级锁

### synchronized原理

### 锁优化

## volatile

### volatile实现机制

### 内存语义

### 内存模型

## DCL

### 单例模型

### DCL

### 解决方案

## 并发编程

### AQS

### CAS

## 锁 LOCK

### ReentrantLock

### 8.7.2. ReentrantReadWriteLock

### 8.7.3. Condition

## 8.8. 并发工具类

### CyclicBarrier

### CountDownLatch

### Semphore

## 8.9. 并发集合

### ConcurrentHashMap

### COncurrentLinkedQueue

## 8.10. 原子操作

### 基本类型

### 数组

### 引用类型

### 原子更新字段类

## 8.11. 线程池

### Executor

### ThreadPoolExecutor

### Callable和Future

### ScheduledExecutorService

### 合理配备线程池

### 线程池监控

### ExecutorCompletionService

## 8.12. 其他

### ThreadLocal

### Fork/Join

## 8.13. java如何实现原子操作

### 使用循环CAS实现原子操作

### 锁

# 性能调优

## 9.1. 理解性能优化

### 性能基准

### 性能优化分析

### 衡量维度

### 知其然，知其所以然

## jvm调优

### 什么是JVM运行时数据区

### 什么是JVM内存模型JMM

### 各种垃圾回收使用场景（Throughput/cms）

### 理解GC日志，从日志汉字看端倪

### 实战MAT分许dump文件

## nginx调优

## mysql调优

### 理解MySql底层B+ Tree机制

### SQL执行计划详解

### 索引优化详解

### SQL语句优化

## tomcat调优

### Tomcat线程模型

### Tomcat系统参数认识及调优

### 基准测试

## java代码优化

### 调用链过长

### 业务复杂多条sql connection链接查询

### 纵表存储inner jion关联过长

### 采用横表一对多，多对多存储

### 代码for循环调用，经常new线程池

### 利群搭建

### 影子表辅助表

### 分库分表

### 缓存，刷新缓存，同步缓存

# 分布式框架

## 分布式问题

### 分布式中间件

### 分布式id生成器

### 分布式通信

### 分布式事务

### RPC应用

### netty分布式通信

#### 浅谈分布式框架

### 初始分布式架构及意义

### 单体应用-->分布式,分布式选型

### 大型分布式架构演进过程

### 构建分布式架构最重要的因素

### 分布式架构设计

## 分布式架构策略-分而治之

### 网络通信探究分布式原理

### 基于消息方式的系统间通信

### 通信协议传输过程中的序列化反序列化

### RPC通信技术

### 传统RPC技术在大型分布式架构下面的问题

### 分布式架构下的RPC解决方案

### 分布式系统的基石-Zookeeper

### Dubbo对单一应用服务化改造

## 分布式架构-中间件

### 分布式通信

### 分布式缓存

### 数据存储

### 后台服务

### 高性能NIO框架

## 分布式解决方案

### 分布式锁

### 分布式全局ID生成方案

### 分布式会话：session跨域共享及企业级单点登录解决方案实战

### 分布式事物解决方案实战

### 高并发下的服务降级，限流

### 分布式框架下分布式锁的解决方案

### 分布式框架下实现分布式定时调度

### 灰度发布

# 平台服务

## liunx

## docker

## redis集群

## nginx集群

## jenkins安装部署

## 工具集

### jdk安装

### xmind使用

### Mybase

### MobaXterm

## 服务器

### tomcat

### jetty

### jboss

### aplach

## 平台服务

### iaas,paas,saas

### Infrastructure-as-a-Service（基础设施即服务）

### Platform-as-a-Service（平台即服务）

### Software-as-a-Service（软件即服务）

## 网络编程

### http

### https

### socket

### webSocket

# 热门知识

## 区块链

## 人工智能

## 大数据

# 商业系统

## 三层架构

## cpq样机项目

## 电商项目

## Netty行情推送

## 股票微服务

# 业务能力

## 业务推动技术

## 单元测试

### 功能测试

### 所有的测试代码方式

### 场景测试

### 压力测试

#### Netty压力测试

用到的工具

* Jmeter的WebSocket Sampler
* Linux连接数量脚本

### 自动化测试

Jemeter的自动化测试

1 校验可以用错误吗校验,如果正确就是正确的校验码200,否则就是错误码code

2 用户登录需要session,可以设置前置处理,先调用接口获取session,下面的接口再取这个session值,用于登录

### 集成测试

### 异常测试

### bug测试

## PPT能力 41

### 痛点

### 解决了什么问题，为什么解决

### 行业对标怎么样

### 解决方式

### 整体架构的把控

### 演讲能力

## 项目经验

### 引用的逻辑

### 多对多版本管理

### Subtopic 3

# 面试经验

## 人事面试

## 技术面试

1 业务架构图

2 项目中遇到难点

3

# 个人能力

# 领导能力