

Java 面试指导

D瓜哥 – <https://www.diguage.com>

目录

关于 D瓜哥

为什么要准备面试?

面试应该重点准备什么?

技术方向应该准备什么?

Java 基础

Java 虚拟机

Java 常用框架

常用数据结构和算法

网络

数据库

NoSQL

建议大家去关注的技术

复习材料精选

Java 基础

Java 虚拟机

设计模式

MySQL 数据库

计算机网络

NoSQL

架构

三个月面试重点

面试准备流程

准备简历 — 如何写好技术简历?

简历的意义

工具推荐

面试技巧

面试常问的技术之外的问题

如何谈待遇?

一些善意的提醒

一些题外话

参考资料

关于 D瓜哥

- 网名：**D瓜哥**
- 码农一枚
- 个人博客："地瓜哥"博客网：<https://www.diguage.com/>

如果您觉得这个“Java 面试指导”对您有所帮助，看在 D瓜哥 码这么多字的辛苦上，欢迎友情支持一下，D瓜哥 感激不尽，



为什么要准备面试?

1. 各个公司侧重点不一样，所需要的知识不一样，准备面试来扩大知识点；
2. 绝大部分的人的语言表述能力需要提高；
3. 刚毕业的，不熟悉公司所需要的知识体型。
4. 工作中用到的知识点相对来说更集中
5. 各个公司所有的技术各有侧重
6. 绝大部分人的语言表达能力有待提高，准备面试可以提高对知识的认识和描述，让自己更专业。
 - a. 一些常用技术了解的不够深入以至于“无话可说”
 - b. 深入了解的技术，但却不能专业地表述出来
7. 绝大部分人对技术的抽象描述能力不够好
8. 一些技术有个更进一步的发展。例如：
 - a. Java 8 中的 HashMap 的重新实现
 - b. Java 8 中的 Lambda 和 Stream API

面试应该重点准备什么？

1. 更基础的技术
2. 更有生命力的技术
3. 整个行业喜闻乐见的技术 — PS: 有点赶时髦，除个别情况，不推荐
4. 主攻方向之外的加分技术

技术方向应该准备什么？

1. Java
2. 数据结构和算法
3. 设计模式
4. 网络协议
5. 数据库，尤其是 MySQL
6. NoSQL



学习一门技术之前，一定要了解这个技术出现的历史背景。

1. 为什么会出现它？
2. 它解决了什么问题？
3. 它自身又带来了什么问题？
4. 有没有更好的解决方案？

这样，经过反复思考，才会对一个技术有一个更好的认识。

Java 基础

1. 集合类
 - a. 实现原理
 - b. 源码
 - c. 类似数据结构不同实现的优缺点
 - d. 更高要求：和常用的集合工具类的对比： Apache Commons Collections、Google Guava
2. Lambda 表达式
3. Stream API
4. 并发原理以及常用工具类
 - a. 并发基础原理
 - i. 锁
 - ii. CAS
 - iii. Java Memory Model
 - b. 常用工具类的源码
 - i. AbstractQueuedSynchronizer
 - ii. ConcurrentHashMap
 - iii. ArrayBlockingQueue
 - iv. ThreadPoolExecutor
 - v. ForkJoinPool
5. Java I/O
 - a. BIO
 - b. NIO
 - c. AIO
 - d. 更高要求：Netty — 目前 Java 中最流行的网络 I/O 库。
6. 反射

a. 反射概念很简单，但是应用很广泛；是 Java 扩展性的基础

7. 动态代理



工作前两年，一定要把《Java编程思想》看三遍。认认真真、仔仔细细看。边看边Code。

Java 虚拟机

1. Java 字节码技术
 - a. Java Agent
 - b. [Byte Buddy](#) — 如果学，重点学这个库。
 - c. [CGLib](#) — Spring 内部使用该库。
 - d. [Javassist](#) — Hibernate 内部使用该库。JBoss 出品的相关框架也多使用该库。
 - e. [ASM](#) — Java 字节码编辑库。太基础，不要学。
2. 类加载器
3. JVM 内存模型
4. 垃圾判断算法 — 什么样的对象是垃圾？
 - a. 引用计数法
 - b. 科大学分析
5. 垃圾回收算法
 - a. 标记-清除算法
 - b. 复制算法
 - c. 标记-整理算法
6. 常用垃圾收集器
 - a. CMS
 - b. G1
7. Java Memory Model
8. JVM 常用故障排查工具



- 请看 《[深入理解Java虚拟机（第2版）](#)》；
- 更新的资料请看： 《[Java性能权威指南](#)》。

Java 常用框架

1. Spring Framework
 - a. IoC
 - b. AOP
 - c. MVC
2. Spring Boot
3. MyBatis
4. Dubbo — Dubbo 的文档值得好好读一读。
5. Spring Cloud
6. Apache Kafka — 抛砖引玉，权当入门大数据的一个引线吧。

常用数据结构和算法

数据结构

1. 链表
2. 队列
3. 堆栈
4. 跳跃表
5. 树

算法

1. 排序
 - a. 冒泡排序
 - b. 选择排序
 - c. 归并排序
 - d. 希尔排序
 - e. 快速排序
2. 查找
 - a. 二分查找
 - b. 堆查找
3. 树的遍历
4. 一致性哈希 — 举个

网络

1. HTTP

- a. 长连接
- b. 管道化
- c. 缓存
- d. 代理
- e. 压缩
- f. HTTP 2
- g. HTTPS
- h. HSTS



请关注一下为了提高性能，HTTP 协议如何一步一步演进的。

2. DNS

- a. CDN 的基础。

3. FTP

4. SSL/TLS

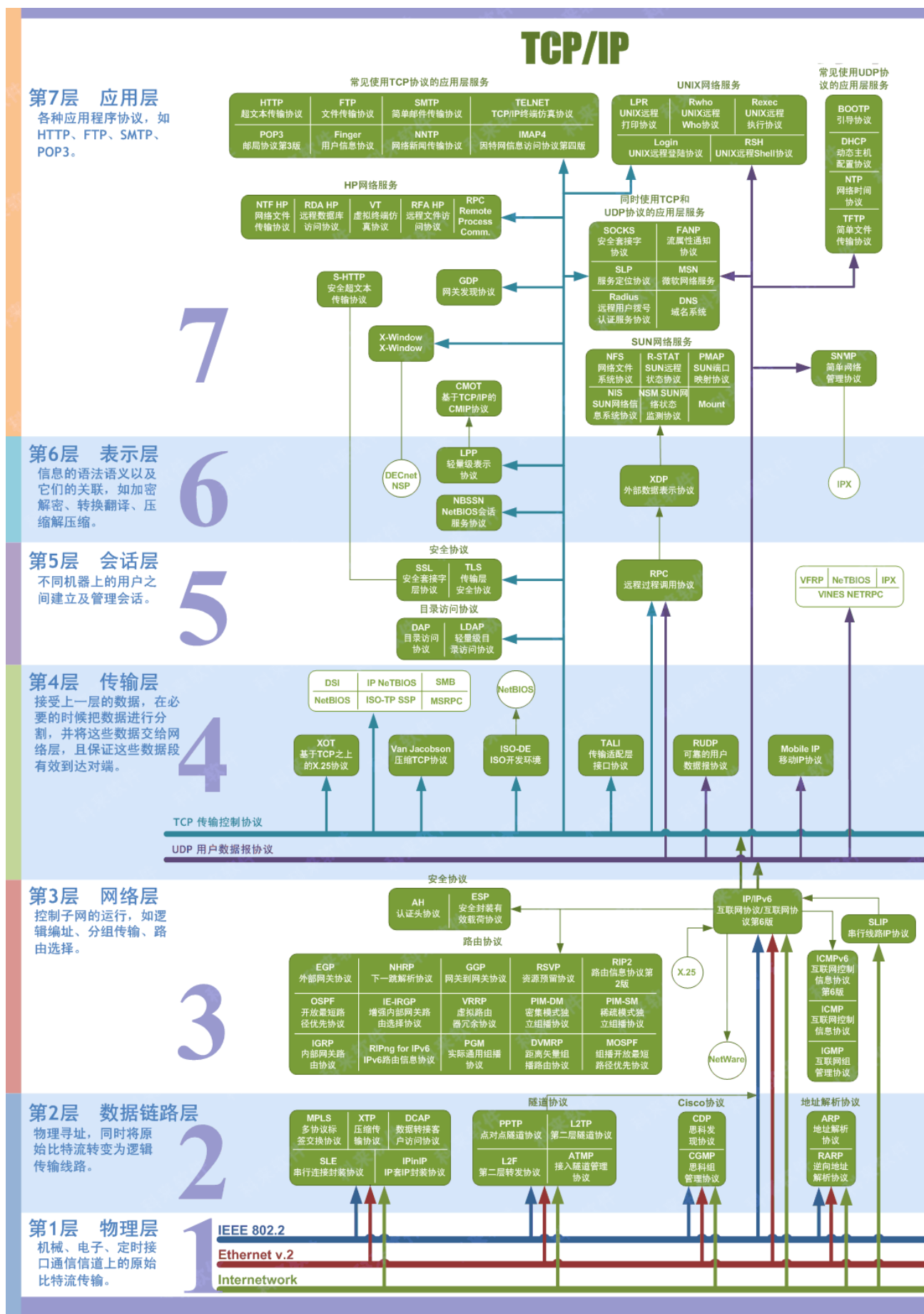
5. TCP

- a. 三次握手
- b. 四次挥手
- c. 慢启动
- d. 滑动窗口

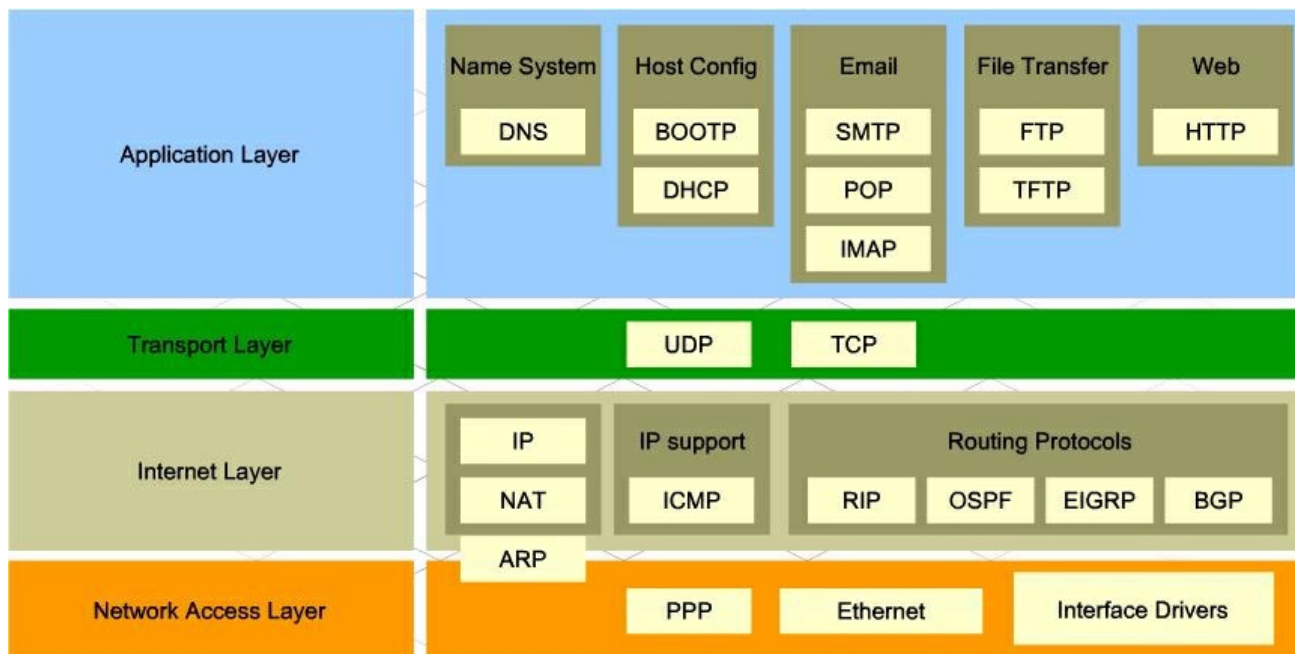
6. IP

- a. 子网划分
- b. 路由交换 — 这个主题内容有点大。

网络协议栈



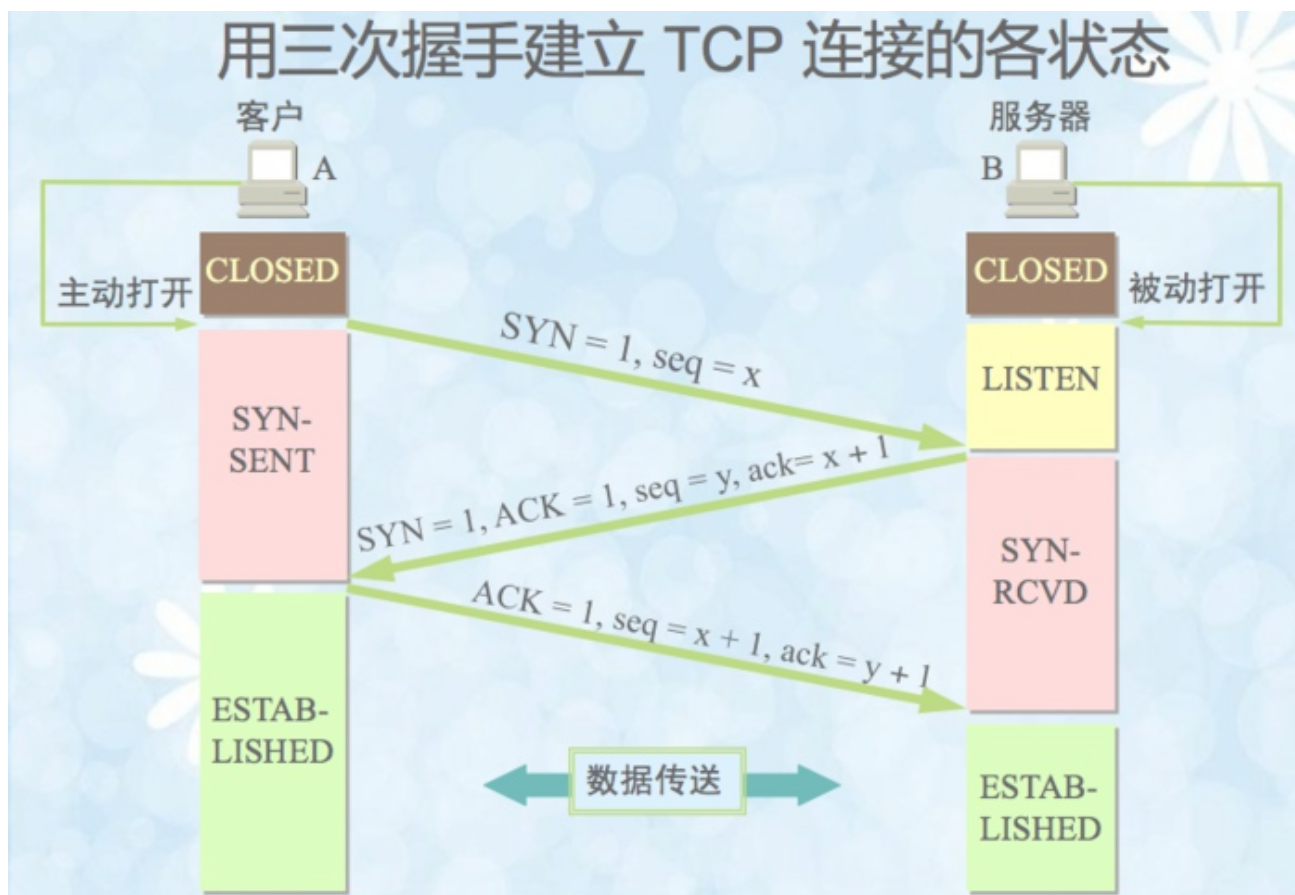
网络协议栈·需要关注的协议



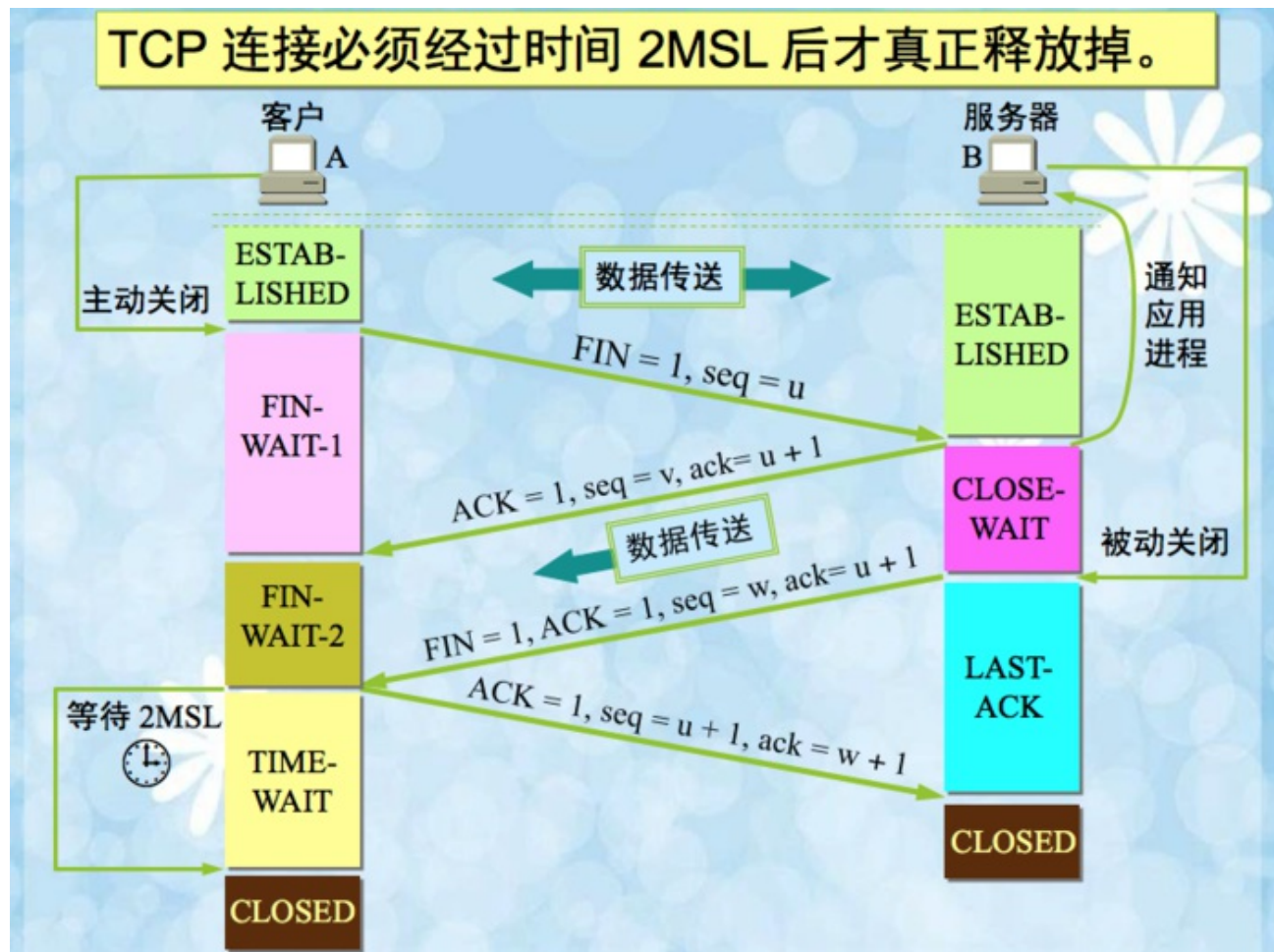
TCP 三次握手



注意“握手”和“挥手”时，TCP 的状态变化。



TCP 四次挥手



数据库

这里重点介绍 MySQL。不过，基础原理性的东西，任何一个数据库都是相通的。

1. ACID
2. 事务隔离级别
3. MVVC
4. 索引的结构
 - a. InnoDB 是聚簇索引
 - b. MyISAM 是 B+树。
5. 查询优化
6. 主从复制原理



这里话题很多，不展开讲了，想深入学习，请看：

1. 《高性能MySQL》——重点看四、五、六章。最少看三遍。
2. 《MySQL技术内幕》
3. 我的学习笔记：[MySQL 学习笔记](#)

NoSQL

1. Redis — 重点学习。支持数据结构更丰富。刚兴趣可以看看内部实现原理，对数据结构和算法理解的更深入。
2. Memcached
3. MongoDB
4. Cassandra
5. HBase — 大数据中使用较多。



《Redis设计与实现》，你值得花时间学习。

建议大家去关注的技术

1. Docker
2. Kubernetes

复习材料精选

Java 基础

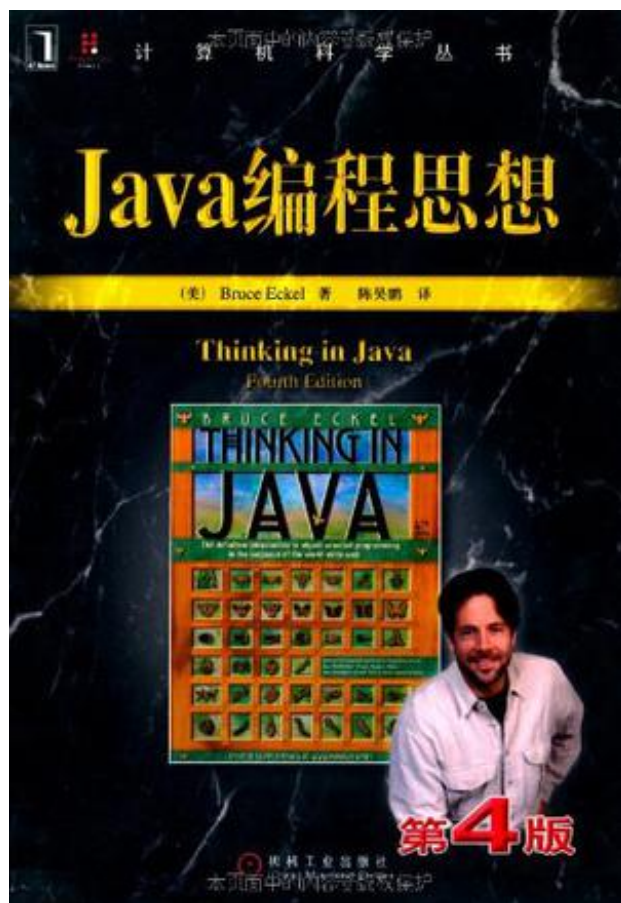
1. 数据结构与算法分析

这本书最大的特点是和 JDK 集合类库贴合的很近。既可以学习数据结构相关知识，还可以借此读一读 JDK 集合类源码。



2. Java编程思想

看集合类、反射、并发相关章节。尤其是并发，非常适合入门。



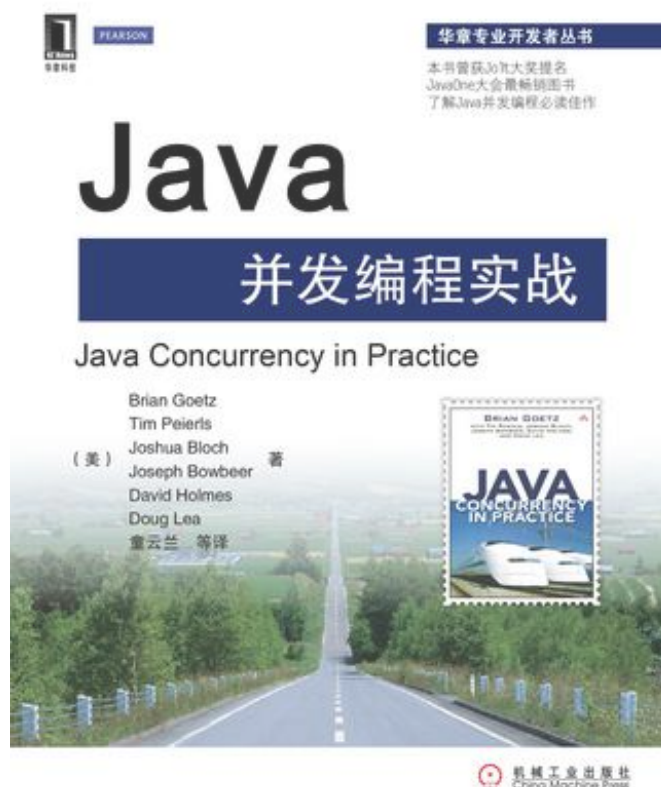
3. Java 8 实战

对 Java 8 中的 Lambda 和 Stream API 做了深入细致的说明。重点学习一下函数式编程思想。



4. Java并发编程实战

重点讲解并发基础理论。适合所有搞开发的从业人员看。



5. 实战Java高并发程序设计

这本书更新一些，讲解了 Java 8 中的很多内容。



6. Java并发编程的艺术

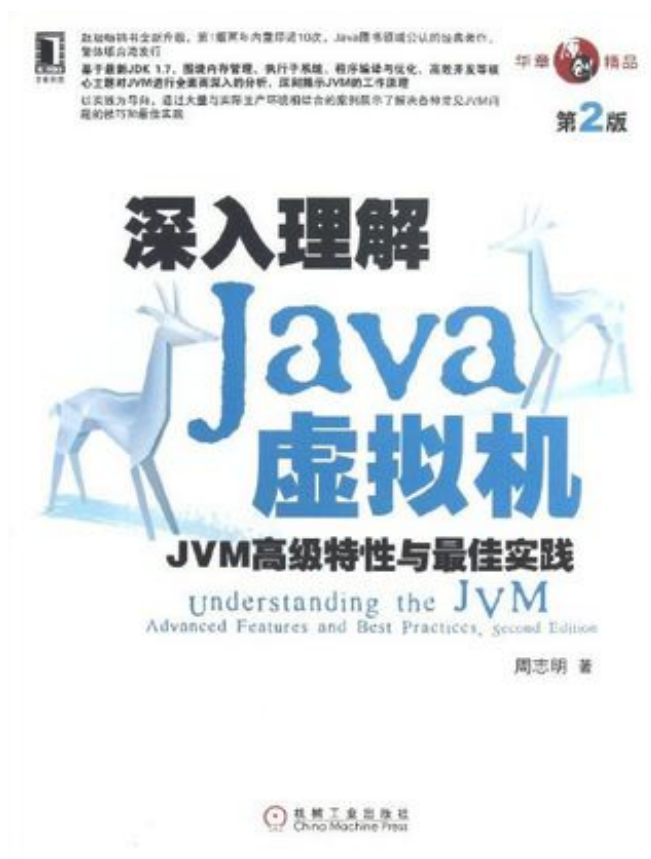
讲解的很细致，也分析了很多 JUC 源码。美中不足是，针对 Java 7 的。目前作者没有升级的计划。



Java 虚拟机

1. 深入理解Java虚拟机

对 Java 虚拟机做了比较全面的介绍。请注意：并发和 Java 虚拟机结合的非常紧密。另外，这本书关于“垃圾回收”算法的图不是很好，请参考 [Java程序性能优化](#) 相关章节的示图。



2. Java性能权威指南

这本书更新，介绍的内容也更新。



设计模式

1. 大话设计模式

幽默风趣又深入浅出。强烈推荐！

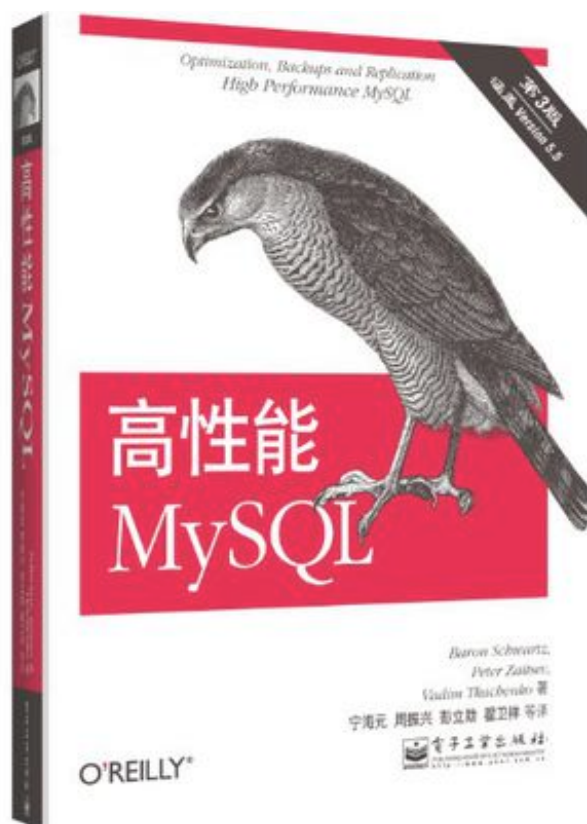


《Head First 设计模式》也非常不错。本人更推荐《大话设计模式》。

MySQL 数据库

1. 高性能MySQL

重点看四、五、六这三章内容。多看两遍。我的笔记：[MySQL 学习笔记](#)。



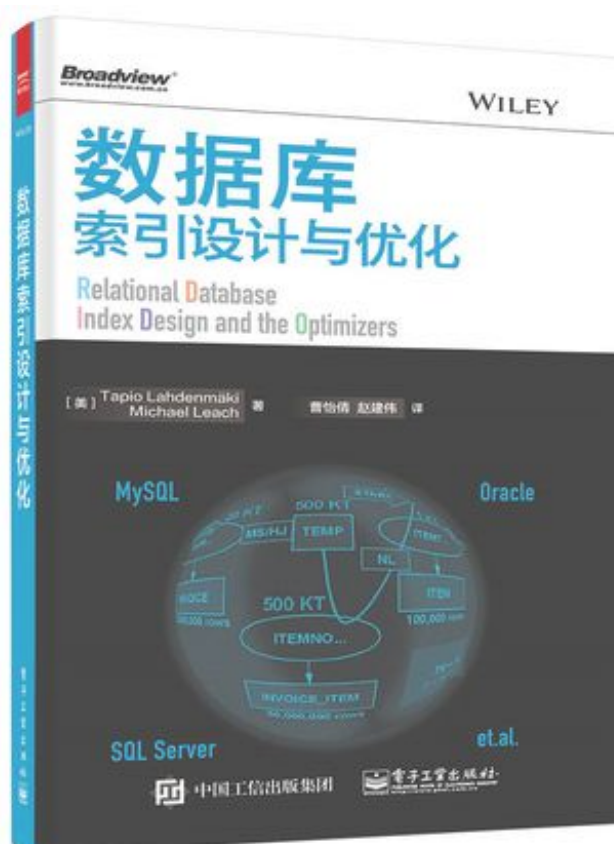
2. MySQL技术内幕

和《高性能MySQL》相互参考着看，效果更好。



3. 数据库索引设计与优化

从各个角度计算一次查询的开销，然后针对性地优化。



计算机网络

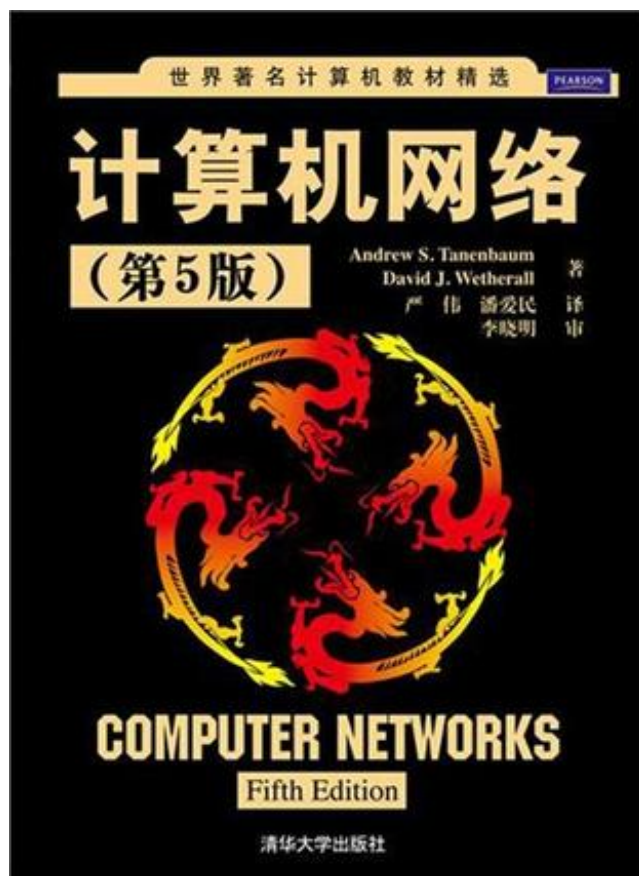
1. 计算机网络-自顶向下方法

自顶向下来讲解网络协议，比较直观易懂。适合入门。



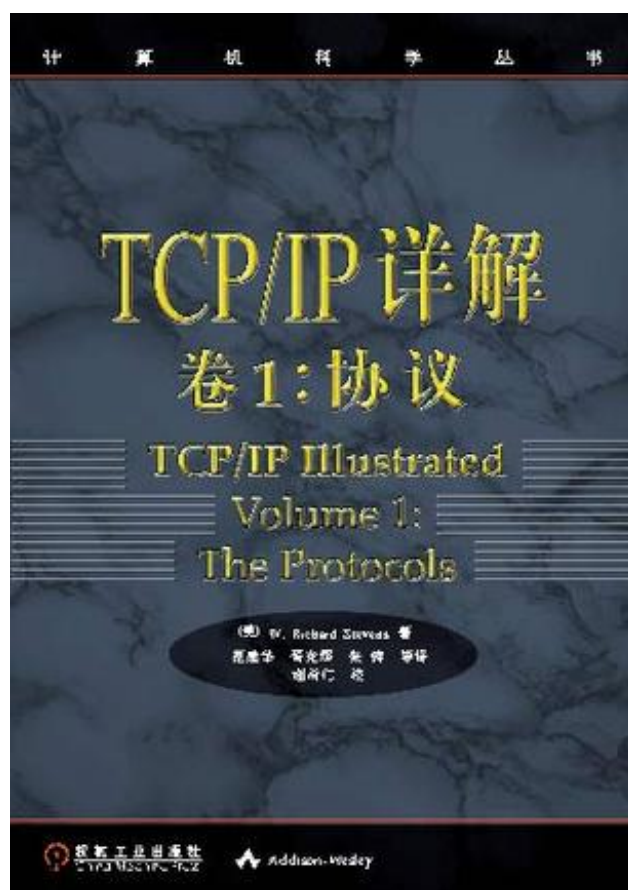
2. 计算机网络

Tanenbaum 老爷子的书，光看人名就值得入手。这本和上面那本，选一本入门就可以了。



3. TCP/IP详解 卷1：协议

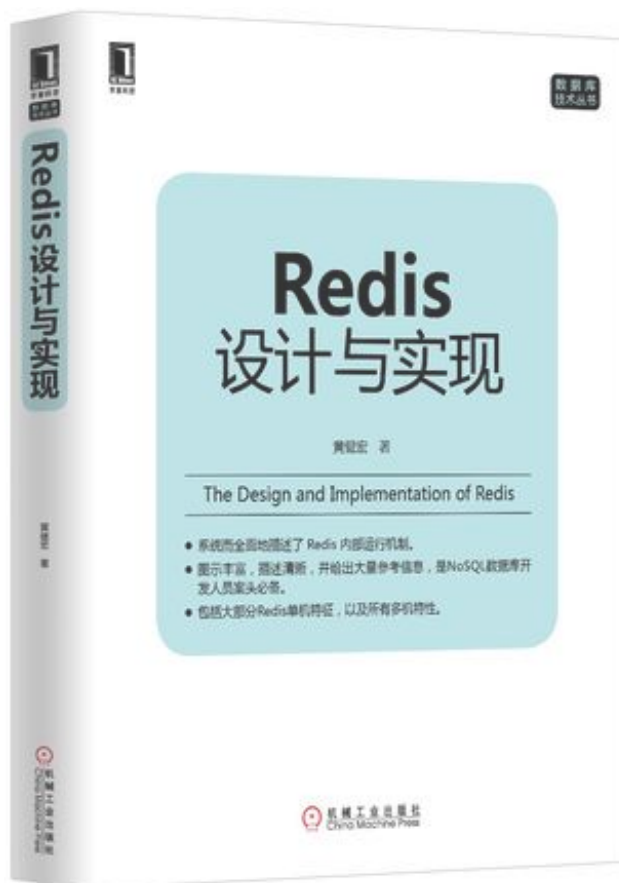
看了这本书就可以当黑客了。 高级内容。



NoSQL

1. Redis设计与实现

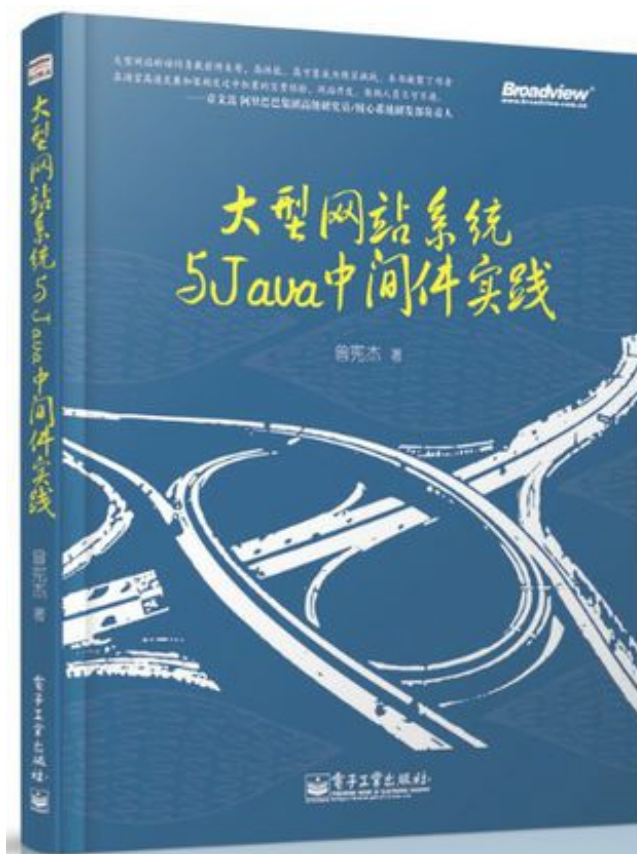
很细致地讲解了 Redis 的内部实现。



架构

1. 大型网站系统与Java中间件开发实践

结合 Dubbo 的文档看起来效果更好。了解一个网站，从小到大的发展历程，才能更好地理解它们选用的技术方案。



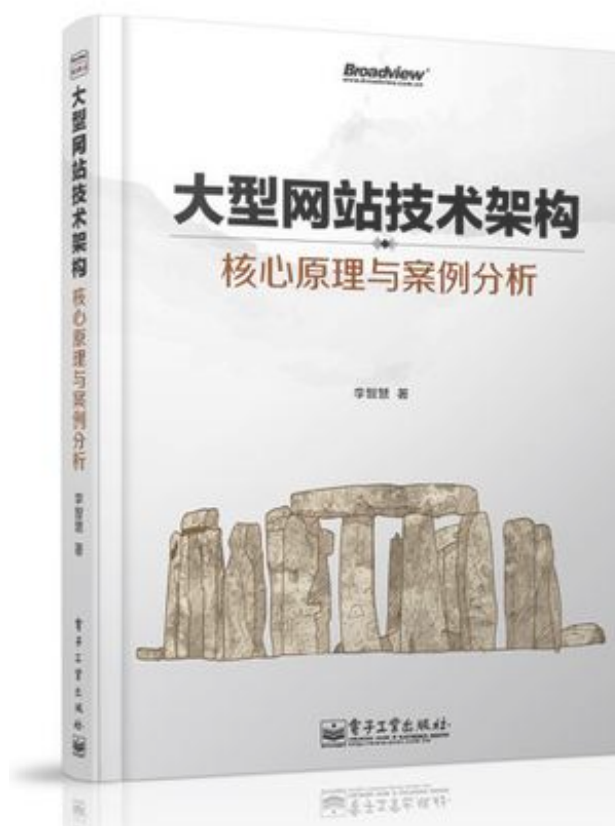
2. 亿级流量网站架构核心技术

对大型网站架构中各个环节有了深入地说明。



3. 大型网站技术架构

从不同维度介绍了大型网站需要考虑的方方面面的特性。



三个月面试重点

上面的书，琳琅满目，完整阅读下列，时间有点长。这里重点整理一下，短时间突击的内容。

1. Java 集合类

- a. 推荐看看《Java 编程思想》中关于容器的两章
- b. 类的继承体系结构
- c. 相同接口不同实现的区别
- d. 常用类的源码
 - i. ArrayList
 - ii. LinkedList
 - iii. HashSet
 - iv. TreeSet
 - v. HashMap
 - vi. TreeMap

2. Java 并发

- a. 推荐看看《Java 编程思想》第21章并发
- b. 常用类的源码
 - i. ThreadPoolExecutor
 - ii. ConcurrentHashMap
 - iii. ArrayBlockingQueue
 - iv. AbstractOwnableSynchronizer — 整个 Java 并发库的基础。看懂这个，就很懂三分之一的并发库了。不过，代码量很大，也很抽象。
 - v. ForkJoinPool — 这里的思想也很精巧。有时间可以看，没时间可以稍后推一推。



《Java 编程思想》已经是十年前的书了，所以最新的一些 Java 并发库演进并没有体现出来。可以看《实战Java高并发程序设计》。不过，注意时间，你只有三个月的时间。

3. Java I/O

- a. BIO
- b. NIO
- c. AIO



先理解三个模式的优缺点和思路；有时间可以看看 Netty 的封装。

4. Java 动态代理

- a. 熟悉代理模式
- b. 了解代理模式的缺点
- c. 动态代理
- d. Proxy 源码 — 有时间可以看看



马士兵有几节视频专门讲到动态代理模式，可以看看。参考下载链接: <https://pan.baidu.com/s/1c1R8l48> 密码: rnm9。

5. Java 反射

- a. 熟悉 API，多实践。
- b. Reflections — 所有时间可以看看这个类库的使用，甚至源码实现。

6. 设计模式

- a. 单例模式
- b. 工厂模式
- c. 代理模式
- d. 装饰模式
- e. 适配器模式

f. 观察者模式

g. 模板方法模式

h. 看《大话设计模式》中相关章节

7. 数据结构和算法

a. 链表基本操作

b. 二分查找

c. 归并排序

d. 快速排序

e. 树的遍历

8. Spring 的原理

a. IoC

b. AOP

i. 动态代理

ii. 修改字节码

9. 计算机网络 — 选学

a. HTTP

i. 长连接

ii. 管道化

iii. 缓存

b. TCP

i. 三次握手

ii. 四次挥手

iii. 慢启动

iv. 滑动窗口

10. 数据库 — 重点

- a. 《高性能 MySQL》重点看四、五、六这三章内容。
- b. [MySQL 学习笔记](#)

面试准备流程

1. 功夫下在平时，多积累多实践，多撸代码，多思考，多总结；
2. 功夫下在平时，多积累多实践，多撸代码，多思考，多总结；
3. 功夫下在平时，多积累多实践，多撸代码，多思考，多总结；
4. 退而求其次，以面试为目标，高效复习，争取更好的Offer。
 - a. 抽三个月的时候，全身心学习上述资料；
 - b. 先学习核心知识点；尤其是大的互联网公司，尤其对并发、网络 I/O 问的比较多。
 - c. 数据库也很重要，把我前面提到的重点内容看了，也有可能成为你的闪光点！
 - d. 再学习加分项
5. 上网收集面试题，查漏补缺，自行预热
6. 收集相关信息
 - a. 企业重点业务
 - b. 企业的技术体系
 - c. 对企业情况预热
 - d. 整理相关疑问
7. 整理简历
 - a. 针对不同岗位，侧重不一样，书写不同的简历
8. 投递简历
 - a. 首选内推
 - b. 其次拉勾上投
 - c. 能力出众、业绩漂亮的，可以上 100Offer 试试。

准备简历 — 如何写好技术简历?

简历的意义

简历的存在只有一个目的 —— 帮你约到面试。只要能达到这个目的，简历可以是一段视频，一个开源项目，一张照片，甚至是一行字，比如：

“*I wrote python.*

— Guido van Rossum

简历，不光要介绍你自己，要推销你的自己才行。

一份好的简历，要低调的告诉招聘方，爷很NB。

和大家分享两个技巧：

首先，一份好的简历不光要说明事实，更要通过FAB法则来增强其说服力。

- Feature：是什么
- Advantage：比别人好在哪些地方
- Benefit：如果雇佣你，招聘方会得到什么好处

其次，写简历和写议论文不同，过分的论证会显得自夸，反而容易引起反感，所以要点到为止。这里的技巧是，提供论据，把论点留给阅读简历的人自己去得出。

工具推荐

用 AsciiDoctor 写，用 PDF 发。

推荐工具：

1. AsciiDoctor

推荐使用 Atom 编辑器，装好插件，语法高亮，支持预览。

上述是 AsciiDoctor 的官网，我翻译了一些它的文档，初学看“语



法快速参考”即可：[AsciiDoc 语法快速参考](#)。

中文网：[Asciidoctor 中文网](#)

2. wkhtmltopdf



wkhtmltopdf 推荐使用 0.12.3 版，在 Mac 系统上，最新版本在转化时有问题。

用 AsciiDoc 格式书写，使用 Asciidoctor 转换成 HTML，再使用 `wkhtmltopdf` 将 HTML 转化成 PDF 格式。

面试技巧

1. 以招聘者的心态来思考面试中的问题；

面试常问的技术之外的问题

1. 介绍一下你自己
2. 未来三五年的规划是什么?
3. 你的优缺点是什么?
4. 你还有什么想问的吗?
5.

如何谈待遇？

1. 找人打听公司情况：薪资、职级、其他福利待遇；
2. 上拉钩上，看相关工作的人的待遇。
3. 根据面试效果来说谈待遇

一些善意的提醒

1. 一定要去一流的大互联网公司发展几年；
 - a. 人脉；
 - b. 机会；
 - c. 视野；
 - d. 技术深度；
 - e. 背景光环；
2. 选择一个方向，深入专研下去；
3. 常见的算法要可以手写出来；

一些题外话

1. 学好英语，读英文原版文档；
2. 学好英语，读英文原版文档；
3. 学好英语，读英文原版文档；
4. 认清时代的发展，选择适应时代发展的方向去努力；
5. 多读书，多实践，多思考；
6. 读论文、RPC；
7. 一定要坚持写自己的博客

8. 针对一个主题，深入挖掘；
9. 多阅读源码：Spring 0.9、MyBATIS、JUnit 3.8.2、Dubbo
10. 工作之余多造“轮子”——把工作中用到的框架，自己都仿制一下。
 - a. DAO 框架
 - i. 自己造轮子
 - ii. 对原有轮子再封装：比如对 MyBATIS 的增强包 [MyBATIS Plus](#)。
 - b. IoC 框架
 - c. AOP 框架
 - d. 测试框架
 - e. 分布式服务框架
 - f. Java Web 容器

参考资料

1. 写给前端面试者
2. 如何写好技术简历 —— 实例、模板及工具
3. 谈Offer15法(15 Rules for Negotiating a Job Offer)
4. . 《JAVA面试题解惑系列》 — 对很多问题讲解的很深入。不过，这个系列专题有些解释有点陈旧了。
5. [Java 集合系列目录](#)
6. [Java 集合系列](#)
7. [java io系列01之 "目录"](#)
8. [数据结构与算法系列](#)
9. [HTTP协议详解](#)
10. [Fiddler 教程](#)
11. [TCP三次握手和四次挥手深入实践 - 简书](#)
12. [通俗大白话来理解TCP协议的三次握手和四次分手](#)
13. [wkhtmltopdf Man Page](#)

Last updated 2017-12-26 16:11:47 CST