

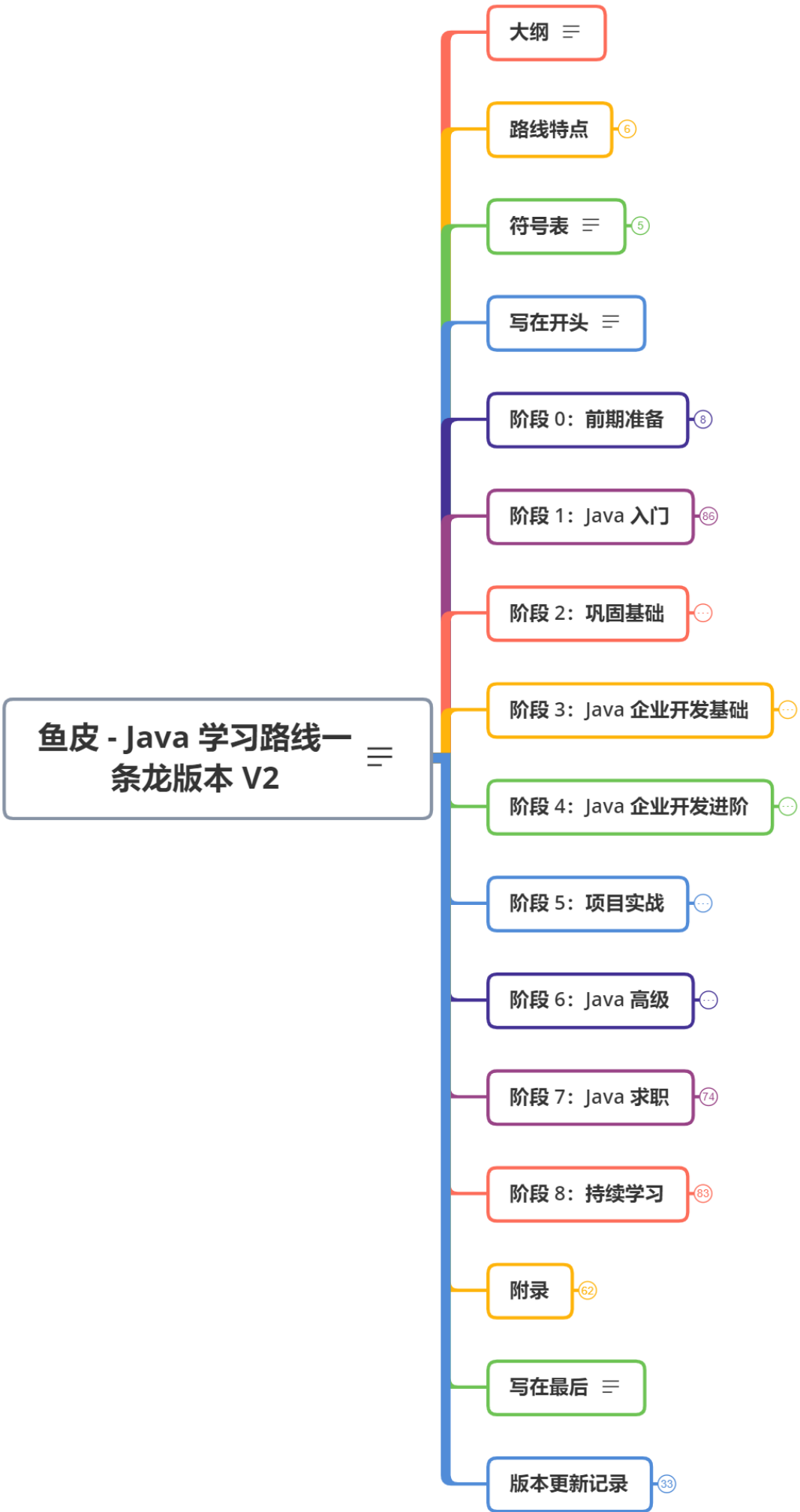
Java 学习路线

Java 学习路线一条龙版 by 程序员鱼皮



建议先观看视频导读：<https://www.bilibili.com/video/BV1Qf4y1K7ff/>

大纲



路线特点

1. 最新，完整一条龙的大厂 Java 学习路线，从入门到入土
2. 同时适用于想全面学习 / 快速求职的同学，可以根据符号来定制自己的专属学习路线
3. 给出目标、学习建议、关键知识点、最优资源以及各类资源推荐（视频、书籍、文档、项目、工具等）
4. 鱼皮自己是 Java 开发者，融入个人自学和从业的经历 + 思考，让路线有灵魂
5. 划分阶段、更有计划，且在最后给出持续学习的方向、探索 Java 程序员发展的无限可能
6. 完全开源，回馈社区，持续更新！

符号表

可以通过路线知识点前的表情字符，根据自己的实际情况选择学习：

- 🍷 所有同学必须学习！！！！
- 🍷 非常急着找工作，才可不学；目标大厂，必须学习！
- 🍷 急着找工作的话，可不学；目标大厂，建议学习
- 🍷 时间充足的话，再去学
- ☆ 表示推荐资源

写在开头

首先呢，我们要了解 Java 的应用场景和就业方向，看看和自己的学习目的是否一致。

目前，Java 由于其优秀的特性以及火爆的生态，岗位需求量大，是后台开发、安卓开发、大数据开发的主流编程语言，因此学习资源也很丰富，值得学习！

有关更多编程语言的选择，大家可以看我的视频了解：<https://www.bilibili.com/video/BV1y3411r7pX/>

阶段 0：前期准备

1. 准备好一款在线、随时随地写代码的工具，不用在本地去安装和配置，是初学者的好帮手。
 - 推荐网站 - 菜鸟工具：<https://c.runoob.com/compile/10>
2. 准备一款记笔记的软件，学编程的过程中老师的思路、自己遇到的问题、解决问题的方法、心得感悟、遇到的好资料，都要记下来哦，好记性不如烂笔头，可以帮助你事半功倍。
 - 推荐本地软件 Typora：<https://www.typora.io/>（线上可以用 xx 文档）
 - 推荐使用 Markdown 语法编写，鱼皮教程：<https://www.bilibili.com/video/BV1no4y1k7YT/>
3. 想学好编程，别背代码，要多写，把这句话刻到脑子里。
4. 选视频和找资料的小技巧：多看评论区、多看隔壁评论区、不要看过期的视频。
5. 如何访问 GitHub：<https://mp.weixin.qq.com/s/uBFsuThsZ4kdkZgqfX12qA>（上面有很多免费的编程学习资源、项目源码，程序员必备）

阶段 1：Java 入门

目标

培养兴趣、快速上手，能运行和编写简单的 Java 程序。

学完本阶段后，可以试着用 Java 解决一些数学计算问题、编写图书管理系统等桌面端 GUI 程序，甚至是五子棋之类的小游戏。

Java 编程基础（45 天）

知识

- Java 特点（看不懂没事，别背！）
- 环境搭建
- IDEA 开发工具
 - 新建项目
 - 运行调试
 - 界面配置

- 插件管理
- **Java 基础语法**
 - 数据类型
 - 流程控制
- 数组
- **面向对象**
 - 方法
 - 重载
 - 封装
 - 继承
 - 多态
- 抽象类
- 接口
- 枚举
- 常用类
 - String
 - 日期时间
- 集合类
- 泛型
- 注解
- 异常处理
- 多线程
- IO 流
- 反射

学习建议

初学一门语言时，一定要持续学习，不能中断！

想要学好编程，一定要多敲代码！建议先跟着书上的例子敲一遍代码，然后试着自主编写代码，并完成课后练习。

不理解代码也没关系，可以学习 Debug 后，一行一行地打断点执行，查看程序的执行过程。千万不要觉得麻烦，养成习惯后真的能节省很多重复学习的时间。

经典面试题

1. 为什么重写 equals 还要重写 hashCode?
2. == 和 equals 比较的区别
3. 为啥有时会出现 $4.0 - 3.6 = 0.40000001$ 这种现象?
4. final 关键字的作用
5. 介绍 Java 的集合类
6. ArrayList 和 LinkedList 的区别

资源

- 视频
 - ☆ 韩顺平 - 零基础 30 天学会 Java: <https://www.bilibili.com/video/BV1fh411y7R8> (900 多集, 顺序安排很合理, 每个知识真正的打碎了, 通俗、有示例、有实战、有思想)
- 文档
 - ☆ 菜鸟教程: <https://www.runoob.com/java/java-tutorial.html> (可以在线练习)
 - 廖雪峰 Java 教程: <https://www.liaoxuefeng.com/wiki/1252599548343744>
 - IDEA 中文教程: <https://github.com/judasn/IntelliJ-IDEA-Tutorial>
 - IDEA Mac 快捷键指南, 地址: <https://www.jianshu.com/p/454c71172c46>
 - IDEA Win 常用快捷键, 地址: <https://www.jianshu.com/p/5de7cca0fefc>
- 书籍
 - 《Head First Java》
- 在线游戏
 - Codegym: <https://codegym.cc/zh> (玩玩前几关培养兴趣不错, 但后面收费了)

Java 8 (3 天)

知识

- Stream API
- Lambda 表达式
- 新日期时间 API
- 接口默认方法

学习建议

Java 8 是如今企业开发中最主流的 Java 稳定版本，在这个版本出现了很多实用的新特性，虽然面试考点不多，但能够提升编程效率，建议学习。

经典面试题

1. java 8 有哪些新特性？
2. HashMap 在 jdk 1.7 和 1.8 的区别？

资源

- 视频
 - ☆ 宋红康 - 全网最全Java零基础入门教程：<https://www.bilibili.com/video/BV1Kb411W75N>（只看 Java 8 部分即可）
- 文档
 - ☆ 菜鸟教程：<https://www.runoob.com/java/java8-new-features.html>
- 书籍
 - 《Java 8 实战》
- 工具
 - ☆ 在线编写运行 Java 8：https://www.tutorialspoint.com/compile_java8_online.php
- 其他
 - Java 8 小代码片段：<https://github.com/biezhi/30-seconds-of-java8>

练手项目

- Java 实现简单计算器：<https://www.lanqiao.cn/courses/185>
- Eclipse 实现 Java 编辑器：<https://www.lanqiao.cn/courses/287>
- 一本糊涂账：<https://how2j.cn/module/104.html>
- Java 五子棋：<https://blog.csdn.net/cnlht/article/details/8176130>
- Java 中国象棋：<https://blog.csdn.net/cnlht/article/details/8205733>
- JAVA GUI 图书馆管理系统：<https://github.com/uboger/LibraryManager>
- JAVA 坦克大战小游戏：<https://github.com/wangzhengyi/TankWar>
- Swing 编写的俄罗斯方块：<https://github.com/HelloClyde/Tetris-Swing>
- 小小记账本：<https://github.com/xenv/SmallAccount>（适合了解数据库的同学）

尾声

学完了 Java 基础后，有些同学会感到迷茫了啊，感觉好像啥也做不出来，不知道下一步做什么，我这一身的本领该如何施展啊？

不要慌，也不要急着去学新技术，接下来我们要多用 Java 来写代码了，巩固基础，但是写什么呢？

当然是数据结构和算法！

阶段 2：巩固基础

目标

想学好编程，计算机基础知识要学好。

比如算法，是程序员的灵魂。学好算法有助于我们理解程序、开拓思路，因此也是很多公司面试时考察的关键，在找工作前，还是要刷个上百道算法题目的。

我们这个阶段的目标是：熟练使用 Java 语言来编写程序，巩固 Java 基础。（那直接用 Java 来写算法题目，一举两得，岂不美哉？）

此外，建议大家利用零碎时间多去了解 **计算机基础知识**，比如操作系统、计算机网络等，对你后面学习开发框架之类的知识都有帮助。

Java 基础（30 天）

学习建议

建议大家去阅读《Java 核心技术卷 1》，这本书堪称经典，是帮助你复习巩固 Java 的不二之选，其中图形界面章节可以选择不看。

之后可以刷牛客的 Java 练习题，检验自己的水平，我当时刷了两遍 1000 题（每天 30 题，1 个月也就刷完了，二刷会更快！）。虽然都是选择题，但能学到很多 Java 语言的特性、避免写代码时容易犯的错误。

如果你时间不够，只是想快速找到工作，那么本章节甚至可以完全跳过，后面慢慢弥补即可。

资源

- 书籍
 - ☆ 《Java 核心技术卷 1》
- 练习
 - ☆ 牛客题库: <https://www.nowcoder.com/intelligentTest>

数据结构与算法

知识

- 时间 / 空间复杂度分析
- 数据结构
 - 数组
 - 字符串
 - 队列
 - 栈
 - 链表
 - 集合
 - 哈希表
 - 树
 - 二叉树
 - 多叉树
 - 字典树
- 算法
 - 排序
 - 双指针
 - 查找
 - 分治
 - 动态规划
 - 递归

- 回溯
- 贪心
- 位运算
- DFS
- BFS
- 图

学习建议

学算法一定要多思考、多练习!!!

在复习 Java、巩固基础的过程中，每天可以坚持用 Java 做 2 - 3 道算法题目。

不用担心看不懂，直接进入 LeetCode 学习板块 LeetBook，提供了免费的教程，文字、图解、动画讲算法、在线练习应有尽有，从 0 开始，跟着学习基础知识、跟着教程刷一些同类题目，培养算法思路。

之后再由简单到复杂、由通过率高到低，自主刷题，多看题解，举一反三。一定要把自己做过的每道题目代码都自行保存好，定期复习。时间足够的话，也可以试着自己写写题解，在网上分享一下自己的刷题思路，找到自己坚持的动力。

有条件的朋友也可以试着参与竞赛，没什么门槛，可以帮助你集中精神，而且周赛的题目还是挺友好的~

经典面试题

- LeetCode 精选 100 道: <https://leetcode-cn.com/problem-list/2cktkvj/>
- LeetCode 精选算法 200 题: <https://leetcode-cn.com/problem-list/qg88wci/>

资源

- 文章
 - ☆ 刷了 1000 多道算法题，一点心得: <https://t.1yb.co/oT9q> (鱼皮原创)
- 在线教程
 - ☆ LeetCode LeetBook: <https://leetcode-cn.com/leetbook/>
- 在线刷题
 - ☆ LeetCode: <https://leetcode-cn.com/>

- 书籍
 - ☆ 《小灰的漫画算法》
 - ☆ 《剑指 Offer》
 - 《程序员代码面试指南》
 - 图解算法数据结构: <https://leetcode-cn.com/leetbook/detail/illustration-of-algorithm/>
 - LeetCode 101 (C++) : https://github.com/changgyhub/leetcode_101
 - LeetCode 题解 (C++) : <https://github.com/soulmachine/leetcode>
 - LeetCode Cookbook (Go 语言) : <https://github.com/halfrost/LeetCode-Go>
- 视频
 - 尚硅谷 Java 数据结构与算法: <https://www.bilibili.com/video/BV1E4411H73v>
(难度比面试的要求大一些, 适合希望更全面学习的朋友)
 - Leetcode 真题解析: <https://www.bilibili.com/video/BV1a54y1b74k>
- 工具
 - VisuAlgo 数据结构和算法动态可视化: <https://visualgo.net/zh>
 - 数据结构可视化: <https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>
 - RegExr: <https://www.code-nav.cn/rd/?rid=79550af2601114e9012110711798772b> (学习、创建和测试正则表达式的在线可视化工具)
- 网站
 - 五分钟学算法: <https://www.cxyxiaowu.com/>

🌐 计算机导论

知识

- 计算机发展历史
- 计算机应用领域
- 计算机发展方向
- 计算机基本组成
- 二进制
- 编程语言发展

学习建议

大学计算机专业的同学一般刚开学就会上这门课，虽说学习它并不会直接提高你的编程技能，但能够让你更了解计算机和编程，从而在一定程度上帮助你培养学习兴趣、确定学习方向。

自学的话，不用刻意去学习计算机导论，而是可以通过看视频、阅读课外读物的方式慢慢地了解计算机的故事。

资源

- 视频
 - 《计算机科学速成课》：<https://www.bilibili.com/video/BV1EW411u7th>（从底层到上层的计算机知识科普，强烈推荐）
- 书籍
 - 《半小时漫画计算机》

操作系统

知识

- 操作系统的组成
- 进程、线程
- 进程 / 线程间通讯方式
- 进程调度算法
- 进程 / 线程同步方式
- 进程 / 线程状态
- 死锁
- 内存管理
- 局部性原理

学习建议

说实话，操作系统这一块知识挺枯燥的。你说说我现在项目都不会做，你又让我看这些理论，是不是想让我头秃？

我的建议是，可以先利用课余时间看一些网课或者有趣的课外书，对一些操作系统的概念先有个大致的印象，比如进程、线程、死锁，等后面有时间了再系统学习、等到找工作了再去背相关八股文。

还在校园就跟着学校的进度学习就行，自学的话可以看下《清华操作系统原理》视频，有实力的小伙伴，能看懂大黑书就更好了，但如果看不懂也别担心，这并不影响你后续知识的学习。

经典面试题

1. 什么是死锁？死锁产生的条件？
2. 线程有哪几种状态？
3. 有哪些进程调度算法？
4. 什么是缓冲区溢出？

资源

- 视频
 - 《清华操作系统原理》：<https://www.bilibili.com/video/BV1uW411f72n>
- 书籍
 - 《编码》
 - 《30天自制操作系统》
 - 《现代操作系统》（难度较大，不推荐新手看）
 - 《深入理解计算机系统》（难度较大，不推荐新手看）
 - 《自己动手写操作系统》（国产好书，网上可以下载）
- 大学课件
 - 浙大操作系统：<https://github.com/QSCTech/zju-icicles>

计算机网络

知识

- 网络分层模型
- 网络传输过程
- IP、端口
- HTTP / HTTPS 协议

- UDP / TCP 协议
- ARP 地址解析协议
- 网络安全
- DNS 域名解析

学习建议

很多学习 Java 开发的同学最后都是从事 **后端开发** 的工作，而计算机网络知识是后端开发的重点。

和操作系统一样，自学网络可能会很枯燥，建议先看有趣的课外书，比如《图解 HTTP》；或者有趣的视频，比如《计算机网络微课堂》。后面要找工作面试前，再重点去背一些八股文就好了。还在学校的同学好好上课一般就没问题。

学习基础能帮助自己今后发展更稳定，且更容易接受新知识，所以请不要相信基础无用论。

经典面试题

1. 计算机网络各层有哪些协议？
2. TCP 和 UDP 协议的区别？
3. TCP 为什么需要三次握手和四次挥手？
4. HTTP 和 HTTPS 协议的区别？

资源

- 视频
 - 《计算机网络微课堂》：<https://www.bilibili.com/video/BV1c4411d7jb>
- 书籍
 - ☆《图解 HTTP》
 - 《网络是怎样连接的》
 - ☆《图解 TCP / IP》
- 大学课件
 - 浙大计算机网络基础：<https://github.com/QSCTech/zju-icicles>

尾声

巩固基础要花至少 1 个月的时间，当你读完《Java 核心技术卷1》并且不用查询文档也能熟练地用 Java 做题时，就可以接着往下了。

阶段 3：Java 企业开发基础

目标

面向薪资编程，学习实际后台开发工作要用的基础技术和框架，并能 **独立** 做出一个具有完整功能的 Java Web 项目。

学完这个阶段后，你应该已经能独立开发出大多数常见的后台系统了，比如各种管理系统、商城系统等。

MySQL 数据库（7 天）

企业中大部分业务数据都是用关系型数据库存储的，因此数据库是后台开发同学的必备技能，其中 MySQL 数据库是目前的主流，也是面试时的重点。

知识

- 基本概念
- MySQL 搭建
- SQL 语句编写
- 约束
- 索引
- 事务
- 锁机制
- 设计数据库表
- 性能优化

学习建议

其中，**SQL 语句编写** 和 **设计数据库表** 这两个能力一定要有！

比如让你做一个学生管理系统，你要能想到需要哪些表，比如学生表、班级表；每个表需要哪些字段、字段类型。

这就要求大家多写 SQL、多根据实际的业务场景去练习设计能力。

经典面试题

1. MySQL 索引的最左原则
2. InnoDB 和 MyISAM 引擎的区别？
3. 有哪些优化数据库性能的方法？
4. 如何定位慢查询？
5. MySQL 支持行锁还是表锁？分别有哪些优缺点？

资源

- 视频
 - ☆ 老杜 - mysql入门基础 + 数据库实战: <https://www.bilibili.com/video/BV1Vy4y1z7EX> (较新、内容相对精炼, 有习题)
 - 尚硅谷 - MySQL基础教程: <https://www.bilibili.com/video/BV1xW411u7ax> (小姐姐讲课, 但感觉音质一般)
- 在线练习
 - ☆ SQL 自学网: <http://xuesql.cn/>
 - ☆ SQL 在线运行: <https://www.bejson.com/runcode/sql/>
- 文档
 - SQL - 菜鸟教程: <https://www.runoob.com/sql/sql-tutorial.html>
 - MySQL - 菜鸟教程: <https://www.runoob.com/mysql/mysql-tutorial.html>
- 网站
 - [数据库大全](#): 果创云收录的各种数据库表设计

🔗 开发框架 (60 天)

Java 之所以能成为主流的企业开发语言，很大一部分原因是它完善的框架生态，用好框架，不仅能够大大提升开发效率，还能提高项目的稳定性、减少维护成本。

开发框架是后台开发工作中不可或缺的，也是面试考察的重点，一定要好好学！

不知道 Java 能做什么的朋友们，学完开发框架，就会有答案啦。

下面给大家推荐的都是企业中应用最多的主流开发框架，知识点比较零碎，就放在一起讲了。

知识

🔗 Java Web

- 描述：Java 网页应用开发基础
- 一丢丢前端基础
- XML
- JSON
- Servlet
- Filter
- Listener
- JSP
- JSTL
- Cookie
- Session

🔗 Spring 5

- 描述：Java 轻量级应用框架
- IOC
- AOP
- 事务

🌀 SpringMVC

- 描述：Java 轻量级 web 开发框架
- 什么是 MVC?
- 请求与响应
- Restful API
- 拦截器
- 配置
- 执行过程

🌀 MyBatis

- 描述：数据访问框架，操作数据库进行增删改查等操作
- 增删改查
- 全局配置
- 动态 SQL
- 缓存
- 和其他框架的整合
- 逆向工程

🌀 MyBatis Plus

- 描述：Mybatis 的增强工具，能够简化开发、提高效率
- 引入
- 通用 CRUD
- 条件构造器
- 代码生成器
- 插件扩展
- 自定义全局操作

🌀 SpringBoot 2

- 描述：简化 Spring 应用的初始搭建以及开发过程，提高效率
- 常用注解
- 资源整合
- 高级特性
- 本地热部署

Spring Security

- 描述：Spring 的安全管理框架
- 用户认证
- 权限管理
- 相关技术：Shiro

Maven / Gradle

- 描述：项目管理工具
- 构建
- 依赖管理
- 插件
- 配置
- 子父工程
- 多模块打包构建
- Nexus 私服搭建

学习建议

由于技术较多，且框架之间存在一定的联系，因此建议大家看同一系列的视频教程（尚硅谷、狂神说等等都可以），以保证学习内容的连续以及体验上的一致。

学这些技术的时候，**千万不能懒**！一定要多记笔记，并且跟着老师写代码。原理部分不要太过纠结，先以能跟着敲出代码、写出可运行的项目为主，有些东西做出来也能帮助你更好地理解理论。

学习顺序挺重要的，建议按我推荐的顺序学，不要一上手就学 Spring Boot。只有先学习下自己整合框架的方法，才能帮你理解 SpringBoot 解决的问题，感受到它的方便和高效。

Maven / Gradle 当成工具用就好，面试基本不问，跟着框架教程去用就行了，急着找工作的话，先不用花太多时间去深入学。大厂面试问这个的也不多。

经典面试题

1. Spring 的 IOC 和 AOP 是什么，有哪些优点？
2. Spring 框架用到了哪些设计模式？
3. 介绍 Spring Bean 的生命周期
4. MyBatis 如何实现延迟加载？
5. 介绍 MyBatis 的多级缓存机制

资源

- 视频（按顺序看）
 - ☆ 尚硅谷最新版 JavaWeb 全套教程：<https://www.bilibili.com/video/BV1Y7411K7zz>（前端部分最好也看下）
 - ☆ 尚硅谷 - Spring 5 框架最新版教程（idea版）：<https://www.bilibili.com/video/BV1Vf4y127N5>
 - ☆ 尚硅谷 - SpringMVC 2021 最新教程：<https://www.bilibili.com/video/BV1Ry4y1574R>
 - ☆ 尚硅谷 - MyBatis 实战教程全套完整版：<https://www.bilibili.com/video/BV1mW411M737>
 - ☆ 尚硅谷 - MyBatisPlus 教程：<https://www.bilibili.com/video/BV1Ds411E76Y>
 - Maven 零基础入门教程：<https://www.bilibili.com/video/BV1TW411g7hP>（搞不懂 Maven 可以看看）
 - ☆ 雷丰阳 2021 版 SpringBoot2 零基础入门：<https://www.bilibili.com/video/BV19K4y1L7MT>
 - 尚硅谷 - SpringSecurity 框架教程：<https://www.bilibili.com/video/BV15a411A7kP>

🎯 开发规范（3 天）

开发不规范，同事两行泪。

开发规范是团队开发中必须遵守的，有利于提高项目的开发效率、降低维护成本。

知识

- 代码规范
 - 代码风格
 - 命名
 - 其他规则
- 代码校验 (CheckStyle)
- 提交规范

学习建议

有时间的话，简单过一遍大厂团队的代码规范手册就好了，以后做项目的时候能想起来的话就去使用，或者从书中、网上查规范文档，再去遵守。

项目做得多了，自然会养成好的习惯，不用刻意去记（毕竟每个团队规范也不完全相同，背了也没用）。也可以直接利用开发工具自带的一些代码检查插件，帮忙养成好的编码习惯。

资源

- 书籍
 - ☆ 阿里巴巴 Java 开发手册：<https://developer.aliyun.com/graph/java>（搜索《Java开发手册》）
- 视频
 - 华山版《Java开发手册》独家讲解：<https://developer.aliyun.com/live/1201>
- 文档
 - Google Java Style Guide：<https://google.github.io/styleguide/javaguide.html>（谷歌 Java 代码规范）

Git (3 天)

此前大家可能听说过 GitHub，一流的代码开源托管平台。

Git 和它可不一样，是一个版本控制工具，可以更好地管理和共享项目代码，比如把自己的代码传到 GitHub 上、或者从远程下载。

无论自己做项目、还是团队开发，Git 都是现在不可或缺的神器。

知识

- 区分 Git 和 GitHub
- 工作区
- 分支
- 代码提交、推送、拉取、回退、重置
- 分支操作
- 代码合并、解决冲突
- 标签
- cherry-pick
- Git Flow
- 相关技术：SVN（比较老）

学习建议

每个命令跟着敲一遍，有个大致的印象，会用即可。

建议平时大家可以多把自己的代码使用 Git 命令上传到 GitHub 上，用的多了自然就熟悉了。

经典面试题

1. 如何解决提交冲突？
2. 提交不小心出现误操作，如何撤销？
3. 什么是 Git Flow，它有什么好处？

资源

- 视频
 - ☆【尚硅谷】5h打通Git全套教程 | 2021最新IDEA版 <https://www.bilibili.com/video/BV1vy4y1s7k6>
- 书籍
 - 猴子都能懂的 Git 入门 <https://backlog.com/git-tutorial/cn/>
 - ☆ GitHub 漫游指南 <https://github.phodal.com/>
- 文档
 - GitHub 官方文档: <https://docs.github.com/cn>

- 游戏
 - Learning Git Branching: https://learngitbranching.js.org/?locale=zh_CN

🐧 Linux (10 天)

企业中的很多前后台项目都是部署在 Linux 服务器上的，因此很有必要熟悉 Linux 的操作和脚本的编写。

后面学微服务、学架构都是在多台服务器操作，如果你不熟悉 Linux，会有点吃力。

知识

- Linux 系统安装
- 环境变量
- 文件管理
- 用户管理
- 内存管理
- 磁盘管理
- 进程管理
- 网络管理
- 软件包管理
- 服务管理
- 日志管理
- Linux 内核
- **常用命令**
- **常用环境搭建**
- **Shell 脚本编程**
- **VIM 的使用**

学习建议

多动手实践，建议自己购买一台云服务器，并且在本地搭建 Linux 虚拟机环境。

一定要自己从 0 开始手敲命令安装软件、部署服务，熟悉整个项目的上线流程。

每个命令至少要跟着敲一遍，了解它们的作用，并通过自然地练习，熟悉常用的 Linux 命令。

记不住没关系，用文档查就行了。

先会用，再理解。一般面试问的 Linux 题目也不会很难，面试前去背一下八股文就没什么问题，感兴趣的话可以去看看 Linux 内核设计。

经典面试题

1. 如何查看某个进程的运行状态？
2. 如何在 Linux 上查看 2 G 的大文件？
3. Linux 软链接和硬链接的区别

资源

- 视频
 - ☆ 2021 韩顺平 一周学会Linux <https://www.bilibili.com/video/BV1Sv411r7vd> (基于 CentOS 7.6 版本较新)
- 书籍
 - 《鸟哥的 Linux 私房菜 —— 基础篇》 http://cn.linux.vbird.org/linux_basic/linux_basic.php (经典)
- 文档
 - Linux 工具快速教程: <https://linuxtools-rst.readthedocs.io> (基础、工具进阶、工具参考)
- 实战
 - ☆ 蓝桥云课 Linux 基础入门: <https://www.lanqiao.cn/courses/1>
 - 腾讯云动手实验室: <https://cloud.tencent.com/developer/labs>
 - 阿里云体验实验室: <https://developer.aliyun.com/adc/labs/>
 - 阿里云知行实验室: <https://start.aliyun.com/>
 - 华为云沙箱实验室: <https://lab.huaweicloud.com/>
- 社区
 - Linux 中国: <https://linux.cn/>
- 工具
 - ☆ Linux 命令搜索: <https://wangchujiang.com/linux-command>
 - Linux 命令大全手册: <https://www.linuxcool.com/>
 - Linux 命令示例: <http://linux-commands-examples.com/>
 - 宝塔 Linux 面板: <https://www.bt.cn/>

🌐 前端基础 (14 天)

虽然 Java 程序员面试时基本不会出现前端相关问题，但是在企业中，往往需要前后端程序员配合完成工作。会一些前端，不仅可以提高你们的协作效率，还能提高自己对整个项目的了解和掌控力，甚至能独立开发出一个完整项目！这点也是能给面试加分的。

知识

- HTML
- CSS
- JavaScript
 - Ajax
- Vue

学习建议

不需要学习太多的前端技术，熟悉下基础的前端三件套，了解前端是如何向后端发送请求来做数据交互的一般就够了。有时间的话可以学下 Vue，是比较容易上手的主流前端开发框架，Vue + SpringBoot 还是很香的。

资源

参见鱼皮原创的前端学习路线：<https://mp.weixin.qq.com/s/HbzLEt6NOP9Is9F3GOnbUQ>

练手项目

更多请见本路线阶段 5：项目源码（50 套）。

- HotelSystem: <https://github.com/misterchaos/HotelSystem>（酒店管理系统 Java,tomcat,mysql,servlet,jsp实现，没有使用任何框架）

- 超市管理系统: <https://github.com/zhanglei-workspace/shopping-management-system>
- Mall4j: <https://github.com/gz-yami/mall4j> (Spring Boot 电商系统)
- newbee-mall: <https://github.com/newbee-ltd/newbee-mall> (基于 Spring Boot 2.X 的全栈电商系统)
- litemall: <https://github.com/linlinjava/litemall> (小商城系统, Spring Boot 后端 + Vue 管理员前端 + 微信小程序用户前端 + Vue 用户移动端)
- forum-java: <https://github.com/Qbian61/forum-java> (一款用 Java Spring Boot 实现的现代化社区系统)

尾声

学完这个阶段的知识后,一定要再串起来回忆一遍,必须自己独立开发一个 Java Web 项目(量级可以不大,但你学过的技术尽可能地用上),能发布到 Linux 服务器上让其他小伙伴访问就更好了~

如果你只是对 Java 感兴趣、或者只是想试着自己开发后台,并不是想靠 Java 找工作的话,学到这里就可以了。可以把更多时间投入到你主方向的学习中。

但如果你是想找 Java 方向的工作,尤其是想进大厂的话,一定要继续努力,用心学习下个阶段的企业开发进阶知识。

阶段 4: Java 企业开发进阶

目标

学习更多企业级开发技术和编程思想,能够结合多种技术,独立开发出架构合理的完整系统, **解决实际问题**。

要了解为什么需要这个技术? 什么时候用这个技术? 某个需求该用哪些技术?

当然,这个阶段的内容有些过于丰富,不是所有的东西都要学,大家可以根据自己的实际情况(时间),有选择地学习。

软件工程

软件开发和管理的一些概念、原则、技术、方法、工具和经验。

知识

- 软件的本质
- 软件特性
- 软件过程
- 软件开发原则
 - 开闭原则
 - 里氏替换原则
 - 依赖倒置原则
 - 单一职责原则
 - 接口隔离原则
 - 迪米特法则
- 软件过程模型
- 敏捷开发
- 软件开发模型
- 需求建模
- 软件设计
- UML
- 体系结构设计
- 设计模式
- 软件质量管理
- 评审
- 软件质量保证
- 软件测试
 - 单元测试
 - 集成测试
 - 系统测试
 - 压力测试
 - 部署测试
- 软件配置管理
- 软件项目管理

- 软件项目估算
- 项目进度安排
- 风险管理
- 软件过程改进
 - 成熟度模型

学习建议

大学软件专业的必修课，偏理论，能学到很多企业软件开发的方法，也是对软件开发同学综合能力的提升，有时间的话可以了解下。但对想要快速找工作的同学来说，忽略即可，面试基本不会问。

资源

- 书籍
 - 《软件工程：实践者的研究方法》：<https://book.douban.com/subject/6047742/>（经典大黑书，讲的非常全面，但同样完整读下来还是有些难度的）
 - 《软件开发的 201 个原则》：<https://book.douban.com/subject/35628886/>（工具书，看一遍挺好的）
- 视频
 - 清华大学软件工程：<https://www.bilibili.com/video/BV1eE411V7Cr>
 - 北京大学软件工程：<https://www.bilibili.com/video/BV1Vt411V72G>
 - 大连工业大学软件工程：<https://www.bilibili.com/video/BV1Ns41177VM>
- 大学课件
 - 浙大计算机软件工程：<https://github.com/QSCTech/zju-icicles>

🧠 设计模式 (21 天)

设计模式是软件开发中解决一类问题的通用方法。

使用设计模式能让你写出更优雅、可维护的代码，也正因如此，很多框架源码都用到了设计模式，你不学很难看懂。

知识

- 创建型模式：对象实例化的模式，创建型模式用于解耦对象的实例化过程
 - 单例模式
 - 工厂方法模式
 - 抽象工厂
 - 建造者模式
 - 原型模式
- 结构型模式：把类或对象结合在一起形成一个更大的结构
 - 适配器模式
 - 组合模式
 - 装饰器模式
 - 代理模式
 - 享元模式
 - 外观模式
 - 桥接模式
- 行为型模式：类和对象如何交互，及划分责任和算法
 - 迭代器模式
 - 模板方法模式
 - 策略模式
 - 命令模式
 - 状态模式
 - 责任链模式
 - 备忘录模式
 - 观察者模式
 - 访问者模式
 - 中介者模式
 - 解释器模式

学习建议

先理解概念，了解每个设计模式的特点和应用场景，再多加练习，运用到实际项目。

经典面试题

1. 单例模式有哪些实现方式？有哪些优缺点？请手写其中一种
2. 你用过哪些设计模式，为什么用它？

资源

- 书籍
 - ☆《图解设计模式》（强烈推荐这本书，正是用 Java 语言实现，图多、有示例代码、有习题和答案，很不错）
 - 《Head First 设计模式》
 - 《大话设计模式》
 - 《设计模式：可复用面向对象软件的基础》（大黑书，有能力的话也可以去读）
- 视频
 - 尚硅谷图解 Java 设计模式：<https://www.bilibili.com/video/BV1G4411c7N4>
- 文档
 - 菜鸟教程：<https://www.runoob.com/design-pattern/design-pattern-tutorial.html>
 - 图说设计模式：https://design-patterns.readthedocs.io/zh_CN/latest/

🎯 缓存 (14 天)

缓存是高并发系统不可或缺的技术，可以提高系统的性能和并发，因此是后台开发必学的知识点，也是面试重点。

知识

- 什么是缓存？
- 本地缓存
 - Caffeine 库
- 多级缓存
- Redis 分布式缓存
 - 数据类型
 - 常用操作
 - Java 操作 Redis
 - Spring Boot Redis Template
 - Redisson
 - 主从模型搭建
 - 哨兵集群搭建
 - 日志持久化

- 缓存 (Redis) 应用场景
 - 数据共享
 - 单点登录
 - 计数器
 - 限流
 - 点赞
 - 实时排行榜
 - 分布式锁
- 缓存常见问题
 - 缓存雪崩
 - 缓存击穿
 - 缓存穿透
 - 缓存更新一致性
- 相关技术: Memcached、Ehcache

学习建议

学会如何简单地使用缓存并不难，和数据库类似，无非就是调用 API 对数据进行增删改查。

因此，建议先能够独立使用它，了解缓存的应用场景；再学习如何在 Java 中操作缓存中间件，并尝试和项目相结合，提高系统的性能。

跟着视频教程实操一遍即可，可以等到面试前再去深入了解原理和高级特性。

经典面试题

1. Redis 为什么快？
2. Redis 有哪些常用的数据结构？
3. Redis RDB 和 AOF 持久化的区别，如何选择？
4. 如何解决缓存击穿、缓存穿透、雪崩问题？
5. 如何用 Redis 实现点赞功能，怎么设计 Key / Value？

资源

- 视频
 - ☆ 尚硅谷 - 2021 最新 Redis 6 入门到精通教程: <https://www.bilibili.com/video/BV1Rv41177Af> (基于 Redis 6 的, 推荐)
- 文档
 - Redis 命令参考: <http://redisdoc.com/>
 - Redis 面试题整理: <https://github.com/lokles/Web-Development-Interview-With-Java/blob/main/Redis%E9%97%AE%E9%A2%98.md>
- 书籍
 - 《Redis 实战》(经典)
- 工具
 - ☆ Redis 在线练习: <https://try.redis.io/> (强烈推荐)

🔴 消息队列 (14 天)

消息队列是用于传输和保存消息的容器, 也是大型分布式系统中常用的技术, 主要解决应用耦合、异步消息、流量削锋等问题。后台开发必学, 也是面试重点。

知识

- 消息队列的作用
- RabbitMQ 消息队列
 - 生产消费模型
 - 交换机模型
 - 死信队列
 - 延迟队列
 - 消息持久化
 - Java 操作
 - 集群搭建
- 相关技术: Kafka、ActiveMQ、TubeMQ、RocketMQ

学习建议

和缓存一样，学会如何使用消息队列并不难，无非就是调用 API 去生产、转发和消费消息。

因此，建议先能够独立使用它，了解消息队列的应用场景；再学习如何在 Java 中操作消息队列中间件，并尝试和项目相结合，感受消息队列带来的好处。

这里我建议初学者先学习 RabbitMQ，比 Kafka 要好理解一些。跟着视频教程实操一遍即可，可以等到面试前再去深入了解原理和高级特性。

经典面试题

1. 使用消息队列有哪些优缺点？
2. 如何保证消息消费的幂等性？
3. 消息队列有哪些路由模型？
4. 你是否用过消息队列，解决过什么问题？

资源

- 视频
 - ☆ 尚硅谷 - 2021 最新 RabbitMQ 教程：<https://www.bilibili.com/video/BV1cb4y1o7zz>（很新很全面）
- 文档
 - RabbitMQ 中文文档：<http://rabbitmq.mr-ping.com/>
- 书籍
 - 《RabbitMQ 实战：高效部署分布式消息队列》（经典）
- 工具
 - ☆ RabbitMQ 在线模拟器：<http://tryrabbitmq.com/>

Nginx (14 天)

Nginx 是主流的、开源的、高性能的 HTTP 和反向代理 web 服务器，可以用于挂载网站、请求转发、负载均衡、网关路由等。前后端开发同学都需要学习，在后端开发的面试中有时会考到。

知识

- Nginx 作用
- 正向代理
- 反向代理（负载均衡）
- 常用命令
- 配置
- 动静分离（网站部署）
- 集群搭建
- 相关技术：HAProxy、Apache

学习建议

Nginx 的基本使用非常简单，甚至不需要看任何的视频，跟着一篇文章就能够用它来提供网站访问能力、实现反向代理。

但是在企业中，Nginx 的使用往往没那么简单，作为负载均衡、请求转发的重要组件，往往需要针对实际场景去写一些特定的配置。因此建议有时间的话，实践下 Nginx 的配置方法，了解 Nginx 基本的设计思想，对今后自己设计系统时也有帮助。

经典面试题

1. Nginx 有哪些作用？
2. Nginx 为什么支持高并发？
3. Nginx 有哪些负载均衡策略？
4. 什么是 Nginx 惊群问题，如何解决它？

资源

- 视频
 - ☆ 尚硅谷 - Nginx 教程由浅入深: <https://www.bilibili.com/video/BV1zJ411w7SV>（讲的比较全面）
 - ☆ Nginx 鱼皮 - 手把手带你从0搭建个人网站: <https://www.bilibili.com/video/BV1rU4y1J785/>（简单演示 Nginx 部署网站）
- 文档
 - Nginx 中文文档: <https://www.nginx.cn/doc/index.html>
- 工具

- Nginx 配置在线生成: <https://www.digitalocean.com/community/tools/nginx?global.app.lang=zhCN>

🌐 Netty 网络编程 (21 天)

开源的 Java 网络编程框架，用于开发高性能（事件驱动、异步非阻塞）、高可靠的网络服务器和客户端程序。

很多网络框架和服务器程序都用到了 Netty 作为底层，学好 Netty 不仅可以让我们自己实现高性能服务器，也能更好地理解其他的框架应用、阅读源码。

知识

- IO 模型 (BIO / NIO)
- Channel
- Buffer
- Selector
- Netty 模型
- WebSocket 编程 (动手做个聊天室)
- 相关技术: Vertx (中文文档: <http://vertxchina.github.io/vertx-translation-chinese/>, 比 Netty 简单多了, 实在看不懂 Netty 也可以学习下这个)

学习建议

不同于之前学的 SSM 框架, Netty 还是需要一定学习成本的, 一方面是国内资源太缺乏, 另一方面很多重要的概念 (比如 NIO) 还是要多动手写代码调试才能理解。

还是建议先从视频入门, 并且不建议在 Netty 上花太多时间, 面试的时候一般也就考察一些 Netty 背后的思想 (比如 NIO) 而非框架本身的语法细节。

经典面试题

1. Netty 有哪些优点?
2. 什么是 NIO?
3. 介绍 Netty 的零拷贝机制

资源

- 视频
 - ☆ 尚硅谷Netty教程: <https://www.bilibili.com/video/BV1DJ411m7NR>
 - Netty 技术演讲 (中文字幕版): <https://www.youtube.com/watch?v=l8yy2Cy7dDI>
- 文档
 - Netty 4 用户指南: https://www.w3cschool.cn/netty_4_user_guide/
- 书籍
 - 《Netty 实战》

🧠 微服务 (60 天)

随着互联网的发展, 项目越来越复杂, 单机且庞大的巨石项目已无法满足开发、运维、并发、可靠性等需求。

因此, 后台架构不断演进, 可以将庞大的项目拆分成一个个职责明确、功能独立的细小模块, 模块可以部署在多台服务器上, 相互配合协作, 提供完整的系统能力。

换言之, 想做大型项目, 这块儿一定要好好学!

知识

Dubbo

- 架构演进
- RPC
- Zookeeper
- 服务提供者
- 服务消费者
- 项目搭建
- 相关技术: DubboX (对 Dubbo 的扩展)

🍉 微服务

- 微服务概念
- Spring Cloud 框架
 - 子父工程
 - 服务注册与发现
 - 注册中心 Eureka、Zookeeper、Consul
 - Ribbon 负载均衡
 - Feign 服务调用
 - Hystrix 服务限流、降级、熔断
 - Resilience4j 服务容错
 - Gateway (Zuul) 微服务网关
 - Config 分布式配置中心
 - 分布式服务总线
 - Sleuth + Zipkin 分布式链路追踪
- Spring Cloud Alibaba
 - Nacos 注册、配置中心
 - OpenFeign 服务调用
 - Sentinel 流控
 - Seata 分布式事务

接口管理

- Swagger 接口文档
- Postman 接口测试
- 相关技术: YApi、ShowDoc

学习建议

时间不急的话，建议先从 Dubbo 学起，对分布式、RPC、微服务有些基本的了解，再去食用 Spring Cloud 全家桶会更香。学完 Spring Cloud 全家桶后，再去学 Spring Cloud Alibaba 就很简单了。

这部分内容的学习，原理 + 实践都很重要，也不要被各种高大上的词汇唬住了，都是上层（应用层）的东西，基本没有什么算法，跟着视频教程学，其实还是很好理解的。

分布式相关知识非常多，但这里不用刻意去背，先通过视频教程实战使用一些微服务框架，也能对其中的概念有基本的了解。

大厂面试的时候很少问 Spring Cloud 框架的细节，更多的是微服务以及各组件的一些思想，比如网关的好处、消息总线的好处等。

经典面试题

1. 什么是微服务，有哪些优缺点？
2. 什么是注册中心，能解决什么问题？

资源

- 视频
 - ☆ 尚硅谷 Dubbo 教程: <https://www.bilibili.com/video/BV1ns411c7jV>
 - ☆ 尚硅谷 SpringCloud (H版&alibaba) 框架开发教程 (微服务分布式架构) : <https://www.bilibili.com/video/BV18E411x7eT> (把国外的 Spring Cloud 和国内的 Spring Cloud Alibaba 结合在一起对比着去讲, 主流技术栈、知识点都讲到了, 真心不错!)
- 文档
 - Apache Dubbo 官方文档: <https://dubbo.apache.org/zh/>
 - Spring Cloud Alibaba 官方文档: <https://github.com/alibaba/spring-cloud-alibaba/blob/master/README-zh.md>
 - ☆ Swagger 教学文档: <https://doc.xiaominfo.com/> (跟着快速开始直接用就好了)

容器 (7 天)

将应用和环境进行封装, 相互隔离、独立部署、便于移植, 提高安全性、提高开发和维护效率。

便于实现微服务、持续集成和交付。

知识

-  Docker
 - 容器概念
 - 镜像

- 部署服务
- Dockerfile
- Docker Compose
- Docker Machine
- Docker Swarm
- 多阶段构建
- 🌐 K8S (Kubernetes)
 - K8S 架构
 - 工作负载
 - 资源类型
 - Pod
 - Pod 生命周期
 - Pod 安全策略
 - K8S 组件
 - K8S 对象
 - 部署应用
 - 服务
 - Ingress
 - Kubectl 命令行
 - 集群管理
- 相关技术：Apache Mesos、Mesosphere

学习建议

业务上云是趋势，但是对于开发同学来说，会用 Docker / K8S 部署项目和服务就行。实际工作中，企业一般都有现成的平台直接用，面试考察的也不多，不用花太多时间。

经典面试题

1. 什么是容器？
2. 使用 Docker 有哪些好处？
3. 如何快速启动多个 Docker 节点？

资源

- 视频
 - ☆【狂神说】Docker 最新超详细版教程通俗易懂: <https://www.bilibili.com/video/BV1og4y1q7M4>
 - K8S 视频教程: <https://www.bilibili.com/video/BV1K5411N7j8>
- 书籍
 - 《深入浅出 Docker》: <https://www.zhihu.com/pub/reader/119632192>
- 文档
 - Docker — 从入门到实践: <https://vuepress.mirror.docker-practice.com/>
 - 菜鸟教程 Docker: <https://www.runoob.com/docker/docker-tutorial.html>
 - 国外 Docker 官方文档: <https://docs.docker.com/get-started/>
- 网站
 - Docker 中文社区: <https://www.docker.org.cn/> (强烈推荐, 有很多 Docker 技术文章和学习笔记)
 - Docker Blog: <https://www.docker.com/blog/> (国外 Docker 技术博客)
- 镜像源
 - Docker Hub: <https://hub.docker.com>
 - 腾讯软件源: <http://mirrors.tencent.com/#/index>
 - 阿里云 Docker 仓库: <https://cr.console.aliyun.com/>

CI / CD (3 天)

持续集成 / 持续交付, 贯穿整个研发到项目上线的过程, 提高效率。

大公司一般都有自己的 CI / CD 平台。

知识

- 什么是 CI / CD
- CI / CD 有什么好处
- 使用任一 CI / CD 平台
- 相关技术: Jenkins、GitLab、微信云托管

学习建议

了解它是什么，并且实战使用任一 CI / CD 平台，感受它和传统开发运维到底有什么不同，就足够了。其实很简单，不要花太多时间。

真正要自己去搭建的时候，跟着官方文档来就行。

资源

- 视频
 - ☆【狂神说】CI/CD到底是什么？十分钟理解企业级DevOps：<https://www.bilibili.com/video/BV1zf4y127vu>（讲的非常棒！带你快速理解 CI / CD 的作用和部分原理）
 - 尚硅谷 - Jenkins 持续集成工具教程：<https://www.bilibili.com/video/BV1GW411w7pn>（实在要用到 Jenkins 再去学）
- 文档
 - Jenkins 官方文档：<https://www.jenkins.io/zh/>（有很多案例，要用的时候查一下就行了）
- 实战平台
 - ☆ 微信云托管：<https://cloud.weixin.qq.com/cloudrun>（直播演示过，鱼皮后面会给大家出视频教程）
 - 前端托管 Webify：<https://cloud.tencent.com/product/webify>（鱼皮给大家出过视频教程的 <https://www.bilibili.com/video/BV11V411p7vf/>）

练手项目

更多请见本路线阶段 5：项目源码（50 套）。

- mall-learning：<https://github.com/macrozheng/mall-learningmall>（学习教程，架构、业务、技术要点全方位解析）
- miaosha：<https://github.com/qiurunze123/miaosha>（秒杀系统设计与实现）
- Mall：<https://github.com/macrozheng/mall>（包括前台商城系统及后台管理系统，基于 SpringBoot+MyBatis 实现）
- Spring-Cloud-Admin：<https://github.com/wxiaopi/Spring-Cloud-AdminCloud-Admin>（基于 Spring Cloud 微服务化开发平台，具有统一授权、认证的后台管理系统）

- paascloud-master: <https://github.com/paascloud/paascloud-master> (基于 spring cloud + vue + oAuth2.0, 前后端分离商城系统)
- mall-swarm: <https://github.com/macrozheng/mall-swarm> (一套微服务商城系统, 采用了 Spring Cloud Greenwich、Spring Boot 2、MyBatis、Docker、Elasticsearch 等核心技术, 同时提供了基于 Vue 的管理后台方便快速搭建系统)
- SecKill: <https://github.com/hfbn/Seckill> (基于 SpringBoot+Mybatis+Redis+RabbitMQ 秒杀系统)
- jeecg-boot: <https://github.com/zhangdaiscott/jeecg-boot> (低代码开发平台)
- PassJava-Platform: <https://github.com/Jackson0714/PassJava-Platform> (一款面试刷题的 Spring Cloud 开源系统)

尾声

看到这里, 相信你已经感叹: 编程语言一辈子学不完了!

但是, 不用担心, 通过对这么多知识点的学习, 相信你已经有了了一定的积累, 也不不知不觉地锻炼了自主学习能力、资源检索能力、代码阅读能力、问题解决能力, 之后的学习会越来越轻松。

接下来, 可以试着用你学到的技术来解决实际的问题, 自主从 0 开始做一些项目, 保持编程手感。

阶段 5: 项目实战

目标

综合所学技术从 0 到 1 开发和上线一个全面、有特色的、可以写进简历的个人项目。

学习建议

其实在之前的框架学习视频中应该就做过几个项目了, 但相对不够完整和体系化。

有想法的同学可以做任何自己想做的项目, 推荐参加一些作品类竞赛, 练手、拿奖、收获项目经历一举三得。

暂时没想法的朋友，建议先跟着专门带做项目的视频教程做 1 - 2 套 Java 完整项目，一定要从 0 到 1 自己手写！

有能力的朋友也可以试着用 GitHub 上的源码来学习啦~

做一个完整的项目的确很不容易，建议大家根据自己的 **时间、兴趣** 选择 **较新的、有配套源码的** 教程，保持耐心。

如何选择编程学习资源，可以看下我的原创文章：<https://mp.weixin.qq.com/s/mIMqI9RJCd7THt6rpGb8UA>

下面推荐一些优质的、较新的项目实战视频教程 + 50 套项目源码。

视频教程

- 尚硅谷 - 谷粒学院 - 微服务 + 全栈 - 在线教育实战项目：<https://www.bilibili.com/video/BV1dQ4y1A75e>（全栈项目，前后端讲得都很全面）
- 尚硅谷 - 尚筹网 - Java项目实战开发教程 - SSM 框架 + 微服务架构：<https://www.bilibili.com/video/BV1bE411T7oZ>（500 多集，包含完整的用户权限管理，功能丰富、讲得很细致全面）
- 黑马 - 24 小时搞定 Java 毕设电商项目：<https://www.bilibili.com/video/BV13f4y1r7fS>
- 黑马程序员 Java 项目《传智健康》，超完整的企业级医疗行业项目：<https://www.bilibili.com/video/BV1Bo4y117zV>
- 黑马程序员 Java 项目《万信金融》企业级开发实战，互联网金融行业解决方案：<https://www.bilibili.com/video/BV1Ub4y1f7rk>
- 黑马程序员 Java 项目 SaaS 移动办公完整版《iHRM 人力资源管理系统》，跨行业 SaaS 办公整合性解决方案：<https://www.bilibili.com/video/BV18A411L7UX>
- 黑马程序员 Java 大型分布式微服务闪聚支付项目，Java 移动支付全生态系统：<https://www.bilibili.com/video/BV17v411V79c>
- 黑马程序员 Java 项目《好客租房》：<https://www.bilibili.com/video/BV1sZ4y1F7PD>

常用类库

工具

- Guava: 谷歌开发的 Java 工具库 (<https://github.com/google/guava>)
- Apache Commons: 各类工具库, 比如 commons-lang、commons-io、commons-collections 等 (<https://github.com/apache/commons-lang>)
- Hutool: Java 工具集库 (<https://github.com/looly/hutool>)
- Lombok: Java 增强库 (<https://github.com/projectlombok/lombok>)
- Apache HttpComponents Client: HTTP 客户端库 (<https://github.com/apache/http-components-client>)
- OkHttp: 适用于 JVM、Android 等平台的 Http 客户端 (<https://github.com/square/okhttp>)
- Gson: 谷歌的 JSON 处理库 (<https://github.com/google/gson>)
- Jcommander: Java 命令行参数解析框架 (<https://github.com/cbeust/jcommander>)
- Apache PDFBox: PDF 操作库 (<https://github.com/apache/pdfbox>)
- EasyExcel: 阿里的 Excel 处理库 (<https://github.com/alibaba/easyexcel>)
- Apache POI: 表格文件处理库 (<https://github.com/apache/poi>)

测试

- JUnit: Java 测试框架 (<https://github.com/junit-team/junit4>)
- Mockito: Java 单元测试 Mock 框架 (<https://github.com/mockito/mockito>)
- Selenium: 浏览器自动化框架 (<https://github.com/SeleniumHQ/selenium>)
- HtmlUnit: Java 模拟浏览器 (<https://github.com/HtmlUnit/htmlunit>)
- TestNG: Java 测试框架 (<https://github.com/cbeust/testng>)
- Jacoco: Java 代码覆盖度库 (<https://github.com/jacoco/jacoco>)

其他

- cglib: 字节码生成库 (<https://github.com/cglib/cglib>)
- Arthas: Java 诊断工具 (<https://github.com/alibaba/arthas>)
- config: 针对 JVM 的配置库 (<https://github.com/lightbend/config>)
- Quasar: Java 线程库 (<https://github.com/puniverse/quasar>)
- drools: Java 规则引擎 (<https://github.com/kiegroup/drools>)
- Caffeine: Java 高性能缓存库 (<https://github.com/ben-manes/caffeine>)
- Disruptor: 高性能线程间消息传递库 (<https://github.com/LMAX-Exchange/disruptor>)
- Knife4j: Swagger 文档增强 (<https://doc.xiaominfo.com/>)

- Thumbnailator: Java 缩略图生成库 (<https://github.com/coobird/thumbnailator>)
- Logback: Java 日志库 (<https://github.com/qos-ch/logback>)
- Apache Camel: 消息传输集成框架 (<https://github.com/apache/camel>)
- Quartz: 定时任务调度库 (<https://github.com/quartz-scheduler/quartz>)
- Apache Mahout: 机器学习库 (<https://github.com/apache/mahout>)
- Apache OpenNLP: NLP 工具库 (<https://github.com/apache/opennlp>)
- RxJava: JVM 反应式编程框架 (<https://github.com/ReactiveX/RxJava>)
- JProfiler: 性能分析库 (<https://www.ej-technologies.com/products/jprofiler/overview.html>)
- jsoup: HTML 文档解析库 (<https://jsoup.org/>)
- webmagic: Java 爬虫框架 (<https://github.com/code4craft/webmagic/>)

IDEA 插件

综合实用性、易用性、流行度、维护频率，精选了以下插件，简单分类为天地二品。

精选地品插件

简单好用、增强功能

- Background Image Plus +
 - 描述: 给编辑器设置背景图片 (小姐姐图片)
 - 官网: <https://plugins.jetbrains.com/plugin/11247-background-image-plus->
- Chinese Language Pack
 - 描述: 中文支持
 - 官网: <https://plugins.jetbrains.com/plugin/13710-chinese-simplified-language-pack--->
- Translation
 - 描述: 翻译插件, 鼠标选中文本, 点击右键即可自动翻译成多国语言。
 - 官网: <https://plugins.jetbrains.com/plugin/8579-translation/>
- Key Promoter X
 - 描述: 快捷键提示插件。当你执行鼠标操作时, 如果该操作可被快捷键代替, 会给出提示, 帮助你自然形成使用快捷键的习惯, 告别死记硬背。
 - 官网: <https://plugins.jetbrains.com/plugin/9792-key-promoter-x/>
- Rainbow Brackets

- 描述：给括号添加彩虹色，使开发者通过颜色区分括号嵌套层级，便于阅读
- 官网：<https://plugins.jetbrains.com/plugin/10080-rainbow-brackets/>
- CodeGlance
 - 描述：在编辑器右侧生成代码小地图，可以拖拽小地图光标快速定位代码，阅读行数很多的代码文件时非常实用。
 - 官网：<https://plugins.jetbrains.com/plugin/7275-codeglance/>
- WakaTime
 - 描述：代码统计和跟踪插件
 - 官网：<https://plugins.jetbrains.com/plugin/7425-wakatime>
- Statistic
 - 描述：代码统计
 - 官网：<https://plugins.jetbrains.com/plugin/4509-statistic>

精选天品插件

提高开发效率和代码质量

- String Manipulation
 - 描述：字符串快捷处理
 - 官网：<https://plugins.jetbrains.com/plugin/2162-string-manipulation>
- Tabnine AI Code Completion
 - 描述：使用 AI 去自动提示和补全代码，比 IDEA 自带的代码补全更加智能化
 - 官网：<https://plugins.jetbrains.com/plugin/12798-tabnine-ai-code-completion-j-s-java-python-ts-rust-go-php-more>
- GsonFormatPlus
 - 描述：根据 json 生成对象
 - 官网：<https://plugins.jetbrains.com/plugin/14949-gsonformatplus>
- JUnitGenerator V2.0
 - 描述：自动生成单元测试
 - 官网：<https://plugins.jetbrains.com/plugin/3064-junitgenerator-v2-0>
- RestfulTool
 - 描述：辅助 web 开发的工具集
 - 官网：<https://plugins.jetbrains.com/plugin/14280-restfultool>
- SequenceDiagram
 - 描述：自动生成方法调用时序图

- 官网: <https://plugins.jetbrains.com/plugin/8286-sequencediagram>
- CheckStyle-IDEA
 - 描述: 自动检查 Java 代码规范
 - 官网: <https://plugins.jetbrains.com/plugin/1065-checkstyle-idea>
- Alibaba Java Coding Guidelines
 - 描述: 代码规范检查插件
 - 官网: <https://plugins.jetbrains.com/plugin/10046-alibaba-java-coding-guidelines>
- SonarLint
 - 描述: 帮助你发现和修复代码的错误和漏洞
 - 官网: <https://plugins.jetbrains.com/plugin/7973-sonarlint>
- MybatisX
 - 描述: MyBatis 增强插件, 支持自动生成 entity、mapper、service 等常用操作的代码, 优化体验
 - 官网: <https://plugins.jetbrains.com/plugin/10119-mybatisx>

美化类

- Extra Icons
 - 描述: 图标美化插件, 支持更多文件类型
 - 官网: <https://plugins.jetbrains.com/plugin/11058-extra-icons>
- Chinese Language Pack
 - 描述: 中文支持
 - 官网: <https://plugins.jetbrains.com/plugin/13710-chinese-simplified-language-pack>
- Grep Console
 - 描述: 自定义设置控制台输出颜色
 - 官网: <https://plugins.jetbrains.com/idea/plugin/7125-grep-console>
- Material Theme UI
 - 描述: 美观的主题
 - 官网: <https://plugins.jetbrains.com/plugin/8006-material-theme-ui>
- Nyan Progress Bar
 - 描述: 进度条美化
 - 官网: <https://plugins.jetbrains.com/plugin/8575-nyan-progress-bar>
- activate-power-mode
 - 描述: 打字特效
 - 官网: <https://plugins.jetbrains.com/plugin/8330-activate-power-mode>

- Power Mode II
 - 描述：打字特效插件升级版
 - 官网：<https://plugins.jetbrains.com/plugin/8251-power-mode-ii>
- Rainbow Brackets
 - 描述：给括号添加彩虹色，使开发者通过颜色区分括号嵌套层级，便于阅读
 - 官网：<https://plugins.jetbrains.com/plugin/10080-rainbow-brackets/>
- Background Image Plus +
 - 描述：给编辑器设置背景图片
 - 官网：<https://plugins.jetbrains.com/plugin/11247-background-image-plus->

代码生成类

- Codota AI Autocomplete for Java and JavaScript
 - 描述：使用 AI 去自动提示和补全代码，比 IDEA 自带的代码补全更加智能化
 - 官网：<https://plugins.jetbrains.com/plugin/7638-codota-ai-autocomplete-for-java-and-javascript>
- Tabnine AI Code Completion
 - 描述：使用 AI 来自动提示和补全代码，比 IDEA 自带的代码补全更加智能化
 - 官网：<https://plugins.jetbrains.com/plugin/12798-tabnine-ai-code-completion-j-s-java-python-ts-rust-go-php-more>
- UUID Generator
 - 描述：生成随机的 UUID
 - 官网：<https://plugins.jetbrains.com/plugin/8320-uuid-generator>
- Randomness
 - 描述：生成随机值
 - 官网：<https://plugins.jetbrains.com/plugin/9836-randomness>
- .ignore
 - 描述：生成不同语言的版本控制忽略文件
 - 官网：<https://plugins.jetbrains.com/plugin/7495--ignore>
- Auto filling Java call arguments
 - 描述：通过快捷键自动补全函数的调用参数
 - 官网：<https://plugins.jetbrains.com/plugin/8638-auto-filling-java-call-argument-s/>
- GenerateAllSetter
 - 描述：快速生成调用对象 setter 方法的代码
 - 官网：<https://plugins.jetbrains.com/plugin/9360-generateallsetter>
- GenerateSerialVersionUID

- 描述：一键为实现 Serializable 接口的类生成 serialVersionUID
 - 官网：<https://plugins.jetbrains.com/plugin/185-generateserialversionuid/>
- GsonFormatPlus
 - 描述：根据 json 生成对象
 - 官网：<https://plugins.jetbrains.com/plugin/14949-gsonformatplus>
- Lombok
 - 描述：配合 Lombok 依赖及注解使用，能够大大减少 Java 对象的代码量。IDEA 已自带，无需安装。
 - 官网：<https://plugins.jetbrains.com/plugin/6317-lombok/>
- Easy Code
 - 描述：根据数据库表生成 Java entity、dao、mapper、service、controller 等代码
 - 官网：<https://plugins.jetbrains.com/plugin/10954-easy-code>
- Easy Javadoc
 - 描述：快速生成 Javadoc 文档
 - 官网：<https://plugins.jetbrains.com/plugin/12977-easy-javadoc>
- JavaDoc
 - 描述：生成 Javadoc 文档
 - 官网：<https://plugins.jetbrains.com/plugin/7157-javadoc>
- Json2Pojo
 - 描述：根据 JSON 生成 POJO，根据 POJO 生成 JSON
 - 官网：<https://plugins.jetbrains.com/plugin/8533-json2pojo>
- JUnitGenerator V2.0
 - 描述：自动生成单元测试
 - 官网：<https://plugins.jetbrains.com/plugin/3064-junitgenerator-v2-0>
- Jindent - Source Code Formatter
 - 描述：根据自定义模板来生成 javadoc 注释
 - 官网：<https://plugins.jetbrains.com/plugin/8309-jindent--source-code-formatter>
- MybatisX
 - 描述：MyBatis 增强插件，支持自动生成 entity、mapper、service 等常用操作的代码，优化体验
 - 官网：<https://plugins.jetbrains.com/plugin/10119-mybatisx>
- MyBatisCodeHelperPro
 - 描述：MyBatis 增强插件，支持自动生成 entity、mapper、service 等常用操作的代码（收费）
 - 官网：<https://plugins.jetbrains.com/plugin/9837-mybatiscodehelperpro>

功能提效类

源码阅读

- Call Graph
 - 描述：生成调用关系图，便于查看源码
 - 官网：<https://plugins.jetbrains.com/plugin/12304-call-graph>
- Jump to Line
 - 描述：支持跳转到任意行进行调试
 - 官网：<https://plugins.jetbrains.com/plugin/14877-jump-to-line>
- SequenceDiagram
 - 描述：自动生成方法调用时序图
 - 官网：<https://plugins.jetbrains.com/plugin/8286-sequencediagram>
- jclasslib Bytecode Viewer
 - 描述：字节码查看器
 - 官网：<https://plugins.jetbrains.com/plugin/9248-jclasslib-bytecode-viewer>
- BrowseWordAtCaret
 - 描述：快速在相同单词间进行跳转
 - 官网：<https://plugins.jetbrains.com/plugin/201-browsewordatcaret>
- CodeGlance
 - 描述：在编辑器右侧生成代码小地图，可以拖拽小地图光标快速定位代码，阅读行数很多的代码文件时非常实用。
 - 官网：<https://plugins.jetbrains.com/plugin/7275-codeglance/>

代码规范

- CheckStyle-IDEA
 - 描述：自动检查 Java 代码规范
 - 官网：<https://plugins.jetbrains.com/plugin/1065-checkstyle-idea>
- Alibaba Java Coding Guidelines
 - 描述：代码规范检查插件
 - 官网：<https://plugins.jetbrains.com/plugin/10046-alibaba-java-coding-guidelines>
- SonarLint
 - 描述：帮助你发现和修复代码的错误和漏洞
 - 官网：<https://plugins.jetbrains.com/plugin/7973-sonarlint>

- FindBugs-IDEA
 - 描述：帮你找出代码中潜藏的 Bug
 - 官网：<https://plugins.jetbrains.com/plugin/3847-findbugs-idea>
- MetricsReloaded
 - 描述：代码复杂度分析
 - 官网：<https://plugins.jetbrains.com/plugin/93?pr=idea>

格式处理

- CamelCase
 - 描述：快速字符串格式转换工具，如驼峰式
 - 官网：<https://plugins.jetbrains.com/plugin/7160-camelcase>
- Json Parser
 - 描述：验证和格式化 Json 字符串的工具
 - 官网：<https://plugins.jetbrains.com/plugin/10650-json-parser>
- Properties to YAML Converter
 - 描述：把 Properties 文件的格式转为 YAML 格式
 - 官网：<https://plugins.jetbrains.com/plugin/8000-properties-to-yaml-converter>
- String Manipulation
 - 描述：字符串快捷处理
 - 官网：<https://plugins.jetbrains.com/plugin/2162-string-manipulation>

代码统计

- WakaTime
 - 描述：代码统计和跟踪插件
 - 官网：<https://plugins.jetbrains.com/plugin/7425-wakatime>
- Statistic
 - 描述：代码统计
 - 官网：<https://plugins.jetbrains.com/plugin/4509-statistic>

代码检索

- OnlineSearch
 - 描述：选择文本后支持不同搜索引擎进行搜索
 - 官网：<https://plugins.jetbrains.com/plugin/8298-onlinesearch/>
- CodeSearch
 - 描述：快捷地打开搜索引擎进行搜索
 - 官网：<https://plugins.jetbrains.com/plugin/12578-codesearch>

其他

- Leetcode Editor
 - 描述：在 IDEA 里刷算法题
 - 官网：<https://plugins.jetbrains.com/plugin/12132-leetcode-editor>
- File Watchers
 - 描述：监听文件，当其发生变化时自动执行命令
 - 官网：<https://plugins.jetbrains.com/plugin/7177-file-watchers>
- PlantUML integration
 - 描述：UML 建模工具，只需简单的代码就能画出复杂的 UML 类图
 - 官网：<https://plugins.jetbrains.com/plugin/7017-plantuml-integration>
- Bytecode Editor
 - 描述：字节码编辑工具
 - 官网：<https://plugins.jetbrains.com/plugin/8461-bytecode-editor>
- Alibaba Cloud Toolkit
 - 描述：阿里云工具集
 - 官网：<https://plugins.jetbrains.com/plugin/11386-alibaba-cloud-toolkit>
- Tencent Cloud Toolkit
 - 描述：腾讯云工具集
 - 官网：<https://plugins.jetbrains.com/plugin/15653-tencent-cloud-toolkit>
- EduTools
 - 描述：面向教育者，允许创建、分享和加入练习
 - 官网：<https://plugins.jetbrains.com/plugin/10081-edutools>
- IDE Features Trainer
 - 描述：IntelliJ IDEA 官方出的学习辅助插件
 - 官网：<https://plugins.jetbrains.com/plugin/8554-ide-features-trainer>
- JRebel Plugin
 - 描述：项目热更新，提高效率（可惜要收费）
 - 官网：<https://plugins.jetbrains.com/plugin/4441-jrebel-and-xrebel-for-intellij>

- Key Promoter X
 - 描述：快捷键提示插件。当你执行鼠标操作时，如果该操作可被快捷键代替，会给出提示，帮助你自然形成使用快捷键的习惯，告别死记硬背。
 - 官网：<https://plugins.jetbrains.com/plugin/9792-key-promoter-x/>
- Translation
 - 描述：翻译插件，鼠标选中文本，点击右键即可自动翻译成多国语言。
 - 官网：<https://plugins.jetbrains.com/plugin/8579-translation/>
- idea_seat
 - 描述：自动提醒，防止你久坐不动
 - 官网：https://github.com/s-unscrupulous/idea_seat
- RestfulTool
 - 描述：辅助 web 开发的工具集
 - 官网：<https://plugins.jetbrains.com/plugin/14280-restfultool>
- Multirun
 - 描述：同时启动多应用
 - 官网：<https://plugins.jetbrains.com/plugin/7248-multirun>

扩展增强类

- Free Mybatis Plugin
 - 描述：MyBatis 增强插件，支持 mapper => xml 的跳转、代码生成等功能
 - 官网：<https://plugins.jetbrains.com/plugin/8321-free-mybatis-plugin>
- MyBatis Log Plugin
 - 描述：MyBatis SQL 提取和格式化输出
 - 官网：<https://plugins.jetbrains.com/plugin/10065-mybatis-log-plugin>
- Maven Helper
 - 描述：Maven 辅助插件
 - 官网：<https://plugins.jetbrains.com/plugin/7179-maven-helper>
- Gradle View
 - 描述：Gradle 项目管理工具的扩展
 - 官网：<https://plugins.jetbrains.com/plugin/7150-gradle-view>
- Arthas Idea
 - 描述：Arthas 命令生成插件。Arthas 是阿里开源的 Java 在线诊断工具，该插件可以自动生成 Arthas 在线 Java 代码诊断命令，不用再到官网翻文档拼命命令啦！
 - 官网：<https://plugins.jetbrains.com/plugin/13581-arthas-idea/>
- GitToolBox

- 描述：Git 增强插件。在自带的 Git 功能之上，新增了查看 Git 状态、自动拉取代码、提交通知等功能。最好用的是可以查看到每一行代码的最近一次提交信息。
 - 官网：<https://plugins.jetbrains.com/plugin/7499-gittoolbox/>
- BashSupport
 - 描述：支持 Bash 脚本文件的高亮和提示等
 - 官网：<https://plugins.jetbrains.com/plugin/4230-bashsupport>
- Git Flow Integration
 - 描述：Git Flow 的图形界面操作
 - 官网：<https://plugins.jetbrains.com/plugin/7315-git-flow-integration>
- Gitee
 - 描述：开源中国的码云插件
 - 官网：<https://plugins.jetbrains.com/plugin/8383-gitee>
- Drools
 - 描述：规则引擎的扩展
 - 官网：<https://plugins.jetbrains.com/plugin/16871-drools>
- EnvFile
 - 描述：对多环境配置文件的支持
 - 官网：<https://plugins.jetbrains.com/plugin/7861-envfile>
- Kubernetes
 - 描述：容器管理的扩展
 - 官网：<https://plugins.jetbrains.com/plugin/10485-kubernetes>
- Vue.js
 - 描述：Vue.js 前端框架的扩展
 - 官网：<https://plugins.jetbrains.com/plugin/9442-vue-js>
- Scala
 - 描述：Scala 语言的扩展
 - 官网：<https://plugins.jetbrains.com/plugin/1347-scala>
- Zookeeper
 - 描述：Zookeeper 中间件的扩展
 - 官网：<https://plugins.jetbrains.com/plugin/7364-zookeeper>
- Jms messenger
 - 描述：对 Java 消息服务的扩展
 - 官网：<https://plugins.jetbrains.com/plugin/10949-jms-messenger>
- MapStruct Support
 - 描述：对 MapStruct 对象转换的支持
 - 官网：<https://plugins.jetbrains.com/plugin/10036-mapstruct-support>
- Big Data Tools

- 描述：提供了 HDFS 等大数据框架的工具集
 - 官网：<https://plugins.jetbrains.com/plugin/12494-big-data-tools>
- CSV
 - 描述：支持 CSV 文件的高亮和提示等
 - 官网：<https://plugins.jetbrains.com/plugin/10037-csv>
- Ideolog
 - 描述：识别和格式化 log 日志文件
 - 官网：<https://plugins.jetbrains.com/plugin/9746-ideolog>
- Toml
 - 描述：支持 toml 文件的高亮和提示等
 - 官网：<https://plugins.jetbrains.com/plugin/8195-toml>
- .env files support
 - 描述：支持 .env 文件的高亮和提示等
 - 官网：<https://plugins.jetbrains.com/plugin/9525-env-files-support>

常用软件

以下部分软件可在【公众号：程序员鱼皮】回复【软件】获取~

开发相关

- JetBrains IDEA：Java 集成开发环境，凭学生邮箱可申请免费使用 (<https://www.jetbrains.com/idea/>)
- Reset JetBrains：你懂的
- Visual Studio Code：插件化代码编辑器 (<https://code.visualstudio.com/>)
- Sublime Text：轻量代码编辑器 (<https://www.sublimetext.com/>)
- Navicat：数据库管理软件 (<https://www.navicat.com.cn/>)
- JMeter：Java 性能测试工具 (<https://jmeter.apache.org/>)
- JVisual VM：Java 运行状态可视化工具 (<https://visualvm.github.io/>)
- XShell：SSH 连接软件 (<https://www.netsarang.com/zh/xshell/>)
- XFTP：FTP 连接软件 (<https://www.netsarang.com/zh/xftp/>)
- Redis Desktop Manager：Redis 可视化管理工具 (<https://github.com/uglide/RedisDesktopManager>)
- Postman：接口测试工具 (<https://www.postman.com/>)
- VMware：虚拟机软件 (<https://www.vmware.com/>)
- Chocolatey：Windows 软件包管理器 (<https://chocolatey.org/>)
- Typora：写文档软件 (<https://typora.io/>)

效率

- Ditto: 强大的剪切板 (<https://ditto-cp.sourceforge.io/>)
- uTools: 插件化的效率工具 (<https://u.tools/>)
- XMind: 思维导图软件 (<https://www.xmind.cn/>)
- Qdir: Windows 多窗口管理器 (<https://q-dir.en.softonic.com/>)

项目源码 (50 套)

电商秒杀

- 天猫整站 J2EE: <https://how2j.cn/module/115.htm>
- 天猫整站 SSM: <https://how2j.cn/module/134.html>
- 天猫整站 Springboot: <https://how2j.cn/module/156.html>
- mall-learning: <https://github.com/macrozheng/mall-learningmall> (学习教程, 架构、业务、技术要点全方位解析)
- SpringBoot 电商商城系统 Mall4j: <https://github.com/gz-yami/mall4j>
- SpringBoot 完整电商系统 Mall: <https://github.com/macrozheng/mall> (包括前台商城系统及后台管理系统, 基于 SpringBoot+MyBatis 实现)
- newbee-mall: <https://github.com/newbee-ltd/newbee-mall> (一套电商系统, 包括 newbee-mall 商城系统及 newbee-mall-admin 商城后台管理系统, 基于 Spring Boot 2.X 及相关技术栈开发)
- paascloud-master: <https://github.com/paascloud/paascloud-master> (基于 spring cloud + vue + oAuth2.0, 前后端分离商城系统)
- mall-swarm: <https://github.com/macrozheng/mall-swarm> (一套微服务商城系统, 采用了 Spring Cloud Greenwich、Spring Boot 2、MyBatis、Docker、Elasticsearch 等核心技术, 同时提供了基于 Vue 的管理后台方便快速搭建系统)
- onemall: <https://github.com/YunaiV/onemall> (mall 商城, 基于微服务的思想, 构建在 B2C 电商场景下的项目实战。核心技术栈, 是 Spring Boot + Dubbo。未来, 会重构成 Spring Cloud Alibaba)
- litemall: <https://github.com/linlinjava/litemall> (又一个小商城, litemall = Spring Boot 后端 + Vue 管理员前端 + 微信小程序用户前端 + Vue 用户移动端)
- xmall: <https://github.com/Exrick/xmall> (基于SOA架构的分布式电商购物商城 前后端分离 前台商城:Vue全家桶 后台管理系统)
- miaosha: <https://github.com/qiurunze123/miaosha> (秒杀系统设计与实现)
- SecKill: <https://github.com/hfbin/Seckill> (基于 SpringBoot+Mybatis+Redis+RabbitMQ 秒杀系统)

博客论坛

- [Mblog](#): 开源 Java 博客系统
- [halo](#): 一个优秀的开源博客发布应用
- [forum-java](#): 一款用 Java (spring boot) 实现的现代化社区 (论坛/问答/BBS/社交网络/博客) 系统平台
- [vhr](#): 微人事是一个前后端分离的人力资源管理系统, 项目采用 SpringBoot+Vue 开发。
- [favorites-web](#): 云收藏 Spring Boot 2.X 开源项目。云收藏是一个使用 Spring Boot 构建的开源网站, 可以让用户在线随时随地收藏的一个网站, 在网站上分类整理收藏的网站或者文章。
- [community](#): 码问, 开源论坛、问答系统, 现有功能提问、回复、通知、最新、最热、消除零回复功能。技术栈 Spring、Spring Boot、MyBatis、MySQL/H2、Bootstrap
- [NiterForum](#): 尼特社区-NiterForum-一个论坛/社区程序。后端 Springboot/MyBatis/Maven/MySQL, 前端Thymeleaf/Layui。可供初学者, 学习、交流使用。
- [VBlog](#): V部落, Vue+SpringBoot实现的多用户博客管理平台!
- [NiceFish](#): SpringBoot/SpringCloud 前后端分离项目
- [My-Blog](#): My Blog 是由 SpringBoot + Mybatis + Thymeleaf 等技术实现的 Java 博客系统, 页面美观、功能齐全、部署简单及完善的代码, 一定会给使用者无与伦比的体验。
- [My-Blog-layui](#): layui 版本的 My-Blog : A simple & beautiful blogging system implemented with spring-boot & layui & thymeleaf & mybatis My Blog 是由 SpringBoot + Layui + Mybatis + Thymeleaf 等技术实现的 Java 博客系统, 页面美观、功能齐全、部署简单及完善的代码, 一定会给使用者无与伦比的体验
- [symphony](#): Java 实现的现代化社区

管理系统

- [Spring-Cloud-Admin](#): Cloud-Admin 是国内首个基于 Spring Cloud 微服务化开发平台, 具有统一授权、认证后台管理系统, 其中包含具备用户管理、资源权限管理、网关 API 管理等多个模块, 支持多业务系统并行开发, 可以作为后端服务的开发脚手架。代码简洁, 架构清晰, 适合学习和直接项目中使用。核心技术采用 Spring Boot2 以及 Spring Cloud Gateway 相关核心组件, 前端采用 vue-element-admin 组件。
- [bootshiro](#): 基于 springboot+shiro+jwt 的资源无状态认证权限管理系统后端
- [悟空CRM](#): 基于jfinal+vue+ElementUI的前后端分离CRM系统
- [EL-ADMIN](#): 基于 SpringBoot 的后台管理系统
- [pig](#): 基于 Spring Boot 2.2、Spring Cloud Hoxton & Alibaba、OAuth2 的 RBAC 权限管理系统。
- [FEBS-Shiro](#): Spring Boot 2.1.3, Shiro1.4.0 & Layui 2.5.4 权限管理系统。

- [Spring Boot-Shiro-Vue](#): 基于Spring Boot-Shiro-Vue 的权限管理
- [studentmanager](#): 基于springboot+mybatis学生管理系统
- [jshERP](#): 华夏ERP基于SpringBoot框架和SaaS模式，立志为中小企业提供开源好用的ERP软件，目前专注进销存+财务功能。主要模块有零售管理、采购管理、销售管理、仓库管理、财务管理、报表查询、系统管理等。支持预付款、收入支出、仓库调拨、组装拆卸、订单等特色功能。拥有库存状况、出入库统计等报表。同时对角色和权限进行了细致全面控制，精确到每个按钮和菜单。
- [HotelSystem](#): 酒店管理系统 Java,tomcat,mysql,servlet,jsp实现，没有使用任何框架

开发平台

- [open-capacity-platform](#): 微服务能力开发平台
- [jeecg-boot](#): JeecgBoot是一款基于BPM的低代码平台！前后端分离架构 SpringBoot 2.x, SpringCloud, Ant Design&Vue, Mybatis-plus, Shiro, JWT, 支持微服务。强大的代码生成器让前后端代码一键生成，实现低代码开发！

其他

- [学之思在线考试系统](#): 一款 java + vue 的前后端分离的考试系统
- [PassJava-Platform](#): 一款面试刷题的 Spring Cloud 开源系统
- [kkFileView](#): 使用spring boot打造文件文档在线预览项目
- [dynamic-datasource](#): 一个基于springboot的快速集成多数据源的启动器
- [moti-cloud](#): 莫提网盘，基于 SpringBoot+MyBatis+ThymeLeaf+BootStrap，适合初学者
- [threadandjuc](#): three-high-import 高可用\高可靠\高性能，三高多线程导入系统（该项目意义为理论贯通）
- [proxyee-down](#): http下载工具，基于http代理，支持多连接分块下载
- [hosp_order](#): 医院预约挂号系统，基于 SSM 框架
- [趋势投资 SpringCloud](#)
- [DiyTomcat](#)

阶段 6: Java 高级

目标

不满足于能做，而是通过更 **深入** 和 **广泛** 的学习，实现高质量的代码和更优秀的架构，培养解决问题的能力。

已经到了这个阶段，建议除了看完成的教程外，平时多自主搜索信息去学习，积少成多。比如遇到了一个不了解的名词，可以去网上搜一下，感兴趣的话再进行下一步的学习。

🌀 并发编程 (21 天)

对 Java 后端开发人员来说，懂得如何利用有限的系统资源来提高系统的性能是很重要的，也是大厂面试考察的重点，因此并发编程（尤其是 Java 并发包的使用）这块的知识很重要。

把它放到高级，是因为在学并发编程前，需要有一定的编程经验、了解一定的操作系统知识。

知识

- 线程和进程
- 线程状态
- 并行和并发
- 同步和异步
- Synchronized
- Volatile 关键字
- Lock 锁
- 死锁
- 可重入锁
- 线程安全
- 线程池
- JUC 的使用
- AQS
- Fork Join
- CAS

学习建议

并发编程入门不难，依然是 **先学会使用** 基础的 Java 并发包，再通过大量地实践和测试，了解一些原理，才能真正掌握何时使用、如何更合理地使用并发编程。而不是张口闭口多线程，上天入地高并发。

经典面试题

1. volatile 关键字的作用
2. 使用线程池有哪些好处？
3. 线程池参数如何设置？
4. 什么是线程安全问题，如何解决？
5. 介绍 synchronized 的锁升级机制
6. CopyOnWriteArrayList 适用于哪种场景？

资源

- 视频
 - ☆【尚硅谷】大厂必备技术之JUC并发编程2021最新版: <https://www.bilibili.com/video/BV1Kw411Z7dF> (特点是短、精、新)
 - 黑马程序员全面深入学习Java并发编程: <https://www.bilibili.com/video/BV16J411h7Rd> (讲的很细、全面深入)
- 书籍
 - ☆《Java并发编程实战》(国外的经典)
 - ☆《Java 并发编程艺术》(国人写的, 理论思想内容较多, 有时间建议反复看)
- 项目
 - ☆ Java 并发知识点总结: <https://github.com/CL0610/Java-concurrency>

🧠 JVM (30 天)

想要深入理解 Java，探秘 Java 跨平台的奥秘，一定要了解 Java 底层的虚拟机技术。

了解虚拟机、掌握虚拟机性能调优方法，有助于你写出更高性能、资源占用更小的优质程序。

在学习 JVM 的过程中，也能学到很多精妙的设计，开拓思路。

知识

- JVM 内存结构
- JVM 生命周期
- 主流虚拟机
- Java 代码执行流程
- 类加载
 - 类加载器
 - 类加载过程
 - 双亲委派机制
- 垃圾回收
 - 垃圾回收器
 - 垃圾回收策略
 - 垃圾回收算法
 - StopTheWorld
- 字节码
- 内存分配和回收
- JVM 性能调优
 - 性能分析方法
 - 常用工具
 - 参数设置
- Java 探针
- 线上故障分析

学习建议

JVM 的知识略显枯燥，建议先看视频，有实操的地方一定要实操！自己多去分析。

第一遍不理解没有关系，可以再看书来巩固，想要真正学好，《深入理解 Java 虚拟机（第三版）》一定要读。

如果只是为了通过面试可以直接看更精简的视频，比如狂神的。

经典面试题

1. 介绍 JVM 的内存模型？
2. JVM 内存为什么要分代？
3. 介绍一次完整的 GC 流程
4. 介绍双亲委派模型，为什么需要它？

资源

- 视频
 - ☆ 尚硅谷宋红康 - JVM 全套教程详解: <https://www.bilibili.com/video/BV1PJ411n7xZ> (讲得相当全面! 附有实操)
 - 【狂神说Java】JVM快速入门篇: <https://www.bilibili.com/video/BV1iJ411d7jS> (讲得有点浅, 但都是面试重点, 时间紧的小伙伴可以直接看这个)
- 书籍
 - ☆ 《深入理解 Java 虚拟机 (第三版) 》 (有理论有实践, 内容丰富, 不可多得的 JVM 学习神书, 就是有点废头发)
- 项目
 - ☆ Java 虚拟机底层原理知识总结: <https://github.com/doocs/jvm>
- 实战
 - 阿里云 JVM 实战: <https://developer.aliyun.com/graph/java/point/64>
- 工具
 - Arthas 开源 Java 诊断工具: <https://github.com/alibaba/arthas>

架构设计

🌀 分布式

- 分布式理论
 - CAP
 - BASE
- 分布式缓存
 - Redis
 - Memcached
 - Etcd
- 一致性算法
 - Raft

- Paxos
 - 一致性哈希
- 分布式事务
 - 解决方案
 - 2PC
 - 3PC
 - TCC
 - 本地消息表
 - MQ 事务消息
 - 最大努力通知
 - LCN 分布式事务框架: <https://github.com/codingapi/tx-lcn>
- 分布式 id 生成
 - Leaf 分布式 id 生成服务: <https://github.com/Meituan-Dianping/Leaf>
- 分布式任务调度
 - XXL-JOB 调度平台: <https://www.xuxueli.com/xxl-job/>
 - elastic-job: <https://gitee.com/elasticjob/elastic-job>
- 分布式服务调用
 - trpc
- 分布式存储
 - HDFS
 - Ceph
- 分布式数据库
 - TiDB
 - OceanBase
- 分布式文件系统
 - HDFS
- 分布式协调
 - Zookeeper
- 分布式监控
 - Prometheus
 - Zabbix
- 分布式消息队列
 - RabbitMQ
 - Kafka
 - Apache Pulsar
- 分布式日志收集
 - Elastic Stack

- Loki
- 分布式搜索引擎
 - Elasticsearch
- 分布式链路追踪
 - Apache SkyWalking
- 分布式配置中心
 - Apollo
 - Nacos

高可用

- 限流
- 降级熔断
- 冷备
- 双机热备
- 同城双活
- 异地双活
- 异地多活
- 容灾备份

高并发

- 数据库
 - 分库分表
 - MyCat 中间件
 - Apache ShardingSphere 中间件
 - 读写分离
- 缓存
 - 缓存雪崩
 - 缓存击穿
 - 缓存穿透
- 负载均衡
 - 负载均衡算法
 - 软硬件负载均衡 (2、3、4、7 层)

🌐 服务网格

服务网格用来描述组成应用程序的微服务网络以及它们之间的交互。服务网格的规模和复杂性不断增长，它将会变得越来越难以理解和管理，常见的需求包括服务发现、负载均衡、故障恢复、度量和监控等。

知识

- Istio
 - 流量管理
 - 安全性
 - 可观测性
- Envoy（开源的边缘和服务代理）

资源

- 资料大全: <https://www.servicemesh.com/awesome-servicemesh/>
- istio 官方文档: <https://preliminary.istio.io/latest/zh>
- istio 视频教程: <https://www.bilibili.com/video/BV1Lf4y1x7j8>
- 在线体验 istio: <https://katacoda.com/courses/istio/>

🌐 DDD 领域驱动设计

将数据、业务流程抽象成容易理解的领域模型，通过用代码实现领域模型，来组成完整的业务系统。

知识

- DDD 的优势
- DDD 的适用场景
- DDD 核心概念
 - 领域模型分类：失血、贫血、充血、涨血
 - 子域划分：核心域、通用域、支撑域
 - 限界上下文
 - 实体和值对象
 - 聚合设计
 - 领域事件
- DDD 实践

资源

- DDD 入门: <https://juejin.cn/post/7004002483601145863>
- DDD 编码实践: <https://juejin.cn/post/6844903903104860174>
- DDD 实践教程: <https://www.bilibili.com/video/BV1Y341167Xp>

🌐 其他

- Sidecar
- Serverless
- 云原生

学习建议

架构设计的学习没有顶点，多看文章，思考每种设计的优缺点和适用场景，有机会的话在企业中实践即可。

还在学校、或者初入这行的同学切记，千万不要一味地去背诵架构设计的八股文。你可以背，但是这一块的知识只有结合具体的项目才有意义，所以要多做项目去实践设计的合理性，而不是什么设计都咋咋往系统里去怼。比如面试问到分布式事务，能结合自己项目中用分布式事务解决问题的经验去回答最好。

🌐 其他技术

- 热数据探测技术: 京东 HotKey
- 数据库流水订阅: 阿里 Canal
- 监控告警
- 应用安全
- 故障演练
- 流量回放

阶段 7: Java 求职

目标

找到好工作

建议

1. 尽早做规划，可以通过大厂招聘官网的岗位描述来了解岗位的要求
2. 雕琢一份优秀的简历，推荐阅读：<https://mp.weixin.qq.com/s/wfabzdpOPdq89faF>
[FpZ4NA](#)
3. 多读面经，坚持刷算法
4. 多参与面试，持续复盘总结

资源

校招岗位

- ☆ 牛客求职校招版：<https://www.nowcoder.com/recommend/campus>
- 阿里 Java 开发：<https://campus.alibaba.com/position.htm?refno=12699>
- 腾讯后台开发：https://join.qq.com/post_detail.html?pid=1&id=101&tid=2
- 腾讯全栈开发：https://join.qq.com/post_detail.html?pid=1&id=137&tid=2
- 腾讯运营开发：https://join.qq.com/post_detail.html?pid=1&id=105&tid=2
- 美团后端开发：<https://campus.meituan.com/jobs?jobFamily=1&jobId=4005&jobType=1&pageNo=2>
- 华为软件开发：<https://career.huawei.com/reccampportal/portal5/campus-recruitment-detail.html?jobId=176044>

社招岗位

- ☆ 牛客求职社招版：<https://www.nowcoder.com/recommend/fulltime>
- 阿里社招：<https://job.alibaba.com/zhaopin/positionList.htm>
- 腾讯社招：<https://careers.tencent.com/search.html>

实习

- ☆ 牛客实习广场: <https://nowpick.nowcoder.com/w/intern/center>
- 实习僧: <https://www.shixiseng.com/>

内推

- 牛客内推: <https://www.nowcoder.com/discuss/referral/index>

鱼皮经历

- 从 0 开始的编程学习进大厂经历
 - ☆ 我学计算机的四年, 共勉: <https://mp.weixin.qq.com/s/Nc7Ee2UyrPg6hGseQUCzeg>
- 实习经历
 - ☆ 我的第一份实习: <https://mp.weixin.qq.com/s/SwGiRzyTkCOOCUXfXPR1Pw>
 - ☆ 我的第二份实习, 字节跳动: <https://mp.weixin.qq.com/s/3q-YIfuZBjPjSnamNwOraw>

知识总结

- ☆ 牛客 Java 面试宝典: <https://www.nowcoder.com/tutorial/94>
- ☆ JavaGuide: <https://snailclimb.gitee.io/javaguide/#/>
- ☆ CS-Notes: <http://www.cyc2018.xyz/>
- 阿里 Java 技术图谱: <https://developer.aliyun.com/graph/java>

面经

- ☆ 牛客面经: <https://www.nowcoder.com/discuss/experience?tagId=639>
- 秋招总结&Java心得 (双非硕士, 已拿11+5家Offer) : <https://www.nowcoder.com/discuss/327041>
- 被疫情逼得gap year的完全菜鸡的春招之路。: <https://www.nowcoder.com/discuss/427812>
- 谨以此文纪念2019年春招: <https://www.nowcoder.com/discuss/181212>
- 2019秋招经验谈: <https://www.nowcoder.com/discuss/122463>
- 秋招心得: <https://www.nowcoder.com/discuss/317888>

- 非科班本科菜鸡投的337家(Java后台): <https://www.nowcoder.com/discuss/133333>
- 2018 这一年: <https://www.nowcoder.com/discuss/137593>
- 双非的秋招, 感谢牛客: <https://www.nowcoder.com/discuss/142181>
- 二本本科Java上岸(滴滴京东58)2019秋招心得+面经: <https://www.nowcoder.com/discuss/333912>
- 碎碎念流水账特别长 of 自己的 Java 秋招: <https://www.nowcoder.com/discuss/354154>
- 秋招之路: 个人历程以及面经总结: <https://www.nowcoder.com/discuss/344311>
- 【秋招面经总结】菜逼的坎坷之路 (JAVA后台研发) : <https://www.nowcoder.com/discuss/352030>
- 阿里11面+EMC+网易+美团面经: <https://www.nowcoder.com/discuss/105302>
- 2018秋招阿里巴巴java笔试试题: <https://www.nowcoder.com/discuss/67379>
- 我的秋招经验分享 (已拿BAT头条网易) : <https://www.nowcoder.com/discuss/117071>
- 20届的阿里 头条 网易 滴滴 百度 小米等Java面经: <https://www.nowcoder.com/discuss/68289>
- 拿了5个offer, 回馈牛友发一波面经 (长文详细) : <https://www.nowcoder.com/discuss/79051>
- 【总结】零基础学习一年拿到BAT三家offer: <https://www.nowcoder.com/discuss/188241>
- 本科生秋招经验总结帖: <https://www.nowcoder.com/discuss/296586>
- 双非渣硕校招总结+Java面经 (网易阿里百度等) : <https://www.nowcoder.com/discuss/134850>
- 2018秋招笔试科大讯飞java笔试试题: <https://www.nowcoder.com/discuss/67684>
- 头条后台研发面经 (共三面) : <https://www.nowcoder.com/discuss/94233>
- 一点点小经验和几家Java面经: <https://www.nowcoder.com/discuss/138434>
- 感谢牛客网! 发一波面经! 阿里、网易游戏、京东等offer。 : <https://www.nowcoder.com/discuss/101712>
- 非科班水硕的春招实习总结- Java 后台开发: <https://www.nowcoder.com/discuss/186528>
- 十面阿里, 七面头条, 六个Offer, 春招结束: <https://www.nowcoder.com/discuss/80156>
- 成功上岸, 最后一波回馈: <https://www.nowcoder.com/discuss/176434>
- 【Java后台开发面经】阿里|网易|头条|腾讯|华为: <https://www.nowcoder.com/discuss/108655>
- 三跨学生的面经 (头条、拼多多、华为、vivo) : <https://www.nowcoder.com/discuss/195640>

面试题解

- ☆ 2021 最新 Java 笔试、面试题合集：https://pan.baidu.com/s/1DE4HAu-DS80n7VOaGpU_KQ（密码：lkw1）
- 牛客网 Java开发校招面试考点汇总（附面试题和答案）：<https://www.nowcoder.com/discuss/161991>
- 50w字+的技术类校招面试题汇总（附答案+考点）：<https://www.nowcoder.com/discuss/164726>
- 硬件，软件（后端开发，算法，数据分析等）历年企业笔试真题汇总：<https://www.nowcoder.com/discuss/212783>

视频

- ☆ 尚硅谷 2021 逆袭版 Java 面试题第三季：<https://www.bilibili.com/video/BV1Hy4y1B78T>
- ☆ 阿里大佬透彻讲解 Java 面试 500 道必考题：<https://www.bilibili.com/video/BV1yU4y1J7Bb>

阶段 8：持续学习

目标

持续追求技术的深度和广度，培养自己的 **核心竞争力** 和 **不可替代性**，学无止境！

学习建议

自主学习，自主搜索教程，多看书，多阅读技术博客，多实践。

学习方向

框架源码

- Spring
- SpringBoot
- SpringMVC
- MyBatis
- Netty
- Dubbo
- SpringCloud

计算机原理

- 《算法导论》：<https://www.bilibili.com/video/av48922404>
- 《现代操作系统》：<https://www.bilibili.com/video/av9555596>
- 《深入理解计算机系统》：<https://www.bilibili.com/video/av31289365>
- 《编译原理》：<https://www.bilibili.com/video/av17649289>
- 《计算机网络：自顶向下方法》：<https://www.bilibili.com/video/BV1JV411t7ow>
- 《计算机程序的构造和解释》：<https://www.bilibili.com/video/av8515129>
- 《数据库系统概论》：<https://www.bilibili.com/video/BV1G54y1d7ZK>

数据库 / 中间件 / 分布式

- 数据库
 - MySQL
 - PostgreSQL
- 缓存
 - Redis
- 队列
 - Apache Kafka
 - Apache Pulsar
- 搜索引擎
 - Elastic Stack
 - Elasticsearch
 - logstash
 - kibana
 - beats
- 容器
 - Docker

- K8S

解决方案

- 广告系统
- 电商系统
- 搜索系统
- 支付转账
- 游戏后台
- 即时通讯
- 社交系统
- CMS 系统
- ERP 系统
- OA 系统
- 代码生成
- 权限管理
- 秒杀活动

架构设计

同阶段 6 架构设计部分

大数据

- 5V 特点
- Hadoop
- HDFS
- MapReduce
- Spark
- Flink
- Storm
- Hive
- HBase
- Druid
- Kylin
- Pig
- Mahout

前沿技术

- 云原生: <https://www.jianshu.com/p/a37baa7c3eff>
 - Quasar Framework: <http://www.quasarchs.com/>
- 服务网格: <https://www.redhat.com/zh/topics/microservices/what-is-a-service-mesh>
 - istio: <https://github.com/istio/istio>
- GraalVM: <https://juejin.cn/post/6850418120570437646> (高性能的通用虚拟机)
 - 官网: <https://www.graalvm.org/>
- ZGC: <https://juejin.cn/post/6859276583656980493> (新一代垃圾回收器)
 - 官网: <http://openjdk.java.net/projects/zgc/>

自学 Java 专题资源

- ☆ GitHub Java 专区: <https://github.com/topics/java>
- ☆ GitHub Java 合集: <https://github.com/akullpp/awesome-java>
- StackOverflow: <https://stackoverflow.com/questions/tagged/java> (解决问题必备)
- ☆ 掘金 Java 专区: <https://juejin.cn/tag/Java>
- 美团技术团队: <https://tech.meituan.com/>
- 阿里技术团队: <https://102.alibaba.com/tech/list/>
- 有赞技术团队: <https://tech.youzan.com/tag/back-end/>

附录

程序员重要能力

想成为大佬，就是要通过学习和积累，提升自己的各方面能力，比较抽象，共勉！

1. 团队协作能力
2. 高效沟通能力
3. 需求分析能力
4. 架构设计能力

5. 抽象复用能力
6. 独立创造能力
7. 问题解决能力
8. 归纳总结能力
9. 自主学习能力
10. 工具利用能力
11. 高效编码能力
12. 信息检索能力
13. 开源建设能力
14. 源码阅读能力
15. 自测审查能力
16. 文档编写能力
17. 知识表达能力
18. 绘图描述能力
19. 兴趣驱动（长期学习能力）
20. 保持好奇心
21. 其他（狗头）
 - 复制粘贴能力
 - 打架能力
 - PPT 制作能力

编程误区

1. 我不适合学编程？ ✕ （兴趣比天赋更重要，不要怀疑自己）
2. 忽视基础，急于求成 ✕ （勿以浮沙筑高台）
3. 数学、英语差，就学不好编程 ✕ （开发工作基本不用高等数学，英语可以使用翻译软件、工作中提升）
4. 编程时，刻意记忆代码 ✕ （知道每行代码的意义，能看文档写出来就行）
5. 每行代码都要追求完美 ✕ （何为完美？每个人的定义不同）
6. 用百度搜索、复制粘贴的都是菜鸡程序员 ✕ （++）
7. 我做的项目太小了，就不整理了 ✕ （自己的每个项目都要整理，可以记录自己的成长）
8. 跟别人交流很麻烦，我更愿意自己写代码 ✕ （ $1 + 1 > 2$ ，切勿闭门造车，不过也要找到合适的同学去交流）

9. 做别人做过的项目没有意义 ✕ (连别人做过的项目都做不出, 还想造新轮子?)
10. 遇到 Bug, 想办法绕过而不是寻找解决方案 ✕ (Bug 最能使人成长)
11. 遇到问题, 第一时间去问别人而非自己解决 ✕ (建议阅读《提问的智慧》)
12. 花了很多时间、看了很多书, 就能学好编程 ✕ (编程最忌讳纸上谈兵)
13. 只向前学习, 不整理学过的知识 ✕ (建议每学完一个知识, 都去回想、总结、写心得, 这个时间的花费是值得的)
14. 单元测试不重要 ✕ (一棍打死)
15. 写文档不重要 ✕ (一棍打死)
16. 和我无关的工作, 就不需要了解 ✕ (了解团队、同行是很重要的, 全局视野)
17. 收藏的资源从来不看, 就不收藏了 ✕ (万一以后用到了呢?)
18. 不去考虑需求的价值, 而是一味寻求解决方案 ✕ (不要成为需求机器, 要有自己的独立思考)

高效法则

1. 不要过分追求完美, 完成比完美更重要。
2. 有现成的代码, 就不要自己写, 避免重复劳动 (学习除外)。
3. 学会使用软件及快捷键来提升自己的编码效率。
4. 做项目前, 要先想清楚怎么做, 做好充分的预研和设计。这样不仅便于后续的程序扩展, 也能避免无意义的返工。
5. 将大的目标进行拆解, 做好计划, 分清主次。不要因为觉得目标遥不可及而拖延, 也不要盲目乐观而松懈。
6. 养成好的作息习惯, 找到自己适合工作的黄金时间。
7. 记性不好, 就多记录、多总结、定期复习。哪怕实在记不住, 也可以借助文档、收藏夹软件来当自己的第二大脑。
8. 积累属于自己的工具库, 配合搜索软件实现随用随取。
9. 多和其他同行交流或订阅技术推送, 拓宽知识渠道, 从而了解更多的工具和方法。有些东西你只要听说过, 要用时也许就能节省大量查找时间。
10. 找到让自己进入专注做事的状态的方法, 比如戴上耳机、嚼口香糖等。
11. 做事高效, 离不开经验的积累, 因此要多写代码、多做项目。

常见问题

1. 按照这份路线学习, 每天要学多少个小时, 总共要学多久?

回答: 每个人学习能力不同, 当然没有固定标准。我当时平均每天可以学习 6 个小时, 寒暑假能保证 8 小时以上的学习, 路线中的时长是我当时学习时长 * 1.5 倍的参考值。实话说, 我的学习效率真的不高, 对编程感兴趣, 多学就好了。与其纠结这个时长, 倒不妨多抽点时间, 每天多学一点。

2. 这份路线上推荐的资源，鱼皮都看完了么？

回答：我是特么神仙啊？怎么可能都看完了！这份路线的本质是以网上现有的、死板的 Java 大学习框架为基础，结合我自己实际学习的知识、经验和感悟给它注入灵魂，从而希望给大家带来更大的帮助和价值。路线中的资源都是利用我自己的资源检索能力和渠道，通过试看、反复对比和参考评价精选出来的，不是主观臆断、空穴来风。

3. 我学到这份路线的哪个位置，可以找工作？

回答：这和你自己的领悟能力和教育背景有关。夸张点说，有些清华大学的同学，可能之前都没学过 Java，照样能找到大厂 Java 开发工作。学到同一阶段的同学，学习过程中的认真程度、积累总结数、做项目个数、基础学习的好坏不同，找到的工作肯定也不同。但一般情况下，想要进大厂，建议至少学完阶段 4 的微服务部分，然后去做一个完整的项目，用上之前学习的技术；不想进大厂的同学，学完第 3 阶段后，学下缓存和队列，一般也能找到 Java 开发的工作。

4. 学完这份路线就能进大厂么？

回答：你先学完，再问这个问题。没有绝对进大厂的保证，但是相信我，把这份路线上的大部分知识学了，再做做项目，哪怕进不了大厂，你拿到的薪水也绝对比松散学习的同学高！

5. 我可以相信你么？

回答：不要相信我，应该相信你自己。想进大厂，就不要怀疑自己的能力！

写在最后

觉得看文字累的朋友，欢迎看下视频版，带大家导读一下：<https://www.bilibili.com/video/BV1Qf4y1K7ff/>

以上就是鱼皮熬了几天大夜、结合个人经验、并且参考了大量网上的文章和评论总结而成的学习路线，真的是非常不容易，如果大家觉得满意请务必 **三连** 支持下。

我是鱼皮，持续分享编程干货的程序员，欢迎关注，下期再见~