

花了一个月零碎的时间，我根据当下 Java 后端求职和招聘的最新要求，对之前写的 Java 后端学习路线进行了全面的优化和改进。

这篇文章可能是你所见过的最用心、最全面的 Java 后端学习路线，共 4w+ 字。不过，也不用担心内容太多学不完，我会按照学习难度给出找一份小厂工作必学的内容以及适合循序渐进提高 Java 后端开发能力的学习路线。

对于初学者，你可以按照这篇文章推荐的学习路线和资料进行系统性的学习；对于有经验的开发者，你可以根据这篇文章更进一步地深入学习 Java 后端开发，提升个人竞争力。

为了保证内容的不至于太杂，这篇文章不会涉及到学习方法以及学习网站的推荐，可以看我之前写的下面这几篇文章：

- [我是怎么学习编程的？](#)
- [我常用的 20+个学习编程的网站！芜湖起飞！](#)
- [程序员如何快速学习新技术？](#)

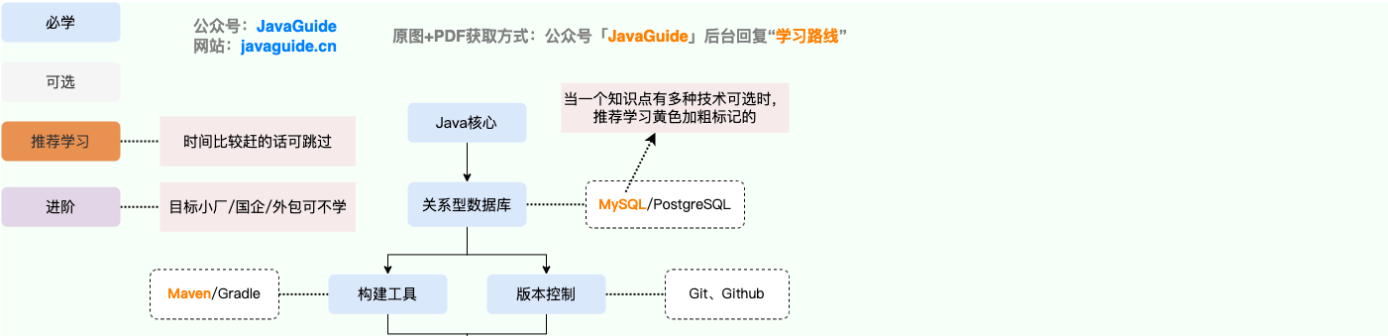
这篇文章也不会涉及到计算机基础的内容，关于计算机基础知识的学习可以参考我的网站上的分享，地址：<https://javaguide.cn/books/cs-basics.html>。

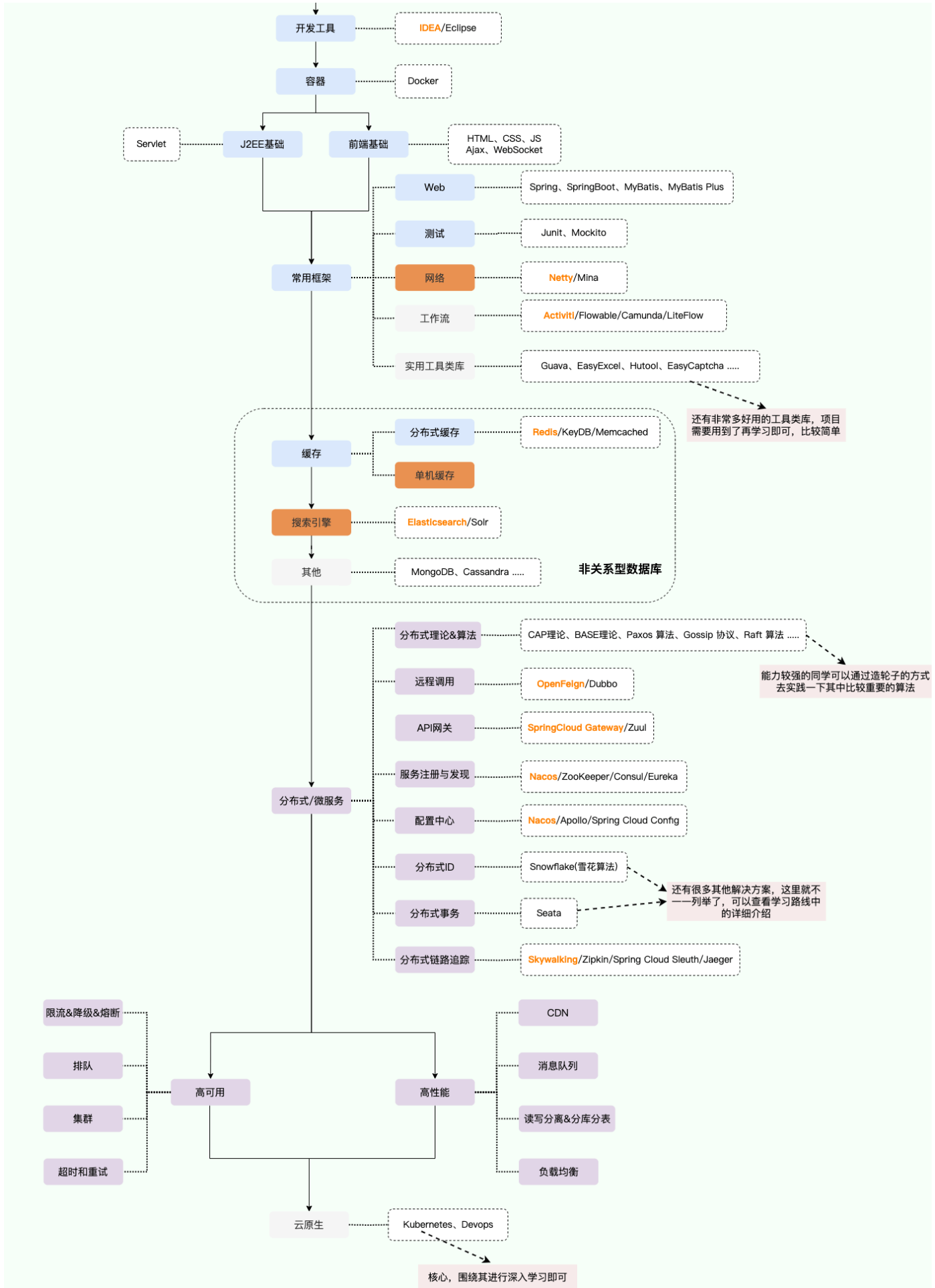
多说一句：对于编程初学者，我不太建议上来通过做项目学习。实践确实很重要，如果你没有编程基础的话，直接上手实战，很容易最后学个四不像。建议你在学习编程的初期尽量多看一些优质视频。跟着视频一步一步走，可以让你少踩很多坑，学习编程的信心也会增加。

Java 后端学习路线概览

我画了一张图，先简单带大家看看 Java 后端学习路线的全貌以及我所推荐的学习顺序。

下图中涉及到的每一个知识点都会在下文中详细介绍（附带学习资源推荐）。





上面这张图片的原图+PDF 版本，可以在公众号「JavaGuide」后台回复“学习路线”获取。

JavaGuide官方公众号 (微信搜索JavaGuide)



- 1、公众号后台回复“PDF”获取原创PDF面试手册
- 2、公众号后台回复“学习路线”获取Java学习路线最新版
- 3、公众号后台回复“开源”获取优质Java开源项目合集
- 4、公众号后台回复“八股文”获取Java面试真题+面经

内容比较多？劝退？如果你只想找到一份小厂的开发工作的话，建议你把重心放在 Java 基础、数据库、常用框架、常用工具上。

像 JVM、分布式、高并发、高可用、微服务这些知识点，如果你想进大厂或者说让自己在求职的时候更有竞争力，那你就也是要多花一点时间来学习的。

现在面试很卷，想要找到一个好工作的话，就需要你去多学一点，多练习一点。虽然，你目前学的很多知识，在你工作之后可能用不到，但是，面试的筛选就需要你会这些。毕竟，很多岗位是很多人一起竞争，为了达到筛选效果，面试难度通常都会比较大的。这也就是所谓的：“面试造火箭，入职拧螺丝”。

Java 核心

Java 基础

如果你之前没有学习过编程的话，我建议你可以看看视频教程。像尚硅谷的《Java 基础教程系列》和韩顺平老师的《零基础 30 天学会 Java》就很不错。



弹幕列表		展开
视频选集 (2/660)		
自动连播		
P1	0001_韩顺平Java_内容介绍	30:04
P2	0002_韩顺平Java_就业方向	02:54
P3	0003_韩顺平Java_开发场景	03:55
P4	0004_韩顺平Java_应用领域	04:01
P5	0005_韩顺平Java_内容梳理	09:08
P6	0006_韩顺平Java_程序举例	11:59
P7	0007_韩顺平Java_Java故事	07:49
P8	0008_韩顺平Java_java特性	07:35
P9	0009_韩顺平Java_sublime	06:20
P10	0010_韩顺平Java_idk介绍	20:03

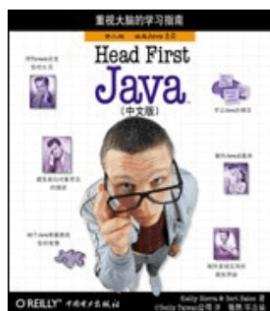
我整理了尚硅谷最新的 Java 后端学习系列完整的视频教程&资料，喜欢看视频的朋友可以点此链接下载：[尚硅谷 Java 后端开发系列视频教程&资料下载](#)。

全部文件 > 尚硅谷Java学科全套教程（总207.77G...	
☐	文件名
☐	尚硅谷全套JAVA教程--选学技术丰富（36.82GB）
☐	4.尚硅谷全套JAVA教程--实战项目（71.89GB）
☐	3.尚硅谷全套JAVA教程--微服务生态（66.68GB）
☐	2.尚硅谷全套JAVA教程--微服务核心（46.39GB）
☐	1.尚硅谷全套JAVA教程--基础必备（67.32GB）

看视频的同时，配套一本好书也是非常有作用的。

优秀的 Java 基础书籍非常多，我这里只推荐 3 本。

1、《Head First Java》

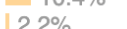


[更新图书信息或封面](#)

作者: Kathy Sierra, Bert Bates 著 / 杨尊一 编译 张然等 改编
出版社: 中国电力出版社
译者: 杨尊一
出版年: 2007-2
页数: 600
定价: 79.00元
装帧: 16开
丛书: O'Reilly Head First系列 (中译本)
ISBN: 9787508344980

豆瓣评分

8.7  1404人评价

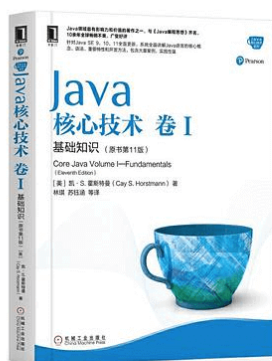
5星  49.9%
4星  36.5%
3星  10.4%
2星  2.2%
1星  1.0%

《Head First Java》这本书的内容很轻松有趣，可以说是我学习编程初期最喜欢的几本书之一了。同时，这本书也是我的 Java 启蒙书籍。我在学习 Java 的初期多亏了这本书的帮助，自己才算是跨进 Java 语言的大门。我在 Java 这块能够坚持下来，这本书有很大的功劳。我身边的很多朋友学习 Java 初期都是看的这本书。

有很多小伙伴就会问了：这本书适不适合编程新手阅读呢？

我个人觉得这本书还是挺适合编程新手阅读的，毕竟是“Head First”系列。

2、《Java 核心技术卷 1+卷 2》



[更新图书信息或封面](#)

作者: [美] 凯 S.霍斯特曼 (Cay S.Horstmann)
出版社: 机械工业出版社
副标题: 基础知识
原名: Core Java Volume I-Fundamentals (Eleven Edition)
译者: 林琪 / 苏钰涵
出版年: 2019-11-25
页数: 636
定价: 149元
装帧: 平装
丛书: Java核心技术系列
ISBN: 9787111636663

豆瓣评分

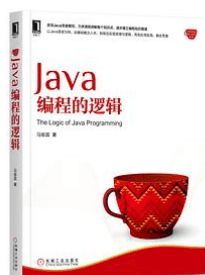
9.4  229人评价

5星  66.4%
4星  26.2%
3星  6.1%
2星  0.4%
1星  0.9%

《Java 核心技术卷 1+卷 2》这两本书的内容很多，全看的话比较费时间，比较适合当工作书。我当时在大学的时候就买了两本放在寝室，没事的时候就翻翻。个人建议有点 Java 基础之后再读这两本，介绍的还是比较深入和全面的。

3、《Java 编程的逻辑》

《Java 编程的逻辑》是一本非常低调的好书，相比于入门书来说，内容更有深度。适合初学者，同时也适合大家拿来复习 Java 基础知识。这篇文章中有这本书的阅读建议：[八股文骚套路之 Java 基础](#)。



作者: 马俊昌
出版社: 机械工业出版社
出版年: 2018-1-1
页数: 675
定价: 99
装帧: 平装
丛书: Java核心技术系列
ISBN: 9787111587729

豆瓣评分

9.2 
234人评价

5星		67.5%
4星		25.6%
3星		6.0%
2星		0.4%
1星		0.4%

学完 Java 基础之后，你可以用自己学的东西实现一个简单的 Java 程序，也可以尝试用 Java 解决一些编程问题，以此来将自己学到的东西付诸于实践。

不太建议学习 Java 基础的之后通过做游戏来巩固。为什么培训班喜欢通过这种方式呢？说白了就是为了找到你的 G 点。新手学习完 Java 基础后做游戏一般是不太现实的，还不如找一些简单的程序问题解决一下比如简单的算法题。

记得多总结！打好基础！把自己重要的东西都记录下来。API 文档放在自己可以看到的地方，以备自己可以随时查阅。为了能让自己写出更优秀的代码，《Effective Java》、《重构》这两本书没事也可以看。

学完这部分内容之后，务必确保自己掌握下面知识点：

- 基本语法、基本数据类型
- 对象、类、接口
- 继承、泛型
- 方法
- 异常、断言
- 集合
-

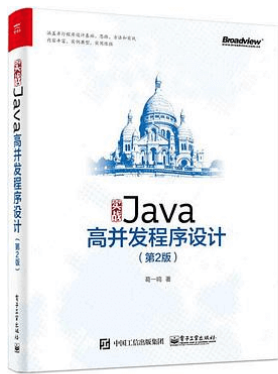
学习的过程中，强烈建议配合上我总结的常见问题和重要知识点（顺便还能准备一下面试常见问题）：

- [Java 基础常见面试题总结（上）](#)
- [Java 基础常见面试题总结（中）](#)
- [Java 基础常见面试题总结（下）](#)
- [Java 集合常见面试题总结（上）](#)
- [Java 集合常见面试题总结（中）](#)
- [Java 集合常见面试题总结（下）](#)

Java 并发（进阶）

并发或者说多线程这部分内容稍微会比较难以理解和实践。如果你刚学完 Java 基础的话，我建议你学习并发这部分内容的时候，可以先简单地了解一下基础知识比如线程和进程的对比。到了后面，你对于 Java 了解的更深了之后，再回来仔细看看这部分的内容。

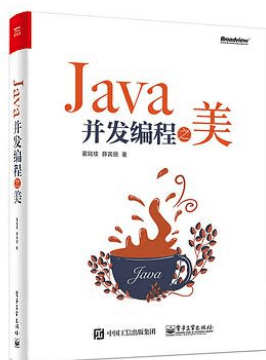
Java 并发书籍的话，挺多写的还不错的，比如《实战 Java 高并发程序设计》、《Java 并发编程之美》、《Java 并发实现原理：JDK 源码剖析》。



作者: 葛一鸣
出版社: 电子工业出版社
出品方: 博文视点
出版年: 2018-10
页数: 416
定价: 89
装帧: 平装
ISBN: 9787121350030



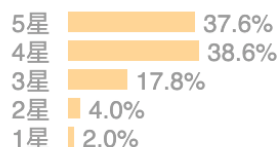
[更新图书信息或封面](#)



作者: 翟陆续 / 薛宾田
出版社: 电子工业出版社
出品方: 博文视点
出版年: 2018-10
页数: 356
定价: 89
ISBN: 9787121349478

豆瓣评分

7.7 ★★★★★
101人评价



Java并发实现原理：JDK源码剖析

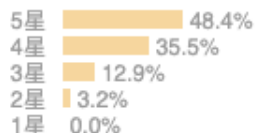


作者: 余春龙
出版社: 电子工业出版社
出版年: 2020-4
页数: 256
定价: 89元
装帧: 平装-胶订
ISBN: 9787121379727

更新图书信息或封面

豆瓣评分

9.0 ★★★★★
31人评价



想要系统学习的话，还是找从里面找一本认真阅读一下。当然，你也可以多选几本结合起来一起看，遇到不懂的知识点再去看看别的书籍的讲解或者找对应的博客讲解。

视频的话，还是推荐尚硅谷周阳老师讲的：<https://www.bilibili.com/video/BV1ar4y1x727/>。

我整理了尚硅谷最新的 Java 后端学习系列完整的视频教程&资料，喜欢看视频的朋友可以点此链接下载：[尚硅谷 Java 后端开发系列视频教程&资料下载](#)。

全部文件 > 尚硅谷Java学科全套教程 (总207.77G...

☐ 文件名 ▾

☐  尚硅谷全套JAVA教程--选学技术丰富 (36.82GB)

☐  4.尚硅谷全套JAVA教程--实战项目 (71.89GB)

☐  3.尚硅谷全套JAVA教程--微服务生态 (66.68GB)

☐  2.尚硅谷全套JAVA教程--微服务核心 (46.39GB)

☐  1.尚硅谷全套JAVA教程--基础必备 (67.32GB)

下面是我总结的一些关于并发的小问题，你可以拿来自测：

1. 什么是线程和进程？线程与进程的关系,区别及优缺点？
2. 说说并发与并行的区别？
3. 为什么要使用多线程呢？
4. 使用多线程可能带来什么问题？（内存泄漏、死锁、线程不安全等等）
5. 创建线程有哪几种方式？（a.继承 Thread 类;b.实现 Runnable 接口;c. 使用 Executor 框架;d.使用 FutureTask）
6. 说说线程的生命周期和状态？
7. 什么是上下文切换？
8. 什么是线程死锁?如何避免死锁？
9. 说说 `sleep()` 方法和 `wait()` 方法区别和共同点？
10. Java 内存模型（JMM）、重排序与 happens-before 原则了解吗？
11. `synchronized` 关键字、`volatile` 关键字
12. `ThreadLocal` 有啥用（解决了什么问题）？怎么用？原理了解吗？内存泄露问题了解吗？
13. 为什么要用线程池？`ThreadPoolExecutor` 类的重要参数了解吗？`ThreadPoolExecutor` 饱和策略了解吗？线程池原理了解吗？几种常见的线程池了解吗？为什么不推荐使用 `FixedThreadPool` ？如何设置线程池的大小？
14. AQS 了解么？原理？AQS 常用组件：`Semaphore`（信号量）、`CountDownLatch`（倒计时器）`CyclicBarrier`（循环栅栏）

15. `ReentrantLock` 、 `ReentrantReadWriteLock` 、 `StampedLock` (JDK8)
16. CAS 了解么？原理？
17. Atomic 原子类
18. 并发容器： `ConcurrentHashMap` 、 `CopyOnWriteArrayList` 、 `ConcurrentLinkedQueue` 、 `BlockingQueue` 、 `ConcurrentSkipListMap`
19. `Future` 和 `CompletableFuture`
20.

学习的过程中，强烈建议配合上我总结的常见问题和重要知识点（里面有上面分享并发小问题的详细参考答案）：

- [Java 并发常见面试题总结（上）](#)
- [Java 并发常见面试题总结（中）](#)
- [Java 并发常见面试题总结（下）](#)

JVM（进阶）

JVM 属于是比并发更高阶一些的内容，学习顺序可以适当延后，比如你可以在框架知识学完之后再回过头来看 JVM。并且，对于目标不是大厂的同学来说，这部分内容可以选择跳过，进而节省时间给更重要的知识点。

不过，我个人建议如果你学有余力的话，还是抽时间学习一下，还是有用的。正所谓只有搞懂了 JVM 才有可能真正把 Java 语言“吃透”。

实际工作中，中小厂一般不会做 JVM 调优，但万一遇到类似 OOM 的问题，你如果知道如何去排查和解决，岂不是更好？

学习 JVM 这部分的内容，一定要注意要实战和理论结合。

书籍的话，《深入理解 Java 虚拟机》这本书是首先要推荐的。

深入理解Java虚拟机（第3版）



更新图书信息或封面

作者: 周志明
出版社: 机械工业出版社
副标题: JVM高级特性与最佳实践
出版年: 2019-12
页数: 540
定价: 129.00元
装帧: 平装
丛书: 华章原创精品
ISBN: 9787111641247

豆瓣评分

9.5  412人评价

5星  80.6%
4星  16.3%
3星  3.2%
2星  0.0%
1星  0.0%

这本书就一句话形容：国产书籍中的战斗机，实实在在的优秀！（真心希望国内能有更多这样的优质书籍出现！加油！💪）

这本书的第三版已经出来挺久了，新增了很多不错的内容比如 ZGC 等新一代 GC 的原理剖析。

不论是你面试还是你想要在 Java 领域学习的更深，你都离不开这本书籍。这本书不光要看，你还要多看几遍，里面都是干货。这本书里面还有一些需要自己实践的东西，我建议你也跟着实践一下。

类似的书籍还有《实战 Java 虚拟机》、《虚拟机设计与实现:以 JVM 为例》，这两本都是非常不错的！

实战Java虚拟机



更新图书信息或封面

作者: 葛一鸣
出版社: 电子工业出版社
出品方: 博文视点
副标题: JVM故障诊断与性能优化
出版年: 2015-3
页数: 452
定价: 79.00元
装帧: 平装
ISBN: 9787121256127

豆瓣评分

8.2  95人评价

5星  35.8%
4星  48.4%
3星  13.7%
2星  0.0%
1星  2.1%

虚拟机设计与实现



更新图书信息或封面

作者: 李晓峰
出版社: 人民邮电出版社
副标题: 以JVM为例
译者: 单业
出版年: 2020-1
页数: 380
定价: 129元
装帧: 平装
丛书: 图灵程序设计丛书·Java系列
ISBN: 9787115527288

豆瓣评分

8.5 ★★★★★
28人评价

5星 46.4%
4星 35.7%
3星 14.3%
2星 3.6%
1星 0.0%

如果你对实战比较感兴趣，想要自己动手写一个简易的 JVM 的话，可以看看 《自己动手写 Java 虚拟机》 这本书。



更新图书信息或封面

作者: 张秀宏
出版社: 机械工业出版社
出版年: 2016-6-1
页数: 267
定价: 69.00元
装帧: 平装
丛书: 自己动手系列
ISBN: 9787111534136

豆瓣评分

8.2 ★★★★★
104人评价

5星 31.7%
4星 42.3%
3星 21.2%
2星 3.8%
1星 1.0%

书中的代码是基于 Go 语言实现的，搞懂了原理之后，你可以使用 Java 语言模仿着写一个，也算是练练手！如果你当前没有能力独立使用 Java 语言模仿着写一个的话，你也可以在网上找到很多基于 Java 语言版本的实现，比如 [《zachaxy 的手写 JVM 系列》](#)。

另外，R 大在豆瓣发的 [《从表到里学习 JVM 实现》](#) 这篇文章中也推荐了很多不错的 JVM 相关的书籍，推荐小伙伴们去看看。

再推荐两个视频给喜欢看视频学习的小伙伴。

第 1 个是尚硅谷的宋红康老师讲的 [《JVM 全套教程》](#)。这个课程的内容非常硬，一共有接近 400 小节（对应的浓缩精华版：[《尚硅谷 JVM 精讲与 GC 调优教程》](#)）。

课程的内容分为 3 部分：

- 1. 《内存与垃圾回收篇》
- 2. 《字节码与类的加载篇》
- 3. 《性能监控与调优篇》

弹幕列表 展开

视频选集 (1/381) 自动连播

- P372 372-GC日志相关VM参数选项的设置 18:07
- P373 373-了解其他VM参数选项 03:11
- P374 374-通过Java代码获取JVM参数 02:41
- P375 375-复习GC的分类 13:44
- P376 376-GC日志的分类 06:45
- P377 377-MinorGC的日志解析 06:56
- P378 378-FullGC的日志解析 04:43
- P379 379-在线日志分析工具: GCEasy 14:29
- P380 380-其它日志分析工具: GCViewe... 04:39
- P381 381-OOM及性能优化案例与解决方案 18:16

未来程序员们，入坑Java想清楚了吗？
bilibili课堂

我整理了尚硅谷最新的 Java 后端学习系列完整的视频教程&资料，喜欢看视频的朋友可以点此链接下载：[尚硅谷 Java 后端开发系列视频教程&资料下载](#)。

全部文件 > 尚硅谷Java学科全套教程（总207.77G...

文件名称

- 尚硅谷全套JAVA教程--选学技术丰富 (36.82GB)
- 4.尚硅谷全套JAVA教程--实战项目 (71.89GB)
- 3.尚硅谷全套JAVA教程--微服务生态 (66.68GB)
- 2.尚硅谷全套JAVA教程--微服务核心 (46.39GB)
- 1.尚硅谷全套JAVA教程--基础必备 (67.32GB)

第 2 个是你假笨大佬的 《JVM 参数【Memory 篇】》 教程，很厉害了！



你假笨

PerfMa KO 系列之 JVM 参数 -
【HeapSize】



你假笨

PerfMa KO 系列之 JVM 参数 -
【HeapSize 动态调整 ①】



你假笨

PerfMa KO 系列之 JVM 参数 -
【HeapSize 动态调整 ②】



你假笨

PerfMa KO 系列之 JVM 参数 -
【NewSize】



你假笨

PerfMa KO 系列之 JVM 参数 -
【Eden/Survivor Size】



你假笨

PerfMa KO 系列之 JVM 参数 -
【Perm Size】



你假笨

PerfMa KO 系列之 JVM 参数 -
【Metaspace Size】



你假笨

PerfMa KO 系列之 JVM 参数 -
【Thread Size】



https://blog.csdn.net/yq_34337272

下面是我总结的一些关于 JVM 的小问题，你可以拿来自测：

1. 什么是虚拟机？
2. Java 内存区域是怎么划分的？大对象放在哪个内存区域？
3. 垃圾回收有哪些算法？GC 的流程
4. 什么是类加载？何时类加载？类加载流程？
5. 知道哪些类加载器。类加载器之间的关系？
6. 类加载器的双亲委派了解么？结合 Tomcat 说一下双亲委派（Tomcat 如何打破双亲委托机制？...）。
7. 常见调优参数有哪些？
8.

学习的过程中，强烈建议配合上我总结的常见问题和重要知识点（里面有上面分享 JVM 小问题的详细参考答案）：

- [Java 内存区域详解（重点）](#)
- [JVM 垃圾回收详解（重点）](#)
- [类文件结构详解](#)
- [类加载过程详解](#)
- [类加载器详解（重点）](#)

数据库

基础（可选）

数据库基础知识点的话，其实是可选择性学习的。对于计算机专业的同学来说，大学的时候应该也学习过。不过，绝大部分学了之后也相当于没学，没学过的也不用担心哈！

这里还是提供一些学习资料给想要学习数据库基础知识的同学把！

书籍的话，强烈推荐《数据库系统概念》，这本书涵盖了数据库系统的全套概念，知识体系清晰，是学习数据库系统非常经典的教材！不是参考书！

数据库系统概念



作者: (美) Abraham Silberschatz / (美) Henry F.Korth / (美) S.Sudarshan
出版社: 机械工业出版社
副标题: (原书第6版)
原作名: Database System Concepts, 6E
译者: 杨冬青 / 李红燕 / 唐世渭
出版年: 2012-3
页数: 805
定价: 99.00元
装帧: 平装
丛书: 计算机科学丛书
ISBN: 9787111375296

豆瓣评分

8.4  416人评价

5星		45.7%
4星		41.1%
3星		11.1%
2星		1.2%
1星		1.0%

如果你觉得书籍比较枯燥，自己坚持不下来的话，我推荐你可以先看看一些不错的视频。就比如北京师范大学的《数据库系统原理》这个就很不错。

这个课程的老师讲的非常详细，而且每一小节的作业设计的也与所讲知识很贴合，后面还有很多配套实验。

北京师范大学

数据库系统原理

国家精品

申请认证证书

评价课程

我的学习

课程

实验 (二) : 认识数据库

实验指导书

实验

实验 认识数据库

1. 实验目的

(1) 初步建立数据库的概念。

(2) 了解数据库中表、行、列、主键的概念。

(3) 了解 QBE 查询。

(4) 感受正能量，传播正能量。

2. 实验环境（写清硬件配置和软件版本）

1. 硬件；

2. 操作系统；

3. 数据库管理系统及图形化管理工具；

3. 实验内容

(1) 熟悉 Access 工作环境。

(2) 创建 Access 数据库表。

(3) 对所创建的表进行查询。

微信扫码课程进度

扫码下载 APP

微信扫码课程进度

扫码下载 APP

如果你比较喜欢动手，对于理论知识比较抵触的话，我推荐你看看《如何开发一个简单的数据库》，这个 project 会手把手教你编写一个简单的数据库。

Let's Build a Simple Database

Writing a sqlite clone from scratch in C

Overview

View on GitHub (pull requests welcome)

How Does a Database Work?

- What format is data saved in? (in memory and on disk)
- When does it move from memory to disk?
- Why can there only be one primary key per table?
- How does rolling back a transaction work?
- How are indexes formatted?
- When and how does a full table scan happen?
- What format is a prepared statement saved in?

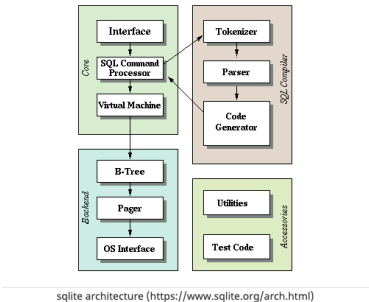
In short, how does a database **work**?

I'm building a clone of **sqlite** from scratch in C in order to understand, and I'm going to document my process as I go.

Table of Contents

- Part 1 - Introduction and Setting up the REPL
- Part 2 - World's Simplest SQL Compiler and Virtual Machine
- Part 3 - An In-Memory, Append-Only, Single-Table Database
- Part 4 - Our First Tests (and Bugs)
- Part 5 - Persistence to Disk
- Part 6 - The Cursor Abstraction
- Part 7 - Introduction to the B-Tree
- Part 8 - B-Tree Leaf Node Format
- Part 9 - Binary Search and Duplicate Keys
- Part 10 - Splitting a Leaf Node
- Part 11 - Recursively Searching the B-Tree
- Part 12 - Scanning a Multi-Level B-Tree
- Part 13 - Updating Parent Node After a Split

"What I cannot create, I do not understand." - Richard Feynman



纸上学来终觉浅 绝知此事要躬行！强烈推荐 CS 专业的小伙伴一定要多多实践！！！！

MySQL

对于 Java 开发来说，虽然 PostgreSQL 也挺火，但 MySQL 是主流，国内的绝大部分企业还是用的 MySQL。

MySQL 入门可以找一些视频看看，比如黑马的《MySQL 数据库入门到精通》。看视频的过程中，可以配套一本 MySQL 入门类的书籍比如《MySQL 必知必会》。

初期不需要学太深了，搞清楚下面这些知识点即可：

1. MySQL 常用命令：

- 安全：登录、增加/删除用户、备份数据和还原数据
- 数据库操作： 建库建表/删库删表、用户权限分配
-

2. MySQL 中常用的数据类型、字符集编码

3. MySQL 简单查询、条件查询、模糊查询、多表查询以及如何对查询结果排序、过滤、分组.....

4. MySQL 中使用索引、视图、存储过程、游标、触发器

5.

更进一步的话，可以找一些优秀的书籍来学习底层原理和性能优化，比如《高性能 MySQL》和《MySQL 技术内幕》。



另外，强推一波《MySQL是怎样运行的》这本书，内容很适合拿来准备面试。讲的很细节，但又不枯燥，内容非常良心！

MySQL是怎样运行的



作者: 小孩子4919
出版社: 人民邮电出版社
出品方: 异步图书
副标题: 从根儿上理解 MySQL
原作名: 小孩子4919
出版年: 2020-11-1
页数: 456
定价: 99.00元
装帧: 平装
ISBN: 9787115547057

豆瓣评分

9.5 ★★★★★
159人评价

5星 78.0%
4星 13.2%
3星 6.9%
2星 0.6%
1星 1.3%

如果你想让自己更加了解 MySQL，同时也是为了准备面试的话，下面这些知识点要格外注意：

1. 索引：索引优缺点、B 树和 B+树、聚集索引与非聚集索引、覆盖索引
2. 事务：事务、数据库事务、ACID、并发事务、事务隔离级别
3. 存储引擎（MyISAM 和 InnoDB）
4. 锁机制与 InnoDB 锁算法

学习的过程中，强烈建议配合上我总结的常见问题和重要知识点：

- [MySQL 常见面试题总结](#)

- [MySQL 索引详解](#)
- [MySQL 三大日志\(binlog、redo log 和 undo log\)详解](#)
- [MySQL 事务隔离级别详解](#)
- [InnoDB 存储引擎对 MVCC 的实现](#)
- [SQL 语句在 MySQL 中的执行过程](#)

PostgreSQL（可选）

和 MySQL 一样，PostgreSQL 也是开源免费且功能强大的关系型数据库。PostgreSQL 的 Slogan 是“世界上最先进的开源关系型数据库”。

<https://www.postgresql.org> :

PostgreSQL: The world's most advanced open source database

PostgreSQL is a powerful, open source object-relational database system with over 30 years of active development that has earned it a strong reputation for ...

客观来说，PostgreSQL 确实比 MySQL 优秀。不过，目前国内 MySQL 还是主流，PostgreSQL 是可选择性学习的。

PostgreSQL 中文文档建议看看：<http://www.postgres.cn/docs/14/index.html>。另外，PostgreSQL 书籍的话，看这里的推荐即可：<https://javaguide.cn/books/database.html#postgresql>。

Redis

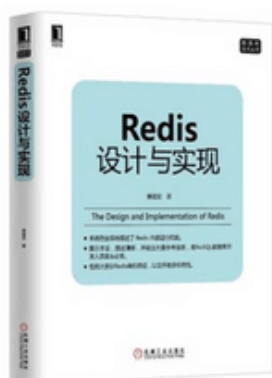
后端项目如果用到分布式缓存的话，一般用的都是 Redis。不过，Redis 不仅仅能做缓存，还能用作分布式锁、延时队列、消息队列等等。

免费视频教程的话，推荐慕课网的《[Redis 入门](#)》（一些简单的使用）和尚硅谷的《[Redis 7 系列最新视频](#)》（阳哥出品，内容更全面，Redis 版本更新，强烈推荐）。

书籍的话，强烈推荐《[Redis 设计与实现](#)》和《[Redis 核心原理与实践](#)》这两本书。

《[Redis 核心原理与实践](#)》这本书出版日期相对近一些，主要是结合源码来分析 Redis 的重要知识点比如各种数据结构和高级特性。

Redis设计与实现



作者: 黄健宏
出版社: 机械工业出版社
出版年: 2014-6
页数: 388
定价: 79.00
装帧: 平装
丛书: 数据库技术丛书
ISBN: 9787111464747

豆瓣评分

8.7 ★★★★★
1487人评价

5星	50.0%
4星	41.9%
3星	7.1%
2星	0.5%
1星	0.5%

Redis核心原理与实践



作者: 梁国斌
出版社: 电子工业出版社
出版年: 2021-8-18
页数: 468
定价: 138.00元
装帧: 平装
ISBN: 9787121415487

豆瓣评分

8.0 ★★★★★
16人评价

5星	50.0%
4星	25.0%
3星	18.8%
2星	0.0%
1星	6.3%

付费专栏的话，推荐一个极客时间的《Redis 核心技术与实战》，虽然未涉及到太多新版 Redis 的内容，但胜在内容全面且清晰易懂。我当时看这个专栏确实学了不少东西，尤其是评论区有很多大佬的精彩评论。

下面是我总结的一些关于并发的小问题，你可以拿来自测：

1. Redis 和 Memcached 的区别和共同点
2. 为什么要用 Redis/为什么要用缓存？
3. 常见的缓存读写策略有哪些？
4. Redis 除了做缓存，还能做什么？如何基于 Redis 实现分布式锁？Redis 可以做消息队列么？
5. Redis 常见数据结构以及使用场景分析
6. Redis 没有使用多线程？为什么不使用多线程？Redis6.0 之后为何引入了多线程？
7. Redis 给缓存数据设置过期时间有啥用？

8. Redis 是如何判断数据是否过期的呢？
9. 过期的数据的删除策略了解么？
10. Redis 内存淘汰机制了解么？
11. Redis 持久化机制(怎么保证 Redis 挂掉之后再重启数据可以进行恢复)
12. Redis 缓存穿透、缓存雪崩？
13. 如何保证缓存和数据库数据的一致性？
14.

学习的过程中，强烈建议配合上我总结的常见问题和重要知识点（里面有上面分享 Redis 小问题的详细参考答案）：

- [缓存基础常见面试题总结](#)
- [Redis 常见面试题总结（上）](#)
- [Redis 常见面试题总结（下）](#)
- [Redis 5 种基本数据类型详解](#)
- [Redis 3 种特殊数据类型详解](#)
- [Redis 持久化机制详解](#)
- [Redis 内存碎片详解](#)

MongoDB（可选）

MongoDB 作为 Java 后端开发来说，是可选择性学习的，用的不多，面试一般也不会问，除非你的项目用到了 MongoDB。

这里就不推荐视频或者书籍了，推荐两篇我写的文章：

- [MongoDB 常见面试题总结（上）](#)
- [MongoDB 常见面试题总结（下）](#)

常用工具

非常重要！非常重要！特别是 Git 和 Docker。

除了下面这些工具之外，我强烈建议你一定要搞懂 Github 的使用。一些使用 Github 的小技巧，你可以看 [《Github 小技巧》](#) 这篇文章。

IDEA

俗话说：“工欲善其事，必先利其器！”。选择一款好的开发工具对于我们高效率编码非常有帮助！

常用的 Java 开发工具就 Eclipse 和 IDEA。就我个人而言 IDEA 是最适合 Java 开发者的 IDE，没有之一（勿杠，你喜欢的就是最好的）。

除了 IDEA 自身对编码优秀的支持（比如智能上下文提示）之外，IDEA 中还有丰富的插件来帮助我们高效开发。

建议你要熟悉 IDEA 的基本操作以及常用快捷。你可以通过 Github 上的开源教程 [《IntelliJ IDEA 简体中文专题教程》](#) 来学习 IDEA 的相关使用。

「[IDEA 高效使用指南](#)」是我创建的一个网站，上面包含了下面这些内容：

- IDEA 使用技巧
- IDEA 必备插件
- IDEA 插件开发入门
- 使用 IDEA 进行重构的小技巧
- 使用 IDEA 进行源码阅读的技巧



Maven

Maven 其实使用起来挺简单的，一两天时间就能入门基本使用了。不过，想要用好还是挺难的，初期的时候会基本使用就好了。

多提一句：学习常用框架之前可以提前花时间学习一下 Maven 的使用，千万不要到处找 Jar 包，下载 Jar 包（如果你做的项目没用上包管理工具，那请你尽快换一个新点的教程看）。

Maven 这里不用推荐什么视频或者书籍了，直接看下面这篇文章即可：

- [Maven 核心概念总结](#)
- [Maven 最佳实践](#)
- [四十五图，一万五千字！一文让你走出迷雾玩转 Maven！](#)

学完之后，务必要搞懂下面这些问题（初学者搞懂前两个问题即可）：

1. Maven 项目如何创建？如何引入依赖？
2. Maven 依赖冲突如何解决？

3. Maven 多模块项目的构建、运行、打包如何做？

4. Maven 私服如何搭建？

Git

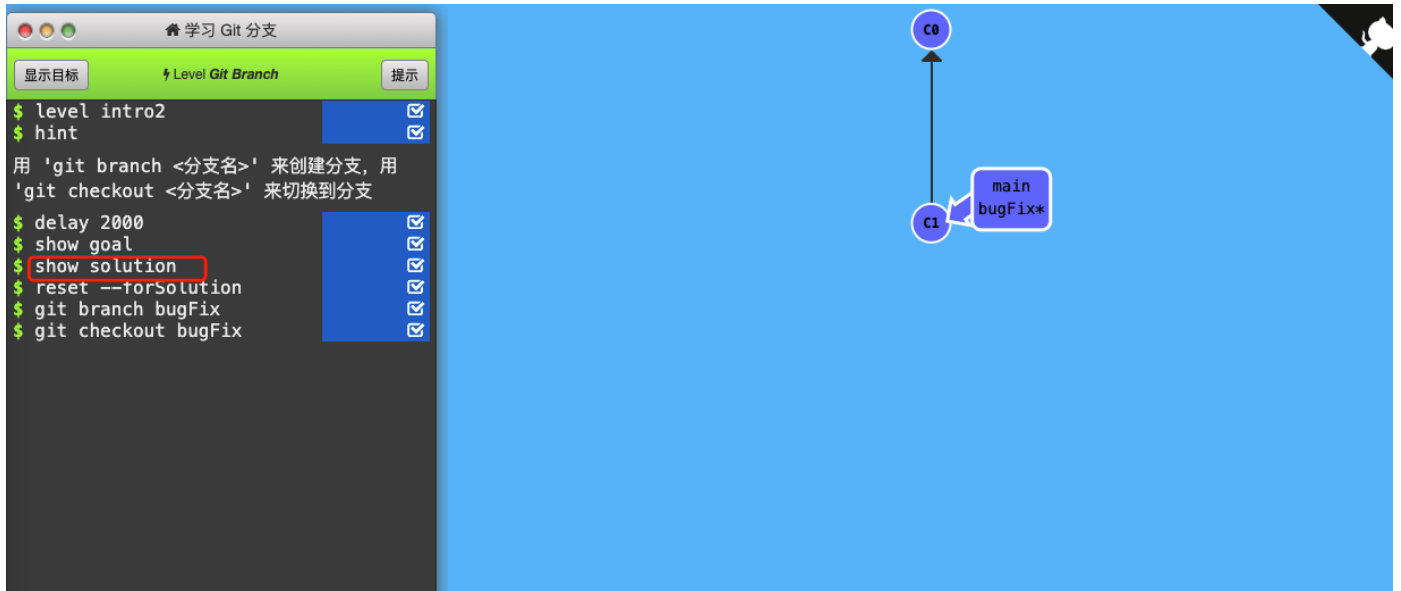
Git 技能对于程序员来说也是必备的！试着在学习的过程中将自己的代码托管在 Github 上，有一个漂亮的 Github 主页在求职面试中是十分加分的。并且，现在的企业都是基于 Git 在 GitHub 或 GitLab 平台上做版本控制。

学习 Git 的话，强烈推荐给大家一个可以交互式学习 Git 的网站 [Learn Git Branching](#)。效果真的非常非常棒，通过游戏的方式让你学习 Git 的常见操作。

整个教程分为很多关，每一关都有非常详细的指导，还会有详细的动图展示结果。并且，你做错了之后还可以使用 `reset` 命令从头开始。



如果你是在不知道答案的话，还可以使用 `show solution` 命令查看答案。



这种即时反馈的学习让过程变得有趣！真心感谢这个网站的作者，太爱了！

另外，你可以看看 Github 上开源的这篇 [《Git 极简入门》](#)，像版本控制和 Git 的相关概念、Git 常见操作这篇文章都有介绍到。

如果想要详细了解 Git 的话，[Git 官方文档教程](#)是肯定要看的，介绍的非常全面，并且有中文版！

About

Documentation

[Reference](#)**Book**[Videos](#)[External Links](#)

Downloads

Community

This book is available in [English](#).

Full translation available in

[azərbaycan dili](#),
[български език](#),
[Deutsch](#),
[Español](#),
[Français](#),
[Ελληνικά](#),
[日本語](#),
[한국어](#),
[Nederlands](#),
[Русский](#),
[Slovenščina](#),
[Tagalog](#),
[Українська](#),
[简体中文](#),

Partial translations available in

[Čeština](#),
[Македонски](#),
[Polski](#),
[Српски](#),
[Ўзбекча](#),
[繁體中文](#),

Translations started for

[Беларуская](#),
[فارسی](#),
[Indonesian](#),
[Italiano](#),
[Bahasa Melayu](#),
[Português \(Brasil\)](#),
[Português \(Portugal\)](#),
[Svenska](#),
[Türkçe](#).

Book

The entire Pro Git book, written by Scott Chacon and Ben Straub and published by Apress, is available here. All content is licensed under the [Creative Commons Attribution Non Commercial Share Alike 3.0 license](#). Print versions of the book are available on [Amazon.com](#).



2nd Edition (2014)

Download Ebook



1. 起步

- 1.1 关于版本控制
- 1.2 Git 简史
- 1.3 Git 是什么?
- 1.4 命令行
- 1.5 安装 Git
- 1.6 初次运行 Git 前的配置
- 1.7 获取帮助
- 1.8 总结

2. Git 基础

- 2.1 获取 Git 仓库
- 2.2 记录每次更新到仓库
- 2.3 查看提交历史
- 2.4 撤销操作
- 2.5 远程仓库的使用
- 2.6 打标签
- 2.7 Git 别名
- 2.8 总结

3. Git 分支

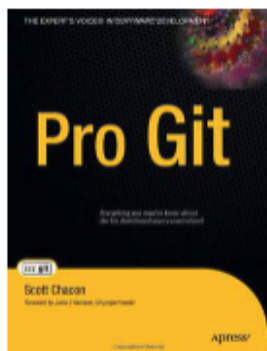
- 3.1 分支简介
- 3.2 分支的新建与合并
- 3.3 分支管理
- 3.4 分支开发工作流
- 3.5 远程分支
- 3.6 变基
- 3.7 总结

4. 服务器上的 Git

- 4.1 协议
- 4.2 在服务器上搭建 Git
- 4.3 生成 SSH 公钥
- 4.4 配置服务器
- 4.5 Git 守护进程
- 4.6 Smart HTTP
- 4.7 GitWeb
- 4.8 GitLab
- 4.9 第三方托管的选择
- 4.10 总结

《Pro Git》这本书也非常不错，还有中文版，内容非常全面，硬核！

Pro Git



作者: [Scott Chacon](#)
出版社: Apress
出版年: 2009-8-27
页数: 288
定价: USD 34.99
装帧: Paperback
ISBN: 9781430218333

豆瓣评分

9.1 
1124人评价

5星  60.0%
4星  34.6%
3星  5.3%
2星 0.1%
1星 0.0%

想读

在读

读过

评价: ☆☆☆☆☆

 写笔记  写书评  加入购书单 分享到 ▾

推荐



如果你比较喜欢看视频教程的话，可以看看极客时间的《玩转 Git 三剑客》，课程的作者是携程代码平台负责人苏玲，讲的挺不错的！

Docker

传统的开发流程中，我们的项目通常需要使用 MySQL、Redis、FastDFS 等等环境，这些环境都是需要我们手动去进行下载并配置的，安装配置流程极其复杂，而且不同系统下的操作也不一样。

Docker 的出现完美地解决了这一问题，我们可以在容器中安装 MySQL、Redis 等软件环境，使得应用和环境架构分开，它的优势在于：

- 1. 一致的运行环境，能够更轻松地迁移
- 2. 对进程进行封装隔离，容器与容器之间互不影响，更高效地利用系统资源
- 3. 可以通过镜像复制多个一致的容器

Docker 常见概念解读，可以看这篇 JavaGuide 的这篇[Docker 基本概念解读](#)，从零到上手实战可以看[Docker 从入门到上手干事](#)这篇文章，内容非常详细！

另外，再给大家推荐一本质量非常高的开源书籍《[Docker 从入门到实践](#)》，这本书的内容非常新，毕竟书籍的内容是开源的，可以随时改进。

 Docker — 从入门到实践

前言

修订记录

如何贡献

Docker 简介

- 什么是 Docker
- 为什么要用 Docker

基本概念

安装 Docker

使用镜像

操作容器

访问仓库

数据管理

使用网络

高级网络配置

Docker Buildx

由 GitBook 提供支持

前言

Stars 23k

release v1.3.0

Based Docker CE v20.10

Docker 技术入门与实践 jd.com

v1.3.0

语言	-
简体中文	阅读

Docker 是个划时代的开源项目，它彻底释放了计算虚拟化的威力，极大提高了应用的维护效率，降低了云计算应用开发的成本！使用 Docker，可以让应用的部署、测试和分发都变得前所未有的高效和轻松！

无论是应用开发者、运维人员、还是其他信息技术从业人员，都有必要认识和掌握 Docker，节约有限的生命。

本书既适用于具备基础 Linux 知识的 Docker 初学者，也希望可供理解原理和实现的高级用户参考。同时，书中给出的实践案例，可供在进行实际部署时借鉴。前六章为基础内容，供用户理解 Docker 的基本概念和操作；7~9 章介绍包括数据管理、网络等高级操作；第 10~12 章介绍了容器生态中的几个核心项目；13、14 章讨论了关于 Docker 安全和实现技术等高级话题。后续章节则分别介绍包括 Etcd、Fedora CoreOS、Kubernetes、容器云等相关热门开源项目。最后，还展示了使用容器技术的典型的应用场景和实践案例。

- 在线阅读：[docker-practice.com](#), [GitBook](#), [Github](#)
- 离线阅读：`$ docker run -it --rm -p 4000:80 ccr.ccs.tencentyun.com/dockerpracticesig/docker_practice:v`

大纲

微信小程序

技术交流

进阶学习

鼓励项目

最后，在学习完 Docker 的常见操作之后，建议大家以一个前后端分离的项目为例，去实践部署一下。比如，你可以选择部署自己的简历项目，这样的话，项目经历部分贴上在线体验地址，也算是一个加分项了！

设计模式

软件开发中有一个概念叫做“**软件复用**”。简单来说，软件复用就是我们在构建一个新的软件的时候，不需要从零开始，通过复用已有的一些轮子（框架、第三方库等）、**设计模式**、设计原则等等现成的物料，我们可以更快地构建出一个满足要求的软件。

软件复用需要设计模式的帮助。因为，在软件开发中，设计模式可以通过封装变化来提高代码的可扩展性和可维护性！

在我们平时工作的业务开发中，如果你不会设计模式，你或许也可以完成项目的功能需求。但是！单纯 CRUD 多没意思啊！我们要思考如何写出质量更高的业务代码。另外，各种框架比如 Spring、MyBatis 中都大量使用了设计模式。如果，你想要搞懂他们的原理，设计模式也是你的必备利器。

设计模式不光需要我们在学习，最重要的还是要不断去实践体会。但是！设计模式不是银弹，**不要为了用设计模式而用设计模式**。

想要看书学习设计模式的话，首推《重学 Java 设计模式》。有趣的例子，配合形象的图片，通过实战案例讲解设计模式的方式秒极了！文中的每一个细节无不透露着作者的用心！每一种设计模式实际都不难理解，大部分读者最需要的还是设计模式的实战经验。如果你能细心思考实践《重学 Java 设计模式》中的每一个案例，我相信，你对设计模式的理解一定会更上一层楼！

重学Java设计模式（全彩）



更新图书信息或封面

作者: 付政委
出版社: 电子工业出版社
原名: 付政委
出版年: 2021-5-1
页数: 367
定价: 128
ISBN: 9787121409387

豆瓣评分

★★★★★
评价人数不足

我读过这本书 2021-07-12 修改 删除

我的评价: ★★★★★ 力荐 标签: Java 程序设计 设计模式 代码质量 计算机 软件工程

@稀土掘金技术社区

想要看视频学习的话，首推 《尚硅谷 Java 设计模式（图解+框架源码剖析）》 这个视频。

尚硅谷Java设计模式（图解+框架源码剖析）

83.3万播放 · 3.6万弹幕 2019-07-05 15:10:45

500瑟瑟发抖

尚硅谷

- Java设计模式内容介绍
- 4) Spring中原型bean的创建，就是原型模式的应用
- 5) 代码分析+Debug源码

```
beans.xml
<bean id="id01" class="com.atguigu.spring.bean.Monster" scope="Test.java"
ApplicationContext applicationContext = new
ClassPathXmlApplicationContext("beans.xml");
//获取monster通过id获取monster
Object bean = applicationContext.getBean("id01");
System.out.println("bean" + bean);

@Override
public Object getBean(String name) throws BeansException {
return doGetBean(name, null, null, false);
}
```

33 人正在看 (全站30+人在看), 861 条弹幕

发个友善的弹幕见证当下

弹幕礼仪 > 发送

9010 9011 3.8万 1374

稿件投诉 笔记

尚硅谷 发消息
程序员标配，人手一套尚硅谷教程，自学一样...
为TA充电 + 关注 62.4万

弹幕列表 展开

视频选集 (1/149)			自动连播
P1	001_尚硅谷_设计模式面试题(1)	11:19	
P2	002_尚硅谷_设计模式面试题(2)	11:22	
P3	003_尚硅谷_何时会用到设计模式	21:32	
P4	004_尚硅谷_内容和授课方式	11:12	
P5	005_尚硅谷_七大设计原则介绍	13:23	
P6	006_尚硅谷_单一职责原则	20:12	
P7	007_尚硅谷_单一职责原则小结	09:42	
P8	008_尚硅谷_接口隔离原则(1)	14:32	
P9	009_尚硅谷_接口隔离原则(2)	10:56	
P10	010_尚硅谷_接口隔离原则小结	05:35	

2021版 马士兵重讲23种设计模式，居然能这么...
马士兵-小美
3.1万播放 · 313弹幕

【狂神说Java】通俗易懂的23种设计模式教学 (...)
遇见狂神说
17.7万播放 · 3070弹幕
@稀土掘金技术社区
【狂神说Java】

这个视频通过图解+框架源码分析的方式全面地讲解了设计模式相关的内容，包括设计模式七大原则、UML 类图-类的六大关系、23 种设计模式及其分类等知识点。

我整理了尚硅谷最新的 Java 后端学习系列完整的视频教程&资料，喜欢看视频的朋友可以点此链接下载：[尚硅谷 Java 后端开发系列视频教程&资料下载](#)。

全部文件 > 尚硅谷Java学科全套教程 (总207.77G...

☐ 文件名 ▾

☐ 尚硅谷全套JAVA教程——选学技术丰富 (36.82GB)

☐ 4.尚硅谷全套JAVA教程——实战项目 (71.89GB)

☐ 3.尚硅谷全套JAVA教程——微服务生态 (66.68GB)

☐ 2.尚硅谷全套JAVA教程——微服务核心 (46.39GB)

☐ 1.尚硅谷全套JAVA教程——基础必备 (67.32GB)

Linux

对于 Java 程序员来说， 我们需要掌握 Linux 基本的使用，尤其是是各种常用的命令比如：目录切换命令、目录操作命令、文件的操作命令、压缩或者解压文件的命令等等。像 Linux 内核架构、底层原理这些底层内容，不是必需的，可以根据自身情况来决定是否学习。

对于想要快速入门 Linux 的同学来说，建议阅读我写的 [Linux 基础知识总结](#) 这篇文章，里面介绍了 Java 程序员必知的 Linux 的一些概念以及常见命令。

对于想要系统学习的同学来说，还是建议看书籍，像《鸟哥的 Linux 私房菜》系列就挺不错的。不过，内容有点太多了，个人还是更建议作为工具书参考或者选择自己感兴趣的内容章节进行学习。

鸟哥的Linux私房菜 基础学习篇 第四版



更新图书信息或封面

作者: 鸟哥

出版社: 人民邮电出版社

出品方: 异步图书

出版年: 2018-10

页数: 796

定价: 118.00元

装帧: 平装

丛书: 鸟哥的Linux私房菜

ISBN: 9787115472588

豆瓣评分

8.4  355人评价

5星  50.4%
4星  33.8%
3星  13.2%
2星  1.7%
1星  0.8%

不要忘记学习一下 Shell 编程了，这个也是必须要掌握的，快速入门可以阅读我写的 [Shell 编程基础知识总结](#) 这篇文章，总结了 Shell 变量、基本运算符、流程控制、函数这些重要的知识点。

前端基础

笔者主要从事 Java 后端开发的，对于前端的了解属于皮毛，刚刚入门的状态（当过一年全栈），这里只是简单聊聊自己的看法。

前端框架更新换代的很快，目前比较流行的是 Vue 和 React。对于国内的同学，Vue 更适合投入精力学习，因为国内用 Vue 的公司更多一些。不过，前端框架并不是必学的，可以根据自身情况来决定是否学习。

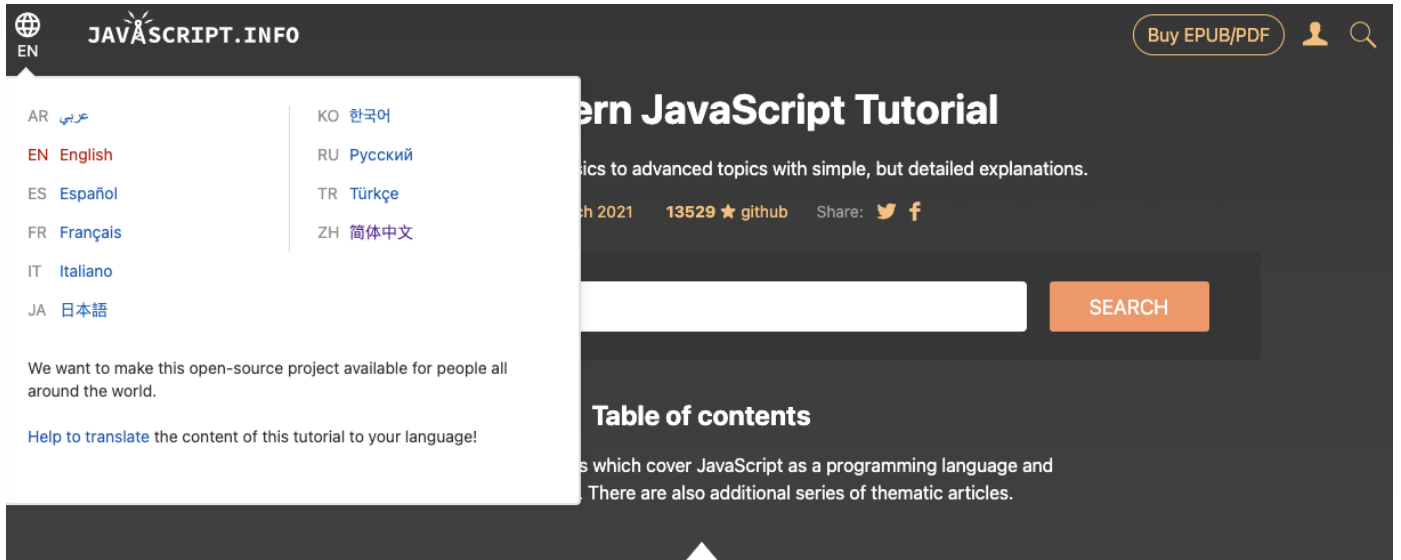
不过，不管前端这些技术怎么变，前端三剑客（HTML、CSS、JavaScript）是不会变的，也是必学的。

HTML 和 CSS 相对 JS 来说就比较简单了。你可以在 [W3school](#) 上学习一些关于 HTML、CSS、JS 的基础知识。然后，通过一个简单的前端项目来实战一下。比如你可以做一个个人简历或者模仿某某官网写个类似的网页。

JavaScript 的水更深，也是前端面试中的重心。

学习 JS 的话，[MDN](#) 上的 JS 相关的内容是必须要看的！上面的内容很全面，质量非常高！

除此之外，开源的 JS 教程《[The Modern JavaScript Tutorial](#)》非常赞！目前的话，这个系列的教程还被翻译成了多国的语言。



PART 1

The JavaScript language

Here we learn JavaScript, starting from scratch and go on to advanced concepts like OOP. We concentrate on the language itself here, with the minimum of environment-specific notes.

An introduction

- 1.1 [An Introduction to JavaScript](#)
- 1.2 [Manuals and specifications](#)
- 1.3 [Code editors](#)
- 1.4 [Developer console](#)

JavaScript Fundamentals

- 2.1 [Hello, world!](#)
- 2.2 [Code structure](#)
- 2.3 [The modern mode, "use strict"](#)
- 2.4 [Variables](#)
- 2.5 [Data types](#)
- 2.6 [Interaction: alert, prompt, confirm](#)
- 2.7 [Type Conversions](#)
- 2.8 [Basic operators, maths](#)
- 2.9 [Comparisons](#)
- 2.10 [Conditional branching: if, '?'](#)
- 2.11 [Logical operators](#)
- 2.12 [Nullish coalescing operator '??'](#)
- 2.13 [Loops: while and for](#)
- 2.14 [The "switch" statement](#)
- 2.15 [Functions](#)
- 2.16 [Function expressions](#)
- 2.17 [Arrow functions, the basics](#)
- 2.18 [JavaScript specials](#)

Code quality

- 3.1 [Debugging in Chrome](#)
- 3.2 [Coding Style](#)
- 3.3 [Comments](#)
- 3.4 [Ninja code](#)
- 3.5 [Automated testing with Mocha](#)
- 3.6 [Polyfills and transpilers](#)

这个教程的内容分为 3 部分

1. JavaScript 编程语言： JavaScript 入门，还会介绍 OOP 等相关高级概念。
2. 浏览器（文档，事件，接口）： 学习如何管理浏览器页面
3. 其他文章： 按需学习其他 JavaScript 高级知识。

另外，除了一些老项目之外，现在一般都是前后端分离开发，也就是前端和后端可以独立开发、测试和部署，两者之间通过 API 进行通信。因此，后端程序员还需要掌握：

- HTTP 协议（计算机网络部分的内容，这里再多提一下）

- RESTful API 的设计和使用
- 前后端通信的常见方式：比如 Ajax（短连接）、WebSocket（长连接，双向的）

J2EE 基础

Servlet

Servlet 属于比较古老的技术了，现在你几乎不会直接使用到 Servlet 相关的 API。不过，学习 Servlet 有助于我们搞清各种封装的比较好的 Web 框架的原理，比如 Spring MVC 不过就是对 Servlet 的封装，它的底层还是依赖于 Servlet。

在 Java Web 程序中，Servlet 主要负责接收用户请求 HttpServletRequest，在 doGet()，doPost() 中做相应的处理，并将回应 HttpServletResponse 反馈给用户。

你可以通过书籍《Head First Servlets & JSP（中文版）》或者《Servlet 和 JSP 学习指南》来学习 Servlet 基础知识。

注意：JSP 就不要学了，过时的技术，已经被淘汰了！

Web 服务器

Tomcat 是 Apache 基金会下的一个项目，主要用作 Web 服务器。

如果你直接学习 Spring Boot 的话，不学习 Tomcat 也没什么影响（建议还是学一学）。因为 Spring Boot（spring-boot-starter-web）使用 Tomcat 作为默认的嵌入式 Servlet 容器，你使用起来是无感知的。

简单来说，Tomcat 主要实现了 2 个核心功能：

1. 处理 Socket 连接，负责网络字节流与 Request 和 Response 对象的转化。
2. 加载和管理 Servlet，以及具体处理 Request 请求。

如果你要深入研究 Tomcat 的话，首选极客时间的《深入拆解 Tomcat & Jetty》这个专栏。这是我看过讲解 Tomcat 底层原理最好的资料，强烈推荐！

这个专栏不光可以加深自己对于 Tomcat 的理解，还能提高自己对于系统架构、性能优化等领域的思考。



深入拆解Tomcat & Jetty

从源码角度深度探索Java中间件

李号双
eBay技术主管

☆ 收藏

开始学习

8292 人已学习 · 已完结

课程介绍

课程目录

你将获得

- 通过Tomcat&Jetty深入理解Java进阶技术；
- 掌握了Tomcat&Jetty中间件的设计思想；
- 解读Servlet最新技术及实际应用；
- 让Web程序飞起来的性能调优策略。

除了 Tomcat 之外，Nginx 也是必须要学习的！

Nginx 是一个高性能的 HTTP 和反向代理服务服务器，经常被拿来做反向代理和负载均衡。

如果你要学习 Nginx 的话，可以看看《[Nginx 核心知识 150 讲](#)》。内容很全面，从概念、代码再到实战，从 HTTP 到 OpenResty 。

常用框架

实际面试中，框架类知识问的不多，学习常用框架更多地是为了满足项目开发需要以及工作要求。

Spring/SpringBoot

没有学习 Spring 可以直接上手学习 SpringBoot 吗？

明确的说，必须可以！目前绝大部分企业都是用的 SpringBoot，Spring 也并不是学习 Spring Boot 的前置基础，相比于 Spring 来说，Spring Boot 要更容易上手一些！如果你只是想使用 Spring Boot 来做项目的话，直接学 Spring Boot 就可以了。

不过，个人还是建议提前搞懂 Spring AOP 和 IoC 这两比较重要的概念之后再去学习 SpringBoot。除此之外，准备面试的话，Spring 中 bean 的作用域与生命周期、SpringMVC 工作原理详解等等知识点都是非常重要的，一定要搞懂。推荐阅读这篇文章：[Spring 常见面试题总结](#)。

学习 Spring Boot 的话，还是建议可以多看看 [《Spring Boot 的官方文档](#)》，写的很详细。

像 SpringBoot 和一些常见技术的整合你也要知道怎么做，比如 SpringBoot 整合 MyBatis、ElasticSearch、SpringSecurity、Redis 等等。尽量还是实践一下，写一些 Demo。到了后期，甚至可以独立做一些小项目把这些知识都应用上。

书籍的话，个人其实并没有什么特别好的推荐，毕竟是框架类知识，更新换代的比较快，很多书籍的内容都已经过时了。

考虑到很多同学比较喜欢阅读书籍，我这里还是简单推荐几本吧！

对于想要实战的同学，我推荐 2023 年出的《Java EE 企业级应用开发实战》，内容相对新一些，也很全面，介绍到了企业级应用的概念、Java EE 的基础知识、Servlet、Spring、Spring Boot、Vue 等内容。并且，在书籍最后还提供了两个可以满足中小型企业级应用实战需求的案例：企业级应用基础开发框架和企业级门户网站。

Java EE企业级应用开发实战（Spring Boot+Vue+Element）



[更新图书信息或封面](#)

作者: 李磊
出版社: 人民邮电出版社
出版年: 2023-4-27
页数: 336
定价: 79.80
装帧: 平装
ISBN: 9787115610218

豆瓣评分

9.6 ★★★★★
31人评价

5星	80.6%
4星	0.0%
3星	0.0%
2星	6.5%
1星	12.9%

对于专研 Spring Boot 底层原理同学，可以看看 [《Spring Boot 编程思想（核心篇）》](#)。



[更新图书信息或封面](#)

作者: 小马哥
出版社: 电子工业出版社
出品方: 博文视点
出版年: 2019-4
页数: 628
定价: 118.00元
装帧: 平装
ISBN: 9787121360398

豆瓣评分

6.5 ★★★★☆
112人评价

5星	22.3%
4星	19.6%
3星	25.0%
2星	14.3%
1星	18.8%

这本书稍微有点啰嗦，不过，原理介绍的比较清楚（不适合初学者）。

如果你比较喜欢看视频的话，推荐尚硅谷雷神的 [《2023 版 Spring Boot3 零基础入门》](#)。这可能是全网质量最高并且免费的 Spring Boot 教程了，好评爆炸！

我整理了尚硅谷最新的 Java 后端学习系列完整的视频教程&资料，喜欢看视频的朋友可以点此链接下载：[尚硅谷 Java 后端开发系列视频教程&资料下载](#)。

全部文件 > 尚硅谷Java学科全套教程 (总207.77G...

☐ 文件名 ▾

- ☐  尚硅谷全套JAVA教程——选学技术丰富 (36.82GB)
- ☐  4.尚硅谷全套JAVA教程——实战项目 (71.89GB)
- ☐  3.尚硅谷全套JAVA教程——微服务生态 (66.68GB)
- ☐  2.尚硅谷全套JAVA教程——微服务核心 (46.39GB)
- ☐  1.尚硅谷全套JAVA教程——基础必备 (67.32GB)

另外，Spring Boot 这块还有很多优质的开源教程，我已经整理好放到 [Java 优质开源技术教程](#) 中了。

■ 技术教程

- 实战项目
- ✖ 系统设计
- 工具类库
- ✎ 开发工具
- 🤖 机器学习
- 📊 大数据

SpringBoot

- [springboot-guide](#)：SpringBoot 核心知识点总结。基于 Spring Boot 2.19+。
- [SpringAll](#)：循序渐进，学习 Spring Boot、Spring Boot & Shiro、Spring Cloud、Spring Security & Spring Security OAuth2，博客 Spring 系列源码。
- [Springboot-Notebook](#)：一系列以 Spring Boot 为基础开发框架，整合 Redis、Rabbitmq、ES、MongoDB、Spring Cloud、Kafka、Skywalking 等互联网主流技术，实现各种常见功能点的综合性案例。
- [springboot-learning-example](#)：Spring Boot 实践学习案例，是 Spring Boot 初学者及核心技术巩固的最佳实践。
- [spring-boot-demo](#)：spring boot demo 是一个用来深度学习并实战 spring boot 的项目，目前总共包含 63 个集成 demo，已经完成 52 个。
- [SpringBoot-Labs](#)：Spring Boot 系列教程。

MyBatis

MyBatis 是国内使用最多的 ORM 框架。在学习 Spring/Spring Boot 的时候，你就要顺带去学习 MyBatis，这个我在上面也提到过。

另外，建议你还要掌握至少一个 MyBatis 增强框架，这里推荐两个国产的：

1. [MyBatis-Plus](#)：简称 MP，在 MyBatis 的基础上只做增强不做改变，为简化开发、提高效率而生。
2. [MyBatis-Flex](#)：非常轻量、同时拥有极高的性能与灵活性的 MyBatis 增强框架。

对于做项目的同学，也可以直接选择学习使用 MyBatis 增强框架。

单元测试

对于单测来说，目前常用的单测框架有：JUnit、Mockito、Spock、PowerMock、JMockit、TestableMock 等等。

JUnit 几乎是默认选择，但是其不支持 Mock，因此我们还需要选择一个 Mock 工具。Mockito 和 Spock 是最主流的两款 Mock 工具，一般都是在这两者中选择。

究竟是选择 Mockito 还是 Spock 呢？我这里做了一些简单的对比分析：

1. Spock 没办法 Mock 静态方法和私有方法，Mockito 3.4.0 以后，支持静态方法的 Mock，具体可以看这个 issue：<https://github.com/mockito/mockito/issues/1013>，具体教程可以看这篇文章：<https://www.baeldung.com/mockito-mock-static-methods>。
2. Spock 基于 Groovy，写出来的测试代码更清晰易读，比较规范(自带 given-when-then 的常用测试结构规范)。Mockito 没有具体的结构规范，需要项目组自己约定一个或者遵守比较好的测试代码实践。通常来说，同样的测试用例，Spock 的代码要更简洁。
3. Mockito 使用的人群更广泛，稳定可靠。并且，Mockito 是 SpringBoot Test 默认集成的 Mock 工具。

Mockito 和 Spock 都是非常不错的 Mock 工具，相对来说，Mockito 的适用性更强一些。

这里顺带推荐一些测试相关的学习资料：

1. [阿里内部单元测试培训教程](#)
2. [单元测试到底是什么？应该怎么做？](#)
3. [Integration Testing in Spring](#)
4. [Testing the Web Layer](#)
5. [可能是全网最好的 Spock 单测入门文章：](#)
6. [单元测试框架 Mockito 落地实践分享](#)
7. [如何写出有效的单元测试](#)

Netty（可选）

Netty 是 Java 网络编程最热门的框架，大家可以根据个人需要决定是否进行学习，实际企业开发中用的不多。

不过，个人建议学有余力的同学还是抽时间认真学习一下，对个人开发能力的提升还是很有帮助的。

- 1. Netty 基于 NIO（NIO 是一种同步非阻塞的 I/O 模型，在 Java 1.4 中引入了 NIO）。使用 Netty 可以极大地简化并简化了 TCP 和 UDP 套接字服务器等网络编程,并且性能以及安全性等很多方面都非常优秀。
- 2. 我们平常经常接触的 Dubbo、RocketMQ、Elasticsearch、gRPC、Spark、Elasticsearch 等等热门开源项目都用到了 Netty。
- 3. 大部分微服务框架底层涉及到网络通信的部分都是基于 Netty 来做的，比如说 Spring Cloud 生态系统中的网关 Spring Cloud Gateway。

下面是一些比较推荐的书籍/专栏。

《Netty 实战》



更新图书信息或封面

作者: 诺曼·毛瑞尔 (Norman Maurer) / 马文·艾伦·沃尔夫泰尔 (Marvin Allen Wolfthal)

出版社: 人民邮电出版社

出品方: 异步图书

副标题: Netty IN ACTION

原作名: Netty IN ACTION

译者: 何品

出版年: 2017-5-1

页数: 276

定价: 69.00

装帧: 平装

ISBN: 9787115453686

豆瓣评分

7.5  339人评价

5星	21.2%
4星	42.5%
3星	28.3%
2星	5.0%
1星	2.9%

这本书可以用来入门 Netty，内容从 BIO 聊到了 NIO、之后才详细介绍为什么有 Netty、Netty 为什么好用以及 Netty 重要的知识点讲解。

这本书基本把 Netty 一些重要的知识点都介绍到了，而且基本都是通过实战的形式讲解。

《Netty 进阶之路：跟着案例学 Netty》



[更新图书信息或封面](#)

作者: 李林锋
出版社: 电子工业出版社
出品方: 博文视点
出版年: 2018-11-1
页数: 340
定价: 79.00元
装帧: 平装
ISBN: 9787121352621

豆瓣评分

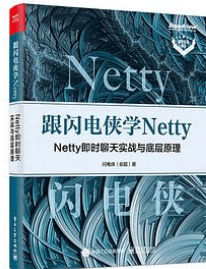
7.5  38人评价

5星  26.3%
4星  42.1%
3星  23.7%
2星  2.6%
1星  5.3%

内容都是关于使用 Netty 的实践案例比如内存泄露这些东西。如果你觉得你的 Netty 已经完全入门了，并且你想要对 Netty 掌握的更深的话，推荐你看一下这本书。

《跟闪电侠学 Netty：Netty 即时聊天实战与底层原理》

跟闪电侠学 Netty：Netty 即时聊天实战与底层原理



[更新图书信息或封面](#)

作者: 俞超
出版社: 电子工业出版社
出版年: 2022-3
页数: 344
定价: 108
ISBN: 9787121426797

豆瓣评分

 评价人数不足

这本书分为上下两篇，上篇通过一个即时聊天系统的实战案例带你入门 Netty，下篇通过 Netty 源码分析带你搞清 Netty 比较重要的底层原理。

视频的话，黑马的 [黑马程序员 Netty 全套教程](#) 就挺不错的，从 Netty 的基础知识 NIO 讲起，比较容易接受。



弹幕列表 展开

视频选集 (1/157) 自动连播

P1	Netty-导学	19:08
P2	第1章_01_nio三大组件-channel-buffer	05:52
P3	第1章_02_nio三大组件-服务器设计-多线程...	06:21
P4	第1章_03_nio三大组件-服务器设计-线程...	05:00
P5	第1章_04_nio三大组件-服务器设计-sele...	06:11
P6	第1章_05_bytebuffer-基本使用	16:55
P7	第1章_06_bytebuffer-内部结构	11:13
P8	第1章_07_bytebuffer-方法演示1	09:37
P9	第1章_08_bytebuffer-方法演示2	09:30
P10	第1章_09_bytebuffer-方法演示3	09:04



IDEA教程
Java
权威教程



Java
并发编程
全面深入 面试必考

Java基础+进阶

家豪的猜想
381.9万 播放 · 17.2万 弹幕

黑马程序员全面深入学习
Java并发编程，JUC并发...
黑马程序员
89.1万 播放 · 3.2万 弹幕

工作流（可选）

国内用的比较多的开源工作流引擎是 Flowable 和 Activiti 这两个，参考资料也蛮多的。Camunda 也不错，更轻量，功能也很完善，性能和稳定性也很不错。关于开源流程引擎的选择，可以参考这篇文章：<https://zhuanlan.zhihu.com/p/369761832>。

ps：Flowable 和 Camunda 都是 Activiti5 的一个分支发展而来，三者的理念有所差别。

国内比较火的工作流引擎 LiteFlow（<https://liteflow.cc/>）只做基于逻辑的流转，而不做基于角色任务的流转。如果你想做基于角色任务的流转，推荐使用 Flowable 和 Activiti 这两个框架。也就是说，像审批流（A 审批完应该是 B 审批，然后再流转到 C 角色）这种 LiteFlow 就不适合了。LiteFlow 适用于拥有复杂逻辑的业务，比如说价格引擎，下单流程等，这些业务往往都拥有很多步骤，这些步骤完全可以按照业务粒度拆分成一个个独立的组件，进行装配复用变更。

这里就不推荐学习资料了，感兴趣的同学可以自己去找一下。

搜索引擎

搜索引擎用于提高搜索效率，功能和浏览器搜索引擎类似。比较常见的搜索引擎是 Elasticsearch（推荐）和 Solr。

如果你要学习 Elasticsearch 的话，[Elastic 中文社区](#) 以及 [Elastic 官方博客](#) 都是非常不错的资源，上面会分享很多具体的实践案例。

视频教程可以看看尚硅谷的 [《ElasticSearch 入门到精通》](#)，前面基于 ElasticSearch 7.x 讲解，后面加更了 Elasticsearch8.x 新特性。

除此之外，极客时间的 [《Elasticsearch 核心技术与实战》](#) 这门课程非常赞！这门课基于 Elasticsearch 7.1 版本讲解，比较新。并且，作者是 eBay 资深技术专家，有 20 年的行业经验，课程质量有保障！



如果你想看书的话，可以考虑一下《Elasticsearch 实战》这本书。不过，需要说明的是，这本书中的 Elasticsearch 版本比较老，你只能通过这本书学习一些理论性的知识。

Elasticsearch 实战



作者: Radu Gheorghe
出版社: 人民邮电出版社
出品方: 异步图书
原作名: Elasticsearch in Action
译者: 黄申
出版年: 2018-10
页数: 337
定价: 79.00
装帧: 平装
ISBN: 9787115449153

豆瓣评分

7.7 ★★★★★
56人评价

5星 16.1%
4星 48.2%
3星 28.6%
2星 3.6%
1星 3.6%

想读 在读 读过 评价: ☆☆☆☆☆

写笔记 写书评 加入购书单 分享到

推荐

ps: 个人不是特别推荐看书学习 Elasticsearch 的, 真不如看我后面推荐的一些文章。

最后, 再推荐一些 ElasticSearch 相关的优秀文章和专辑来帮助你学习和更好的使用 ElasticSearch:

- [Elasticsearch 常见面试题总结 - JavaGuide](#)
- [Elasticsearch 基础入门详文 - 腾讯技术工程](#)
- [在工作中 ElasticSearch 的一些使用规范](#)
- [《滴滴技术的 ES 系列》](#)
- [《死磕 Elasticsearch 系列》](#) (上百篇 ES 的理论+实战文章, 全网最全面的 ES 教程。部分内容对应的视频教程: <https://space.bilibili.com/471049389>)

分布式/微服务 (进阶)

这部门内容涉及到的知识点比较多, 我这里只列举比较重要的部分比如分布式算法和协议、配置中心、分布式事务。

学习分布式知识, 个人比较建议阅读书籍和博客。当然了, 如果比较喜欢看视频的话, 也可以找一些不错的教程视频或者公开课来看, 用适合自己的学习方式去学习即可!

书籍推荐：

《深入理解分布式系统》这本书非常不错。这本书的作者用了大量篇幅来介绍分布式领域中非常重要的共识算法，并且还会基于 Go 语言带着你从零实现了一个共识算法的鼻祖 Paxos 算法。

深入理解分布式系统



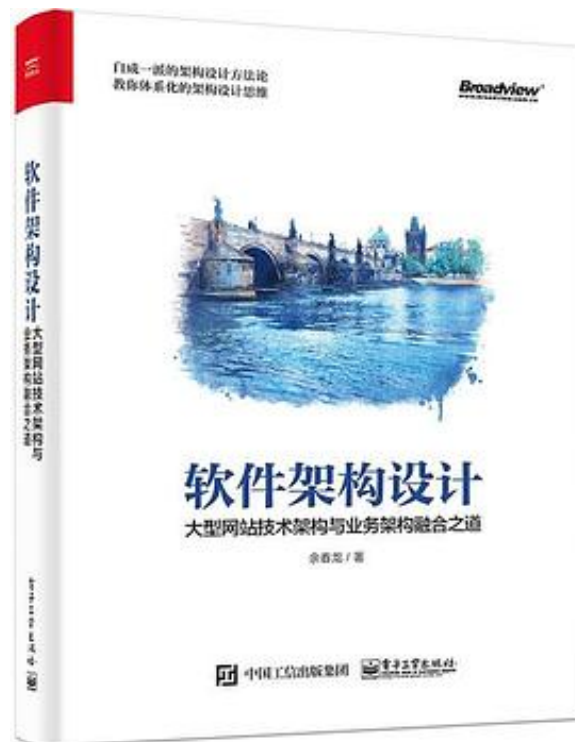
作者: 唐伟志
出版社: 电子工业出版社
出版年: 2022-3
页数: 316
定价: 108
ISBN: 9787121428111



《从零开始学架构》这本书的内容比较全面，分布式、微服务、高并发、高可用这些都有涉及到。这本书对应的是极客时间的专栏—《从零开始学架构》，里面的很多内容都是这个专栏里面的，两者选一个阅读就行了。



余老师的《软件架构设计：大型网站技术架构与业务架构融合之道》这本书类似于《从零开始学架构》，内容同样比较全面，也很不错。



公开课推荐：

MIT6.824: Distributed System 这门公开课挺经典的。这门课每节课都会精读一篇分布式系统领域的经典论文，并由此传授分布式系统设计与实现的重要原则和关键技术。

- [如何的才能更好地学习 MIT6.824 分布式系统课程？](#)
- [MIT6.824: Distributed System \(中文翻译 wiki\)](#)
- [MIT6.824: Distributed System - CS 自学指南](#)

理论&算法&协议

比较重要的分布式理论&算法&协议有：CAP 理论、BASE 理论、Paxos 算法、Gossip 协议、Raft 算法等等。

文章推荐：

- [CAP & BASE 理论详解](#)
- [Paxos 算法详解](#)
- [Raft 算法详解](#)
- [Gossip 协议详解](#)

远程调用

不同服务之间的调用一般有两种方法：

- **RPC**：RPC（Remote Procedure Call）即远程过程调用，通过 RPC 可以帮助我们调用远程计算机上某个服务的方法，这个过程就像调用本地方法一样简单。Dubbo 是一款国产的 RPC 框架，由阿里开源，国内用的最多。
- **HTTP 客户端**：通过 HTTP 协议调用其他服务的 RESTful API。Feign 和 OpenFeign（Spring Cloud 官方基于 Feign 开发的，用于替代已经进入停更维护状态的 Feign）是目前最常用的 HTTP 客户端。

OpenFeign 和 Dubbo 都是目前广泛应用于微服务架构的远程调用框架，但两者实现方式不同（OpenFeign 基于 HTTP 协议，Dubbo 支持多种协议，还可以自定义协议），适合的场景也略有区别。Spring Cloud 微服务项目现在用的比较多的是基于 Rest 风格的调用方式的 OpenFeign，个人比较建议学习这个。

不过，如果你跟着教程做的项目用的是 Dubbo 或者工作需要用到 Dubbo 的话，那你可以主要学习 Dubbo。推荐一下我写的总结：

- [RPC 基础知识总结](#)
- [Dubbo 常见问题总结](#)

另外，Dubbo 官方文档是一定要看的，地址：<https://cn.dubbo.apache.org/zh-cn/overview/home/>。

服务注册与发现

Eureka、Zookeeper、Consul、Nacos 都可以提供服务注册与发现的功能。

个人比较建议学习 Nacos，国内用的比较多，功能也更强大！除了提供服务注册与发现功能之外，还可以作为配置中心使用。

	Nacos	Eureka	Consul	Zookeeper
一致性协议	CP+AP	AP	CP	CP
健康检查	TCP/HTTP/MYSQL/Client Beat	Client Beat	TCP/HTTP/gRPC/Cmd	Keep Alive
负载均衡策略	权重/ metadata/Selector	Ribbon	Fabio	—
雪崩保护	有	有	无	无
自动注销实例	支持	支持	不支持	支持
访问协议	HTTP/DNS	HTTP	HTTP/DNS	TCP
监听支持	支持	支持	支持	支持
多数据中心	支持	支持	支持	不支持
跨注册中心同步	支持	不支持	支持	不支持
SpringCloud集成	支持	支持	支持	不支持
Dubbo集成	支持	不支持	不支持	支持
K8S集成	支持	不支持	支持	不支持

学习 Nacos 的话，官方文档是一定要看的：<https://nacos.io/zh-cn/docs/v2/quickstart/quick-start.html>。

另外，再推荐一些我觉得还不错的学习资料：

- [Nacos 架构&原理 - 阿里藏经阁](#)（推荐，像 Nacos 内核设计、底层原理、最佳实践）
- [55 张图吃透 Nacos - 不才陈某](#)
- [图文解析 Nacos 配置中心的实现 - 掘金](#)（没有过多代码粘贴，原理讲的很清楚）
- [Nacos 帮我们解决什么问题？—— 配置管理篇 - 阿里巴巴中间件](#)

API 网关

网关可以为我们提供请求转发、安全认证（身份/权限认证）、流量控制、负载均衡、降级熔断、日志、监控、参数校验、协议转换等功能。

关于 API 网关的基础知识和技术选型推荐阅读我写的 [API 网关基础知识总结](#) 这篇文章。

Spring Cloud 微服务项目比较推荐实用 SpringCloud Gateway 作为 API 网关，这是 Spring Cloud 的一个全新项目，为了取代 Netflix Zuul。为了提升网关的性能，SpringCloud Gateway 是基于 WebFlux 实现。Spring Cloud Gateway 的目标是不仅提供统一的路由方式，并且基于 Filter 链的方式提供了网关基本的功能，例如：安全，监控/指标，和限流。

下面这些是我觉得还不错的学习资料：

- [Spring Cloud Gateway 常见问题总结 - JavaGuide](#)
- [6000 字 | 16 图 | 深入理解 Spring Cloud Gateway 的原理 - 悟空聊架构](#)
- [Spring Cloud Gateway 夺命连环 10 问? - 不才陈某](#)
- [Spring Cloud Gateway 整合阿里 Sentinel 网关限流实战! - 不才陈某](#)
- [实战 Spring Cloud Gateway 之限流篇 - aneasystone](#)（对于常见的限流算法和组件都有介绍到）

配置中心

微服务下，业务的发展一般会导致服务数量的增加，进而导致程序配置（服务地址、数据库参数等等）增多。

传统的配置文件的方式已经无法满足当前需求，主要有两点原因：一是安全性得不到保障（配置放在代码库中容易泄露）；二是时效性不行（修改配置需要重启服务才能生效）。

Spring Cloud Config、Nacos 、Apollo、K8s ConfigMap 都可以用来做配置中心。

Apollo 和 Nacos 我个人更喜欢。Nacos 使用起来更加顺手，还能顺便作为注册中心使用，Apollo 在配置管理方面做的更加全面。

个人还是更建议学习 Nacos ,学习资料在上面的服务注册与发现已经推荐过了。

分布式 ID

ID 是数据的唯一标识，分布式 ID 是分布式系统下的 ID。

分布式 ID 的解决方案有很多，比如：

- 算法：UUID、Snowflake(雪花算法)

- 开源框架： UidGenerator(百度)、Leaf(美团)、Tinyid(滴滴)、IdGenerator(个人)

这块内容比较简单，推荐阅读下面这两篇文章进行学习：

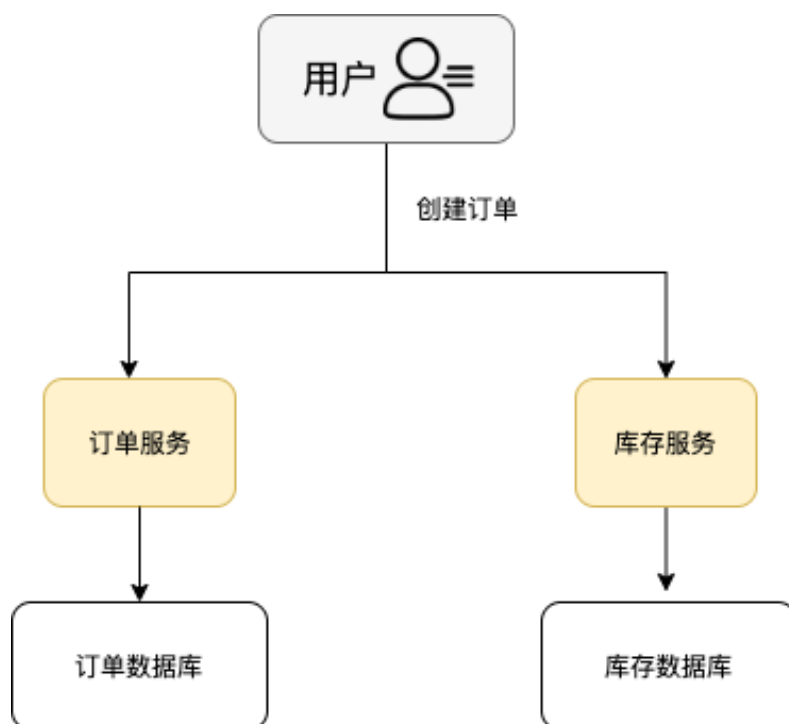
- [分布式 ID 介绍&实现方案总结](#)
- [分布式 ID 设计指南](#)

分布式事务

微服务架构下，一个系统被拆分为多个小的微服务。

每个微服务都可能存在不同的机器上，并且每个微服务可能都有一个单独的数据库供自己使用。这种情况下，一组操作可能会涉及到多个微服务以及多个数据库。

举个例子：电商系统中，你创建一个订单往往会涉及到订单服务（订单数加一）、库存服务（库存减一）等等服务，这些服务会有供自己单独使用的数据库。



那么如何保证这一组操作要么都执行成功，要么都执行失败呢？

这个时候单单依靠数据库事务就不行了！我们就需要引入 **分布式事务** 这个概念了！

常用分布式事务解决方案有 Seata 和 Hmily。

1. **Seata** :Seata 是一款开源的分布式事务解决方案，致力于在微服务架构下提供高性能和简单易用的分布式事务服务。
2. **Hmily** : 金融级分布式事务解决方案。

目前国内用的比较多的是 Seata，建议学习这个。

分布式链路追踪

不同于单体架构，在分布式架构下，请求需要在多个服务之间调用，排查问题会非常麻烦。我们需要分布式链路追踪系统来解决这个痛点。

目前分布式链路追踪系统基本都是根据谷歌的《Dapper 大规模分布式系统的跟踪系统》这篇论文发展而来，主流的有 Pinpoint，Skywalking，CAT（当然也有其他的例如 Zipkin，Jaeger 等产品，不过总体来说不如前面选取的 3 个完成度高）等。

Zipkin 是 Twitter 公司开源的一个分布式链路追踪工具，Spring Cloud Sleuth 实际是基于 Zipkin 的。

SkyWalking 是国人吴晟（华为）开源的一款分布式追踪，分析，告警的工具，现在是 Apache 旗下开源项目。

目前国内用的比较多的是 SkyWalking，建议学习这个。

高性能（进阶）

CDN（掌握概念和原理即可）

CDN 就是将静态资源分发到多个不同的地方以实现就近访问，进而加快静态资源的访问速度，减轻服务器以及带宽的负担。

我们只需要掌握 CDN 的基本概念和原理以及会用云厂商提供的现成 CDN 服务即可，花费不了太多时间。推荐阅读我写的[CDN 常见问题总结](#)这篇文章。

消息队列

消息队列在分布式系统中主要是为了异步、解耦和削峰。

常用的消息队列如下：

1. **RocketMQ**：阿里巴巴开源的一款高性能、高吞吐量的分布式消息中间件。
2. **Kafka**: Kafka 是一种分布式的，基于发布 / 订阅的消息系统。
3. **RabbitMQ** :基于 erlang 开发的基于 AMQP（Advanced Message Queue 高级消息队列协议）协议实现的消息队列。
4. **Pulsar**：下一代云原生分布式消息流平台。

建议选择 RocketMQ 和 Kafka 其中的一个进行深入学习，其他消息队列了解即可。

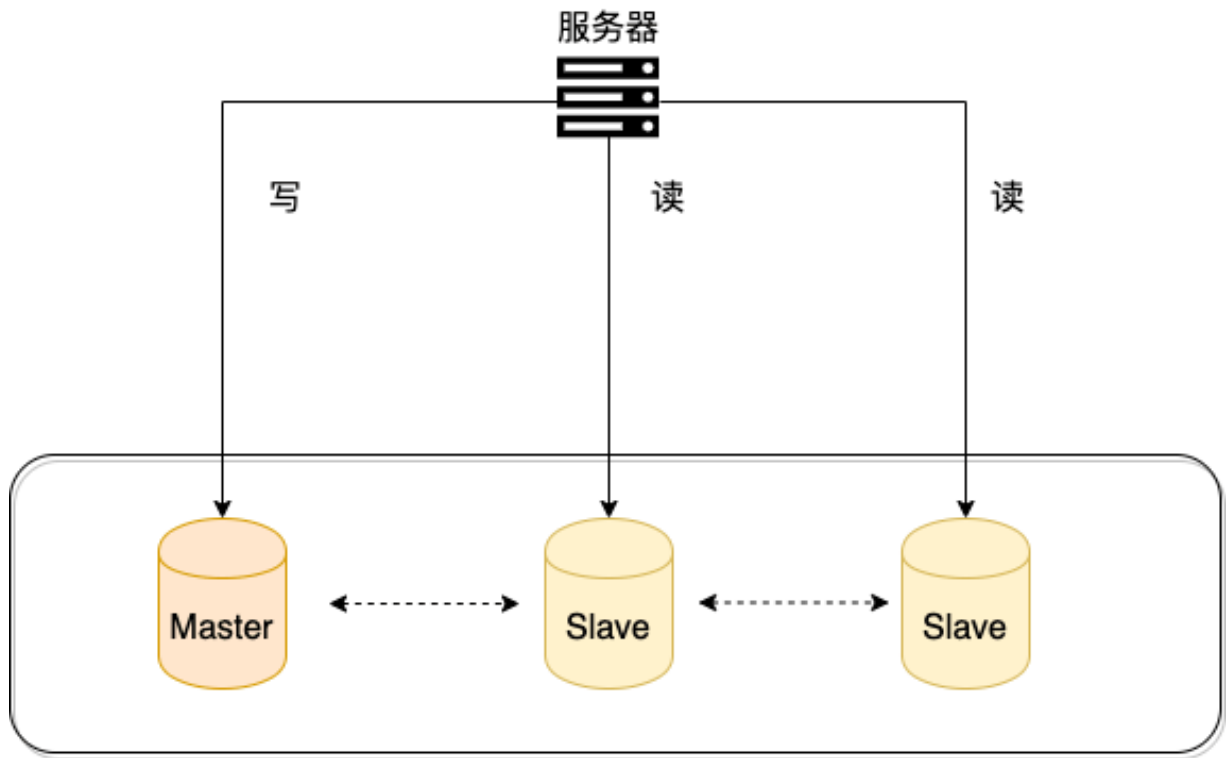
关于消息队列基础概念、技术选型方面的介绍，建议阅读我写的[消息队列基础知识总结](#)这篇文章。

Kafka、RocketMQ、RabbitMQ 学习资源推荐请看[知识星球](#)的这篇帖子：<https://t.zsxq.com/0bEDFwgon>。

读写分离&分库分表（掌握概念和原理即可）

读写分离主要是为了将数据库的读和写操作分不到不同的数据库节点上。主服务器负责写，从服务器负责读。另外，一主一从或者一主多从都可以。

读写分离可以大幅提高读性能，小幅提高写的性能。因此，读写分离更适合单机并发读请求比较多的场景。



分库分表是为了解决由于库、表数据量过大，而导致数据库性能持续下降的问题。

常见的分库分表工具有：sharding-jdbc（当当）、TSharding（蘑菇街）、MyCAT（基于Cobar）、Cobar（阿里巴巴）...。推荐使用 sharding-jdbc。因为，sharding-jdbc 是一款轻量级 Java 框架，以 jar 包形式提供服务，不要我们做额外的运维工作，并且兼容性也很好。

				垂直拆分	
				nickname	description
水平拆分	ID	name	age	sex	
	1				
	2				
					
	999999				
	1000000				
	1000001				
					
	9999999				

现在很多公司都是用的类似于 TiDB 这种分布式关系型数据库，不需要我们手动进行分库分表，因此我们只需要掌握读写分离&分库分表的常见概念和原理即可，不需要花费太多时间去实践，推荐阅读我写的 [读写分离&分库分表常见问题总结](#) 这篇文章。

负载均衡

负载均衡系统通常用于将任务比如用户请求处理分配到多个服务器处理以提高网站、应用或者数据库的性能和可靠性。

开发过程中，我们接触到的负载均衡可以简单分为 **服务端负载均衡** 和 **客户端负载均衡** 这两种。服务端负载均衡可以通过硬件（比如 F5、A10、Array）或者软件（比如 LVS、Nginx、HAProxy）实现。Java 领域主流的微服务框架 Dubbo、Spring Cloud 等都内置了开箱即用的客户端负载均衡实现。Dubbo 属于是默认自带了负载均衡功能，Spring Cloud 是通过组件的形式实现的负载均衡，属于可选项，比较常用的是 Spring Cloud Load Balancer（官方，推荐）和 Ribbon（Netflix，已被启用）。

个人建议学习一下 Nginx 和 Spring Cloud Load Balancer。

负载均衡的常见概念、算法和技术方案可以看看这篇文章：[负载均衡常见问题总结](#)。

高可用（进阶）

高可用描述的是一个系统在大部分时间都是可用的，可以为我们提供服务的。高可用代表系统即使在发生硬件故障或者系统升级的时候，服务仍然是可用的。

限流&降级&熔断

限流是从用户访问压力的角度来考虑如何应对系统故障。限流为了对服务端的接口接受请求的频率进行限制，防止服务挂掉。比如某一接口的请求限制为 100 个每秒，对超过限制的请求放弃处理或者放到队列中等待处理。限流可以有效应对突发请求过多。

关于服务限流的介绍推荐阅读我写的[服务限流详解](#)这篇文章，里面有介绍常见的限流算法以及单机限流和分布式限流的技术方案。

降级是从系统功能优先级的角度考虑如何应对系统故障。服务降级指的是当服务器压力剧增的情况下，根据当前业务情况及流量对一些服务和页面有策略的降级，以此释放服务器资源以保证核心任务的正常运行。

熔断和降级是两个比较容易混淆的概念，两者的含义并不相同。降级的目的在于应对系统自身的故障，而熔断的目的在于应对当前系统依赖的外部系统或者第三方系统的故障。

Netflix 开源的 [Hystrix](#) 和阿里开源的 [Sentinel](#) 都能实现限流、降级、熔断。不过，Hystrix 已经停止维护了，更建议使用功能更为强大的 Sentinel。另外，Sentinel 的 Wiki 中对比了常用限流降级组件，感兴趣的可以看看，传送门：[常用限流降级组件对比](#)。

[Sentinel 的 wiki](#) 中已经详细描述了其与 Hystrix 的区别，你可以看看。

学习 Sentinel 的话，官方文档是一定要看的：<https://sentinelguard.io/zh-cn/docs/introduction.html>。

另外，再推荐一些我觉得还不错的学习资料：

- [阿里限流神器 Sentinel 夺命连环 17 问？ - 不才陈某](#)
- [Sentinel 为什么这么强，我扒了扒背后的实现原理 - 三友的 java 日志](#)
- [Sentinel 流控滑动窗口算法设计 - 老周聊架构](#)

排队

另类的一种限流，类比于现实世界的排队。玩过英雄联盟的小伙伴应该有体会，每次一有活动，就要经历一波排队才能进入游戏。

实现排队的方法有很多种，比如我们可以借助消息队列、JDK 中的各种阻塞队列。

集群

相同的服务部署多份，避免单点故障。

超时和重试机制

一旦用户的请求超过某个时间得不到响应就结束此次请求并抛出异常。如果不进行超时设置可能会导致请求响应速度慢，甚至导致请求堆积进而让系统无法在处理请求。

另外，重试的次数一般设为 3 次，再多次的重试没有好处，反而会加重服务器压力（部分场景使用失败重试机制会不太适合）。

云原生（可选）

提示：云原生开发对能力要求很高，Java 后端岗位通常也不会要求云原生开发技能。因此，这部分内容不推荐对云原生开发不感兴趣或者不了解的同学学习，可以选择跳过。

云原生就是在云中构建、运行应用程序的一套完整的技术体系和方法论。这里的技术体系和方法论就目前来说指的是 微服务+DevOps+持续交付+容器化。

越来越多的编程语言、框架开始拥抱云原生，例如 Spring 推出了面向云原生的技术 Spring Native、RedHat 开源了 Java 云原生服务框架 Quarkus。

如果你对云原生领域比较感兴趣的话，建议你重点关注下面这些技术：

1. 微服务：SpringCloud 或者 SpringCloud Alibaba 其实是不用学习的，云原生下一般基于后面提到的 Kubernetes 来构建微服务。
2. 网关：网关是整个微服务架构的流量入口，负责认证授权、请求分发、认证授权、限流、API 管理、负载均衡等工作，是微服务架构中非常重要的一个组件。因此，我这里专门单独将网关拿出来提一嘴。
3. 容器：容器技术是云原生发展的基石，以 Docker 为首的容器工具提出了“一次构建，到处运行”的口号。
4. Kubernetes：K8s 被称为云原生时代的操作系统，云原生应用的优势与其提供的功能息息相关。
5. DevOps：DevOps 关注的是如何实现应用程序的全生命周期（开发，测试，运维）自动化管理，从而实现更快速、更高质量、更频繁、更稳定的软件交付。DevOps 团队通常会使用微服务架构来构建应用程序，借助于持续集成和持续部署（CI/CD）来实施 DevOps。

6. ServiceMesh：你可以将 Service Mesh 看作是为了简化开发工作专门抽象出来的一层，通常作为透明的一层接入到现有的分布式应用程序里。
7.

其中，比较重要的是 Kubernetes。如果你做项目的话，建议优先考虑 Kubernetes 相关的项目。

我之前写过一篇文章来介绍云原生，可以看看：[云原生时代，程序员应该掌握哪些能力？](#)。

总结

这是一份非常详细的学习路线，把上面的内容学完之后，找到一份比较好的工作已经比较容易。

另外，我在上面也说了，如果你觉得内容比较多自己学不完或者如果你只想找到一份小厂的开发工作的话，建议你把重心放在 Java 基础、数据库、常用框架、常用工具上。

像 JVM、分布式、高并发、高可用、微服务这些知识点，如果你想进大厂或者说让自己在求职的时候更有竞争力，那你就也是要多花一点时间来学习的。

现在面试很卷，想要找到一个好工作的话，就需要你去多学一点，多练习一点。虽然，你目前学的很多知识，在你工作之后可能用不到，但是，面试的筛选就需要你会这些。毕竟，很多岗位是很多人一起竞争，为了达到筛选效果，面试难度通常都会比较大的。这也就是所谓的：“面试造火箭，入职拧螺丝”。

公众号(推荐)

学习路线的最新更新会第一时间同步在公众号，推荐大家关注一波！

JavaGuide官方公众号

(微信搜索JavaGuide)



- 1、公众号后台回复“PDF”获取原创PDF面试手册
- 2、公众号后台回复“学习路线”获取Java学习路线最新版
- 3、公众号后台回复“开源”获取优质Java开源项目合集
- 4、公众号后台回复“八股文”获取Java面试真题+面经

知识星球

为了帮助更多同学准备 Java 面试以及学习 Java ， 我创建了一个纯粹的Java 面试知识星球。虽然收费只有培训班/训练营的百分之一， 但是知识星球里的内容质量更高， 提供的服务也更全面， 非常适合准备 Java 面试和学习 Java 的同学。

欢迎准备 Java 面试以及学习 Java 的同学加入我的 知识星球， 干货非常多， 学习氛围也很不错！ 收费虽然是白菜价， 但星球里的内容或许比你参加上万的培训班质量还要高。

我的知识星球能为你提供什么？

努力做一个最优质的 Java 面试交流星球！加入到我的星球之后，你将获得：

1. 6 个高质量的专栏永久阅读，内容涵盖面试，源码解析，项目实战等内容！价值远超门票！
2. 多本原创 PDF 版本面试手册。
3. 免费的简历修改服务（已经累计帮助 4000+ 位球友修改简历）。
4. 一对一免费提问交流（专属建议，走心回答）。
5. 专属求职指南和建议，帮助你逆袭大厂！
6. 海量 Java 优质面试资源分享！价值远超门票！
7. 读书交流，学习交流，让我们共同努力创建一个纯粹的学习交流社区。
8. 不定期福利：节日抽奖、送书送课、球友线下聚会等等。
9.

其中的任何一项服务单独拎出来价值都远超星球门票了。