



链滴

分享一些 Java 无关基础方面的书籍

作者: [ideal](#)

原文链接: <https://ld246.com/article/1504015484751>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<h2 id="个人认为看书有两个点好处- ">个人认为看书有两个点好处： </h2>

<p>1.能出版出来的书一定是经过反复思考，雕琢和审核的，因此从专业性的角度来说，一本好书的值超其他资料</p>

<p>2.对着书上的代码自己敲的时候方便</p>

<p>“看完书之后再次提升自我的最好途径是看一些相关的好博文”，我个人认为这是学习的第二步因为一本书往往有好几百页，好的博文是自己看书学习之后的一些总结和提炼，对于梳理学习的内容多好处，当然这些不是自己的学习方法，就不在扯下去了。</p>

<p>很多程序员们往往有看书的冲动，但不知道看哪些书，下面我就给各位的 Java 的程序员们推荐些好书（每本书都有加粗），其中我绝大多数都是我自己平时再看的书，对于我平时都的书做一个次总结和读后感吧。</p>

<blockquote>

<p>首先推荐的不是一本书，而是一个博客，也是我们博客园另外一位博友 java my li e。</p>

</blockquote>

<p>目前市面上讲解设计模式的书很多，虽然我前面讲了看书是最好的，但是对设计模式感兴趣的朋友们，我推荐的是这个博客。这位博友的设计模式讲得非常非常好，我认为 90% 的内容都是没有问题很值得学习的，其讲解设计模式的大体路线是：</p>

<p>随便开篇点明该设计模式的定义</p>

<p>图文并茂讲解该设计模式中的结构</p>

<p>以详细的代码形式写一下该种设计模式的实现</p>

<p>补充内容</p>

<p>讲解该设计模式的优缺点</p>

<p>对于一个设计模式我们关注、学习的知识点，不就是上面这些吗？</p>

<p>不过我要重点提醒一下网友们，同一种设计模式的写法有多种，并不是说只有按某种写法来写才这种设计模式。比方说适配器模式，我们关注适配器模式一定要关注的是什么是适配器模式不是怎么适配器模式，不要认为某段代码不是按照适配器模式的写法写下来的它就不是适配器模式了，记住这点，你在学习设计模式的时候一定会对代码中用到的设计模式有更深入的理解。</p>

<h3 id="---深入理解-Java-虚拟机-JVM-高级特性与最佳实践- ">一、《深入理解 Java 虚拟机： JV 高级特性与最佳实践》</h3>

<p>如果你不满足于做一个只会写 if...else... 的 Java 程序员，而是希望更进一步，我随便举几个例子：</p>

<p>1、了解 Java 代码的底层运行机制</p>

<p>2、定位性能问题</p>

<p>3、对整个系统进行性能调优</p>

<p>4、解决各种奇奇怪怪的线上线下问题</p>

<p>5、更加高级别的，为自己的项目量身定做一款适合自己项目的虚拟机</p>

<p>那么 Java 虚拟机是你必学的一门技术。《深入理解 Java 虚拟机： JVM 高级特性与最佳实践》者是周志明，这本书可以说是国内写得最好的有关 Java 虚拟机的书籍，近半年，前前后后这本书我码看了有 5 遍。国内写虚拟机的书除了这本，其实还有一些其他的，我也买过，不过粗略看下来，很内容也是《深入理解 Java 虚拟机： JVM 高级特性与最佳实践》此书里面的。</p>

<p>另外值得一提的是，《深入理解 Java 虚拟机： JVM 高级特性与最佳实践》这本书，有电子版，网上搜一下就能下载到了。不过建议有兴趣的朋友还是去买书看，电子版本下载到的一般是比较老

版本，相比最新修订版的《深入理解 Java 虚拟机：JVM 高级特性与最佳实践》，有很多作者新补充知识点是没有的。

二、《HotSpot 实战》

所有的 Java 虚拟机都是遵循着 Java 虚拟机规范来的，市面上的 Java 虚拟机几十款，《深入理解 Java 虚拟机：JVM 高级特性与最佳实践》一书里面讲的虚拟机并不针对某种特定的虚拟机，而是从 Java 虚拟机规范的角度来讲解 Java 虚拟机。

我们平时使用的乃至商用的大多数 Java 虚拟机都是 Sun 公司的 HotSpot，大家 cmd 进入命令，使用“java -version”命令就可以看到了。如果希望在 Java 虚拟机规范的基础上更加深入地去理解虚拟机的一些细节是怎么实现的，就可以看一下《HotSpot 实战》一书，作者是陈涛。不过由于 HotSpot 的源码都是 C/C++ 写的，所以要求读者有非常好的 C/C++ 基础，如果对这两门语言不是很熟的朋友，看这本书可能对你帮助不是很大。

最后提一句，如果有兴趣的朋友，不妨先去网上下载一个 openJDK，HotSpot 的源码就在里面。

三、《Java 并发编程实战》

这本书常常被列入 Java 程序员必读十大书籍排行榜前几位，不过个人不是很推荐这本书。

《Java 并发编程实战》作者是 Brian Goetz，怎么说呢，这本书前前后后我也看了两左右，个人感受是：

1、文字多代码少
2、讲解多实践少

我觉得这可能就是老外写书的特点吧，因为 Java 是北美国家（加拿大、美国）开发和维护的，以老对外对 Java 方方面面的理论知识体系都掌握得是非常清楚和透彻的。翻开这本书看，多线程什么、什么是死锁、什么是竞争、什么是线程安全等等，方方面面的知识点都用大量的文字篇幅讲解，不让人感觉十分枯燥，也难让读者有实质性的进步。我这本书看了两遍也属于一目十行意思，有兴趣的朋友就重点看一下。

无论如何，作为一本常常位于 Java 程序员必读十大书籍排行榜前几名的书，还是一定要推荐给大家的。

四、《Java 多线程编程核心技术》

《Java 多线程编程核心技术》作者高洪岩。想要学习多线程的朋友，这本书是我大力推荐的，个人博客里面二十多篇的多线程博文都是基于此书，并且在这本书的基础上进行提炼和总结而写出的。

此书和《Java 并发编程实战》相反，这本书的特点是大篇幅的代码 + 小篇幅的精讲解，可能这是中国人写的书比较偏向实用主义的风格有关。本书关于线程安全、synchronized、Reentrant、Timer 等等都用详细的代码进行了讲解，而且每个大知识点下的多个小知识点都会详细讲解到，非常有实践值。

有兴趣的朋友们，我相信只要你们跟着这本书里面的代码敲、运行、思考，三步走，对于多线程使用与理解一定会进几大步。

不过这本书的缺点就是对于 Java 并发包下的一些类像 CountDownLatch、Semaphore、CyclicBarrier、Future、Callable 等都没有讲到，重点的 CAS 和 AQS 也没有触及，重点类的实现原理也没提。当然，这很深入了，在学习了这本书之后如果能再去对这些知识进行一些学习、研究的话，你一定会慢慢成长为一个很厉害的多线程高手。

五、《Effective Java 中文版》

这是唯一一本我没有买的书。初识这本书，是在我的博文 Java 代码优化（长期更新）里面，底评论的时候有朋友提到了这本书，当时我说要去买，不过这两个月一直都没时间去逛书店，甚是遗憾之后肯定会找时间去买这本书的。

《Effective Java 中文版》的作者是 Joshua Bloch，这个人就很厉害了，他是谷歌的首席架构师属于超级技术大牛级别了吧，呵呵。由于没有看过这本书，所以我不好发表评论，但是从这本书的知名度以及其作者的来头来看（多提一句，这本书也是 Java 之父 James Gosling 博士推崇的一本书），相信这一定是一本值得一看的好书。

好的代码是每个 Java 程序员都应该去追求的，不是说我今天写一段好代码相比写一段烂代码对可能会有多大的提升，更多的应该是提升了代码的可读性以及可以规避许多潜在的、未知的问题，避免

码上线之后出问题而花时间去维护——无论从时间成本、人力成本还是风险成本来说，这都是非常高的。

</p>

<h3 id="六--深入分析-Java-Web-技术内幕->>六、《深入分析 Java Web 技术内幕》</h3>

<p></p>

<p>《深入分析 Java Web 技术内幕》，作者许令波，淘宝工程师。</p>

<p>这本书我用一个字概括就是：全。真的非常全，HTTP、DNS、CDN、静态化、Jetty、Tomcat、Servlet、Spring、MyBatis 等等，什么都有，涉及知识面非常广，但又不像专门精讲某个知识点的书籍一样讲得非常深入，感觉这本书就是尽量去用短的篇幅讲清楚一些 Java Web 使用到的技术的内幕，让读者对这些知识点的技术内幕有一个理性的认识。</p>

<p>不过，尽管每个知识点的篇幅都不多，但是重点都基本讲到了，是一本让人真正有收获的书。如想进一步了解这些技术的技术内幕，就要自己去买相关书籍或者自己上网查资料了，有种抛砖引玉，或者说师傅领进门、修行在个人的感觉。</p>

<h3 id="七--大型网站技术架构-核心原理与案例分析->>七、《大型网站技术架构 核心原理与案例分析》</h3>

<p></p>

<p>一个字评价这本书，屌；两个字评价这本书，很屌；三个字评价这本书，非常屌。呵呵，好了，说下去可能别人以为我是水军了。</p>

<p>《大型网站技术架构 核心原理与案例分析》的作者是李智慧，原阿里巴巴技术专家。</p>

<p>Java 的大多数应用都是用在 Web 上的，现在只要稍微大型一点的 Web 应用，都一定是一个分布式系统，那么一个分布式系统用到了哪些技术？一个大型网站是如何从一个小型网站成长起来的？如何保证你的网站安全？分布式系统使用到了缓存，有哪些缓存？缓存的使用有哪些值得注意的事项？</p>

<p>关于分布式的知识点，都在这本书里面有体现，只有你想不到，没有他写不到，而且写得非常易懂，基本属于看一两遍，再记一些笔记就知道是怎么一回事儿了。多看几遍，对分布式的理解一定会加不少。而且里面不仅仅是分布式的知识，还非常接地气地写了如何做一个好的架构师，其实我认为这仅仅是写给想做架构师的读者看的，就是给读者一些建议，如何更好地提出意见、如何更让别人关注的声音、如何看到他人的优点，入木三分，让人获益匪浅。</p>

<h3 id="八--大型网站系统与-Java-中间件实践->>八、《大型网站系统与 Java 中间件实践》</h3>

<p></p>

<p>《大型网站系统与 Java 中间件实践》作者曾宪杰，是淘宝的技术总监，算起来应该在阿里有至少 P8 的级别了吧。</p>

<p>这本书的部分内容和上面一本李智慧的《大型网站技术架构 核心原理与案例分析》有所重合，分布式系统的演化、CDN、CAP 理论和 BASE 理论等等，这也更说明这些都是分布式系统或者说是个大型网站重点关注的内容，当作一次再学习也不错。</p>

<p>本书要突出的重点是中间件三个字，中间件是分布式系统中一个非常重要的东西，其最重要的作用应该就是解耦，降低模块与模块之间的强依赖，不同的模块之间的依赖度降低，便可以各自独立地开发自己的功能，这也可以说是软件工程发展的目标和驱动力。</p>

<p>因此，本书有一部分的内容就是基于中间件，详细讲解了中间件与 JMS 的各种知识，适合对分布式系统比较熟悉并且想要往中间件方面有一定研究的读者。</p>

<h3 id="九--从-Paxos-到-ZooKeeper-分布式一致性原理与实践->>九、《从 Paxos 到 ZooKeeper 分布式一致性原理与实践》</h3>

<p></p>

<p>《从 Paxos 到 ZooKeeper 分布式一致性原理与实践》，作者倪超，阿里巴巴工程师。</p>

<p>这本书是我最近在研读的一本书，和上面的《大型网站系统与 Java 中间件实践》一样，属于分布式组件的范畴，属于有些深入的内容，当然也是我自己的个人兴趣。当然，如果有志向做一个出色的大型网站架构师、公司的技术总监之类，这些知识当然是必须掌握的。</p>

<p>本书从分布式系统基本理论开始讲起，讲到 Paxos 算法，最后慢慢引入到 Zookeeper，循序渐进。当然，更多的我目前还不方便发表什么看法，因为这本书的第二张 Paxos 算法我都还没有弄懂（Paxos 算法确实有些难以理解和不太易懂），接下来的章节还没有看下去。</p>

<p>如果网友们所在的公司在使用 Zookeeper，并且你又对 Zookeeper 感兴趣想要研究一下它的原理，这本书将是不二之选。</p>

<h3 id="十--MySQL5-6-从零开始学->>十、《MySQL5.6 从零开始学》</h3>

<p></p>

<p>《MySQL5.6 从零开始学》，作者刘增杰和李坤。</p>

<p>作为一名 Java 程序员，我认为我们千万不要觉得数据库是 DBA 的事情，数据库对一个 Java 程员来说也是必须掌握的一门知识，丰富的数据库性能优化经验是一个顶尖程序员必备技能。</p>

<p>目前主流的数据库有 Oracle 和 MySQL，当然推荐大家的是 MySQL，主要原因我认为有两点：</p>

<p>1、MySQL 相比 Oracle 更轻量级、更小、安装和卸载更方便，SQL 其实都是差不多的，如果想数据库，学 MySQL 就可以了，在家里面可以自己方便地研究，如果你的公司使用 Oracle，只要再对比学习法，关注一下 Oracle 和 MySQL 的差别即可</p>

<p>2、随着 2009 年阿里巴巴去 IOE 的运动的进行，目前国内的很多互联网公司都会选择 MySQL 为它们使用的数据库，因为 MySQL 免费，所以既省钱又不需要出了问题就依赖甲骨文公司</p>

<p>MySQL 学习我推荐的是这本我自己学习看的《MySQL5.6 从零开始学》，我是觉得挺好的这本，书里面的知识点很细致、很全面，读者选择书籍的标准大多不就是这两点吗？</p>

十一、《Spring 源码深度解析》

<p></p>

<p>Spring 这个框架做得太好了，功能太强大了，以至于很多开发者都只知 Spring，不知什么是工、什么是单例、什么是代理（我面试别人的真实体会）。这种功能强大的框架内部一定是很复杂的实，这就导致一旦你的程序使用 Spring，出了问题，可能是 Error、可能是 Exception、可能是程序运结果不是你的预期的，出现诸如此类问题的时候，将会让你感到困惑，除了上网查资料或者问别人似没有更好的解决办法。</p>

<p>研读 Spring 的源代码不失为一种很好的学习方法，我个人认为这有很多好处：</p>

<p>1、理解框架内部的实现之后，可以主动去解决问题，而不需要依赖别人</p>

<p>2、Spring 框架内部实现用到了很多设计模式，很好的代码设计思路，这将会对你写代码、对你解设计模式有很大的提高</p>

<p>3、研究 Spring 框架将会大大增强你读代码的能力，我相信只要你能研究清楚 Spring 内部是如何实现的，其他任何一个框架的源代码都难不倒你</p>

<p>总而言之，我认为读代码的能力是一个普通的程序员和一个好的程序员之间最大的差别之一，前者只会把别人写好的东西拿来用，后者不仅能用好，还清楚知道别人写好的东西底层是如何实现的，在现问题的时候可以轻松解决。</p>

<p>Spring 源代码，个人推荐《Spring 源码深度解析》一书，真要研究透并且写清楚 Spring 源代码，恐怕三四本书都不够，作者在近 400 页的篇幅中尽量去讲解 Spring 源代码是如何实现的，殊为不易，尽管无法讲得完全，但是相信作者的讲解配合上读者自己的研究，一定可以对 Spring 的实现有更深入的理解。</p>

<p>后记</p>

<p>以上就是我推荐给 Java 开发者们的一些值得一看的好书。但是这些书里面并没有 Java 基础、Java 教程之类的书，不是我不推荐，而是离我自己学习 Java 基础技术也过去好几年了，我学习的时候的什么也忘了，所以我不能不负责任地推荐一些我自己都没有看过的书给大家。

非原创</p>