

分享下多年积累的对 JAVA 程序员成长之路的总结

我也搞了几年 JAVA 了，由于一向懒惰，没有成为大牛，只是一普通程序猿，不爱玩社交网站，不爱玩微博，唯独喜欢百度贴吧，潜水很久了，手痒来给新人分享下从新手成长为老鸟的已见，也刷刷存在感，应该不比曝照差吧。

首先初识语法的阶段，必须要学会怎么操作对象，操作 if 和 for，操作 list set map，然后是线程、IO 和 jdbc 什么的，其余的，若是一时不理解，可以后边需要时再学。

这阶段完了，你可以写些能在控制台打印出来的小程序，锻炼下逻辑思维。也就是号称 JAVASE 毕业了，其实不过是入门而已，如果要往 WEB 方向发展，这些倒是基本足够了。

接下来要学 HTML JSP SERVLET 数据库 JAVASCRIPT TOMCAT, 目标, 写出第一个动态网站，也许只是个登陆功能，只能展示下个人资料，但这是很重要的一步，你要弄清楚的是，一个用户的点击产生的请求，是从哪里发起，哪里接收，哪里处理，哪里返回，你得理解浏览器和服务器的关系和分工，cookie 和 session，request 和 response。这个是个 WEB 开发的学习初级阶段，这都是些 JAVA 诞生以来最原始的最官方的 WEB 开发技术，当然现在真正的项目很少直接采用这些技术了，为了不断提高技术的易用性、开发效率和可维护性、可扩展性，无数开源项目都是在这些原始技术的基础上封装、改进。所以这个阶段不要盲目乐观地跑去找工作或者对实习挑三拣四，你会被打击到的。

好，初始阶段完成，开始进入 WEB 开发的正题，首先是传说中的三大框架，struts hibernate spring 成为熟练的增删改查程序员是必须的，在这个阶段你还要理解为什么要用这仨而不是那些 JSP SERVLET JDBC，你要体会到写 MVC、三层架构的好处。这个阶段不要轻易质疑三大框架的价值，也许刚开始你会觉得麻烦，觉得他们有时候是多此一举，一两年后回过头来你会觉得少了这些框架你已经很难干活了。这个阶段还要避免的心态是能熟练地增删改查了，就认为写程序不过如此，然后往简历上填个精通，这也是新手面试被批得体无完肤的原因之一。如果你学得好，这会应该能熟练地写个博客啊，小论坛啊之类的 WEB 项目了，也就是达到了就业的基本要求。

接下来就没有固定模式了，你的第一家企业对你的影响会很大，我见识过烂企业是如何毁新人的，用着设计恶劣的架构，老掉牙的技术，不断地耽误新人的成长，有的员工在这种企业一呆数年，思维已经完全被错误观点定型，深陷于公司业务和老系统中疲于奔命，看似能解决问题，却无法给公司带来真正的进步，就像一个用了数年砍刀的人，就算再熟练，刀法再好，也打不过用枪的。如果离开了这个公司，恐怕很难在这个行业再立足。记得有个说法很好，一个人用 10 年的时间重复做头一两年做的事，你的真实工作经验其实还是一两年。烂公司的老员工就是如此。在这头一年里，定个标准吧，你应该能做到的有：

- 1、把一些常用的琐碎技术学会了，比如 xml 的作用和实际运用，如何操作文件，如何运用日志，如何简化封装常用的工具类，比如日期操作、编码处理、分页模块等等，这些小东西会占用你大量的时间去处理摸索。
- 2、把数据库用好。数据库不是会增删改查就可以了的，你要关注事务设置是否合理，是否有可能产生脏数据，是否要动用乐观锁悲观锁等等，重要数据出了错，无论是技术上还是业务上都是很麻烦的事。
- 3、懂得什么是集群，什么是分布式，懂得怎么向别的系统发请求，交换数据，webservice。

4、如果你也写页面，你要会用 JAVASCRIPT，不一定要很深入，你可以用 JQUERY 代替，做到无刷新的页面操作以及 AJAX 请求。

第一年能达到这个程度，你绝对已经在平均水准之上了。在我达到这个水平时，我有种似乎一切网站开发都不在话下的感觉，当然后来我很快就发现了我还差得太远太远....这个阶段想面个高薪的工作还是很难的，只能说达到温饱线了，不用为柴米油盐发愁。

下一个阶段，你应该考虑你的代码质量了。

1、多线程和异步、并发的概念会忽然变得很重要，因为在一个对性能有要求的系统里，是不会允许你把各种琐事都排着队一件件去办的，你不能老写些 a 方法调 b 方法再调 c 方法再调 d 方法然后返回的代码。你开始想起有队列这种东西，有 runnable 这种东西，你开始觉得自己的代码高端起来了，和当年的增删改查程序员有天壤之别。但同时麻烦也来了，你开始遇到线程安全问题，尤其是当你的程序跑在一个多机环境而不是单机环境里的时候，你会对这些很头痛。对于这些问题，Jms 技术是个很有代表性的东西，当然并不像 ssh 那么常用。

2、你会发现数据库性能实在不行，出于不甘现状，或者赶时髦，你该去了解 nosql 了，memcached redis mongodb，眼花缭乱，没关系，试着用用，能很好地缓解 mysql 之类关系数据库的压力。

3、出于公司某些业务需要，你可能会钻研一个特定技术领域，比如全文搜索技术，lucene\solr，比如工作流 jbpm，比如权限系统，单点登陆之类的，又或者某些特定业务领域的算法研究，这些是你的加分项。

4、你开始发现你的代码很乱，久了以后自己都看不懂，重复的，难以重用的代码堆积如山，不想维护，BUG 百出。于是你要开始重视设计模式，合理地改造下自己的代码习惯，不再被僵化的 SSH，MVC 三层架构束缚住。

5、Google 和各种资料是你进步的动力，极少再遇到必须请教别人才能解决的问题，如果你这个阶段还老是问别人，你的技术生涯也就快到头了。

这个阶段，如果你的交际和自我推销能力不是太差，你的收入将是白领水平，至少接近了。

我觉得多数程序员在工作多年之后都能达到这个水平，有的人只需要两年，有的人却要用上五六年，在这个阶段落伍的人，有的是出于天赋和思维所限，有的是出于对技术缺乏热情，有的是出于工作内容的制约。等到年近中年，再也拼不过年轻人，被淘汰出局，只能在自嘲为屌丝和码农的无奈中黯然转行。

这是个很重要的分水岭，你能不能继续进步，能不能在 30 岁以后继续从事技术工作，能不能在公司里独当一面，我觉得就看你能不能超越这个阶段了。我前面提到的烂公司里的工作数年的项目经理，连这个层次都还没达到...

接下来又是个全新的阶段。

1、你要读读优秀开源项目的源码，你要搞懂一些当年不懂的基础知识，你开始理解 thinking in java 的精髓，你能写一些底层的代码，有时还会觉得自己封装的比某些开源工具更好用更简单。

2、WEB 的难点和重点永远都在于性能、负载能力上，而现在网络的发达造成了数据量和操作密度的大大上升，硬件却没有相应的进步，你得试着更好地运用更多的服务器来协同工作，

从 WEB 端到服务端到数据库，全都需要集群，需要分布式，需要合理的控制数据的流向，掌握好网站上下，一大堆机器的平衡，找出性能的瓶颈，找出稳定性和安全性的瓶颈，硬件出现故障，第三方技术出现错误，将被当成家常便饭融入到你的系统和代码里仔细考虑。

3、开始觉得方案无比地重要，一将无能累死千军将不断应验，一个不好的设计，一个不好的方案，会让一群优秀的程序员工作成果大打折扣。你要关注架构知识，不能再满足于 SSH 三层架构到底。领域驱动设计，面向事件开发，敏捷开发等等一系列的思想在关键时刻能决定你项目的生死，这个阶段不再有标准范例让你照抄，你只知道思想和原理，实践却需要自己不断尝试和改进。

4、多关注各种杂七杂八的开源技术，有些你可能前面已经接触过了，和通信有关的，和集成开发环境有关的，和架构有关的，各个领域你都应该能信口说上几个主流技术，虽然你可能只是听说过，了解，但关键时刻你得知道如何去选择技术，并快速掌握它。你还会去考虑尝试下别的语言，这里不是说转向什么 C++ C#之类的，那和 JAVA 程序员不相干，我说的是些运行于 JVM 之上的语言，比如 scala 和 groovy，初识他们时你会觉得 java 真的老了。但当你回到一个综合性的大型项目中，又会觉得 java 积累下来的整个体系技术是如此完善，就像一个工业化标准一样。

你能把这个阶段实践好，胜任项目经理，乃至中小公司的技术总监都是可以的。

不知不觉写了这么多了，以上是我个人眼里的一般向 JAVA EE 发展线路，限于篇幅也不全面，实际个人成长路线可能因为工作内容的不同差异会很大，有的人偏向底层研究，有的人偏向业务需求设计，有的带有浓重的行业色彩，而且技术之外，还有很多知识也很重要。w 做 JAVA 没有轻松的方向，但一个对技术抱有兴趣的人，走到这一步时，仍然会对开发抱有热情，想要写出好的项目。纯为了生计而工作的程序猿是走不到这一步的。

暂且收手，希望以上能让 J8 的新人们了解将来要面对的大致状况，同时也抱着打击下小朋友们幼小心灵的邪恶心理就此按下发表按钮。

能完整看完的同学，在此表示钦佩，我想当年的我也未必能看完这种又臭又长的叙述...