写给大三了还迷茫的你

# 前言：

大三了，作为一名java程序员，你是否觉得自己在学校里只做过一些小玩具的程序，学校里学的课程都是从理论出发，作业什么的都感觉不到有什么用，就算是一个课程设计也不过是小打小闹的基础总结，不知道如何去学习更多的知识，对以后出路感到了迷茫。又或抑你在大一大二的时候都是玩乐中度过的，现在想去努力一把，但却又不知道该如何去做。如果你是这两者中任意的一个，请继续看下去，希望以下的学习内容能给你提供帮助。(注：本文为大二大三并是学习java web 开发方向的同学而写)

# 基础篇

1. 学习Web的基础知识

现在的界面几乎被两个东西主宰，1）Web ， 2）移动端（Android 和iOS）。

学习html的基本语法，css的基本样式， js的基础，学会用 Firefox + Firebug调试Javascript代码。还有时间有兴趣的话可以了解下PHP。

1. 熟练一种编辑器（非IDE）如Linux里的Vim, Windows里的Notepad++，可以快速编辑器文本。
2. 学习Oracle ，了解MySQL，对于java开发方向来说， Oracle数据库运用相对较多， MySQL可以当兴趣来了解，MySQL的性能虽然比不上Oracle 但是开源免费让它能有更大的发展空间，现在它也是众多个人网站的首选数据库（免费~~你懂的）。

如果你对以上基础篇的内容都基本掌握的话可以进一步学习下面相关知识：

1. java I/O系统，特别是二进制数据的存储读取和 序列化与反序列化
2. java 多线程以及java网络编程
3. java 内部类（android学习过程中会遇到大量的内部类代码）

# 提升篇

1. 熟练掌握一个java开发的IDE,推荐MyEclipse ，熟悉各种常用的快捷键，Ctrl + 单击函数方法是必须掌握的，Shift + Ctrl + o 之类的日常了解就好，有效地利用快捷键能提高编码的速度。
2. 学习JavaEE内容：1）jsp（Tomcat6.0+的使用， 输出jsp表达式， request, response, session）, 2) Servlet(javaBean, MVC的基本思想以及简单代码实现)。个人推荐可以观看李华兴的Jsp视频教程，说得挺好的， 重要的是可以让你感受下什么叫记事本编程，什么叫说话的速度比不上敲代码的速度….膜拜吧，少年们~~
3. SSH（Struts2（V） 🡪 Hibernate（M）🡪Spring(C)）按顺序学习，简称查删改查，java码农必学神器，要理解SHH中MVC的工作原理以及具体的过程。（重点）
4. SSH的整合，尝试用MyEclipse + Tomcat + ssh整合来实现些小项目，如blog 或者 订书系统，投票系统之类的。（重点）
5. 加深Javascript学习，如果细心的同学肯定会发现在的js扮演着越来越重要的角色，从web到移动端你都可以看到它的身影，以及在各种javaEE方向招聘上都要求招聘者对js有较深的掌握。其次还要去学习相关的js库，如Jquery(推荐《《锋利的jquery 第2版》》，简单快速入门) 或者 EXTjs，我初学Jquery的时候才发现用以前的代码写得有多烂，刚接触Jquery的时候有种让人爽爆的感觉。

# 进阶篇

SSH是当前做管理信息系统最基本和最流行的通用技术，这几个技术出来已经很多年了，现在已经相当普及了。每个做增删改查的人基本都会SSH，所以，SSH没什么了不起的，SSH对个人发展来说，无法成为核心竞争力。一些公司内部所谓的核心技术，就是那些非SSH的知识。但是大多数OA类的项目都是增删改查，虽然没有技术含量，但是用工需求量大，这也就是能诞生出IT蚁族和IT民工的由来。但是，java不只是有SSH这套技术，如果你不想成为IT蚁族，如果你不想厌烦软件开发，那就不能只学SSH，这个没有太多技术含量的单一熟练工种，你可以考虑选择学习下面的一些知识：

1. Linux系统下的基本操作以及能掌握在Linux系统下开发
2. 学习版本控制工具，如svn（集中式）， git（分布式）。一个大项目，永远都不会是一两个人就能解决，众人共同开发，版本控制就必不可少的了，学会使用版本控制工具是必不可少的技能。

补充：

集中式控制系统的版本库是集中存放在中央服务器的，而工作时候是用自己的电脑，所以需要中央服务器取得最新的版本，然后开始工作，完成后再把自己的版本推送到中央服务器。而分布式版本控制系统根本没有“中央服务器”，每个人的电脑上都有一个完整的版本库，这样在工作的时候就不需要联网了，因为版本库就在你自己的电脑上。既然每个人电脑上都有一个完整的版本库，那多个人如何协作呢？比如说小A改了B文件，小C也改了B文件，他们只需要将修改的那部分推送给对方就能看到彼此的修改了。和集中式版本控制系统相比，分布式版本控制系统的安全性要高很多，因为每个人电脑里都有完整的版本库，某一个人的电脑坏掉了不要紧，随便从其他人那里复制一个就可以了。而集中式版本控制系统的中央服务器要是出了问题，所有人都没法干活了。所以个人还是推荐使用git， 重要的是git还是开源免费的，有兴趣的同学可以到 代码托管 - 开源中国社区 了解下。

1. Hadoop, 大数据时代，不解释，入门推荐看《《Hadoop 实战（第2版）》》。
2. JVM相关知识， 一般情况下一个程序员只要了解了必要的Java API ，Java语法，以及学习适当的第三方框架，就能满足日常开发需要，了解java 虚拟机的动作并不是一般的开发人员必须掌握的内容， 但是你真你希望自己只是一个一般的程序员吗？个人也是刚开始接触相关内容，不多说。
3. 数据库的优化，双11又到了， 有没有想过淘宝的服务器要怎么才能处理在一分38秒就提交了10亿元订单的大量数据？

# 总结：

本文为个人一些学习经验的一些总结，因为个人水平有限，写出来的东西可能有多的不足，望广大同学朋友指正。

邓永豪

2014.11.11