**学习线路**

**DONE DOING**

**前端：**

1. **Webstorm（前端IDE工具）、bower、gulp、firebug、requireJS工具下载使用**

注： node.js(JavaScript本来只是一个客户端脚本语言，node.js是其在服务器端的延伸)NPM是旗下一个包管理工具(类似java包功能)，不太适合前端，但bower（用node.js写的，所以安装前需要先安装node.js）用来管理前端合适，是一个前端包管理工具（操作目录需要在项目根目录下进行,在项目下建立.bowerrc配置下载文件目录 .json文件放置配置依赖信息，可以根据github路径、http路径、包名字来install，以后相同的项目只需要拷贝.json文件，则bower自动下载相应的包，不用直接拷贝包了）。Gulp是一个管理项目的命令集合。

2、**JavaScript语言精粹（书）ECMA、BOM和DOM编程**

JavaScript权威指南（查阅）、bom和dom在权威上学习。

JSLint是一个js代码检查插件（webstorm下自动集成，只需设置enable即可）。相对而言，JSHint实用。

1. **JavaScript异步编程**（主要用在node.js服务器端）**(书)**TP312JA/3590
2. **Ajax学习（了解，重点学习Jquery中封装的Ajax）（**前端通过js脚本与服务器端请求交互（核心是XMLHttpRequest）实现局部更新，本来只能通过form表单提交数据，然后刷新页面获取服务器端[数据） （是基于java技术下web2.0](http://请求）（是基于java技术下web2.0)的核心，结合了xml、java、js等多种技术的技术）
3. **Jquery学习（了解）**（一个JS类库,封装了**Ajax**和JS功能 ，提供更加简化的操作。）

Ps：书《锋利的Jquery》TP312JA/3194=2结合API文档查看，imook视频

1. **Bootstrap学习 （重点）html/css学习（了解）**

工具包：html/css/js的前端框架：很多功能效果直接利用库即可,其中js插件封装了jquery，所以jquery需要在boostrap之前引用。

注：imook网学习+官网学习

7、**AngularJS学习**(**难点、重点**是封装好的**Ajax**)

Ps：这是一个javascript的MVC模型的框架，imook视频和官网学习、权威指南。

1. **数据传输 JSON**

ps: JSON使用的是javascript声明对象字面量的做法，所以当从服务器端接收到JSON数据时只需要赋值给一个变量，则就可以使用JSON数据了，但通常接收到的是JSON格式的字符串，则需要先解析（**eval() $.parseJSON()**等方法都行）成**JSON对象**也就是**原生js中对象字面量**。JSON是Ajax传输数据的最好选择(前端接收到数据，用js处理非常方便，后端java有对应的包实现JSON数据包装)

1. **接口规范 RESTful**

PS: 书籍学习《java web service》：是一套客户端与服务器端交互的设计原则和规范

**10、Git和Github使用**

使用教程：先配置好SSH协议(git远程提交github的协议，这样才能push代码，否则只能clone项目，多人协作则在该github上添加多个SSH密钥)

1 先在github创建一个新的repository仓库

2 git切换到本地项目workbench路径下，**$ git init** 即可初始化该目录下文件成git代码库

3 **$ git add .** 把workbench目录下文件提交到暂存区

4 **$git commit –m “message”** 提交暂存区到本地仓库

5 **$ git remote add origin** [**git@github.com:xiyihai/PicLabel.git**](mailto:git@github.com:xiyihai/PicLabel.git)关联远程仓库(一次即可)

6 **$ git push origin master** 把本地代码push到远程master分支(操作目录下已建立git仓库)

7 **$ git pull origin master** 把远程最新代码更新到本地

8 **$ git clone git@github.com:xiyihai /PicLabel.git** 把项目clone到本地，此时该项目已经是git仓库，且关联好了远程仓库，之后若想提交代码只需commit，push即可。

11、**D3学习、Echarts学习、百度地图api学习、JTS学习**

**后端：**

1. **学习HDFS**(主要是java分布式IO流的操作)
2. **Mysql操作原理、Tomcat操作原理**
3. **JavaEE框架**(strcts2+spring+Hibernate）（系统的学习web整体构建）
4. **Maven、Ant使用**
5. **WinSCP(win linux传输工具)、putty(SSH协议)工具使用**
6. **Redis内存数据库**(不同于mysql等关系型数据库，不能用来持久化，是一种HashMap型可主动操作的key-value缓存数据库，以往的缓存无法主动操作)。
7. **Log4j日志使用(**SSH+Log4j是一种经典的Web解决方案)