1 http相关知识： 看了前面几章内容，了解各种方法使用场景 对应的语意，状态码，请求头 请求体，响应体等

2 代码规范：

两周后，导师分配了修复movedata 和 stat 项目 sonar 问题的任务，通过查看各个问题中sonar提出的需要注意的地方，来学习如何编写 可读性高，不易出问题的代码。当然在这一块上还是有些欠缺，希望在不断的实践中完善

3.1

简单用一项技术的话可以通过demo或博客快速上手，深入使用的话还是看官方文档靠谱点（学OkHttp的时候有个问题在网上一直没找到，最后在官方文档找到解决方法）

3.2

从文档中获取信息（开发过程中有张表的字段不明确是什么意思，代码中比较难猜测出来，通过看数据字典文档得知）

3.3

现在一般遇到一些问题的话都会及时的记录下来，因为有时候确实容易忘事，所以有个文档可以在忘记的时候可以帮助查找 还是很重要的。

3.4

特别是对于不懂的地方，更是要多问，自己想办法解决的话 可能会耗费很多时间，这样效率就不高了。（当时做增量同步记录完善，配合测试的时候，有个客户端一直有问题，自己在项目代码中排查半天没发现问题，后来导师看了下 直接就发现是客户端那边的问题）

4

这两个功能的实现是对 之前所学到的知识或者是能力的一个比较综合的运用和体现；虽然还是遇到了挺多问题，也是在不断的总结和完善。

优势和不足：

1 趁着还没毕业，社会对我们还有足够的宽容度前 抓紧时间多学点东西充实自己；不要等到到时步入社会后 才发现啥都不会。

2 虽然遇到难解决的问题 也会焦虑苦恼， 可是解决问题后的喜悦更加让我享受

3 适应能力的话 想举个例子说下： 我初中上高中的时候 全级120多名考上去的，在我那个班是属于垫底的， 但在第一次期中考的时候考了全级 第7； 高考的时候又是以垫底的分数进入我那个专业，然后拿了两年奖学金+国励，今年应该也可以拿个三等；讲这么多的话 是很想借助公司这个平台 快速的成长起来。

前面厚着脸皮大肆吹捧了自己，当然也存在很多不足的地方了；

1 以前做的几个项目都是比较简单的项目， 测试直接用浏览器点鼠标来测的，而且用户量也不大；接触到公司的项目后就感觉大不同了，经验还是比较欠缺。

2 有些问题的话 之前没有接触过 所以很容易会疏忽掉

3 思维比较直，没有考虑到有些问题其实是可以迂回解决的

未来展望：

1 代码是为业务服务的，了解业务才能写出更有用的代码。

2 吸收前人的智慧，取其精华去其糟粕的过程

4 近几年内深耕技术，钻研技术扩宽技术领域，当技术水平到达一定层次高度后再来考虑一个更细的方向去发展。 这是个人基于当前认知做出的想法，可能会随着认知的改变做相应的调整