## 课程总结

成为一名优秀的软件工程师是每一个软件工程师的目标，但是在实际中，优秀的软件工程师和一般的软件工程师的表现肯定是参差不齐的，这是什么原因造成的呢？这是一个我们必须反思的问题，同时还是一个关乎团队建设和个人成长的重要问题。

1.编写高质量的代码

良好的编程习惯不但有助于代码的移植和纠错，也有助于不同技术人员之间的协作。优秀的软件工程师都具有良好的编程习惯，他们编写尽量少的代码即可满足需求，还能够引入恰当的抽象层次，重构代码，将业务逻辑和其他程序逻辑分离。

一般的软件工程师编写不出高质量的代码，他们只关心自己的代码是否能够达到预期的输出结果，只要有输出结果，就认为自己的任务完成了，根本不在乎代码的可维护性和可读性。

2.快速、高质量的产出

优秀的软件工程师能够在代码质量和开发所耗时间上做出很好的权衡，能够严格控制技术风险，不断完善自己，能够在有限的时间内写出高质量的代码。 一般的工程师写出的代码质量很低，还声称是最优的设计，时间限制常常是他们怠工的借口。他们的代码会给企业带来很大的负担，并给团队成员留下一大堆问题。3.解决问题能力

优秀的软件工程师不会盲目地编写项目代码，他们会确定每个问题的目标，找出问题的根本原因，并找到解决问题的合理方案。优秀的软件工程师会将大问题分拆为小问题，针对小问题进行理解、构思、计划、维护和重用。一般的软件工程师只是匆忙编码，解决所分配的问题，不会进一步思考正在解决的问题，从不考虑长期影响。

4.正确的决定

优秀的软件工程师深知没有完美的设计、服务和工具，他们还深刻理解可选方案以及实际状况的约束并能做出适当的权衡。

一般的软件工程师只是盲目地使用当前比较流行的服务、设计和工具，而不做更深一步的理解和研究。

5.团队精神

团队精神和协作能力是程序员应该具备的最基本素质，优秀的软件工程师所写的代码不仅能够让机器高效地执行，更重要的是能够方便其他工程师阅读、改进和使用。他们写测试用例，确保所写代码的正确性，还能够方便其他人对程序进行改进。他们充分利用审查和使用代码的机会，学习其他部分的代码、分享代码风格、控制整个代码的质量。他们认识到并且坚信，代码审查的过程是相互学习和提高整个工程团队产出质量的最佳时机之一。他们的评论能够帮助团队成员成长，他们考虑整体设计、边缘情况、格式改进及提高代码整体质量的其他优化措施。

一般的软件工程师根本就不关心自己代码的可维护性和可读性，他们很少写测试用例，除非写测试用例是指派的任务。

6.洞察力

优秀的软件工程师具有长远的眼光，使他们能够承担起各种困难的软件项目，他们经历过无数的非预期任务，如被喊去解决线上服务器的问题，处理积累了很多技术债的代码以及面对较为紧张的期限。

一般的软件工程师对发生的问题从不积极响应，还责怪别人处理不当；当事情进展不顺时，他们很容易失去信心或推卸责任，不主动、不认真地解决问题。