# 定义高级

所谓高级开发者，与中级这一代码熟手的区别就在于，观察问题的角度不同，高级开发者会了解业务价值，并且关心产品。

有一种误解，有的人以为我了解了很多编程语言，知道很多酷炫技术，但是如果只是花哨，而不能给客户提供实际价值，那么就不能称为高级开发人员，高级开发不只是职位，而更应该能给利益相关者带来实际的收益。

所以问问自己是不是高级开发，先自问一下有没有构建过一个从未来角度（而不是仅仅是眼前）考虑问题的产品，并且要为需求方提供最大的利益。所以说，高级开发人员最重要的因素就是，您为公司，企业，客户，需求方带来了什么样的价值。

# 高级开发者的特征

一个高级开发，只拥有一项技能肯定是不够的，为了提升职业生涯，个人能力必须多元化，并且积极工作。高级开发这拥有以下这些特征，要达成高级开发的成就，可以参照以下几点能力去执行。

## 技术能力

编程的语言和技术都是基本功，必须要有，但是不够，高级开发人员还会：

1. 花时间了解为什么要在特定产品中使用特定的技术？
2. 为什么要选择特定的技术或者语言来构建特定的产品？
3. 为什么存在某些问题？
4. 为什么存在某些框架？

高级开发人员善于选择特定产品的正确工具和技术，使用这些工具可以给客户带来最大的利益。所以，高级开发们在为任何项目选择技术之前，都应仔细考量这些技术的优缺点。并且他们会对有益于开发生命周期的工具和想法有很好的认识，并且会不断改进交付过程以提供更好的产品。

了解产品的完整架构以及事务之间的连接方式，还考虑了产品在将来增长或者扩展的时候潜在的问题。技术和框架每次都会发生变化，但是高级开发人员应该了解更基础的知识，比如数据结构，算法，计算机原理，框架的本质万变不离其宗，框架的更替只是因为出现了一种让使用方更加省事的用法。

## 团队合作能力

大清亡了，再也不是单打独斗的年代了。好的团队合作可以为公司带来最大的价值，高级开发者会为其他成员提供帮助，如果自己不了解某些内容，也会试图学习它，高级开发不会表现出很大的自负，总是乐于接受别人的反馈。 一个成熟的高级开发不会为了炫技而编写复杂的代码，他们编码简单，干净，可读性高，让其他成员可以轻松理解。简而言之，和高级开发者合作的那些人，能感觉到相处愉快。

## 沟通能力

你能否正确与用户沟通并且理解他们的需求，用户面临的问题如何给出一个最佳的解决方案。你需要成为一个出色的倾听者，并且需要针对客户的情况提出一些专业的问题。 高级开发人员与客户互动，了解需求和问题，讨论预算，然后在组织能力范围内给出最佳的解决方案。您需要与他们建立良好的关系，给客户提出想法（但是别提意见），给他们适当的详细程度，并且帮助他们做出明智的决定。您还有可能需要向非技术人员解释复杂的编程问题。

## 学习能力

自身人员通常也是出色的自学者，他们始终对技术开发充满热情，他们相信每天都会学习新知识，并且对编程世界中的所有事物都感到好奇，他们喜欢探索新主题，也喜欢与其他开发人员交谈。他们阅读博客，提出问题，并且具有成长心态，不断地学习与好奇心使得他们和其他开发者有所不同。 如果高级开发人员不知道某个问题的答案，那么他们将尝试从各种不同的资源中寻求解决方案，与合适的人交谈。他们知道如何通过不断的学习来成长。对于任何技术，语言，框架，他们没有任何偏见，他们明白一切都有其优缺点，如果一项技术不合时宜，他们会给他人提供不适宜的理由。

## 分享能力

高级开发人员乐于与业界共享信息，并喜欢加入开发人员社区，无论是聚会还是讲座，他们总喜欢与他们互动，他们选择走出去与其他社区进行互动，而不是关在一个盒子里闭门造车。这有助于他们获得更多知识。除此之外，他们还喜欢加入在线社区，并解决世界各地其他开发人员的问题。他们善于通过博客文章来共享信息，提供问题的解决方法，与其他开发人员讨论问题的时候，他们总是尝试使用各种线上资源来找到最佳的解决方案。

# 怎么做去达成高级

## 技术能力

除了技术功底之外，高级开发更善于思考，并且更深入技术底层。我们应该转变思维，摒弃以前见山是山的思维方式，站在不同的维度，高度去思考问题，让自己的思考结果更加完善，并且：

看问题要站在一定的高度，俯瞰全局，特别是在 设计解决方案的时候。

遇到问题要尽可能深挖，找到解决方案了还没结束，挖穿问题，看看是不是类似的问题都有相同的结论。

## 团队合作能力

与任何人何做都要本着合作共赢的心态，给他人提供尽可能的帮助

心态平和，接受批评反馈

团队合作编码，不要为了炫技而用一些生涩的语法，要简单明了，他人一眼看明白

## 沟通能力

客户，用户，需求方 就是上帝，善于倾听他们的意见

上帝并不会个个都懂技术，要选择适当的详细程度来解释技术问题

在自己可以触达的能力范围之内给出最佳的解决方案，如果达不到，那也能说明自己尽力了

## 学习能力

保持好奇心，每天去技术社区猎奇，找到合适的主题就着手研究

多与合适的人交谈，聊天过程中很多人问题都有可能闪现灵感

## 分享能力

加入线下交流会，学习高手的思维方式

自己的技术成果，要总结成文，发表到开源社区，获得更多人的认可或者反馈