1. 职业规划

预则立，不预则废。

职业规划，这个词对于大学生或者刚刚大学毕业的人来说，非常迷惑。这是什么东西，除了面试扯淡，真有用？世界变化快，规划就是鬼话。

我也年轻过，我也曾经这样想。人生还长，当下最好，想这些没用的。

问自己几个问题：

5年后，10年后，你想做什么事情？

你了解你说的那件事吗？

说出那件事的最大价值。

详细描述，做那件事，你需要具体做什么工作，需要什么能力。

我敢说，如果你还非常年轻，你的答案是不会很详尽，很接近真实的。不要误会，我不是在PUA年轻人，我的意思是，这样的问题，需要社会阅历，需要生活经历，才能回答得好。

持续思考这些问题，就是职业规划。

职业规划，是关于你的目标，和到达目标的路径。中间就是你每天日常的行为。

**职业规划，不是职业梦想**。那为啥年轻的时候都容易把规划弄成梦想呢？还是因为没经历过。你会说，那还有梦想成真，真就5年当上技术总监的呢？对，应该有这样的人，这样的人是我们所有人的榜样，可是大多数人，不要这样去想。为什么？不是想阻止你成功，而是想阻止你被打击得太惨。

那么普通人的职业规划到底是什么？**职业规划，是指对你所处职业和行业的环境，职业的价值，职业所需的能力，和对自己在环境中所处位置的动态认知，以及基于这样的认知，选择你每天，每个时刻做什么，怎么做。这叫规划，不是梦想。**

分三层意思。

**第一层意思，你得有认知。**职业规划的是你的行动，而不是吹牛或者面试的话术。吹牛怎么吹都行，反正也不去做。行动就要有依据，不然就是盲目而动。这个依据，就是你的认知。你要了解你的行业，你的职业，这个职业的历史，前因后果，行业中的不同的人，不同的公司，不同的客户，哪些是前驱者开拓者，哪些是追随者。这叫做环境。这个环境是我们大家共同生存的依托，它同时也是影响我们最深，但是我们最难察觉的因素。比方说，为什么现在做区块链工资高，是因为创造价值大？不是，是因为资本逐利，有溢价。但是这个能持久吗？要去想。还要认知的，是你职业的价值。你工资高，是因为程序写得好吗？也没错，可是如果继续追问，是因为你用计算机技术，解决了某种问题，需求最终来源于人。你得要有价值观，技术要向善。

**第二层意思，认知得是动态的。**社会在变，技术在变，公司在变，如果你不变，就很可能被淘汰。你至少眼界应该越来越宽，能逐步看到整个行业/公司的全貌，也能看到不同岗位所需要的能力。做到这一点很不容易。第一难的，是你不关心，只管自己一亩三分地，程序写得好，bug修得快，管那么多干啥。第二难的，是傻逼思维，认为只有写程序最牛。可不是吗？产品经理屁大点需求都弄不明白，架构师天天就画图吹水，项目管理天天就知道搞数据，研发经理更是不干人事，至于更高的领导，他们就是命好，天天就开会，拿我十倍百倍工资。这个动态认知决定了什么？你觉得只有写程序是最牛的，那可不就只能当码农。写不了什么具有影响力的程序出来，就是码农。你以为Linus只是C语言牛逼？他如果不能深刻懂得操作系统这个产品，他光算法写得好有屁用。他说过一句话，没有任何一个软件项目比它的用户还重要，你能理解吗？可是我接触过的国内的程序员，没几个有这样的思想，认为写程序就是写代码本身，很少去想用户的重要性。所以，丢掉那些面子思想，直面自己的无知，虚心才能受教。乔布斯有那么句话，Stay Hungry, Stay Foolish。还是中文美：求知若渴，虚怀若谷。

**第三层意思，主动选择你的行为。**领会了上面两层意思并能时时坚持，你的行为选择自然会有你的认知去主宰。其实，能时常主动构建自己的认知，本身就是一种行为，我理解为这就是知行合一。如果你行动失当，其实就说明认知有误。具体一点，你如果有全局观，价值观，不会计较一时得失，也不会关注什么内卷，就会专注提高自己创造能力和在环境中的贡献能力。这就是主动选择。横向比较只能说明没有真正的创造力。Linus会不会去和James Gosling比较？各做各的就好嘛。典型没出息的行为就在不去多想创新做好东西，就知道抄袭同质化竞争。当别人这样想时，一笑了之可好？不要和别人比，这是老生常谈，我说一点小道理：你和谁在比？身边的人吧？那你看不到，不认识的人呢？单说竞争，难道你就是和身边人竞争吗？你所处的是一个大行业，几千万人所在的行业。眼界放宽一点，心就没那么狭窄。竞争是存在的，但不是和身边的人，也不是看一年两年的短期，看谁做得快，谁能在领导前挣个头彩。有点出息，就眼光长远一点，看看自己三五年能做出来什么成果，或者能在哪些方面真正的提高，这样的竞争，不是同质化竞争，不是做试卷似的竞争，而是比谁更能专注于自己的目标，这样不就可以各得其所，百花争鸣吗。

最后说说**怎么做职业规划**。

持续地，反复地观察和思考，不断地更新答案，针对最开始提的那几个问题。

具体方法：

**从身边向外观察。这是最切实际，成本最低的方法。**你一定所处一个小组，一个部门，一个产品线，一个公司，一个行业。那你可以从近到远，从小到大去思考。这里的其他人，其他组织，他们关心什么，他们在做什么，这里的价值是怎么被创造，又是怎么被分配的。这样你可以理解每个岗位，每种组织的价值所在，也去理解岗位所需的能力，组织被赋予的责任。更具体的方法，比如开会的时候仔细听，哪怕和你的工作没有直接关联，试着站在对方立场去思考。

**多看看编程技术之外的书或者文章，比如产品，测试，项目管理，企业经营等等。**不能当书呆子，你明确自己看这些的目的是要**构建自己的理解**，而不是填鸭更多的知识。看这些是为了增加一些感觉，让你不那么固守在自己的岗位上，而是明白，你的工作其实只是一个很大产业中非常微小的一环而已。

**思考的时候拔高一点自己的站位**，比如，把整个产业作为观察对象，把一个很成功的产品作为观察对象，把一个你觉得很厉害的牛人作为观察对象。

**持续地做这件事，不要想一蹴而就。**既然是认知，就是需要长时间地不停更新自我。

本星球的大部分内容，将会是关于具体的技术讲解和分析。学习方法很重要，思考认知很重要，但是这些“务虚”的话题，却不需要很多篇幅，因为直击要害即可，不必翻过来覆过去的讲。之所以把职业规划作为第一篇专栏文章，因为这是提纲挈领的话题。没有规划，终究迷茫。