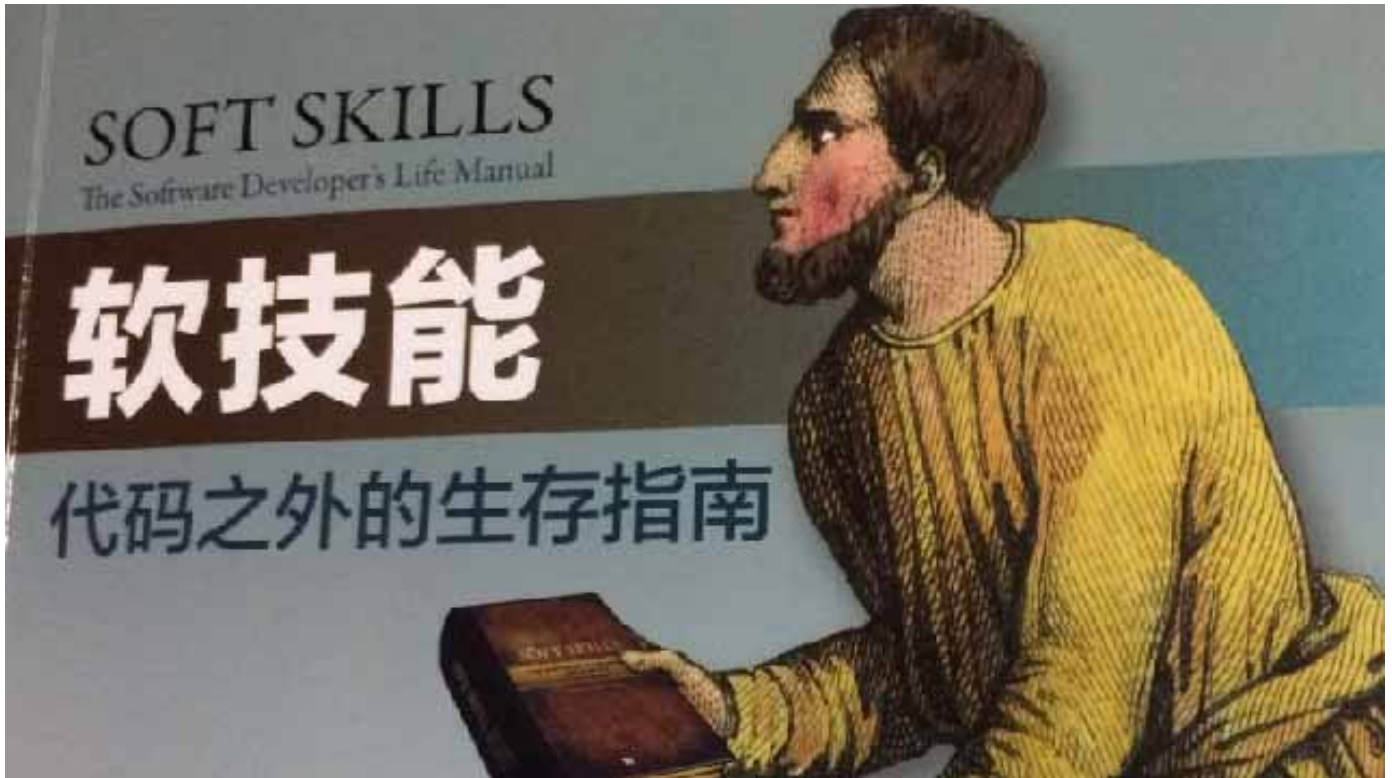


程序员必会的十步快速学习法 (附书单)

2021 年 2 月 5 日

[筆記](#)



本文对《软技能：代码之外的生存指南》书中的“十步学习法”展开讨论，该书作者是John Z. Sonmez，一名程序员，书中有很多面试、专业等建议。有一些建议相见恨晚。某天在喵哥群里看到了本书的推荐，非常感谢推荐。

本文末尾附上了John Z. Sonme推荐的书单（自我提升、软件开发、投资类），感兴趣的可以看一下。

虽然本书说的是“程序员必备的软技能”，但我认为这是所有职业都通用的软技能。该书中总共分成七个部分，分别是：

职业、自我营销、学习、生产力、理财、健身、精神。我做了下面的思维导图，可以概括个大概，可以根据需要随意读取某个章节：



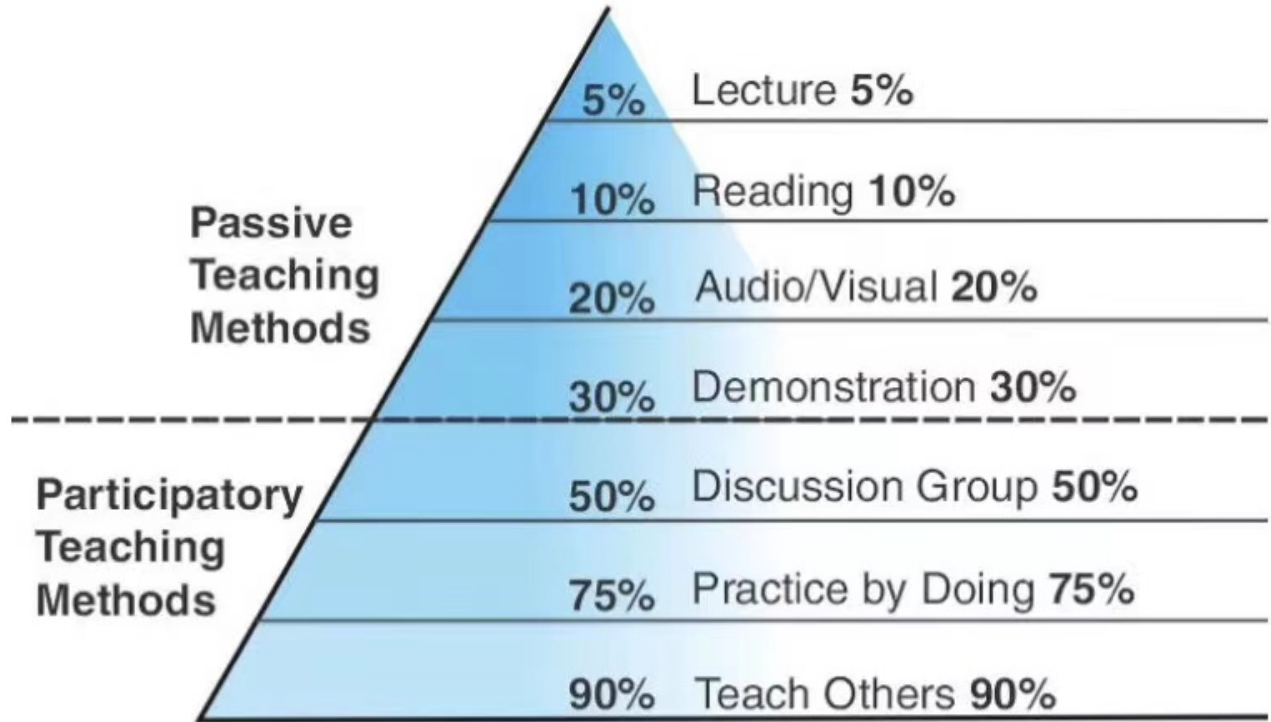
本书总共71篇文章，可以把每一篇都看成一篇高质量的博客，也就是说如果你能够写出71篇高质量的博客，你也就能够出书了（手动狗头）。

这位作者老哥很会玩，他有自己的youtube频道，网址 youtube.com/jsonmez 我点进他的油管主页看了一下，好家伙，作者本人还是个猛男：



十步学习法

你是不是和我一样，学习知识的时候就喜欢拿着一本书“从封面到封底”仔细阅读，这样可以学到东西，但是效率很低。因为我们往往只会用到书中的一部分内容而已，没用到的内容会迅速忘记。这是关键，没用到的内容会迅速忘记.....通过不同的方法，对于知识的掌握程度也不同，可以参考下面这张图：



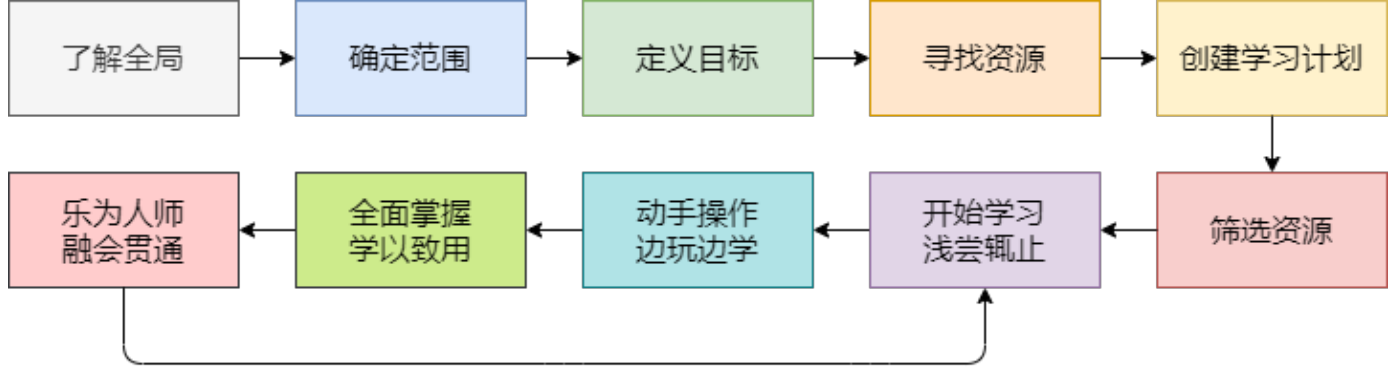
Adapted from National Training Laboratories. Bethel, Maine

仅仅通过阅读，学习信息的留存度只有10%。因为我们并没有掌握那些没被用到的知识。学到了某个知识，叫懂得；练习过学到了的知识，这叫掌握。掌握了的知识，不容易忘。

懂得很多道理，但是依然过不好这一生。——因为这些道理只是头脑上的懂得，但并没有身体力行的掌握.....

学习二字，学——头脑上的懂得，习——头脑和身体上的练习。练习中的体验，极大加深了该知识的记忆。我们都有背单词的经历，但真正掌握这个单词的时候，还是和这个单词在某个地方不期而遇时，比如某一天下午你在看国外新闻时，看到了这个单词，这个时候才发现这个单词还有这个用法。我们学会了使用它，直接使用它，也就记得了它。

十步学习法如下：



不要被这么多步骤给吓到了哈，总共分成两个部分：

第一步到第六步：学习前的准备部分，即为前期调研以及设定目标。

第七步到第十步：学习过程部分，即这四步：学习-实践-学习-分享教授（Learning, Doing, Learning, and Teaching）

第1步，了解全局

这些年有个很流行的认知的四个阶段：不知道自己不知道什么、知道自己不知道什么、知道自己知道什么、不知道自己知道什么。

“了解全局”就是从第一阶段到达第二阶段的过程。如果想要学习一门编程语言，那么首先得知道计算机的编程语言都可以做些什么，市面上有哪些编程语言，这些编程语言之间又有什么关系等等。关于这些我们可以在网上搜索一些文章，或者请教相关的人，了解个全局。

等到我们了解个差不多的时候，我们终于知道了这些东西大概是个什么，但是具体是啥我们是不知道的。这就进入了“知道自己不知道什么”的状态。

了解全局是必要的，如下面：

Baseline vs. Progressive



如果我们按照图的上半部分，在没有了解全局，就开始精细地学，这样恐怕不会持久。

但是像下半部分那三张图一样，先了解一个模糊的大局，再逐渐精细地了解，这样的效果会更好

第2步：确定范围

我们大致了解全局之后，需要确定范围。因为计算机知识浩如烟海，我们无法掌握全部——穷尽一生也不可能。甚至就算单是学习一门编程语言，比如如果想要精通C++，那没个

好几年是不可能的。但是我现在的学习时间只有一两周，那只能把范围缩小，这样的范围可以是，使用学习某个编程语言中的：

- 图形界面相关的编程。
- 文件操作相关的编程。
- 网络通信相关的编程。

短时间内我们可以聚焦在以上三个主题中的一个。一个主题就是一个聚焦范围。

为什么要聚焦，我们啃知识的过程就想打怪。还记得《灌篮高手》里面，宫城良田是怎么把三井寿的门牙打掉的吗？其实三井寿是带兄弟围殴的，如果宫城不聚焦地去打三井，那么他就会被围殴的毫无还手之力。

我们打知识也要聚焦，不然会被知识围殴地毫无还手之力。

第3步：定义目标

确定一个小范围之后可以定一个小目标，比如下面的任意一个都是小目标：

- 写出一个图形界面
- 用程序操作文件
- 用程序进行PC之间的通信。

第4步：寻找资源（学习材料）

学习一个知识点的时候，最好不要就只用一本书，可以参考不同各个书上的指定的章节，因为很多时候不同的书都在描述同一个东西，但是描述的角度不一样。

当我们从**A**角度看某个东西的时候，可能会消除在**B**角度看相同东西时产生的那个疑惑。尽可能寻找更多资源。

信息来源：

- 图书
- 博客文章
- 在线视频，公开课
- 询问这方面的专家
- 播客
- 源代码
- 示例项目
- 在线文档
- 各种网页
- 付费视频、课程

第5步：创建学习计划

创建学习计划，就算我们已经在第2步里面缩小了学习范围，可能这个范围内还有比如六个如A,B,C,D,E,F个知识点，而有些知识点是最基本的，比如知识点A，所有其他知识点B、C、D都依赖于A。这个时候我们肯定要学最基础的A，再学其他不那么基础的点。

就像上图中的螺旋上升的阶梯，往上的阶梯需要稳固的基础，也就是一个起点。

很多时候可以直接参考书的目录，基础的点都在前面，进阶的内容都在后头。

第6步：筛选资源

因为我们在第4步的时候已经寻找了很多资源，那么这个时候可以进行筛选，去掉重复性比较多的，或者不太适合自己的那些书，找出最物超所值的那几本——可以看看网络上对该书的评价。

第7步：开始学习，浅尝辄止

以上学习前的准备过程就差不多了，接下来就是学习过程，学习-实践-学习-分享教授（Learning, Doing, Learning, and Teaching）。

这一步的学习是为了下一步的“动手操作”。

这个时候我们就要清楚地知道，为了下一步的操作，我们需要的“最小量的启动知识”有多少。度很关键，学太多或者学太少都影响效率。

第8步：动手操作，边玩边学

如果是学习代码，这一步可以按照示例代码自己试一遍，或

者自己改一遍。

如果是学习英语单词，就使用这个单词造句、练习发音，并根据不同的释义想象可以在什么场景下可以使用。

以上过程也是根据不同的例子来强化自己的记忆。我知道有一个挺有名的程序员章亦春，他喜欢在学习的时候，手抄代码。第一次听到这事的时候，我震惊了，手抄，那效率得有多低？后来我在刷leetcode的时候，遇到看不懂的那些题解代码，复制粘贴后上程序跑一遍还是不懂。然后我就在纸上手抄这些代码，抄着抄着的过程，竟然真的慢慢懂了，真香！

第9步：全面掌握，学以致用

我们可能会在上一步的操作过程中遇到问题，这个时候可以回到资料里面，去寻找答案。又或者这个时候我们再看之前看过的内容，也会有不同的体会。

第10步：乐为人师、融会贯通

原书中是这样写的：

你可以用多种方式将自己所学教给他人。你可以写博客，也可以制作YouTube视频。你也可以跟自己的爱人探讨，将自己所学解释给他/她。重点在于，你要花时间将自己学到的东西从大脑中提取出来，以别人能够理解的方式组织起来。在经历了整个这个过程之后你会发现，有很多你以为自己明白了的知识点，其实并没有摸透。于是你

会将那些以前自己没太明白的东西联系起来，并且简化自己大脑中已有的信息，将它们浓缩并经常复习。

费曼学习法也大概是这个意思，诺贝尔物理奖得主费曼说：“如果你没有办法用简单的语言表述你所学的知识，你就没有真正学会它。”著名的费曼学习法，推崇以教为学。

这段视频解释得相当透彻：[为什么费曼技巧被称为终极学习方法](#)

首先我们通过练习大量的例子，从而总结出这些例子的通用规律。然后将这些例子以他人听得懂的方式表述给他人，在这个过程中你再一次对这个通用规律进行了总结——或者说压缩。同时在表述的过程中，进行了逻辑上的验证。

有时候，逻辑验证会出现卡壳，会发现有很多你以为自己明白了的知识点，其实并没有摸透。

这还没完，在表述给别人的时候，别人会产生疑问，这时候可以将别人的例子，套用在你已经总结出的规律中，进一步验证这个规律的准确性。

分享知识的途径：

- 写博客文章
- 创建视频教程
- 发表演讲
- 与朋友或爱人进行对话探讨
- 线上或者线下回答他人的问题

只要善于总结，就超越了很多。当年RTS游戏盛行的时代，WAR3（魔兽争霸3）游戏的王者，中国人皇SKY李晓峰。他当年在网吧接受采访的时候，秀出了他一个记满技术要点的小本子，这是他对很很多场对战的关键总结。说实话，字写得真的丑，写的句子也和“文笔”两字丝毫不沾边，但这并不能阻止他成为世界人皇。

最后，祝各位前程似锦。

最后的最后附书单

文末附上原书《软技能：代码之外的生存指南》作者推荐的书籍汇总。各位也可以参考以上学习方法，选择某一主题的书，进行学习哈。

自我提升和励志类图书

- Steven Pressfield的The War of Art [Black Irish Books, 2002]
- Dale Carnegie的《人性的弱点》(How to Win Friends and Influence People)[Gallery Books, 1998再版] 没错就是卡耐基的书，英文名直译有点意思：如果获得朋友和有影响力的人
- Napoleon Hill的《思考致富》（Think and Grow Rich）[Wilder Publications, 2007]
- Maxwell Maltz的《心理控制方法》（PsychoCybematics）[Pocket Books, 1989再版]
- Norman Vincent Peale的《积极思考就是力量》（The

Power of Positive Thinking) [Touchstone 2003再版]

- Ayn Rand的《阿特拉斯耸耸肩》 (Atlas Shrugged) [Signet, 2005再版]

软件开发类图书

- Steve McConnell的《代码大全》
- Robert Martin的《代码整洁之道》
- Eric Freeman, Elisabeth Robson, Bert Bates和Kathy Sierra的《Head First设计模式》

投资类图书

- Gary Keller的The Millionaire Real Estate Investor [McGraw-Hill, 2005]
- Robert Kiyosaki的《富爸爸，穷爸爸》
- Kerry Given的No-Hype Options Trading: Myths, Realities, and Strategies That Really Work [Wiley, 2011]