

写在前面的话

不知不觉，专栏已经更新了一半有余了。在两个月的时间里，我收到了很多同学的留言，很开心可以跟大家保持交流，也很惊喜见证了大家的成长。

在整个学习过程中，你有没有一些学习感悟呢，有哪些新的学习收获，又有哪些实践感悟？有没有对前端的知识架构有了更为深刻的认识，是否规划好了自己的前端之路。

如果你有想分享的经验，可以在文章下留言，如果你的留言被选中作为用户故事，我们将会为你送出“极客时间”& winter联名T恤一件。分享你的收获，我们一起成长。



下面我们一起来看看三位同学的分享。

极客时间用户@阿成 分享

在我进入知乎社区的时候，winter老师已经不在这个社区了，虽然是退隐了江湖，但是江湖上还流传着他的传说，后来我在Google Feed中看到winter老师的开篇词，看完就感觉是我要的东西，还正好赶上打折，就毫不犹豫地买了。

我本科学的是光电信息，所以并不算是科班出身的程序员（虽然也有学C啦）。我本身对计算机程序一直很有兴趣，所以从大三开始，我就决定放弃从事本专业，做一名程序员。要说为什么选择前端领域，自然也是烂俗的“门槛低”理由。

不过，我没想到的是，前端的学习曲线到后面是如此陡峭，并且前端发展的速度如此之快，让人目不暇接，也让很多人产生了“技术焦虑”。

在这样的环境中，我也产生了迷茫，作为一名前端开发人员，我到底应该学什么，怎么学。毕业后两年多的时间里，我从使用原生的HTML/CSS/JavaScript，到现在流行的视图库Vue/React，看似我好像学会了很多，但是离开这些库，或者说将来这些库被淘汰了，我又剩下了什么呢？

我逐渐意识到，我需要的可能是一些更“底层”的东西。

那么，什么是“底层”的东西呢，这里可不是指计算机底层，而是指不会被淘汰的东西，永远有用的东西。我认为对于任何程序员来说，最重要的就是：迅速获取某项知识的能力以及动手解决问题的能力。

前者需要我们建立自己的知识体系，也就是winter老师在第一讲中提到的，我们需要完善的知识架构。

我们不需要也不可能记住整个架构的知识，但我们却可以在需要的时候想起来这块知识的位置，是干什么的，能帮我们解决什么样的问题，可以通过什么关键词检索到，这就够了。

其实你让我回忆winter老师这么多节课的所有内容，我也是好多想不起来的，但多少都留下了印象。我们的大脑容量是有限的，所以“按需加载”也许对人脑也是个不错的策略。

后者其实就归功于winter老师精心设计的课后题，先看这些课文总归是差了点什么，只有你动手去做，才能真正理解这节课所讲的内容。

在做的过程中，你多半会遇到困难（这其实也是winter老师的本意），如果你通过努力解决掉了，这个提升才是最大的。你的方案不一定是最好的，或者你可能实在没什么头绪，这时候也可以利用搜索引擎去看看别人的方案。

因为我早上到公司要坐一个半小时左右的公交，路上我基本上会用耳机听完课（不伤眼），然后剩下的时间去思考回顾一下，想想作业怎么完成。

在中午吃完饭休息的时候，动手做一下，有时候时间不一定够，就利用晚上和第二天的时间完成。每个人的情况都不一样，不过，想要有收获，就得付出多一些。

希望大家都能学有所成，有所收获。

极客时间用户@峰峰峰 分享

先感谢极客时间给了我这次学习分享的机会，如果文中有不足之处，希望小伙伴们可以多多指导。

我是转行过来学习前端的，目前在前端这个岗位上工作了大概一年半的样子，最初知道重学前端这个专栏，也是因为朋友圈里面的一些大牛分享的，比较好奇，并且老师又是非常有名的winter老师，就报了这门课程。

我本人是对前端的技术非常感兴趣的，尤其是原生基础知识，因为我觉得一个人基础的扎实程度，能决定这个人在这个领域所能发展的高度。

刚开始看到课程目录的时候我很吃惊，因为这个目录和我之前所学的所有的课的目录都不一样，从文法、运行时的角度去解释JavaScript这门语言，这让我非常兴奋，因为我知道，这正是我所需要的。

通过跟随winter老师不断学习，我发现这把我原有的一些知识结构打乱了，但这并不是坏事，因为我以前的学习和知识结构都非常的混乱。

通过这次学习，让我对JavaScript的理解更加深入了一些，同时，重新梳理我的知识点也让我的知识结构更加的清晰了起来。

可能是我工作时间尚短的原因吧，我觉得老师的每一节课对于我来说都是一个新的东西。举个例子，就拿第3讲和第4讲来说吧，标签语义化我一直都知道，但是我在工作中用得很少，并且在理解上也只停留在非常浅的层面上，但是通过老师的讲解，让我理解到了语义化标签真正的含义。

我现在在工作都会开始使用一些我比较熟悉的语义化标签。包括一些浏览器的工作原理，关于JavaScript这门语言是面向对象还是面向类的解释，以及JavaScript的执行等等，这些知识点都让我受益良多，非常感谢winter老师，希望以后还能有机会跟着老师继续学习。

最后，分享一下我的学习方法吧。

1. 先提升知识的广度，再去拓展知识的深度。
2. 养成随时记录的习惯，可以是笔记，可以是代码（一定要写上注释）。
3. 多逛逛技术论坛，有条件的情况下多去实验一下论坛中自己感兴趣的代码。
4. 最后一条也是最重要的一条，坚持每天八小时工作外的学习和锻炼。

以上就是我的学习心得，不足之处还请多多见谅，希望对小伙伴们有帮助。

极客时间用户@木易杨 分享

我看了老师在“极客live”的直播，总结了他提到的一些观点。

1. 对于框架的使用没必要花太多时间，应该多研究一下三大框架背后的设计思想。
2. 当一个程序员对算法、语言标准、底层、原生、英文文档这些词汇产生恐惧感的时候，他的技术生命已经走到尽头。
3. 前端架构主要解决的是高复用性，架构能力提升方向主要是组件库开发、前端框架实现等。
4. 对于前端进阶这个问题，其实看书的作用和意义已经不太明显，需要寻找好的平台和合适的项目，在项目中不断克服难题并挑战自己，遇到问题再去查资料总结。如果只是闭门看书那很难成为高手，书只是基础而已，真正的应用还是在项目中。
5. 寒冬中能做的只有提升自己，但是光靠技术是不行的。
6. 推荐 TensorFlow、可视化切图、PWA、WebGL。
 - TensorFlow 可以了解使用并做点东西出来，原理很难但不影响使用。
 - PWA 有望进一步发展。
 - WebGL 在未来会是一个很好的方向，它可以实现任何你想要的界面效果，但重点需要多掌握图形学的基础知识，它和算法，数据结构一样重要。
 - Weex和 RN 虽然都叫 Hybrid 但不太一样，前者适合大厂主要是嵌入 APP 中使用，后者更适合创业公司。
7. 不推荐 SSR、TypeScript、函数式编程。
 - SSR 不太看好，其主要是用于SEO，不太建议用做服务端渲染，其能够使用的场景不多，而且成本代价太大。
 - TypeScript 是好东西，是很有前景的语言，但适用于十万行以上代码级别的大型项目，小项目并不适合，反而徒增复杂。
 - 用 JavaScript 做函数式编程并不靠谱，Map/Reduce/Redux/Hooks等并不是函数式编程，只是长得像而已。

PS：笔记内容由自己和群友提供，仅供参考。

我的一些思考和尝试

听完 winter 老师直播后对其观点很是赞同，但因为是面对所有人讲解，所以内容有点多范围有些广，但对个人来说还是找适合自己的方向，在自己相对熟悉的领域再去扩展去突破。横向只是拓宽你的眼界，纵向才是你的核心竞争力。

对我来说感触最大的就是 3、4、5 这几点，因为我一直在思考以下几个问题。

1. 我现在是高级前端，但又感觉自身很弱，那我如何才能夯实我高级的地基然后成为资深前端呢？
2. 项目迭代节奏快日常加班又多，那我如何做才能平衡工作和学习？
3. 单单提升技术好像还是有很大瓶颈，那我如何提升我的职场核心竞争力？

结合 winter 老师的专栏和最近看的几篇文章，说说我对这几个问题的思考。

对于第一个问题，在 2 个多月前我还是不会写文章的小白，那个时候受到一些文章的触动，尝试把我的前端之路记录下来。

我的想法很简单，就是把前端进阶 28 期的重难点知识全部讲完，目前已经进行到第 4 期了，通过写作把知识通俗易懂的介绍给别人，在这个过程自己肯定会收获很多很多。

这就是我目前在尝试的学习方式，通过写作建立自己的知识架构，并且在这个架构上不断地进行优化，时间到了自然就进阶了。

对于第二个问题，刚开始写作时精力充沛时间也很多，每天都会更新技术文章，但随着项目迭代压力增大，文章更新速度相应就变慢了一些。

虽然文章质量提升了很多，但更新速度从日更变成了周更再变成了双周更，这个说实话我自身已经很难接受了，因为速度变慢导致我年初的计划要打折扣。

所以说做项目和自我学习要如何权衡，是否说我要减少项目难度和时间并在工作中摸鱼去学习呢？

我的结论是不，因为一句话，“最好的学习就是在项目中锻炼自己”。既然我有这么好的项目去锻炼，那为什么还要摸鱼去学习呢，这不就是南辕北辙嘛，道理很简单但不是所有人都懂。

我最近在执行的方法是专注 + 锻炼 + 利用周末。

专注即工作时专注于工作，努力做好每次迭代，遇到难题迎难而上，工作时不开微信，勤用笔记安排日常工作并整理文档。

锻炼即一周抽出三天时间每次去健身房锻炼1小时，强壮的体魄才能撑住高强度的工作和学习，因为网易有健身房所以冬天锻炼也没什么问题，没办法去健身房的小伙伴可以尝试跑步或者在瑜伽垫上做Keep。

利用周末即加大周末和平常晚上熬夜的时间去写文章，减少娱乐的时间，正所谓时间挤一挤总会有的，但这件事情比较反人性，所以重在坚持。公众号写文章更容易坚持，原因在于文章发布后有正反

馈，比如粉丝的增长，留言和鼓励，赞赏等等，这些都更能促使你坚持下去，时间长了自然就养成了习惯。

对于第三个问题，winter 老师也说了，寒冬中光靠技术是不行的，那应该靠什么呢？幸好在最近看到的一篇文章中找到了答案，那就是表达能力。

表达能力是形成自己的框架系统，有理有据并且逻辑清晰，而且能让外人听懂，大部分优秀的人都具备这样的能力。反观自己并没有这样的能力，所以我要努力提升这块，其中最重要的方法就是写作训练。

兜兜转转那么多，其实我一直在尝试的进阶方法就是通过写作建立自己的知识架构体系，同时提高自身的表达能力，通过正反馈机制和锻炼保证我长久的坚持下去并最终养成习惯，习惯这一模式之后变成优秀的人也只是时间问题罢了。

希望我的一些思考和尝试能对你有所帮助。

（直播回顾：<https://v.qq.com/x/page/x0830fc8xoj.html>）

恭喜幸运用户 @阿成、@峰峰峰、@木易杨，你们的经验被选取成为本期的用户故事。极客时间将送出“极客时间”& winter联名T恤一件。1个工作日之内，工作人员会与你取得联系。

读完了三位同学的故事后，你有哪些感受呢，留下你的收获，我们一起成长。

写在前面的话

不知不觉，专栏已经更新了一半有余了。在两个月的时间里，我收到了很多同学的留言，很开心可以跟大家保持交流，也很惊喜见证了大家的成长。

在整个学习过程中，你有没有一些学习感悟呢，有哪些新的学习收获，又有哪些实践感悟？有没有对前端的知识架构有了更为深刻的认识，是否规划好了自己的前端之路。

如果你有想分享的经验，可以在文章下留言，如果你的留言被选中作为用户故事，我们将会为你送出“极客时间”& winter联名T恤一件。分享你的收获，我们一起成长。



下面我们一起来看看三位同学的分享。

极客时间用户@阿成 分享

在我进入知乎社区的时候，winter 老师就已经不在这个社区了，虽然是退隐了江湖，但是江湖上还流传着他的传说，后来我在 Google Feed 中看到 winter 老师的开篇词，看完就感觉是我要的东西，还正好赶上打折，就毫不犹豫地买了。

我本科学的是光电信息，所以并不算是科班出身的程序员（虽然也有学C啦）。我本身对计算机程序一直很有兴趣，所以从大三开始，我就决定放弃从事本专业，做一名程序员。要说为什么选择前端领域，自然也是烂俗的“门槛低”理由。

不过，我没想到的是，前端的学习曲线到后面是如此陡峭，并且前端发展的速度如此之快，让人目不暇接，也让很多人产生了“技术焦虑”。

在这样的环境中，我也产生了迷茫，作为一名前端开发人员，我到底应该学什么，怎么学。毕业后两年多的时间里，我从使用原生的HTML/CSS/JavaScript，到现在流行的视图库Vue/React，看似我好像学会了很多，但是离开这些库，或者说将来这些库被淘汰了，我又剩下了什么呢？

我逐渐意识到，我需要的可能是一些更“底层”的东西。

那么,什么是“底层”的东西呢，这里可不是指计算机底层，而是指不会被淘汰的东西，永远有用的东西。我认为对于任何程序员来说，最重要的就是：迅速获取某项知识的能力以及动手解决问题的能力。

前者需要我们建立自己的知识体系，也就是winter老师在第一讲中提到的，我们需要完善的知识架构。

我们不需要也不可能记住整个架构的知识，但我们却可以在需要的时候想起来这块知识的位置，是干什么的，能帮我们解决什么样的问题，可以通过什么关键词检索到，这就够了。

其实你让我回忆winter老师这么多节课的所有内容，我也是好多想不起来的，但多少都留下了印象。我们的大脑容量是有限的，所以“按需加载”也许对人脑也是个不错的策略。

后者其实就归功于winter老师精心设计的课后题，先看这些课文总归是差了点什么，只有你动手去做，才能真正理解这节课所讲的内容。

在做的过程中，你多半会遇到困难（这其实也是winter老师的本意），如果你通过努力解决掉了，这个提升才是最大的。你的方案不一定是最好的，或者你可能实在没什么头绪，这时候也可以利用搜索引擎去看看别人的方案。

因为我早上到公司要坐一个半小时左右的公交，路上我基本上会用耳机听完课（不伤眼），然后剩下的时间去思考回顾一下，想想作业怎么完成。

在中午吃完饭休息的时候，动手做一下，有时候时间不一定够，就利用晚上和第二天的时间完成。每个人的情况都不一样，不过，想要有收获，就得付出多一些。

希望大家都学有所成，有所收获。

极客时间用户@峰峰峰 分享

先感谢极客时间给了我这次学习分享的机会，如果文中有不足之处，希望小伙伴们可以多多指导。

我是转行过来学习前端的，目前在前端这个岗位上工作了大概一年半的样子，最初知道重学前端这个专栏，也是因为朋友圈里面的一些大牛分享的，比较好奇，并且老师又是非常有名的winter老师，就报了这门课程。

我本人是对前端的技术非常感兴趣的，尤其是原生基础知识，因为我觉得一个人基础的扎实程度，能决定这个人在这个领域所能发展的高度。

刚开始看到课程目录的时候我很吃惊，因为这个目录和我之前所学的所有的课的目录都不一样，从文法、运行时的角度去解释JavaScript这门语言，这让我非常兴奋，因为我知道，这正是我所需要的。

通过跟随winter老师不断学习，我发现这把我原有的一些知识结构打乱了，但这并不是坏事，因为我以前的学习和知识结构都非常的混乱。

通过这次学习，让我对JavaScript的理解更加深入了一些，同时，重新梳理我的知识点也让我的知识结构更加的清晰了起来。

可能是我工作时间尚短的原因吧，我觉得老师的每一节课对于我来说都是一个新的东西。举个例子，就拿第3讲和第4讲来说吧，标签语义化我一直都知道，但是我在工作中用得很少，并且在理解上也只停留在非常浅的层面上，但是通过老师的讲解，让我理解到了语义化标签真正的含义。

我现在在工作都会开始使用一些我比较熟悉的语义化标签。包括一些浏览器的工作原理，关于JavaScript这门语言是面向对象还是面向类的解释，以及JavaScript的执行等等，这些知识点都让我受益良多，非常感谢winter老师，希望以后还能有机会跟着老师继续学习。

最后，分享一下我的学习方法吧。

1. 先提升知识的广度，再去拓展知识的深度。
2. 养成随时记录的习惯，可以是笔记，可以是代码（一定要写上注释）。
3. 多逛逛技术论坛，有条件的情况下多去实验一下论坛中自己感兴趣的代码。
4. 最后一条也是最重要的一条，坚持每天八小时工作外的学习和锻炼。

以上就是我的学习心得，不足之处还请多多见谅，希望对小伙伴们有帮助。

极客时间用户@木易杨 分享

我看了老师在“极客live”的直播，总结了他提到的一些观点。

1. 对于框架的使用没必要花太多时间，应该多研究一下三大框架背后的设计思想。
2. 当一个程序员对算法、语言标准、底层、原生、英文文档这些词汇产生恐惧感的时候，他的技术生命已经走到尽头。
3. 前端架构主要解决的是高复用性，架构能力提升方向主要是组件库开发、前端框架实现等。
4. 对于前端进阶这个问题，其实看书的作用和意义已经不太明显，需要寻找好的平台和合适的项目，在项目中不断克服难题并挑战自己，遇到问题再去查资料总结。如果只是闭门看书那很难成为高手，书只是基础而已，真正的应用还是在项目中。
5. 寒冬中能做的只有提升自己，但是光靠技术是不行的。
6. 推荐 TensorFlow、可视化切图、PWA、WebGL。
 - TensorFlow 可以了解使用并做点东西出来，原理很难但不影响使用。
 - PWA 有望进一步发展。
 - WebGL 在未来会是一个很好的方向，它可以实现任何你想要的界面效果，但重点需要多掌握图形学的基础知识，它和算法，数据结构一样重要。
 - Weex和 RN 虽然都叫 Hybrid 但不太一样，前者适合大厂主要是嵌入 APP 中使用，后者更适合创业公司。
7. 不推荐 SSR、TypeScript、函数式编程。
 - SSR 不太看好，其主要是用于SEO，不太建议用做服务端渲染，其能够使用的场景不多，而且成本代价太大。
 - TypeScript 是好东西，是很有前景的语言，但适用于十万行以上代码级别的大型项目，小项目并不适合，反而使徒复杂。
 - 用 JavaScript 做函数式编程并不靠谱，Map/Reduce/Redux/Hooks等并不是函数式编程，只是长得像而已。

PS：笔记内容由自己和群友提供，仅供参考。

我的一些思考和尝试

听完 winter 老师直播后对其观点很是赞同，但因为是面对所有人讲解，所以内容有点多范围有些广，但对个人来说还是找适合自己的方向，在自己相对熟悉的领域再去扩展去突破。横向只是拓宽你的眼界，纵向才是你的核心竞争力。

对我来说感触最大的就是3、4、5这几点，因为我一直在思考以下几个问题。

1. 我现在是高级前端，但又感觉自身很弱，那我如何才能夯实我高级的地基然后成为资深前端呢？
2. 项目迭代节奏快日常加班又多，那我如何做才能平衡工作和学习？
3. 单单提升技术好像还是有很大瓶颈，那我如何提升我的职场核心竞争力？

结合 winter老师的专栏和最近看的几篇文章，说说我对这几个问题的思考。

对于第一个问题，在2个多月前我还是不会写文章的小白，那个时候受到一些文章的触动，尝试把我的前端之路记录下来。

我的想法很简单，就是把前端进阶 28 期的重难点知识全部讲完，目前已经进行到第 4 期了，通过写作把知识通俗易懂的介绍给别人，在这个过程自己肯定会收获很多很多。

这就是我目前在尝试的学习方式，通过写作建立自己的知识架构，并且在这个架构上不断地进行优化，时间到了自然就进阶了。

对于第二个问题，刚开始写作时精力充沛时间也很多，每天都会更新技术文章，但随着项目迭代压力增大，文章更新速度相应就变慢了一些。

虽然文章质量提升了很多，但更新速度从日更变成了周更再变成了双周更，这个说实话我自身已经很难接受了，因为速度变慢导致我年初的计划要打折扣。

所以说做项目和自我学习要如何权衡，是否说我要减少项目难度和时间并在工作中摸鱼去学习呢？

我的结论是不，因为一句话，“最好的学习就是在项目中锻炼自己”。既然我有这么好的项目去锻炼，那为什么还要摸鱼去学习呢，这不就是南辕北辙嘛，道理很简单但不是所有人都懂。

我最近在执行的方法是专注 + 锻炼 + 利用周末。

专注即工作时专注于工作，努力做好每次迭代，遇到难题迎难而上，工作时不开微信，勤用笔记安排日常工作并整理文档。

锻炼即一周抽出三天时间每次去健身房锻炼1小时，强壮的体魄才能撑住高强度的工作和学习，因为网易有健身房所以冬天锻炼也没什么问题，没办法去健身房的小伙伴可以尝试跑步或者在瑜伽垫上做Keep。

利用周末即加大周末和平常晚上熬夜的时间去写文章，减少娱乐的时间，正所谓时间挤一挤总会有的，但这件事情比较反人性，所以重在坚持。公众号写文章更容易坚持，原因在于文章发布后有正反馈，比如粉丝的增长，留言和鼓励，赞赏等等，这些都更能促使你坚持下去，时间长了自然就养成了习惯。

对于第三个问题，winter 老师也说了，寒冬中光靠技术是不行的，那应该靠什么呢？幸好在最近看到的一篇文章中找到了答案，那就是表达能力。

表达能力是形成自己的框架系统，有理有据并且逻辑清晰，而且能让外人听懂，大部分优秀的人都具备这样的能力。反观自己并没有这样的能力，所以我要努力提升这块，其中最重要的方法就是写作训练。

兜兜转转那么多，其实我一直在尝试的进阶方法就是通过写作建立自己的知识架构体系，同时提高自身的表达能力，通过正反馈机制和锻炼保证我长久的坚持下去并最终养成习惯，习惯这一模式之后变成优秀的人也只是时间问题罢了。

希望我的一些思考和尝试能对你有所帮助。

（直播回顾：<https://v.qq.com/x/page/x0830f8xoj.html>）

恭喜幸运用户 @阿成、@峰峰峰、@木易杨，你们的经验被选取成为本期的用户故事。极客时间将送出“极客时间”& winter联名T恤一件。1个工作日之内，工作人员会与你取得联系。

读完了三位同学的故事后，你有哪些感受呢，留下你的收获，我们一起成长。

写在前面的话

不知不觉，专栏已经更新了一半有余了。在两个月的时间里，我收到了很多同学的留言，很开心可以跟大家保持交流，也很惊喜见证了大家的成长。

在整个学习过程中，你有没有一些学习感悟呢，有哪些新的学习收获，又有哪些实践感悟？有没有对前端的知识架构有了更为深刻的认识，是否规划好了自己的前端之路。

如果你有想分享的经验，可以在文章下留言，如果你的留言被选中作为用户故事，我们将会为你送出“极客时间”& winter联名T恤一件。分享你的收获，我们一起成长。



下面我们一起来看看三位同学的分享。

极客时间用户@阿成 分享

在我进入知乎社区的时候，winter 老师就已经不在这个社区了，虽然是退隐了江湖，但是江湖上还流传着他的传说，后来我在 Google Feed 中看到 winter 老师的开篇词，看完就感觉是我要的东西，还正好赶上打折，就毫不犹豫地买了。

我本科学的是光电信息，所以并不算是科班出身的程序员（虽然也有学C啦）。我本身对计算机程序一直很有兴趣，所以从大三开始，我就决定放弃从事本专业，做一名程序员。要说为什么选择前端领域，自然也是烂俗的“门槛低”理由。

不过，我没想到的是，前端的学习曲线到后面是如此陡峭，并且前端发展的速度如此之快，让人目不暇接，也让很多人产生了“技术焦虑”。

在这样的环境中，我也产生了迷茫，作为一名前端开发人员，我到底应该学什么，怎么学。毕业后两年多的时间里，我从使用原生的HTML/CSS/JavaScript，到现在流行的视图库Vue/React，看似我好像学会了很多，但是离开这些库，或者说将来这些库被淘汰了，我又剩下了什么呢？

我逐渐意识到，我需要的可能是一些更“底层”的东西。

那么，什么是“底层”的东西呢，这里可不是指计算机底层，而是指不会被淘汰的东西，永远有用的东西。我认为对于任何程序员来说，最重要的就是：迅速获取某项知识的能力以及动手解决问题的能力。

前者需要我们建立自己的知识体系，也就是winter老师在第一讲中提到的，我们需要完善的知识架构。

我们不需要也不可能记住整个架构的知识，但我们却可以在需要的时候想起来这块知识的位置，是干什么的，能帮我们解决什么样的问题，可以通过什么关键词检索到，这就够了。

其实你让我回忆winter老师这么多节课的所有内容，我也是好多想不起来的，但多少都留下了印象。我们的大脑容量是有限的，所以“按需加载”也许对人脑也是个不错的策略。

后者其实就归功于winter老师精心设计的课后题，光看这些课文总归是差了点什么，只有你动手去做，才能真正理解这节课所讲的内容。

在做过程中，你多半会遇到困难（这其实也是winter老师的本意），如果你通过努力解决掉了，这个提升才是最大的。你的方案不一定是最好的，或者你可能实在没什么头绪，这时候也可以利用搜索引擎去看看别人的方案。

因为我早上到公司要坐一个半小时左右的公交，路上我基本上会用耳机听完课（不伤眼），然后剩下的时间去思考回顾一下，想想作业怎么完成。

在中午吃完饭休息的时候，动手做一下，有时候时间不一定够，就利用晚上和第二天的时间完成。每个人的情况都不一样，不过，想要有收获，就得付出多一些。

希望大家都能学有所成，有所收获。

极客时间用户@峰峰峰 分享

先感谢极客时间给了我这次学习分享的机会，如果文中有不足之处，希望小伙伴们可以多多指导。

我是转行过来学习前端的，目前在前端这个岗位上工作了大概一年半的样子，最初知道重学前端这个专栏，也是因为朋友圈里面的一些大牛分享的，比较好奇，并且老师又是非常有名的winter老师，就报了这门课程。

我本人是对前端的技术非常感兴趣的，尤其是原生基础知识，因为我觉得一个人基础的扎实程度，能决定这个人这个领域所能发展的高度。

刚开始看到课程目录的时候我很吃惊，因为这个目录和我之前所学的所有的课的目录都不一样，从文法、运行时的角度去解释JavaScript这门语言，这让我非常兴奋，因为我知道，这正是我所需要的。

通过跟随winter老师不断学习，我发现这把我有的一些知识结构打乱了，但这并不是坏事，因为我以前的学习和知识结构都非常的混乱。

通过这次学习，让我对JavaScript的理解更加深入了一些，同时，重新梳理我的知识点也让我的知识结构更加的清晰了起来。

可能是我工作时间尚短的原因吧，我觉得老师的每一节课对于我来说都是一个新的东西。举个例子，就拿第3讲和第4讲来说吧，标签语义化我一直都知道，但是我在工作中用得很少，并且在理解上也只停留在非常浅的层面上，但是通过老师的讲解，让我理解到了语义化标签真正的含义。

我现在在工作都会开始使用一些我比较熟悉的语义化标签。包括一些浏览器的工作原理，关于JavaScript这门语言是面向对象还是面向类的解释，以及JavaScript的执行等等，这些知识点都让我受益良多，非常感谢winter老师，希望以后还能有机会跟着老师继续学习。

最后，分享一下我的学习方法吧。

1. 先提升知识的广度，再去拓展知识的深度。
2. 养成随时记录的习惯，可以是笔记，可以是代码（一定要写上注释）。
3. 多逛逛技术论坛，有条件情况下多去实验一下论坛中自己感兴趣的代码。
4. 最后一条也是最重要的一条，坚持每天八小时工作外的学习和锻炼。

以上就是我的学习心得，不足之处还请多多见谅，希望对小伙伴们有帮助。

极客时间用户@木易杨 分享

我看了老师在“极客live”的直播，总结了他提到的一些观点。

1. 对于框架的使用没必要花太多时间，应该多研究一下三大框架背后的设计思想。
2. 当一个程序员对算法、语言标准、底层、原生、英文文档这些词汇产生恐惧感的时候，他的技术生命已经走到尽头。
3. 前端架构主要解决的是高复用性，架构能力提升方向主要是组件库开发、前端框架实现等。
4. 对于前端进阶这个问题，其实看书的作用和意义已经不太明显，需要寻找好的平台和合适的项目，在项目中不断克服难题并挑战自己，遇到问题再去查资料总结。如果只是闭门看书那很难成为高手，书只是基础而已，真正的应用还是在项目中。
5. 寒冬中能做的只有提升自己，但是光靠技术是不行的。
6. 推荐 TensorFlow、可视化切图、PWA、WebGL。
 - TensorFlow 可以了解使用并做点东西出来，原理很难但不影响使用。
 - PWA 有望进一步发展。
 - WebGL 在未来会是一个很好的方向，它可以实现任何你想要的界面效果，但重点需要多掌握图形学的基础知识，它和算法，数据结构一样重要。
 - Weex和 RN 虽然都叫 Hybrid 但不太一样，前者适合大厂主要是嵌入 APP 中使用，后者更适合创业公司。
7. 不推荐 SSR、TypeScript、函数式编程。
 - SSR 不太看好，其主要是用于SEO，不太建议用做服务端渲染，其能够使用的场景不多，而且成本代价太大。
 - TypeScript 是好东西，是很有前景的语言，但适用于十万行以上代码级别的大型项目，小项目并不适合，反而使逻辑复杂。
 - 用 JavaScript 做函数式编程并不靠谱，Map/Reduce/Redux/Hooks等并不是函数式编程，只是长得像而已。

PS：笔记内容由自己和群友提供，仅供参考。

我的一些思考和尝试

听完 winter 老师直播后对其观点很是赞同，但因为是面对所有人讲解，所以内容有点多范围有些广，但对个人来说还是找适合自己的方向，在自己相对熟悉的领域再去扩展去突破。横向只是拓宽你的眼界，纵向才是你的核心竞争力。

对我来说感触最大的就是 3、4、5 这几点，因为我一直在思考以下几个问题。

1. 我现在是高级前端，但又感觉自身很弱，那我如何才能夯实我高级的地基然后成为资深前端呢？
2. 项目迭代节奏快日常加班又多，那我如何做才能平衡工作和学习？
3. 单单提升技术好像还是有很大瓶颈，那我如何提升我的职场核心竞争力？

结合 winter 老师的专栏和最近看的几篇文章，说说我对这几个问题的思考。

对于第一个问题，在 2 个多月前我还是不会写文章的小白，那个时候受到一些文章的触动，尝试把我的前端之路记录下来。

我的想法很简单，就是把前端进阶 28 期的重难点知识全部讲完，目前已经进行到第 4 期了，通过写作把知识通俗易懂的介绍给别人，在这个过程中自己肯定会收获很多很多。

这就是我目前在尝试的学习方式，通过写作建立自己的知识架构，并且在这个架构上不断地进行优化，时间到了自然就进阶了。

对于第二个问题，刚开始写作时精力充沛时间也很多，每天都会更新技术文章，但随着项目迭代压力增大，文章更新速度相应就变慢了一些。

虽然文章质量提升了很多，但更新速度从日更变成了周更再变成了双周更，这个说实话我自己已经很难接受了，因为速度变慢导致我年初的计划要打折扣。

所以说做项目和自我学习要如何权衡，是否说我要减少项目难度和时间并在工作摸鱼去学习呢？

我的结论是不，因为一句话，“最好的学习就是在项目中锻炼自己”。既然我有这么好的项目去锻炼，那为什么还要摸鱼去学习呢，这不就是南辕北辙嘛，道理很简单但不是所有人都懂。

我最近在执行的方法是专注 + 锻炼 + 利用周末。

专注即工作时专注于工作，努力做好每次迭代，遇到难题迎难而上，工作时不开微信，勤用笔记安排日常工作并整理文档。

锻炼即一周抽出三天时间每次去健身房锻炼1小时，强壮的体魄才能撑住高强度的工作和学习，因为网易有健身房所以冬天锻炼也没什么问题，没办法去健身房的小伙伴可以尝试跑步或者在瑜伽垫上做Keep。

利用周末即加大周末和平常晚上熬夜的时间去写文章，减少娱乐的时间，正所谓时间挤一挤总会有的，但这件事情比较反人性，所以重在坚持。公众号写文章更容易坚持，原因在于文章发布后有正反馈，比如粉丝的增长，留言和鼓励，赞赏等等，这些都更能促使你坚持下去，时间长了自然就养成了习惯。

对于第三个问题，winter 老师也说了，寒冬中光靠技术是不行的，那应该靠什么呢？幸好在最近看到的一篇文章中找到了答案，那就是表达能力。

表达能力是形成自己的框架系统，有理有据并且逻辑清晰，而且能让外人听懂，大部分优秀的人都具备这样的能力。反观自己并没有这样的能力，所以我要努力提升这块，其中最重要的方法就是写作训练。

兜兜转转那么多，其实我一直在尝试的进阶方法就是通过写作建立自己的知识架构体系，同时提高自身的表达能力，通过正反馈机制和锻炼保证我长久的坚持下去并最终养成习惯，习惯这一模式之后变

成优秀的人也只是时间问题罢了。

希望我的一些思考和尝试能对你有所帮助。

（直播回顾：<https://v.qq.com/x/page/x0830f68xoj.html>）

恭喜幸运用户 @阿成、@峰峰峰、@木易杨，你们的经验被选取成为本期的用户故事。极客时间将送出“极客时间”& winter联名T恤一件。1个工作日之内，工作人员会与你取得联系。

读完了三位同学的故事后，你有哪些感受呢，留下你的收获，我们一起成长。

写在前面的话

不知不觉，专栏已经更新了一半有余了。在两个月的时间里，我收到了很多同学的留言，很开心可以跟大家保持交流，也很惊喜见证了大家的成长。

在整个学习过程中，你有没有一些学习感悟呢，有哪些新的学习收获，又有哪些实践感悟？有没有对前端的知识架构有了更为深刻的认识，是否规划好了自己的前端之路。

如果你有想分享的经验，可以在文章下留言，如果你的留言被选中作为用户故事，我们将会为你送出“极客时间”& winter联名T恤一件。分享你的收获，我们一起成长。



下面我们一起来看看三位同学的分享。

极客时间用户@阿成 分享

在我进入知乎社区的时候，winter 老师就已经不在这个社区了，虽然是退隐了江湖，但是江湖上还流传着他的传说，后来我在 Google Feed 中看到 winter 老师的开篇词，看完就感觉是我要的东西，还正好赶上打折，就毫不犹豫地买了。

我本科学的是光电信息，所以并不算是科班出身的程序员（虽然也有学C啦）。我本身对计算机程序一直很有兴趣，所以从大三开始，我就决定放弃从事本专业，做一名程序员。要说为什么选择前端领域，自然也是烂俗的“门槛低”理由。

不过，我没想到的是，前端的学习曲线到后面是如此陡峭，并且前端发展的速度如此之快，让人目不暇接，也让很多人产生了“技术焦虑”。

在这样的环境中，我也产生了迷茫，作为一名前端开发人员，我到底应该学什么，怎么学。毕业后两年多的时间里，我从使用原生的HTML/CSS/JavaScript，到现在流行的视图库Vue/React，看似我好像学会了很多，但是离开这些库，或者说将来这些库被淘汰了，我又剩下了什么呢？

我逐渐意识到，我需要的可能是一些更“底层”的东西。

那么，什么是“底层”的东西呢，这里可不是指计算机底层，而是指不会被淘汰的东西，永远有用的东西。我认为对于任何程序员来说，最重要的就是：迅速获取某项知识的能力以及动手解决问题的能力。

前者需要我们建立自己的知识体系，也就是winter老师在第一讲中提到的，我们需要完善的知识架构。

我们不需要也不可能记住整个架构的知识，但我们却可以在需要的时候想起来这块知识的位置，是干什么的，能帮我们解决什么样的问题，可以通过什么关键词检索到，这就够了。

其实你让我回忆winter老师这么多节课的所有内容，我也是好多想不起来的，但多少都留下了印象。我们的大脑容量是有限的，所以“按需加载”也许对人脑也是个不错的策略。

后者其实就归功于winter老师精心设计的课后题，先看这些课文总归是差了点什么，只有你动手去做，才能真正理解这节课所讲的内容。

在做的过程中，你多半会遇到困难（这其实也是winter老师的本意），如果你通过努力解决掉了，这个提升才是最大的。你的方案不一定是最好的，或者你可能实在没什么头绪，这时候也可以利用搜索引擎去看看别人的方案。

因为我早上到公司要坐一个半小时左右的公交，路上我基本上会用耳机听完课（不伤眼），然后剩下的时间去思考回顾一下，想想作业怎么完成。

在中午吃完饭休息的时候，动手做一下，有时候时间不一定够，就利用晚上和第二天的时间完成。每个人的情况都不一样，不过，想要有收获，就得付出多一些。

希望大家都能学有所成，有所收获。

极客时间用户@峰峰峰 分享

先感谢极客时间给了我这次学习分享的机会，如果文中有不足之处，希望小伙伴们可以多多指导。

我是转行过来学习前端的，目前在前端这个岗位上工作了大概一年半的样子，最初知道重学前端这个专栏，也是因为朋友圈里面的一些大牛分享的，比较好奇，并且老师又是非常有名的winter老师，就报了这门课程。

我本人是对前端的技术非常感兴趣的，尤其是原生基础知识，因为我覺得一个人基础的扎实程度，能决定这个人在这个领域所能发展的高度。

刚开始看到课程目录的时候我很吃惊，因为这个目录和我之前所学的所有的课的目录都不一样，从文法、运行时的角度去解释JavaScript这门语言，这让我非常兴奋，因为我知道，这正是我所需要的。

通过跟随winter老师不断学习，我发现这把我有的一些知识结构打乱了，但这并不是坏事，因为我以前的学习和知识结构都非常的混乱。

通过这次学习，让我对JavaScript的理解更加深入了一些，同时，重新梳理我的知识点也让我的知识结构更加的清晰了起来。

可能是我工作时间尚短的原因吧，我觉得老师的每一节课对于我来说都是一个新的东西。举个例子，就拿第3讲和第4讲来说吧，标签语义化我一直都知道，但是我在工作中用得很少，并且在理解上也只停留在非常浅的层面上，但是通过老师的讲解，让我理解到了语义化标签真正的含义。

我现在在工作都会开始使用一些我比较熟悉的语义化标签。包括一些浏览器的工作原理，关于JavaScript这门语言是面向对象还是面向类的解释，以及JavaScript的执行等等，这些知识点都让我受益良多，非常感谢winter老师，希望以后还能有机会跟着老师继续学习。

最后，分享一下我的学习方法吧。

1. 先提升知识的广度，再去拓展知识的深度。
2. 养成随时记录的习惯，可以是笔记，可以是代码（一定要写上注释）。
3. 多逛逛技术论坛，有条件的情况下多去实验一下论坛中自己感兴趣的代码。
4. 最后一条也是最重要的一条，坚持每天八小时工作外的学习和锻炼。

以上就是我的学习心得，不足之处还请多多见谅，希望对小伙伴们有帮助。

极客时间用户@木易杨 分享

我看了老师在“极客live”的直播，总结了他提到的一些观点。

1. 对于框架的使用没必要花太多时间，应该多研究一下三大框架背后的设计思想。
2. 当一个程序员对算法、语言标准、底层、原生、英文文档这些词汇产生恐惧感的时候，他的技术生命已经走到尽头。
3. 前端架构主要解决的是高复用性，架构能力提升方向主要是组件库开发、前端框架实现等。
4. 对于前端进阶这个问题，其实看书的作用和意义已经不太明显，需要寻找好的平台和合适的项目，在项目中不断克服难题并挑战自己，遇到问题再去查资料总结。如果只是闭门看书那很难成为高手，书只是基础而已，真正的应用还是在项目中。
5. 寒冬中能做的只有提升自己，但是光靠技术是不行的。
6. 推荐 TensorFlow、可视化切图、PWA、WebGL。
 - TensorFlow 可以了解使用并做点东西出来，原理很难但不影响使用。
 - PWA 有望进一步发展。
 - WebGL 在未来会是一个很好的方向，它可以实现任何你想要的界面效果，但重点需要多掌握图形学的基础知识，它和算法，数据结构一样重要。
 - Weex和 RN 虽然都叫 Hybrid 但不太一样，前者适合大厂主要是嵌入 APP 中使用，后者更适合创业公司。
7. 不推荐 SSR、TypeScript、函数式编程。
 - SSR 不太看好，其主要用于SEO，不太建议用做服务端渲染，其能够使用的场景不多，而且成本代价太大。
 - TypeScript 是好东西，是很有前景的语言，但适用于十万行以上代码级别的大型项目，小项目并不适合，反而徒增复杂。
 - 用 JavaScript 做函数式编程并不靠谱，Map/Reduce/Redux/Hooks等并不是函数式编程，只是长得像而已。

PS：笔记内容由自己和群友提供，仅供参考。

我的一些思考和尝试

听完 winter 老师直播后对其观点很是赞同，但因为是面对所有人讲解，所以内容有点多范围有些广，但对个人来说还是找适合自己的方向，在自己相对熟悉的领域再去扩展去突破。横向只是拓宽你的眼界，纵向才是你的核心竞争力。

对我来说感触最大的就是 3、4、5 这几点，因为我一直在思考以下几个问题。

1. 我现在是高级前端，但又感觉自身很弱，那我如何才能夯实我高级的地基然后成为资深前端呢？
2. 项目迭代节奏快日常加班又多，那我如何做才能平衡工作和学习？
3. 单单提升技术好像还是有很大瓶颈，那我如何提升我的职场核心竞争力？

结合 winter 老师的专栏和最近看的几篇文章，说说我对这几个问题的思考。

对于第一个问题，在 2 个多月前我还是不会写文章的小白，那个时候受到一些文章的触动，尝试把我的前端之路记录下来。

我的想法很简单，就是把前端进阶 28 期的重难点知识全部讲完，目前已经进行到第 4 期了，通过写作把知识通俗易懂的介绍给别人，在这个过程自己肯定会收获很多很多。

这就是我目前在尝试的学习方式，通过写作建立自己的知识架构，并且在这个架构上不断地进行优化，时间到了自然就进阶了。

对于第二个问题，刚开始写作时精力充沛时间也很多，每天都会更新技术文章，但随着项目迭代压力增大，文章更新速度相应就变慢了一些。

虽然文章质量提升了很多，但更新速度从日更变成了周更再变成了双周更，这个说实话我自己已经很难接受了，因为速度变慢导致我年初的计划要打折扣。

所以说做项目和自我学习要如何权衡，是否说我要减少项目难度和时间并在工作中摸鱼去学习呢？

我的结论是不，因为一句话，“最好的学习就是在项目中锻炼自己”。既然我有这么好的项目去锻炼，那为什么还要摸鱼去学习呢，这不就是南辕北辙嘛，道理很简单但不是所有人都懂。

我最近在执行的方法是专注 + 锻炼 + 利用周末。

专注即工作时专注于工作，努力做好每次迭代，遇到难题迎难而上，工作时不开微信，勤用笔记安排日常工作并整理文档。

锻炼即一周抽出三天时间每次去健身房锻炼1小时，强壮的体魄才能撑住高强度的工作和学习，因为网易有健身房所以冬天锻炼也没什么问题，没办法去健身房的小伙伴可以尝试跑步或者在瑜伽垫上做Keep。

利用周末即加大周末和平常晚上熬夜的时间去写文章，减少娱乐的时间，正所谓时间挤一挤总会有的，但这件事情比较反人性，所以重在坚持。公众号写文章更容易坚持，原因在于文章发布后有正反馈，比如粉丝的增长，留言和鼓励，赞赏等等，这些都更能促使你坚持下去，时间长了自然就养成了习惯。

对于第三个问题，winter 老师也说了，寒冬中光靠技术是不行的，那应该靠什么呢？幸好在最近看到的一篇文章中找到了答案，那就是表达能力。

表达能力是形成自己的框架系统，有理有据并且逻辑清晰，而且能让外人听懂，大部分优秀的人都具备这样的能力。反观自己并没有这样的能力，所以我要努力提升这块，其中最重要的方法就是写作训练。

兜兜转转那么多，其实我一直在尝试的进阶方法就是通过写作建立自己的知识架构体系，同时提高自身的表达能力，通过正反馈机制和锻炼保证我长久的坚持下去并最终养成习惯，习惯这一模式之后变成优秀的人也只是时间问题罢了。

希望我的一些思考和尝试能对你有所帮助。

（直播回顾：<https://v.qq.com/x/page/x0830f8xoj.html>）

恭喜幸运用户 @阿成、@峰峰峰、@木易杨，你们的经验被选取成为本期的用户故事。极客时间将送出“极客时间”& winter联名T恤一件。1个工作日之内，工作人员会与你取得联系。

读完了三位同学的故事后，你有哪些感受呢，留下你的收获，我们一起成长。

写在前面的话

不知不觉，专栏已经更新了一半有余了。在两个月的时间里，我收到了很多同学的留言，很开心可以跟大家保持交流，也很惊喜见证了大家的成长。

在整个学习过程中，你有没有一些学习感悟呢，有哪些新的学习收获，又有哪些实践感悟？有没有对前端的知识架构有了更为深刻的认识，是否规划好了自己的前端之路。

如果你有想分享的经验，可以在文章下留言，如果你的留言被选中作为用户故事，我们将会为你送出“极客时间”& winter联名T恤一件。分享你的收获，我们一起成长。



下面我们一起来看看三位同学的分享。

极客时间用户@阿成 分享

在我进入知乎社区的时候，winter 老师就已经不在这个社区了，虽然是退隐了江湖，但是江湖上还流传着他的传说，后来我在 Google Feed 中看到 winter 老师的开篇词，看完就感觉是我要的东西，还正好赶上打折，就毫不犹豫地买了。

我本科学的是光电信息，所以并不算是科班出身的程序员（虽然也有学C啦）。我本身对计算机程序一直很有兴趣，所以从大三开始，我就决定放弃从事本专业，做一名程序员。要说为什么选择前端领域，自然也是烂俗的“门槛低”理由。

不过，我没想到的是，前端的学习曲线到后面是如此陡峭，并且前端发展的速度如此之快，让人目不暇接，也让很多人产生了“技术焦虑”。

在这样的环境中，我也产生了迷茫，作为一名前端开发人员，我到底应该学什么，怎么学。毕业后两年多的时间里，我从使用原生的HTML/CSS/JavaScript，到现在流行的视图库Vue/React，看似我好像学会了很多，但是离开这些库，或者说将来这些库被淘汰了，我又剩下了什么呢？

我逐渐意识到，我需要的可能是一些更“底层”的东西。

那么，什么是“底层”的东西呢，这里可不是指计算机底层，而是指不会被淘汰的东西，永远有用的东西。我认为对于任何程序员来说，最重要的就是：迅速获取某项知识的能力以及动手解决问题的能力。

前者需要我们建立自己的知识体系，也就是winter老师在第一讲中提到的，我们需要完善的知识架构。

我们不需要也不可能记住整个架构的知识，但我们却可以在需要的时候想起来这块知识的位置，是干什么的，能帮我们解决什么样的问题，可以通过什么关键词检索到，这就够了。

其实你让我回忆winter老师这么多节课的所有内容，我也是好多想不起来的，但多少都留下了印象。我们的大脑容量是有限的，所以“按需加载”也许对人脑也是个不错的策略。

后者其实就归功于winter老师精心设计的课后题，光看这些课文总归是差了点什么，只有你动手去做，才能真正理解这节课所讲的内容。

在做过程中，你多半会遇到困难（这其实也是winter老师的本意），如果你通过努力解决掉了，这个提升才是最大的。你的方案不一定是最好的，或者你可能实在没什么头绪，这时候也可以利用搜索引擎去看看别人的方案。

因为我早上到公司要坐一个半小时左右的公交，路上我基本上会用耳机听完课（不伤眼），然后剩下的时间去思考回顾一下，想想作业怎么完成。

在中午吃完饭休息的时候，动手做一下，有时候时间不一定够，就利用晚上和第二天的时间完成。每个人的情况都不一样，不过，想要有收获，就得付出多一些。

希望大家都能学有所成，有所收获。

极客时间用户@峰峰峰 分享

先感谢极客时间给了我这次学习分享的机会，如果文中有不足之处，希望小伙伴们可以多多指导。

我是转行过来学习前端的，目前在前端这个岗位上工作了大概一年半的样子，最初知道重学前端这个专栏，也是因为朋友圈里面的一些大牛分享的，比较好奇，并且老师又是非常有名的winter老师，就报了这门课程。

我本人是对前端的技术非常感兴趣的，尤其是原生基础知识，因为我觉得一个人基础的扎实程度，能决定这个人在这个领域所能发展的高度。

刚开始看到课程目录的时候我很吃惊，因为这个目录和我之前所学的所有的课的目录都不一样，从文法、运行时的角度去解释JavaScript这门语言，这让我非常兴奋，因为我知道，这正是我所需要的。

通过跟随winter老师不断学习，我发现这把我有的一些知识结构打乱了，但这并不是坏事，因为我以前的学习和知识结构都非常的混乱。

通过这次学习，让我对JavaScript的理解更加深入了一些，同时，重新梳理我的知识点也让我的知识结构更加的清晰了起来。

可能是我工作时间尚短的原因吧，我觉得老师的每一节课对于我来说都是一个新的东西。举个例子，就拿第3讲和第4讲来说吧，标签语义化我一直都知道，但是我在工作中用得很少，并且在理解上也只停留在非常浅的层面上，但是通过老师的讲解，让我理解到了语义化标签真正的含义。

我现在在工作都会开始使用一些我比较熟悉的语义化标签。包括一些浏览器的工作原理，关于JavaScript这门语言是面向对象还是面向类的解释，以及JavaScript的执行等等，这些知识点都让我受益良多，非常感谢winter老师，希望以后还能有机会跟着老师继续学习。

最后，分享一下我的学习方法吧。

1. 先提升知识的广度，再去拓展知识的深度。
2. 养成随时记录的习惯，可以是笔记，可以是代码（一定要写上注释）。
3. 多逛逛技术论坛，有条件情况下多去实验一下论坛中自己感兴趣的代码。
4. 最后一条也是最重要的一条，坚持每天八小时工作外的学习和锻炼。

以上就是我的学习心得，不足之处还请多多见谅，希望对小伙伴们有帮助。

极客时间用户@木易杨 分享

我看了老师在“极客live”的直播，总结了他提到的一些观点。

1. 对于框架的使用没必要花太多时间，应该多研究一下三大框架背后的设计思想。
2. 当一个程序员对算法、语言标准、底层、原生、英文文档这些词汇产生恐惧感的时候，他的技术生命已经走到尽头。
3. 前端架构主要解决的是高复用性，架构能力提升方向主要是组件库开发、前端框架实现等。
4. 对于前端进阶这个问题，其实看书的作用和意义已经不太明显，需要寻找好的平台和合适的项目，在项目中不断克服难题并挑战自己，遇到问题再去查资料总结。如果只是闭门看书那很难成为高手，书只是基础而已，真正的应用还是在项目中。
5. 寒冬中能做的只有提升自己，但是光靠技术是不行的。
6. 推荐 TensorFlow、可视化切图、PWA、WebGL。
 - TensorFlow 可以了解使用并做点东西出来，原理很难但不影响使用。
 - PWA 有望进一步发展。
 - WebGL 在未来会是一个很好的方向，它可以实现任何你想要的界面效果，但重点需要多掌握图形学的基础知识，它和算法，数据结构一样重要。
 - Weex和 RN 虽然都叫 Hybrid 但不太一样，前者适合大厂主要是嵌入 APP 中使用，后者更适合创业公司。
7. 不推荐 SSR、TypeScript、函数式编程。
 - SSR 不太看好，其主要是用于SEO，不太建议用做服务端渲染，其能够使用的场景不多，而且成本代价太大。
 - TypeScript 是好东西，是很有前景的语言，但适用于十万行以上代码级别的大型项目，小项目并不适合，反而会使复杂。
 - 用 JavaScript 做函数式编程并不靠谱，Map/Reduce/Redux/Hooks等并不是函数式编程，只是长得像而已。

PS：笔记内容由自己和群友提供，仅供参考。

我的一些思考和尝试

听完 winter 老师直播后对其观点很是赞同，但因为是面对所有人讲解，所以内容有点多范围有些广，但对个人来说还是找适合自己的方向，在自己相对熟悉的领域再去扩展去突破。横向只是拓宽你的眼界，纵向才是你的核心竞争力。

对我来说感触最大的就是 3、4、5 这几点，因为我一直在思考以下几个问题。

1. 我现在是高级前端，但又感觉自身很弱，那我如何才能夯实我高级的地基然后成为资深前端呢？
2. 项目迭代节奏快日常加班又多，那我如何做才能平衡工作和学习？
3. 单单提升技术好像还是有很大瓶颈，那我如何提升我的职场核心竞争力？

结合 winter 老师的专栏和最近看的几篇文章，说说我对这几个问题的思考。

对于第一个问题，在 2 个多月前我还是不会写文章的小白，那个时候受到一些文章的触动，尝试把我的前端之路记录下来。

我的想法很简单，就是把前端进阶 28 期的重难点知识全部讲完，目前已经进行到第 4 期了，通过写作把知识通俗易懂的介绍给别人，在这个过程中自己肯定会收获很多很多。

这就是我目前在尝试的学习方式，通过写作建立自己的知识架构，并且在这个架构上不断地进行优化，时间到了自然就进阶了。

对于第二个问题，刚开始写作时精力充沛时间也很多，每天都会更新技术文章，但随着项目迭代压力增大，文章更新速度相应就变慢了一些。

虽然文章质量提升了很多，但更新速度从日更变成了周更再变成了双周更，这个说实话我自己已经很难接受了，因为速度变慢导致我年初的计划要打折扣。

所以说做项目和自我学习要如何权衡，是否说我要减少项目难度和时间并在工作摸鱼去学习呢？

我的结论是不，因为一句话，“最好的学习就是在项目中锻炼自己”。既然我有这么好的项目去锻炼，那为什么还要摸鱼去学习呢，这不就是南辕北辙嘛，道理很简单但不是所有人都懂。

我最近在执行的方法是专注 + 锻炼 + 利用周末。

专注即工作时专注于工作，努力做好每次迭代，遇到难题迎难而上，工作时不开微信，勤用笔记安排日常工作并整理文档。

锻炼即一周抽出三天时间每次去健身房锻炼1小时，强壮的体魄才能撑住高强度的工作和学习，因为网易有健身房所以冬天锻炼也没什么问题，没办法去健身房的小伙伴可以尝试跑步或者在瑜伽垫上做Keep。

利用周末即加大周末和平常晚上熬夜的时间去写文章，减少娱乐的时间，正所谓时间挤一挤总会有的，但这件事情比较反人性，所以重在坚持。公众号写文章更容易坚持，原因在于文章发布后有正反馈，比如粉丝的增长，留言和鼓励，赞赏等等，这些都更能促使你坚持下去，时间长了自然就养成了习惯。

对于第三个问题，winter 老师也说了，寒冬中光靠技术是不行的，那应该靠什么呢？幸好在最近看到的一篇文章中找到了答案，那就是表达能力。

表达能力是形成自己的框架系统，有理有据并且逻辑清晰，而且能让外人听懂，大部分优秀的人都具备这样的能力。反观自己并没有这样的能力，所以我要努力提升这块，其中最重要的方法就是写作训练。

兜兜转转那么多，其实我一直在尝试的进阶方法就是通过写作建立自己的知识架构体系，同时提高自身的表达能力，通过正反馈机制和锻炼保证我长久的坚持下去并最终养成习惯，习惯这一模式之后变

成优秀的人也只是时间问题罢了。

希望我的一些思考和尝试能对你有所帮助。

（直播回顾：<https://v.qq.com/x/page/x0830f68xoj.html>）

恭喜幸运用户 @阿成、@峰峰峰、@木易杨，你们的经验被选取成为本期的用户故事。极客时间将送出“极客时间”& winter联名T恤一件。1个工作日之内，工作人员会与你取得联系。

读完了三位同学的故事后，你有哪些感受呢，留下你的收获，我们一起成长。

写在前面的话

不知不觉，专栏已经更新了一半有余了。在两个月的时间里，我收到了很多同学的留言，很开心可以跟大家保持交流，也很惊喜见证了大家的成长。

在整个学习过程中，你有没有一些学习感悟呢，有哪些新的学习收获，又有哪些实践感悟？有没有对前端的知识架构有了更为深刻的认识，是否规划好了自己的前端之路。

如果你有想分享的经验，可以在文章下留言，如果你的留言被选中作为用户故事，我们将会为你送出“极客时间”& winter联名T恤一件。分享你的收获，我们一起成长。



下面我们一起来看看三位同学的分享。

极客时间用户@阿成 分享

在我进入知乎社区的时候，winter 老师就已经不在这个社区了，虽然是退隐了江湖，但是江湖上还流传着他的传说，后来我在 Google Feed 中看到 winter 老师的开篇词，看完就感觉是我要的东西，还正好赶上打折，就毫不犹豫地买了。

我本科学的是光电信息，所以并不算是科班出身的程序员（虽然也有学C啦）。我本身对计算机程序一直很有兴趣，所以从大三开始，我就决定放弃从事本专业，做一名程序员。要说为什么选择前端领域，自然也是烂俗的“门槛低”理由。

不过，我没想到的是，前端的学习曲线到后面是如此陡峭，并且前端发展的速度如此之快，让人目不暇接，也让很多人产生了“技术焦虑”。

在这样的环境中，我也产生了迷茫，作为一名前端开发人员，我到底应该学什么，怎么学。毕业后两年多的时间里，我从使用原生的HTML/CSS/JavaScript，到现在流行的视图库Vue/React，看似我好像学会了，但是离开这些库，或者说将来这些库被淘汰了，我又剩下了什么呢？

我逐渐意识到，我需要的可能是一些更“底层”的东西。

那么，什么是“底层”的东西呢，这里可不是指计算机底层，而是指不会被淘汰的东西，永远有用的东西。我认为对于任何程序员来说，最重要的就是：迅速获取某项知识的能力以及动手解决问题的能力。

前者需要我们建立自己的知识体系，也就是winter老师在第一讲中提到的，我们需要完善的知识架构。

我们不需要也不可能记住整个架构的知识，但我们却可以在需要的时候想起来这块知识的位置，是干什么的，能帮我们解决什么样的问题，可以通过什么关键词检索到，这就够了。

其实你让我回忆winter老师这么多节课的所有内容，我也是好多想不起来的，但多少都留下了印象。我们的大脑容量是有限的，所以“按需加载”也许对人脑也是个不错的策略。

后者其实就归功于winter老师精心设计的课后题，先看这些课文总归是差了点什么，只有你动手去做，才能真正理解这节课所讲的内容。

在做的过程中，你多半会遇到困难（这其实也是winter老师的本意），如果你通过努力解决掉了，这个提升才是最大的。你的方案不一定是最好的，或者你可能实在没什么头绪，这时候也可以利用搜索引擎去看看别人的方案。

因为我早上到公司要坐一个半小时左右的公交，路上我基本上会用耳机听完课（不伤眼），然后剩下的时间去思考回顾一下，想想作业怎么完成。

在中午吃完饭休息的时候，动手做一下，有时候时间不一定够，就利用晚上和第二天的时间完成。每个人的情况都不一样，不过，想要有收获，就得付出多一些。

希望大家都能学有所成，有所收获。

极客时间用户@峰峰峰 分享

先感谢极客时间给了我这次学习分享的机会，如果文中有不足之处，希望小伙伴们可以多多指导。

我是转行过来学习前端的，目前在前端这个岗位上工作了大概一年半的样子，最初知道重学前端这个专栏，也是因为朋友圈里面的一些大牛分享的，比较好奇，并且老师又是非常有名的winter老师，就报了这门课程。

我本人是对前端的技术非常感兴趣的，尤其是原生基础知识，因为我覺得一个人基础的扎实程度，能决定这个人在这个领域所能发展的高度。

刚开始看到课程目录的时候我很吃惊，因为这个目录和我之前所学的所有的课的目录都不一样，从文法、运行时的角度去解释JavaScript这门语言，这让我非常兴奋，因为我知道，这正是我所需要的。

通过跟随winter老师不断学习，我发现这把我有的一些知识结构打乱了，但这并不是坏事，因为我以前的学习和知识结构都非常的混乱。

通过这次学习，让我对JavaScript的理解更加深入了一些，同时，重新梳理我的知识点也让我的知识结构更加的清晰了起来。

可能是我工作时间尚短的原因吧，我觉得老师的每一节课对于我来说都是一个新的东西。举个例子，就拿第3讲和第4讲来说吧，标签语义化我一直都知道，但是我在工作中用得很少，并且在理解上也只停留在非常浅的层面上，但是通过老师的讲解，让我理解到了语义化标签真正的含义。

我现在在工作都会开始使用一些我比较熟悉的语义化标签。包括一些浏览器的工作原理，关于JavaScript这门语言是面向对象还是面向类的解释，以及JavaScript的执行等等，这些知识点都让我受益良多，非常感谢winter老师，希望以后还能有机会跟着老师继续学习。

最后，分享一下我的学习方法吧。

1. 先提升知识的广度，再去拓展知识的深度。
2. 养成随时记录的习惯，可以是笔记，可以是代码（一定要写上注释）。
3. 多逛逛技术论坛，有条件的情况下多去实验一下论坛中自己感兴趣的代码。
4. 最后一条也是最重要的一条，坚持每天八小时工作外的学习和锻炼。

以上就是我的学习心得，不足之处还请多多见谅，希望对小伙伴们有帮助。

极客时间用户@木易杨 分享

我看了老师在“极客live”的直播，总结了他提到的一些观点。

1. 对于框架的使用没必要花太多时间，应该多研究一下三大框架背后的设计思想。
2. 当一个程序员对算法、语言标准、底层、原生、英文文档这些词汇产生恐惧感的时候，他的技术生命已经走到尽头。
3. 前端架构主要解决的是高复用性，架构能力提升方向主要是组件库开发、前端框架实现等。
4. 对于前端进阶这个问题，其实看书的作用和意义已经不太明显，需要寻找好的平台和合适的项目，在项目中不断克服难题并挑战自己，遇到问题再去查资料总结。如果只是闭门看书那很难成为高手，书只是基础而已，真正的应用还是在项目中。
5. 寒冬中能做的只有提升自己，但是光靠技术是不行的。
6. 推荐 TensorFlow、可视化切图、PWA、WebGL。
 - TensorFlow 可以了解使用并做点东西出来，原理很难但不影响使用。
 - PWA 有望进一步发展。
 - WebGL 在未来会是一个很好的方向，它可以实现任何你想要的界面效果，但重点需要多掌握图形学的基础知识，它和算法，数据结构一样重要。
 - Weex和RN虽然都叫Hybird但不太一样，前者适合大厂主要是嵌入APP中使用，后者更适合创业公司。
7. 不推荐 SSR、TypeScript、函数式编程。
 - SSR 不太看好，其主要用于SEO，不太建议用做服务端渲染，其能够使用的场景不多，而且成本代价太大。
 - TypeScript 是好东西，是很有前景的语言，但适用于十万行以上代码级别的大型项目，小项目并不适合，反而徒增复杂。
 - 用JavaScript做函数式编程并不靠谱，Map/Reduce/Redux/Hooks等并不是函数式编程，只是长得像而已。

PS：笔记内容由自己和群友提供，仅供参考。

我的一些思考和尝试

听完 winter 老师直播后对其观点很是赞同，但因为是面对所有人讲解，所以内容有点多范围有些广，但对个人来说还是找适合自己的方向，在自己相对熟悉的领域再去扩展去突破。横向只是拓宽你的眼界，纵向才是你的核心竞争力。

对我来说感触最大的就是3、4、5这几点，因为我一直在思考以下几个问题。

1. 我现在是高级前端，但又感觉自身很弱，那我如何才能夯实我高级的地基然后成为资深前端呢？
2. 项目迭代节奏快日常加班又多，那我如何做才能平衡工作和学习？
3. 单单提升技术好像还是有很大瓶颈，那我如何提升我的职场核心竞争力？

结合 winter 老师的专栏和最近看的几篇文章，说说我对这几个问题的思考。

对于第一个问题，在2个多月前我还是不会写文章的小白，那个时候受到一些文章的触动，尝试把我的前端之路记录下来。

我的想法很简单，就是把前端进阶28期的重难点知识全部讲完，目前已经进行到第4期了，通过写作把知识通俗易懂的介绍给别人，在这个过程自己肯定会收获很多很多。

这就是我目前在尝试的学习方式，通过写作建立自己的知识架构，并且在这个架构上不断地进行优化，时间到了自然就进阶了。

对于第二个问题，刚开始写作时精力充沛时间也很多，每天都会更新技术文章，但随着项目迭代压力增大，文章更新速度相应就变慢了一些。

虽然文章质量提升了很多，但更新速度从日更变成了周更再变成了双周更，这个说实话我自己已经很难接受了，因为速度变慢导致我年初的计划要打折扣。

所以说做项目和自我学习要如何权衡，是否说我要减少项目难度和时间并在工作中摸鱼去学习呢？

我的结论是不，因为一句话，“最好的学习就是在项目中锻炼自己”。既然我有这么好的项目去锻炼，那为什么还要摸鱼去学习呢，这不就是南辕北辙嘛，道理很简单但不是所有人都懂。

我最近在执行的方法是专注 + 锻炼 + 利用周末。

专注即工作时专注于工作，努力做好每次迭代，遇到难题迎难而上，工作时不开微信，勤用笔记安排日常工作并整理文档。

锻炼即一周抽出三天时间每次去健身房锻炼1小时，强壮的体魄才能撑住高强度的工作和学习，因为网易有健身房所以冬天锻炼也没什么问题，没办法去健身房的小伙伴可以尝试跑步或者在瑜伽垫上做Keep。

利用周末即加大周末和平常晚上熬夜的时间去写文章，减少娱乐的时间，正所谓时间挤一挤总会有的，但这件事情比较反人性，所以重在坚持。公众号写文章更容易坚持，原因在于文章发布后有正反馈，比如粉丝的增长，留言和鼓励，赞赏等等，这些都更能促使你坚持下去，时间长了自然就养成了习惯。

对于第三个问题，winter 老师也说了，寒冬中光靠技术是不行的，那应该靠什么呢？幸好在最近看到的一篇文章中找到了答案，那就是表达能力。

表达能力是形成自己的框架系统，有理有据并且逻辑清晰，而且能让外人听懂，大部分优秀的人都具备这样的能力。反观自己并没有这样的能力，所以我要努力提升这块，其中最重要的方法就是写作训练。

兜兜转转那么多，其实我一直在尝试的进阶方法就是通过写作建立自己的知识架构体系，同时提高自身的表达能力，通过正反馈机制和锻炼保证我长久的坚持下去并最终养成习惯，习惯这一模式之后变成优秀的人也只是时间问题罢了。

希望我的一些思考和尝试能对你有所帮助。

（直播回顾：<https://v.qq.com/x/page/x0830f8xoj.html>）

恭喜幸运用户 @阿成、@峰峰峰、@木易杨，你们的经验被选取成为本期的用户故事。极客时间将送出“极客时间”& winter联名T恤一件。1个工作日之内，工作人员会与你取得联系。

读完了三位同学的故事后，你有哪些感受呢，留下你的收获，我们一起成长。

写在前面的话

不知不觉，专栏已经更新了一半有余了。在两个月的时间里，我收到了很多同学的留言，很开心可以跟大家保持交流，也很惊喜见证了大家的成长。

在整个学习过程中，你有没有一些学习感悟呢，有哪些新的学习收获，又有哪些实践感悟？有没有对前端的知识架构有了更为深刻的认识，是否规划好了自己的前端之路。

如果你有想分享的经验，可以在文章下留言，如果你的留言被选中作为用户故事，我们将会为你送出“极客时间”& winter联名T恤一件。分享你的收获，我们一起成长。



下面我们一起来看看三位同学的分享。

极客时间用户@阿成 分享

在我进入知乎社区的时候，winter 老师就已经不在这个社区了，虽然是退隐了江湖，但是江湖上还流传着他的传说，后来我在 Google Feed 中看到 winter 老师的开篇词，看完就感觉是我要的东西，还正好赶上打折，就毫不犹豫地买了。

我本科学的是光电信息，所以并不算是科班出身的程序员（虽然也有学C啦）。我本身对计算机程序一直很有兴趣，所以从大三开始，我就决定放弃从事本专业，做一名程序员。要说为什么选择前端领域，自然也是烂俗的“门槛低”理由。

不过，我没想到的是，前端的学习曲线到后面是如此陡峭，并且前端发展的速度如此之快，让人目不暇接，也让很多人产生了“技术焦虑”。

在这样的环境中，我也产生了迷茫，作为一名前端开发人员，我到底应该学什么，怎么学。毕业后两年多的时间里，我从使用原生的HTML/CSS/JavaScript，到现在流行的视图库Vue/React，看似我好像学会了很多，但是离开这些库，或者说将来这些库被淘汰了，我又剩下了什么呢？

我逐渐意识到，我需要的可能是一些更“底层”的东西。

那么，什么是“底层”的东西呢，这里可不是指计算机底层，而是指不会被淘汰的东西，永远有用的东西。我认为对于任何程序员来说，最重要的就是：迅速获取某项知识的能力以及动手解决问题的能力。

前者需要我们建立自己的知识体系，也就是winter老师在第一讲中提到的，我们需要完善的知识架构。

我们不需要也不可能记住整个架构的知识，但我们却可以在需要的时候想起来这块知识的位置，是干什么的，能帮我们解决什么样的问题，可以通过什么关键词检索到，这就够了。

其实你让我回忆winter老师这么多节课的所有内容，我也是好多想不起来的，但多少都留下了印象。我们的大脑容量是有限的，所以“按需加载”也许对人脑也是个不错的策略。

后者其实就归功于winter老师精心设计的课后题，光看这些课文总归是差了点什么，只有你动手去做，才能真正理解这节课所讲的内容。

在做过程中，你多半会遇到困难（这其实也是winter老师的本意），如果你通过努力解决掉了，这个提升才是最大的。你的方案不一定是最好的，或者你可能实在没什么头绪，这时候也可以利用搜索引擎去看看别人的方案。

因为我早上到公司要坐一个半小时左右的公交，路上我基本上会用耳机听完课（不伤眼），然后剩下的时间去思考回顾一下，想想作业怎么完成。

在中午吃完饭休息的时候，动手做一下，有时候时间不一定够，就利用晚上和第二天的时间完成。每个人的情况都不一样，不过，想要有收获，就得付出多一些。

希望大家都能学有所成，有所收获。

极客时间用户@峰峰峰 分享

先感谢极客时间给了我这次学习分享的机会，如果文中有不足之处，希望小伙伴们可以多多指导。

我是转行过来学习前端的，目前在前端这个岗位上工作了大概一年半的样子，最初知道重学前端这个专栏，也是因为朋友圈里面的一些大牛分享的，比较好奇，并且老师又是非常有名的winter老师，就报了这门课程。

我本人是对前端的技术非常感兴趣的，尤其是原生基础知识，因为我觉得一个人基础的扎实程度，能决定这个人在这个领域所能发展的高度。

刚开始看到课程目录的时候我很吃惊，因为这个目录和我之前所学的所有的课的目录都不一样，从文法、运行时的角度去解释JavaScript这门语言，这让我非常兴奋，因为我知道，这正是我所需要的。

通过跟随winter老师不断学习，我发现这把我原有的一些知识结构打乱了，但这并不是坏事，因为我以前的学习和知识结构都非常的混乱。

通过这次学习，让我对JavaScript的理解更加深入了一些，同时，重新梳理我的知识点也让我的知识结构更加的清晰了起来。

可能是我工作时间尚短的原因吧，我觉得老师的每一节课对于我来说都是一个新的东西。举个例子，就拿第3讲和第4讲来说吧，标签语义化我一直都知道，但是我在工作中用得很少，并且在理解上也只停留在非常浅的层面上，但是通过老师的讲解，让我理解到了语义化标签真正的含义。

我现在在工作都会开始使用一些我比较熟悉的语义化标签。包括一些浏览器的工作原理，关于JavaScript这门语言是面向对象还是面向类的解释，以及JavaScript的执行等等，这些知识点都让我受益良多，非常感谢winter老师，希望以后还能有机会跟着老师继续学习。

最后，分享一下我的学习方法吧。

1. 先提升知识的广度，再去拓展知识的深度。
2. 养成随时记录的习惯，可以是笔记，可以是代码（一定要写上注释）。
3. 多逛逛技术论坛，有条件的话多去实验一下论坛中自己感兴趣的代码。
4. 最后一条也是最重要的一条，坚持每天八小时工作外的学习和锻炼。

以上就是我的学习心得，不足之处还请多多见谅，希望对小伙伴们有帮助。

极客时间用户@木易杨 分享

我看了老师在“极客live”的直播，总结了他提到的一些观点。

1. 对于框架的使用没必要花太多时间，应该多研究一下三大框架背后的设计思想。
2. 当一个程序员对算法、语言标准、底层、原生、英文文档这些词汇产生恐惧感的时候，他的技术生命已经走到尽头。
3. 前端架构主要解决的是高复用性，架构能力提升方向主要是组件库开发、前端框架实现等。
4. 对于前端进阶这个问题，其实看书的作用和意义已经不太明显，需要寻找好的平台和合适的项目，在项目中不断克服难题并挑战自己，遇到问题再去查资料总结。如果只是闭门看书那很难成为高手，书只是基础而已，真正的应用还是在项目中。
5. 寒冬中能做的只有提升自己，但是光靠技术是不行的。
6. 推荐 TensorFlow、可视化切图、PWA、WebGL。
 - TensorFlow 可以了解使用并做点东西出来，原理很难但不影响使用。
 - PWA 有望进一步发展。
 - WebGL 在未来会是一个很好的方向，它可以实现任何你想要的界面效果，但重点需要多掌握图形学的基础知识，它和算法，数据结构一样重要。
 - Weex和 RN 虽然都叫 Hybrid 但不太一样，前者适合大厂主要是嵌入 APP 中使用，后者更适合创业公司。
7. 不推荐 SSR、TypeScript、函数式编程。
 - SSR 不太看好，其主要是用于SEO，不太建议用做服务端渲染，其能够使用的场景不多，而且成本代价太大。
 - TypeScript 是好东西，是很有前景的语言，但适用于十万行以上代码级别的大型项目，小项目并不适合，反而会使复杂。
 - 用 JavaScript 做函数式编程并不靠谱，Map/Reduce/Redux/Hooks等并不是函数式编程，只是长得像而已。

PS：笔记内容由自己和群友提供，仅供参考。

我的一些思考和尝试

听完 winter 老师直播后对其观点很是赞同，但因为是面对所有人讲解，所以内容有点多范围有些广，但对个人来说还是找适合自己的方向，在自己相对熟悉的领域再去扩展去突破。横向只是拓宽你的眼界，纵向才是你的核心竞争力。

对我来说感触最大的就是 3、4、5 这几点，因为我一直在思考以下几个问题。

1. 我现在是高级前端，但又感觉自身很弱，那我如何才能夯实我高级的地基然后成为资深前端呢？
2. 项目迭代节奏快日常加班又多，那我如何做才能平衡工作和学习？
3. 单单提升技术好像还是有很大瓶颈，那我如何提升我的职场核心竞争力？

结合 winter 老师的专栏和最近看的几篇文章，说说我对这几个问题的思考。

对于第一个问题，在 2 个多月前我还是不会写文章的小白，那个时候受到一些文章的触动，尝试把我的前端之路记录下来。

我的想法很简单，就是把前端进阶 28 期的重难点知识全部讲完，目前已经进行到第 4 期了，通过写作把知识通俗易懂的介绍给别人，在这个过程中自己肯定会收获很多很多。

这就是我目前在尝试的学习方式，通过写作建立自己的知识架构，并且在这个架构上不断地进行优化，时间到了自然就进阶了。

对于第二个问题，刚开始写作时精力充沛时间也很多，每天都会更新技术文章，但随着项目迭代压力增大，文章更新速度相应就变慢了一些。

虽然文章质量提升了很多，但更新速度从日更变成了周更再变成了双周更，这个说实话我自己已经很难接受了，因为速度变慢导致我年初的计划要打折扣。

所以说做项目和自我学习要如何权衡，是否说我要减少项目难度和时间并在工作摸鱼去学习呢？

我的结论是不，因为一句话，“最好的学习就是在项目中锻炼自己”。既然我有这么好的项目去锻炼，那为什么还要摸鱼去学习呢，这不就是南辕北辙嘛，道理很简单但不是所有人都懂。

我最近在执行的方法是专注 + 锻炼 + 利用周末。

专注即工作时专注于工作，努力做好每次迭代，遇到难题迎难而上，工作时不开微信，勤用笔记安排日常工作并整理文档。

锻炼即一周抽出三天时间每次去健身房锻炼1小时，强壮的体魄才能撑住高强度的工作和学习，因为网易有健身房所以冬天锻炼也没什么问题，没办法去健身房的小伙伴可以尝试跑步或者在瑜伽垫上做Keep。

利用周末即加大周末和平常晚上熬夜的时间去写文章，减少娱乐的时间，正所谓时间挤一挤总会有的，但这件事情比较反人性，所以重在坚持。公众号写文章更容易坚持，原因在于文章发布后有正反馈，比如粉丝的增长，留言和鼓励，赞赏等等，这些都更能促使你坚持下去，时间长了自然就养成了习惯。

对于第三个问题，winter 老师也说了，寒冬中光靠技术是不行的，那应该靠什么呢？幸好在最近看到的一篇文章中找到了答案，那就是表达能力。

表达能力是形成自己的框架系统，有理有据并且逻辑清晰，而且能让外人听懂，大部分优秀的人都具备这样的能力。反观自己并没有这样的能力，所以我要努力提升这块，其中最重要的方法就是写作训练。

兜兜转转那么多，其实我一直在尝试的进阶方法就是通过写作建立自己的知识架构体系，同时提高自身的表达能力，通过正反馈机制和锻炼保证我长久的坚持下去并最终养成习惯，习惯这一模式之后变

成优秀的人也只是时间问题罢了。

希望我的一些思考和尝试能对你有所帮助。

（直播回顾：<https://v.qq.com/x/page/x0830f68xoj.html>）

恭喜幸运用户 @阿成、@峰峰峰、@木易杨，你们的经验被选取成为本期的用户故事。极客时间将送出“极客时间”& winter联名T恤一件。1个工作日之内，工作人员会与你取得联系。

读完了三位同学的故事后，你有哪些感受呢，留下你的收获，我们一起成长。

写在前面的话

不知不觉，专栏已经更新了一半有余了。在两个月的时间里，我收到了很多同学的留言，很开心可以跟大家保持交流，也很惊喜见证了大家的成长。

在整个学习过程中，你有没有一些学习感悟呢，有哪些新的学习收获，又有哪些实践感悟？有没有对前端的知识架构有了更为深刻的认识，是否规划好了自己的前端之路。

如果你有想分享的经验，可以在文章下留言，如果你的留言被选中作为用户故事，我们将会为你送出“极客时间”& winter联名T恤一件。分享你的收获，我们一起成长。



下面我们一起来看看三位同学的分享。

极客时间用户@阿成 分享

在我进入知乎社区的时候，winter 老师就已经不在这个社区了，虽然是退隐了江湖，但是江湖上还流传着他的传说，后来我在 Google Feed 中看到 winter 老师的开篇词，看完就感觉是我要的东西，还正好赶上打折，就毫不犹豫地买了。

我本科学的是光电信息，所以并不算是科班出身的程序员（虽然也有学C啦）。我本身对计算机程序一直很有兴趣，所以从大三开始，我就决定放弃从事本专业，做一名程序员。要说为什么选择前端领域，自然也是烂俗的“门槛低”理由。

不过，我没想到的是，前端的学习曲线到后面是如此陡峭，并且前端发展的速度如此之快，让人目不暇接，也让很多人产生了“技术焦虑”。

在这样的环境中，我也产生了迷茫，作为一名前端开发人员，我到底应该学什么，怎么学。毕业后两年多的时间里，我从使用原生的HTML/CSS/JavaScript，到现在流行的视图库Vue/React，看似我好像学会了很多，但是离开这些库，或者说将来这些库被淘汰了，我又剩下了什么呢？

我逐渐意识到，我需要的可能是一些更“底层”的东西。

那么，什么是“底层”的东西呢，这里可不是指计算机底层，而是指不会被淘汰的东西，永远有用的东西。我认为对于任何程序员来说，最重要的就是：迅速获取某项知识的能力以及动手解决问题的能力。

前者需要我们建立自己的知识体系，也就是winter老师在第一讲中提到的，我们需要完善的知识架构。

我们不需要也不可能记住整个架构的知识，但我们却可以在需要的时候想起来这块知识的位置，是干什么的，能帮我们解决什么样的问题，可以通过什么关键词检索到，这就够了。

其实你让我回忆winter老师这么多节课的所有内容，我也是好多想不起来的，但多少都留下了印象。我们的大脑容量是有限的，所以“按需加载”也许对人脑也是个不错的策略。

后者其实就归功于winter老师精心设计的课后题，先看这些课文总归是差了点什么，只有你动手去做，才能真正理解这节课所讲的内容。

在做过程中，你多半会遇到困难（这其实也是winter老师的本意），如果你通过努力解决掉了，这个提升才是最大的。你的方案不一定是最好的，或者你可能实在没什么头绪，这时候也可以利用搜索引擎去看看别人的方案。

因为我早上到公司要坐一个半小时左右的公交，路上我基本上会用耳机听完课（不伤眼），然后剩下的时间去思考回顾一下，想想作业怎么完成。

在中午吃完饭休息的时候，动手做一下，有时候时间不一定够，就利用晚上和第二天的时间完成。每个人的情况都不一样，不过，想要有收获，就得付出多一些。

希望大家都能学有所成，有所收获。

极客时间用户@峰峰峰 分享

先感谢极客时间给了我这次学习分享的机会，如果文中有不足之处，希望小伙伴们可以多多指导。

我是转行过来学习前端的，目前在前端这个岗位上工作了大概一年半的样子，最初知道重学前端这个专栏，也是因为朋友圈里面的一些大牛分享的，比较好奇，并且老师又是非常有名的winter老师，就报了这门课程。

我本人是对前端的技术非常感兴趣的，尤其是原生基础知识，因为我覺得一个人基础的扎实程度，能决定这个人在这个领域所能发展的高度。

刚开始看到课程目录的时候我很吃惊，因为这个目录和我之前所学的所有的课的目录都不一样，从文法、运行时的角度去解释JavaScript这门语言，这让我非常兴奋，因为我知道，这正是我所需要的。

通过跟随winter老师不断学习，我发现这把我有的一些知识结构打乱了，但这并不是坏事，因为我以前的学习和知识结构都非常的混乱。

通过这次学习，让我对JavaScript的理解更加深入了一些，同时，重新梳理我的知识点也让我的知识结构更加的清晰了起来。

可能是我工作时间尚短的原因吧，我觉得老师的每一节课对于我来说都是一个新的东西。举个例子，就拿第3讲和第4讲来说吧，标签语义化我一直都知道，但是我在工作中用得很少，并且在理解上也只停留在非常浅的层面上，但是通过老师的讲解，让我理解到了语义化标签真正的含义。

我现在在工作都会开始使用一些我比较熟悉的语义化标签。包括一些浏览器的工作原理，关于JavaScript这门语言是面向对象还是面向类的解释，以及JavaScript的执行等等，这些知识点都让我受益良多，非常感谢winter老师，希望以后还能有机会跟着老师继续学习。

最后，分享一下我的学习方法吧。

1. 先提升知识的广度，再去拓展知识的深度。
2. 养成随时记录的习惯，可以是笔记，可以是代码（一定要写上注释）。
3. 多逛逛技术论坛，有条件的话多去实验一下论坛中自己感兴趣的代码。
4. 最后一条也是最重要的一条，坚持每天八小时工作外的学习和锻炼。

以上就是我的学习心得，不足之处还请多多见谅，希望对小伙伴们有帮助。

极客时间用户@木易杨 分享

我看了老师在“极客live”的直播，总结了他提到的一些观点。

1. 对于框架的使用没必要花太多时间，应该多研究一下三大框架背后的设计思想。
2. 当一个程序员对算法、语言标准、底层、原生、英文文档这些词汇产生恐惧感的时候，他的技术生命已经走到尽头。
3. 前端架构主要解决的是高复用性，架构能力提升方向主要是组件库开发、前端框架实现等。
4. 对于前端进阶这个问题，其实看书的作用和意义已经不太明显，需要寻找好的平台和合适的项目，在项目中不断克服难题并挑战自己，遇到问题再去查资料总结。如果只是闭门看书那很难成为高手，书只是基础而已，真正的应用还是在项目中。
5. 寒冬中能做的只有提升自己，但是光靠技术是不行的。
6. 推荐 TensorFlow、可视化切图、PWA、WebGL。
 - TensorFlow 可以了解使用并做点东西出来，原理很难但不影响使用。
 - PWA 有望进一步发展。
 - WebGL 在未来会是一个很好的方向，它可以实现任何你想要的界面效果，但重点需要多掌握图形学的基础知识，它和算法，数据结构一样重要。
 - Weex和RN虽然都叫Hybird但不太一样，前者适合大厂主要是嵌入APP中使用，后者更适合创业公司。
7. 不推荐 SSR、TypeScript、函数式编程。
 - SSR 不太看好，其主要用于SEO，不太建议用做服务端渲染，其能够使用的场景不多，而且成本代价太大。
 - TypeScript 是好东西，是很有前景的语言，但适用于十万行以上代码级别的大型项目，小项目并不适合，反而徒增复杂。
 - 用JavaScript做函数式编程并不靠谱，Map/Reduce/Redux/Hooks等并不是函数式编程，只是长得像而已。

PS：笔记内容由自己和群友提供，仅供参考。

我的一些思考和尝试

听完 winter 老师直播后对其观点很是赞同，但因为是面对所有人讲解，所以内容有点多范围有些广，但对个人来说还是找适合自己的方向，在自己相对熟悉的领域再去扩展去突破。横向只是拓宽你的眼界，纵向才是你的核心竞争力。

对我来说感触最大的就是3、4、5这几点，因为我一直在思考以下几个问题。

1. 我现在是高级前端，但又感觉自身很弱，那我如何才能夯实我高级的地基然后成为资深前端呢？
2. 项目迭代节奏快日常加班又多，那我如何做才能平衡工作和学习？
3. 单单提升技术好像还是有很大瓶颈，那我如何提升我的职场核心竞争力？

结合 winter 老师的专栏和最近看的几篇文章，说说我对这几个问题的思考。

对于第一个问题，在2个多月前我还是不会写文章的小白，那个时候受到一些文章的触动，尝试把我的前端之路记录下来。

我的想法很简单，就是把前端进阶28期的重难点知识全部讲完，目前已经进行到第4期了，通过写作把知识通俗易懂的介绍给别人，在这个过程自己肯定会收获很多很多。

这就是我目前在尝试的学习方式，通过写作建立自己的知识架构，并且在这个架构上不断地进行优化，时间到了自然就进阶了。

对于第二个问题，刚开始写作时精力充沛时间也很多，每天都会更新技术文章，但随着项目迭代压力增大，文章更新速度相应就变慢了一些。

虽然文章质量提升了很多，但更新速度从日更变成了周更再变成了双周更，这个说实话我自己已经很难接受了，因为速度变慢导致我年初的计划要打折扣。

所以说做项目和自我学习要如何权衡，是否说我要减少项目难度和时间并在工作中摸鱼去学习呢？

我的结论是不，因为一句话，“最好的学习就是在项目中锻炼自己”。既然我有这么好的项目去锻炼，那为什么还要摸鱼去学习呢，这不就是南辕北辙嘛，道理很简单但不是所有人都懂。

我最近在执行的方法是专注 + 锻炼 + 利用周末。

专注即工作时专注于工作，努力做好每次迭代，遇到难题迎难而上，工作时不开微信，勤用笔记安排日常工作并整理文档。

锻炼即一周抽出三天时间每次去健身房锻炼1小时，强壮的体魄才能撑住高强度的工作和学习，因为网易有健身房所以冬天锻炼也没什么问题，没办法去健身房的小伙伴可以尝试跑步或者在瑜伽垫上做Keep。

利用周末即加大周末和平常晚上熬夜的时间去写文章，减少娱乐的时间，正所谓时间挤一挤总会有的，但这件事情比较反人性，所以重在坚持。公众号写文章更容易坚持，原因在于文章发布后有正反馈，比如粉丝的增长，留言和鼓励，赞赏等等，这些都更能促使你坚持下去，时间长了自然就养成了习惯。

对于第三个问题，winter 老师也说了，寒冬中光靠技术是不行的，那应该靠什么呢？幸好在最近看到的一篇文章中找到了答案，那就是表达能力。

表达能力是形成自己的框架系统，有理有据并且逻辑清晰，而且能让外人听懂，大部分优秀的人都具备这样的能力。反观自己并没有这样的能力，所以我要努力提升这块，其中最重要的方法就是写作训练。

兜兜转转那么多，其实我一直在尝试的进阶方法就是通过写作建立自己的知识架构体系，同时提高自身的表达能力，通过正反馈机制和锻炼保证我长久的坚持下去并最终养成习惯，习惯这一模式之后变成优秀的人也只是时间问题罢了。

希望我的一些思考和尝试能对你有所帮助。

（直播回顾：<https://v.qq.com/x/page/x0830f8xoj.html>）

恭喜幸运用户 @阿成、@峰峰峰、@木易杨，你们的经验被选取成为本期的用户故事。极客时间将送出“极客时间”& winter联名T恤一件。1个工作日之内，工作人员会与你取得联系。

读完了三位同学的故事后，你有哪些感受呢，留下你的收获，我们一起成长。