

两年内收集的30条小Tips

jsonchao 鸿洋 2020-02-06

本文作者

作者: **jsonchao**

链接:

<https://juejin.im/post/5e13e8265188253a887bb44c>

本文由作者授权发布。

1.一个浪头打过来，最好的办法是迎上去了解个究竟，而不是漠视，或者干脆当事情没有发生。

2.记住这个教训：别猜，去问！去查！

3.隔一段时间重顾知识是记忆的关键方法。

不要一次性学完一项知识就不管了，这样你无法真正的掌握。隔一段时间回顾一下，每次重复，你都会加深自己的理解，次数多了，你就会理解每个细节，成为真正的专家。

4.对于一个技术通常我们需要抱有这样的疑问

它是什么 一句话概括 解决了什么问题

存在的意义

怎样去使用它

怎样解决了问题 内部的实现

它的缺点是什么 多角度分析

5.高效率的学习方法。

理清楚概念很重要

做好控制变量法

多动手实践，与理论结合

抓住重点，剔除干扰因素

6.关于如何选择第三方库

确定这个库是否是必需的

这个库能否带来开发效率的提升，降低代码的维护成本

这个库的学习成本如何 比如rxjava其实学习成本会相对高一些。

这个库的质量如何，不要仅仅看star，更要看issue的处理情况。

7.怎样练习算法题？

每道算法题都先自己去实现，先写思路，然后自己去实现一遍，然后再看看答案，记住答案的思路，第二天再重新按照答案的思路实现一遍。

8."另类"的学习方法 抄书的奥妙——那就是延缓阅读速度，不至遗漏每一个重要的细节：眼到，手到，心到，其实不仅书抄得，代码(优秀源码)也抄得。

9.建议：建立逐字稿

计算机网络相关的逐字稿

计算机操作系统相关的逐字稿

数据库相关的逐字稿

设计模式相关的逐字稿

数据结构与算法相关的逐字稿

Java知识相关的逐字稿

Kotlin知识相关的逐字稿

Python知识相关的逐字稿

React Native知识相关的逐字稿

Flutter知识相关的逐字稿

小程序知识相关的逐字稿

JS知识相关的逐字稿

Android知识相关的逐字稿

与技术无关的逐字稿

10.如何提升你的阅读能力？

只字不差的反复阅读

真正的获取知识，是通过阅读，深入思考与践行

11.如何高效阅读一篇文章？

由主题扩展为知识树

尝试描述

尝试记忆

12.如何阅读源码？

流程：

- 1.寻找驱动力
- 2.浏览官方文档，对开源项目的功能、架构有大概的印象
- 3.在工作中或实践中使用开源项目
- 4.网上搜索针对该开源项目进行分析的优秀文章
- 5.对开源项目提出自己的疑问
- 6.把开源项目下载到本地，并导入IDE，方便调试、测试
- 7.带着疑问阅读源码
- 8.阅读源码过程中多添加注释、多做笔记
- 9.做阅读总结，吸收和再创造

准备：

Java设计模式(模板方法，单例，观察者，工厂方法，代理，策略，装饰者)

Java高级相关

熟练掌握这个库

先Google了解软件的整体架构设计

搭建系统，把源码跑起来

开始阅读：

根据你对系统的理解，设计几个主要的测试案例，定义好输入，输出。(Debug一遍肯定是不行的，需要Debug很多遍)

第一篇抛弃细节，抓住主要流程，第二篇，第三篇，再去看各个部分的细节。

阅读的时候同时使用UML画出系统的类图。

主要的测试案例明白了，丰富测试案例，考虑一些分支。

这一步会非常非常地花费时间，但是你做完了，对系统的理解绝对有质的飞跃。

13.想象一个来自未来的自己，他非常自信，非常成功，拥有你现在所希望的一切，他会对你现在的你说些什么？他怎么说，你就怎么去做，10年之后，你就变成了他。

14.重视实践，充分运用感性认知潜能，在项目中磨炼自己，才是正确的学习之道。在实践中，在某些关键动作上刻意练习，也会取得事半功倍的效果。

15.我们需要从别人身上学习。从老师、领导、同事、下属甚至对手身上学习，是快速成长的重要手段。

16.多多总结，多多分享，善莫大焉。

17.解答别人的问题也是个人成长的重要手段。有时候，某个问题自己本来不太懂，但是在给别人讲解的时候却豁然开朗。所以，“诲人不倦”利人惠己。

18.学习计划最好能结合工作计划，理论联系实际结合，快速学以致用。

19.学习就是不断地刻意联系，刻意练习，就是有目的的练习，先规划好，再去练习。首先给自己定一个目标，目标可以有效的引导你学习，然后使用3F练习法：

1：专注（Focus），专注在眼前的任务上，在学习过程中保持专注，可以尝试使用番茄工作法。

2：反馈（Feedback），意识到自己的不足，学习完之后进行反思，思考下自己哪些方面不足，为什么不足，

3：修正（Fix），改进自己的不足。不停的练习和思考可以改变大脑结构，大脑像肌肉一样，挑战越大，影响越大，学习更高效，并且也会产生突破性。

20.写一篇博客的过程，其实就是对一件事情，学习、理解、思考、转化，最终输出成一篇博客的过程。

21.深入浅出SQL给出的学习法则：

如何快速记忆知识？将文字转换为图片（将文字嵌入图片效果比较好），如果能转换为令人惊奇，有趣的情景模拟，那么可以让你的大脑意识到这是重要的东西，记忆效果会更好。

1.慢慢来，理解越多，需要强记得就越少。

2.勤做笔记，写下你的心得笔记。

3.你的大脑会需要一段时间来消化新知识，如果之后再学别的知识，会使之前的记忆效果减弱，因此，在睡前看最重要的知识。

4.喝水，多喝水。

5.大声说出你想要记忆的知识，如果能与别人进行一问一答则效果更佳。

6.当学习知识时，达到了漫不经心或过目即忘的状态，则应该让大脑好好休息。

7.用心感受，让你的大脑知道这很重要，将学习的内容尽量以情景化+惊奇+幽默的形式展示出来。

8.用学到的知识解决实际难题（真实的情景演练）。

22.想要学习新技术，想要提升自己，不是看见新技术就去学，沉下心来认真钻研 才行，吃透它，不再为缓解焦虑而学习。

23.业务代码一样很牛逼

1.使用封装和抽象可以使业务代码更具扩展性。

2.多和产品交流以便更好地理解和实现业务。

3.日志记录好了问题定位效率可以提升10倍。

24.在工作中学习、实战提升是效果最好的，其余时间可有目的去碎片化学习 一整块知识，也可以快速构建牢固的知识体系。

25.做更多：工作中熟悉多个业务代码，端到端（前后端）的业务代码，自学。

做更好：

1.提升项目稳定性，引进单元测试和UI测试。

2.重构解耦项目。

3.性能优化。

4.设计模式去除重复代码。

做练习：1.学习 2.尝试 3.教学

26.一项新技术的出现，应该先去了解它，看它是否对自身的技术成长有比较大的帮助，有的话按优先级加入计划表

27.重复记忆时间间隔：1小时、早上/晚上、1天、3天、7天、1个月、3个月形成长期记忆~

28.学习一个新的知识点的流程：what、why、how、原理/源码、优缺点~ 注意“先主后从”原则，多实践加深理解。

29.真正地掌握一个知识点：

1.看书、博客和源码学习

2.看的过程中要把书中的例子都敲一遍，所有的代码都要亲自敲，不要使用复制 粘贴，相信我，复制粘贴达不到你想要的效果

3.看的过程中多思考，多总结，多验证，把关键点和自己的思考总结写成博客或者笔记，于人于己都是好事

4.一个个的知识点重复上述的三个过程，并坚持下来

30.牢记三大沟通方式：尊重对方、倾听对方和情绪控制。

沟通的目的不是为了附和对方，而是产生一种更完整更全面的认知。

只有当双方都愿意接受不同的观点 时，此时的沟通才会迸发出更多的火花，而这一切都需要发生在相互尊重的基础之上。

31.学习的第一步是知道自己学习的这个知识问题是什么，答案是什么。

然后找到这些问题和答案之间的关系，这个关系是我们需要学习的东西。

最后能把这个关系通过通俗易懂的语言输出出来，那么这个知识你一定学会了。

推荐阅读：

[LayoutManager](#)高端玩家，实现花式表格

[OpenGL ES 怎么学？打开未知技术的大门](#)



扫一扫 关注我的公众号
如果你想要跟大家分享你的文章，欢迎投稿~

٩(^ 0 ^)♂明天见！

阅读原文

喜欢此内容的人还喜欢

大厂分享...不，大佬分享...
鸿洋

当代年轻人吃饭原则：排队越长，东西越香！
少女兔

我才不要当全职妈妈呢！
阿司匹林博物馆