

从零开始学iOS7开发系列3-我的地盘我做主-番外篇

原文及示例代码来自raywenderlich store中的iOS Apprentice 系列3教程，经过翻译和改编。

版权归作者所有，本系列教程仅供学习参考使用，感兴趣的朋友建议购买原英文教程教程(The iOS Apprentice Second Edition: Learn iPhone and iPad Programming via Tutorials!)

购买链接：

<http://www.raywenderlich.com/store>

加尔鲁什，你不配统治部落！



好了，伴随着这个NB橙卡人物凯恩.血蹄的登场怒吼，我们即将开始新的学习。不过在正式学习之前，先分享下在微信订阅号MOOC上看到的一个观点。

大家都知道，在传统的学习方式中，无论是现实中的课堂，还是在线的视频，文本，书籍，信息传递往往是单向的。也就是说知识或者技能是从老师（不管是现实的班主任还是活在二次元的苍老师）单向传递给学生。即便会有一些反馈和提问，但总的来说也是由老师教给学生。但在这个知识大爆炸的时代，这种传统的学习方式是低效的。为什么？因为学生是非常被动的。首先他们不能选择自己要学的内容，其次他们只能被动的接受，再次学习的内容是枯燥的。

那么，什么是有效的学习方式呢？

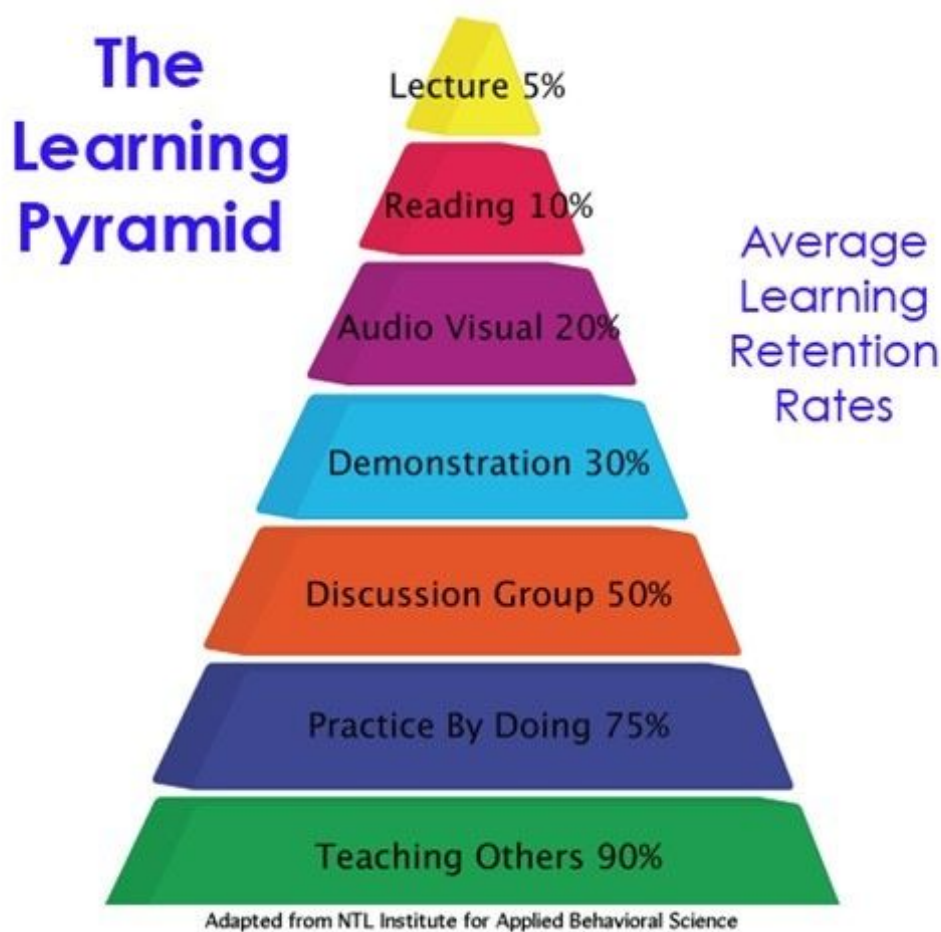
首先要有passionate，这个就不说了，属于虚的东西。如果对一个东西根本没兴趣还看得下去吗？

其次是要主动式学习。什么是主动式学习？简单来说，就是第一时间开始学以致用，然后发现自己的不足再主动学习。而不是被动接受一大堆知识之后再从中筛选自己所需要的。

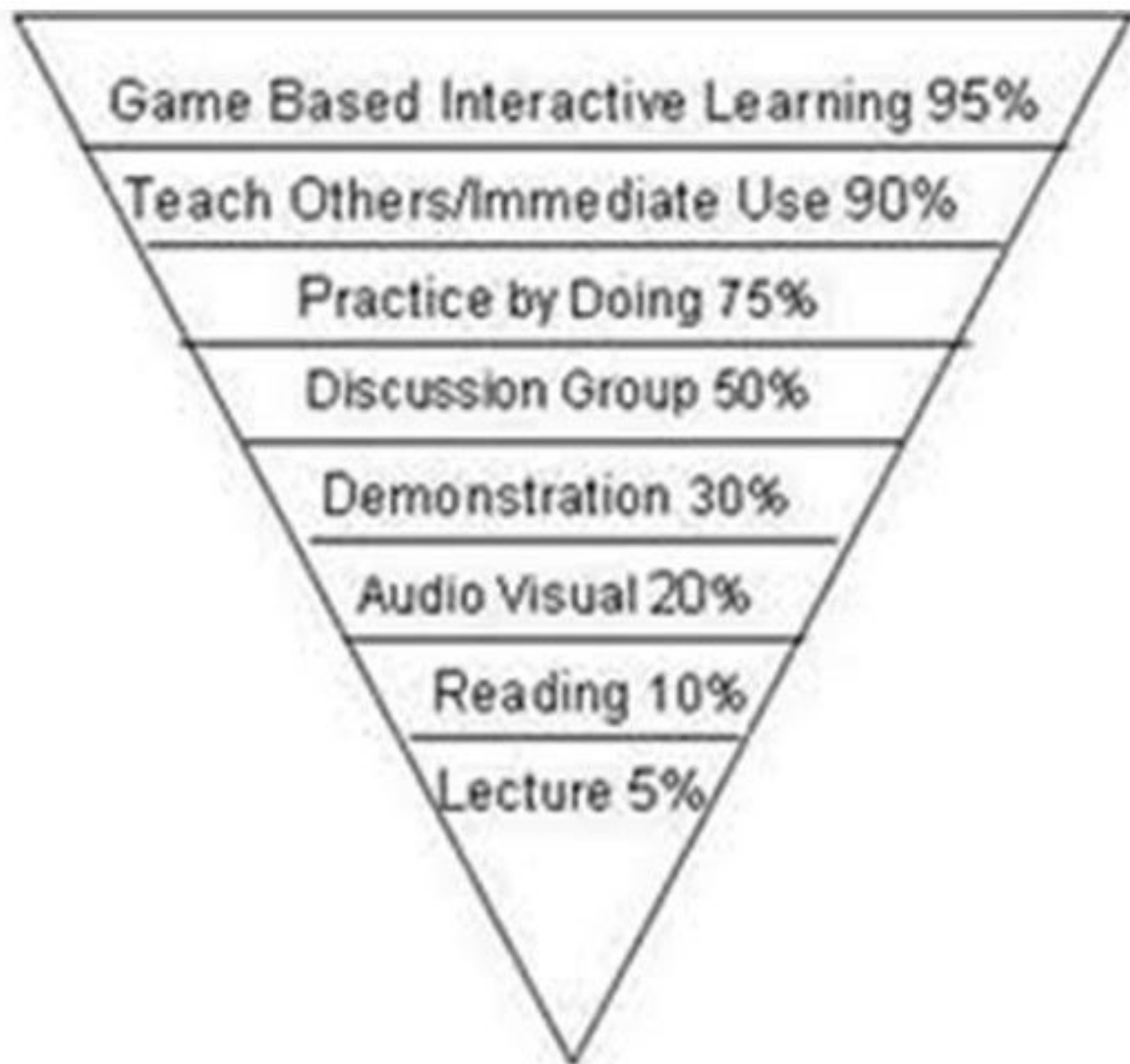
最后是要快乐的学习。这个就更难了，很多知识和技能本身是很复杂的，再加上老师或教程提供者本身对受众的需求不了解，一上来就给你最晦涩难懂的概念，还美其名曰这些都是基础知识，必须打牢基础神马的，先把你吓得尿遁再说。

所以我一向推荐大家直接看e文的教程，或者itunes-u上面的常春藤名校的视频教程，至少可以满足最后一点。老外的入门教程是真正的入门教程，因为小学以上文化基本都能看懂。难道是他们IQ太低？我就不说诺贝尔奖获得者的数据了，单就计算机科学来说，有几个知名的编程语言和框架是天朝程序猿发明的？当然有，但太少！

下图是个学习方式的金字塔，是教育心理学家在研究中发现的学习方法和知识留存率的关系：可以看到常规听课的学习方式效率最低，知识留存率只有5%。也就是我们常说的“之前学的东西早就还给老师了”。阅读可以留下10%，看视频可以留下20%，做演示可以留下30%，参加讨论小组可以留下50%（这就是大家一起学习的好处），实战练习可以留下75%，教别人你想学的东西可以留下90%（神马！！！自己都不会还去教别人？？？）你还真别不相信，要不然知乎和quora会这么火爆？这是有科学依据的（方肘子兄别过来打脸啊，我知道肘子会说这个NTL institute是火星上的野鸡学院，里面的applied behavioral science是子虚乌有的云云~）对数据来源感兴趣的可以直接咨询他们：<http://www.ntl.org>



所以说，如果你要高效学习某个东西，最好可以采用金字塔下方的方法，否则就是事倍功半了。再来看另外一个图。



神马？竟然还有比教别人学和实战练习效率更高的学习方式？当然有，比如我们青少年中二病御宅族最爱的游戏式互动教学。

就拿我最近最爱的炉石传说来说吧，有几个小学以上文化的朋友觉得很难上手的？

因为游戏开发商是这个世界上最了解人性需求的几类人。其他几类人是谁？比如公务猿类，心理学家，传销大湿。。。

游戏式互动教学可以充分利用人性的贪嗔痴（荣耀、互动、独占）、DS对美好虚幻世界的憧憬和向往，并提供实时的奖励或者惩罚（比如最近很变态的Floppy birds），让你在不知不觉中接触了一个全新的世界，并了解了游戏的机制和玩法。

魔兽世界很难玩吗？炉石传授很难玩吗？DNF和LOL很难玩吗？

所以，最有效的教学/学习方式排名如下：

- 1.游戏式互动教学
- 2.自学的同时教别人学
- 3.尽早进入实战
- 4.多加入同类人的讨论群组

至于给别人做演示，看视频（听讲座），阅读书本教材，课堂听讲算是最低效的方式了。很遗憾的是，大家目前主要都是在用最低效的学习方式来学习。

以2013年的苹果年度应用Duolingo为例，它就很成功的融合了上面所提到的几种最高效学习方式。首先，它融合了游戏机制。其实很多儿童教育应用也在尝试这么做，可惜只是借鉴了游戏的皮毛。更准确的说，只是换了一张卡通皮而已。游戏中常用的荣耀、互动、独占、实时奖励、排行、抽奖、节假日活动等都还远远没有用上。

当然，它最吸引人的一点是利用了3。任何一个加入学习的人都会在第一时间接到实际的翻译任务，而不是让你去被动的背单词、语法之类的。比如我们熟知的拓词、猿题库等产品其实也很优秀了，但是在2和3方面做的还远远不足。游戏化互动教学是有了，但没有2和3，更缺少4，是个不小的缺憾。

当然，Duolingo在2方面有所不同，同时在1的游戏式互动教学机制上也还有很大的空间。

结合到我们初学者入门学编程，该怎么来做呢？

1.游戏式互动教学

这方面貌似只有codecademy这个网站算是比较互动了。希望有牛人可以开发出比较NB的学习社交网站（游戏）神马的。这也是我个人一直以来的梦想。

说到这，推荐一本我最近非常喜欢的科幻小说—《游戏玩家》

<http://book.douban.com/review/5874561/>



在依安·班克斯创造的这个文明世界里，所有的学习都是在游戏中进行的，甚至包括课堂点名！当然，类似这样世界的存在需要Oculus Rift之类的设备迅速成熟和流行。



2.自学的同时教别人学

这个其实没那么复杂。比如你可以忽悠身边的女神来学web开发、iPhone应用或者移动游戏开发，然后边教边学。可以完美解决动机和激励的问题，恐怕比游戏式互动教学还要高效吧。弗洛伊德诚不我欺啊！要不然莞式ISO服务会那么火爆？！当然，通过写博客写文章分享心得的方式也可以归为此类方法吧。

3.尽早进入实战

之前提到了，尽量带着一个目标来学。比如你想给女神做个情人节示爱小游戏（比如Rose Crush），那么以此为目标，迅速进入实战，主动学习。不要像在学校里面那样被动的等着老师教，那可是最最低效的学习方式哦！

4.和同类人互动

这个也很简单，经常去stack overflow,github,cocoachina,csdn这些论坛或social coding网站上混个脸熟。

另外现在有很多MOOC网站，比如Coursera,edX,udacity,学堂在线，MOOC学院神马的，可以去了解下。

当然，如果你更喜欢传统的学习方式，那就从传统的听课、看书和教材、看视频听讲座和做展示开始。毕竟每个人的大脑构造不同，有的人就是喜欢更传统的学习方式，也未尝不可。这个世界上可从来不可缺少学霸。

好了，番外篇就这些了。

这些东西或许对你有用，或许没用。不过我还是那个观点，学习某个具体的技能不是关键，毕竟即便如iOS开发这么火的技能5年后很可能也是无人问津。关键的是激发对某件事情的热情，并设定明确的目标，同时知道自己可以怎样来学习新的知识。

这也算是我要在知乎写这些专栏文章的目的吧。

最后回答一个大家常问的问题，我没有在培训机构里面上过班，到目前为止也没有过这种想法。而且我也真心不知道这些培训学校哪个靠谱，最好大家问在里面学过的人或者教过的人，简单搜一下得到的信息还不一定靠谱~

另外，因为时间和精力的原因，组织或参加一些线下活动对我来说暂时是个奢侈的事情。不过欢迎大家随意使用我原创或翻译的这些教程去教别人，也欢迎大家以自己的方式来向K12青少年普及学编程（注意是学编程而不是学写代码）。

下个月会开通个微信订阅号，放一些关于k12青少年如何学编程的文章。

当然，不是K12青少年也没问题，不管你是UI/UX，产品经理，公司保安，清洁工，只要对编程，对创造感兴趣都可以来了解下。

或许编程不会成为你今后的工作或事业，但技多不压身吗。起码以后可以多个谈资，可以了解到习惯生活在二次元世界的宅男程序猿是怎么思考和解决问题的。

想来必然会为生活增添一些乐趣，这就够了~