

序言

公司介绍

路飞学城 (www.luffycity.com) 是由老男孩教育携手一众信任老男孩的学员众筹发起的在线教育平台，致力于帮助有志向的年轻人通过努力学习获得体面工作与品质生活，坚守初心，砥砺前行。



扫描上方二维码加我微信

Alex金角大王

路飞学城创始人/CEO;
中国首位Python讲师
2012年创办国内首个PY培训班
8年面授学员超2万人
教学视频在B站、51cto等学习人数超5百万
曾任职公安部/飞信/汽车之家等公司
51CTO学院最受学员喜爱的10大讲师之一

人生迷茫问题/技术答疑/职业规划都可以加 Alex
个人微信寻求鸡汤解药, 我在这里等你 ☕

本书介绍

- 关于面试题

作者耗费大量时间对近 100+ 套企业面试真题进行归纳和梳理，最终提取 620+ 道面试题目抄录在本书中。

- 关于答案

本书仅含面试题题目，随后会通过【路飞 Python 小课】公众号发布本书的所有答案，请微信扫码关注。



- 关于更新

由于路飞学城 Python 学科的毕业生比较多，回馈给讲师的面试题也在不断更新，本书的内容会进行持续更新，并且会在公众号进行发布，请关注公众号更新。

目录

第一章 Python 基础.....	3
第二章 函数.....	17
第三章 模块.....	29
第四章 面向对象.....	31
第五章 网络和并发编程.....	37
第六章 数据库和缓存.....	41
第七章 前端.....	47
第八章 django	51
第九章 Flask.....	54
第十章 tornado	55
第十一章 api	56
第十二章 git	57
第十三章 爬虫.....	58
第十四章 算法和数据结构.....	59
第十五章 Linux.....	66
第十六章 设计题	70
第十七章 客观题.....	72
第十八章 其他知识点.....	74
第十九章 100+套企业面试真题.....	75
第二十章 附赠.....	76
1. 人生没有白走的路，每一步都算数.....	76
2. 不敢想	79
3. 给一位做技术迷茫的同学回信.....	82
4. 白领如何才能买得起特斯拉.....	84
5. 关于认知、隔绝多维度发展的感触.....	87
6. 自学和报班你想不明白，必然是智商不够.....	89
7. 普通运维就是秋后的蚂蚱	91
8. 哥们别逗了，写个脚本真不叫运维自动化.....	92

第一章 Python 基础

1. 为什么学习 Python?
2. 通过什么途径学习的 Python?
3. 公司线上和开发环境使用的什么系统?
4. Python 和 Java、PHP、C、C#、C++等其他语言的对比?
5. 简述解释型和编译型编程语言?
6. Python 解释器种类以及特点?
7. 位和字节的关系?
8. b、B、KB、MB、GB 的关系?
9. 请至少列举 5 个 PEP8 规范（越多越好）
10. 求结果：or and

```
1. v1 = 1 or 3
2. v2 = 1 and 3
3. v3 = 0 and 2 and 1
4. v4 = 0 and 2 or 1
5. v5 = 0 and 2 or 1 or 4
6. v6 = 0 or False and 1
```

11. ascii、unicode、utf-8、gbk 区别?
12. 字节码和机器码的区别?
13. 三元运算编写格式。
14. 列举你了解的所有 Python2 和 Python3 的区别?
15. Py2 项目如何迁移成 py3?
16. 用一行代码实现数值交换:

```
1. a = 1
2. b = 2
```

17. Python3 和 Python2 中 int 和 long 的区别?

18. xrange 和 range 的区别?
19. 如何实现字符串的反转? 如: name = "wupeiqi" 请反转为 name = "iqiepuw"。
20. 文件操作时: xreadlines 和 readlines 的区别?
21. 列举布尔值为 False 的常见值?
22. 列举字符串、列表、元组、字典每个常用的 5 个方法?
23. is 和 == 的区别?
24. 1、2、3、4、5 能组成多少个互不相同且无重复的三位数
25. 什么是反射? 以及应用场景?
26. 简述 Python 的深浅拷贝?
27. Python 垃圾回收机制?
28. Python 的可变类型和不可变类型的区别?
29. 求结果

```
1. v = dict.fromkeys(['k1', 'k2'], [])
2. v['k1'].append(666)
3. print(v)
4. v['k1'] = 777
5. print(v)
```

30. 一行代码实现删除列表中重复的值?
31. 如何实现 "1,2,3" 变成 ['1','2','3']
32. 如何实现 ['1','2','3'] 变成 [1,2,3]
33. 比较: a = [1,2,3] 和 b = [(1),(2),(3)] 以及 c = [(1),(2),(3)] 的区别?
34. 如何用一行代码生成 [1,4,9,16,25,36,49,64,81,100] ?
35. 常用字符串格式化哪几种?
36. 什么是断言 (assert)? 应用场景?
37. 有两个字符串列表 a 和 b, 每个字符串是由逗号分隔的一些字符:

```

1. a = [ 'a,1',
2.      'b,3,22',
3.      'c,3,4',
4. ]
5.
6. b = [
7.      'a,2',
8.      'b,1',
9.      'd,2',
10. ]
11. 按每个字符串的第一个值，合并 a 和 b 到 c
12.
13. c = [
14.      'a,1,2',
15.      'b,3,22,1',
16.      'c,3,4',
17.      'd,2'
18. ]

```

38. 有一个多层嵌套的列表 A=[1,2,[3,4,["434",...]]], 请写一段代码遍历 A 中的每一个元素并打印出来
39. a = range(10),a[::-3] 的结果是 ____
40. 下面那个命令可以从虚拟环境中退出
- A. deactivate
 - B. exit
 - C. quit
 - D. 以上均可
41. 将列表内的元素,根据位数合并成字典

```

1. lst = [1,2,4,8,16,32,64,128,256,512,1024,32769,65536,4294967296]
2.
3. # 输出
4. {
5.     1:[1,2,3,8],
6.     2:[16,32,64],
7.     3:[128,256,512],
8.     4:[1024,],
9.     5:[32769,65536],
10.    10:[4294967296]
11. }

```

42. 请尽量用简洁的方法将二维数组转换成一维数组

例：转换前 `lst=[[1,2,3],[4,5,6],[7,8,9]]`

转换后 `lst = [1,2,3,4,5,6,7,8,9]`

43. 将列表按下列规则排序, 补全代码

1. 正数在前负数在后
2. 正数从小到大
3. 负数从大到小

例：

排序前 `[7, -8, 5, 4, 0, -2, -5]`

排序后 `[0, 4, 5, 7, -2, -5, -8]`

补全代码：

```
1. sorted(lst, key=lambda x:_____)
```

44. 哈希冲突回避算法有哪几种, 分别有什么特点?

45. 简述 Python 的字符串驻留机制?

46. 以下代码输出是什么? `list=['a','b','c','d','e'] print list[10:]`

- A. `[]`
- B. 程序异常
- C. `['a','b','c','d','e']`
- D. 输出空

47. Python 语言中那些数据类型能做为字典的 key?

- A. 没有限制
- B. 字母数字下划线
- C. 字母
- D. 可被 hash 的类型

48. 以下两段代码的输出一样吗, 占用系统资源一样吗, 什么时候要用 `xrange` 代替 `range`

```
1. for i in range(1): print i  
2.   
3. for i in xrange(1): print i
```

49. 如下代码段

```
1. a = [1,2,3,[4,5],6]
2. b = a
3. c = copy.copy(a)
4. d = copy.deepcopy(a)
5. b.append(10)
6. c[3].append(11)
7. d[3].append(12)
8. 请问 a,b,c,d 的值为
```

50. 现有字典 `d={"a":26,"g":20,"e":20,"c":24,"d":23,"f":21,"b":25}` 请按照字段中的 `value` 字段进行排序。
51. 给定两个 `listA,B`, 请用 `Python` 找出 `A,B` 中相同的元素, `A,B` 中不同的元素
52. 下列叙述中错误的是
- A. 栈是线性结构
 - B. 队列是线性结构
 - C. 线性列表是线性结构
 - D. 二叉树是线性结构
53. 一个栈的输入序列为 `1,2,3,4,5`, 则下列序列中不可能是栈的输出序列的是
- A. `1 5 4 3 2`
 - B. `2 3 4 1 5`
 - C. `1 5 4 2 3`
 - D. `2 3 1 4 5`
54. 下图那些 `PEP` 被认为涉及到了代码规范
- 1. `PEP7`
 - 2. `PEP8`
 - 3. `PEP20`
 - 4. `PEP257`
55. 下面那些是 `Python` 合法的标识符? 那些是 `Python` 的关键字?
- 1. `int32`
 - 2. `40XL`
 - 3. `saving$`
 - 4. `ptint`
 - 5. `this`
 - 6. `self`
 - 7. `0x40L`
 - 8. `true`
 - 9. `big-daddy`
 - 10. `True`
 - 11. `if`

12. do
13. yield
56. 从 0-99 这 100 个数中随机取出 10 个, 要求不能重复, 可以自己设计数据结构
57. python 判断一个字典中是否有这些 key: "AAA",'BB','C',"DD",'EEE'(不使用 for while)
58. 有一个 list["This","is","a","Boy","!"], 所有元素都是字符串, 对他进行大小写无关的排序
59. 描述下 dict 的 item()方法与 iteritems()的不同
60. 请列举你所知道的 Python 代码检测工具及他们间的区别?
61. 介绍一下 try except 的用法和作用?
62. 输入一个字符串, 返回倒序排列的结果 如: abcdef, 返回 fedcba
63. 阅读以下代码, 并写出程序的输出结果

```
1. alist = [2,4,5,6,7]
2. for var in alist:
3.     if var %2 ==0:
4.         alist.remove(var)
5. alist 的最终结果是
```

64. 现有列表 alist=[3,1,-4,-2],按其元素的绝对值大小进行排序?
65. 填空题

1. 表达式 3<4<5 是哪一个表达式的缩写_____.
2. 获取 Python 解释器版本的方法是:_____.
3. 如果模块是被导入的, __name__ 的值是_____, 如果模块是被直接执行的 __name__ 的值是_____.
4. Python 的内存管理中, 为了追踪内存中的对象, 使用了_____这一简单技术
5. Python 的标准解释器是有 C 语言实现的, 称为_____, 有 Java 实现的被称为_____.
6. Python 中, _____语句能直接显示的释放内存资源
7. Python 的乘方运算符是_____

66. 现有字典 `mydict` 和变量 `onekey`, 请写出从 `mydict` 中取出 `onekey` 值的方法 (不止一种写法, 多写加分, 并请叙述不同写法的区别, `mydict` 中是否存在 `onekey` 的键值, 不确定)
67. 现有一列表 `alist`, 请写出两种去除 `alist` 中重复元素的方法, 其中:
- 要求保持原有列表中元素的排列顺序。
 - 无需考虑原有列表中元素的排列顺序。
68. 那些情况下, $y \neq x - (x-y)$ 会成立?
69. 用 Python 实现 99 乘法表(用两种不同的方法实现)
70. 获取 `list` 中的元素个数, 和向末尾追加元素所用的方法分别是什么?
71. 判断 `dict` 中有没有某个 `key` 用的方法是什么?
72. 填空
- ```
l=range(100)
```
1. 如何取第一个到第三个元素用的是\_\_\_\_\_
  2. 如何取倒数第二个元素\_\_\_\_\_
  3. 如何取后十个\_\_\_\_\_
73. 如何判断一个变量是否是字符串?
74. `list` 和 `tuple` 有什么不同?
75. `a = dict(zip(["a","b","c","d","e"],(1,2,3,4,5)))` 请问 `a` 是什么?
76. 一行代码生成列表 `[1,4,9,16,25,36,49,64,81,100]`。
77. 以下叙述正确的是
- A. `continue` 语句的作用是结束整个循环的执行
  - B. 只能在循环体和 `switch` 语句体内使用 `break` 语句
  - C. 在循环体内使用 `break` 语句或者 `continue` 语句的作用相同
  - D. 从多层循环嵌套中退出时, 只能使用 `goto` 语句
78. 读代码

```
1. for i in range(5,0,-1):
2. print(i)
3. 请在下面写出打印结果
```

79. 写结果

```
1. x= "foo"
2. y = 2
3. print x+y
```

- A. foo
- B. foo foo
- C. foo 2
- D. 2
- E. An exception is thrown

80. 求结果

```
1. kvps = {"1":1,"2":2}
2. theCopy = kvps
3. kvps["1"] = 5
4. sum = kvps["1"] + theCopy ["1"]
5. print sum
```

- A. 1
- B. 2
- C. 7
- D. 10

81. python 里如何实现 tuple 和 list 的转化

82. type(1+2L\*3.14)的结果是

- A. int
- B. long
- C. float
- D. str

83. 若 k 为整型, 下列 while 循环执行的次数为

```
1. k = 1000
2. while k>1:
3. print k
4. k = k/2
```

84. 以下何者是不合法的布尔表达式

- A. x in range(6)
- B. 3 = a
- C. e>5 and 4==f
- D. (x-6)>5

85. python 不支持的数据类型有
- A. char
  - B. int
  - C. float
  - D. list
86. 如何在 Python 中拷贝一个对象, 并说明他们之间的区别?
87. 99(10 进制)的八进制表示是什么?
88. 下列 Python 语句正确的是
- 1. min = x is x<y=y
  - 2. max = x>y?x:y
  - 3. if(x>y) print x
  - 4. while True:pass
89. list 对象 alist =  
[{'name':'a','age':20},{ 'name':'b','age':30},{ 'name':'v','age':25},]按 alist 中元素的 age 由大到小排序。
90. 关于 Python 程序的运行性能方面, 有什么手段能提升性能?
91. Python 是如何进行内存管理的? Python 的程序会内存泄漏吗?说说有没有什么方面阻止或检测内存泄漏
92. 详细说说 tuple,list,dict 的用法, 他们的特点
93. 一个大小为 100G 的文件 etl\_log.txt, 要读取文件中的内容, 写出具体过程代码?
94. 已知 Alist=[1,2,3,1,2,1,3],如何根据 Alist 得到[1,2,3]
95. 已知 stra = 'wqedsfsdrfweedqwedqw'
- 1. 如何获取最后两个字符
  - 2. 如何获取第二个和第三个字符
96. 已知 Alist = ["a","b","c"],将 Alist 转化为'a,b,c'的实现过程
97. 已知 ip='192.168.0.100' 代码实现提取各部分并写入列表。
98. python 代码如何获取命令行参数?
99. 写代码

```

1. tupleA = ("a", "b", "c", "d", "e")
2. tupleB = (1, 2, 3, 4, 5)
3. RES = {"a": 1, "b": 2, "c": 3, "d": 4, "e": 5}

```

写出由 tupleA 和 tupleB 得到 res 的及具体实现过程

100. 选出一下表达式表述正确的选项

- A. {1:0, 2:0, 3:0}
- B. {'1':0, '2':0, '3':0}
- C. {(1,2):0, (4,3):0}
- D. {[1,2]:0, [4,3]:0}
- E. {{1,2}:0, {4,3}:0}

101. what gets printde() ?

```

1. kvps = {"1":1, '2':2}
2. theCopy = kvps.copy()
3. kvps["1"] = 5
4. sum = kvps["1"] + theCopy["1"]
5. print sum

```

- A. 1
- B. 2
- C. 6
- D. 10
- E. An execption is thrown

102. what gets printde() ?

```

1. numbers = [1,2,3,4]
2. numbers.append([5,6,7,8])
3. print len(numbers)

```

- A. 4
- B. 5
- C. 8
- D. 12
- E. An exception is thrown

103. what getsprintde() ?

```

1. names1 = ["Amir", "Barry", "Chaies", "Dao"]
2. if "amir" in names1:
3. print 1
4. else:
5. print 2

```

- A. 1

- B. 2
- C. An exception is thrown

104. what getsprintde() ? Assuming ptrhon version2.x()

```
1. print(type(1/2))
```

- A. int
- B. float
- C. 0
- D. 1
- E. 0.5

105. 以下用来管理 Python 库管理工具的是

- A. APT
- B. PIP
- C. YUM
- D. MAVEN

106. which numbers are printed ()?

```
1. for i in range(2):
2. print i
3.
4. for i in range(4,6):
5. print i
```

- A. 2,4,6
- B. 0,1,2,4,5,6
- C. 0,1,4,5
- D. 0,1,4,5,6,7,8,9
- E. 1,2,4,5,6

107. 求结果

```
1. import math
2. print (math.floor(5.5))
```

- A. 5
- B. 5.0
- C. 5.5
- D. 6
- E. 6.0

108. 关于 Python 的内存管理, 下列说法错误的是

- A. 变量不必事先声明
- B. 变量无需先创建和赋值而直接使用

- C. 变量无需指定类型
- D. 可以使用 `del` 释放资源

109. 下面那个不是 Python 合法的标识符

- A. `int32`
- B. `40x1`
- C. `self`
- D. `name`

110. 下列哪种说法是错误的

- A. 除字典类型外, 所有标准对象均可用于布尔测试
- B. 空字符串的布尔值是 `False`
- C. 空列表对象的布尔值是 `False`
- D. 值为 `0` 的任何数字对象的布尔值是 `False`

111. 下列表达是的值为 `True` 的是

- A. `5+4j > 2-3j`
- B. `3>2>2`
- C. `(3,2)<("a","b")`
- D. `"abc" > 'xyz'`

112. 关于 Python 的复数, 下列说法错误的是

- A. 表示复数的语法是 `real+imagej`
- B. 实部和虚部都是浮点数
- C. 虚部后缀必须是 `j`, 且必须小写
- D. 方法 `conjugate` 返回复数的共轭复数

113. 关于字符串下列说法错误的是

- A. 字符应视为长度为 `1` 的字符串
- B. 字符串以 `\0` 标志字符串的结束
- C. 既可以用单引号, 也可以用双引号创建字符串
- D. 在三引号字符串中可以包含换行回车等特殊字符

114. 以下不能创建一个字典的语句是

- A. `dic1 = {}`
- B. `dic2 = {3:5}`
- C. `dic3 = {[1,2,3]: "usetc"}`
- D. `dic4 = {(1,2,3): "usetc"}`

115. python 里面如何拷贝一个对象?(赋值,浅拷贝,深拷贝的区别)

116. 描述在 python 中的元祖,列表,字典的区别,并且分别写一段定义,添加,删除操作的代码片段。

117. 选择结果

```

1. names1 = ["Amir", "Barry", "Chales", "Dao"]
2. names2 = names1
3. names3 = names1[:]
4. name2[0] = "Alice"
5. names3[1] = "Bob"
6. sum = 0
7. for ls in (names1, names2, names3):
8. if ls[0] == "Alice":
9. sum+=1
10. if ls[1]=="Bob":
11. sum+=10
12. print sum

```

- A. 11
- B. 12
- C. 21
- D. 22
- E. 23

118. 下面程序的输出结果是

```

1. x = True
2. y = False
3. z = False
4.
5. if x or y and z:
6. print 'yes'
7. else:
8. print 'no'

```

119. 1 or 2 和 1 and 2 输出分别是什么? 为什么

120. 1 < (2 == 2) 和 1 < 2 == 2 的结果分别是什么, 为什么

121. 如何打乱一个排好序的 list 对象 alist

122. 如何查找一个字符串中特定的字符? find 和 index 的差异?

123. 把 aaabbcccd 这种形式的字符串压缩成 a3b2c3d1 这种形式。

124. Python 一个数如果恰好等于它的因子之和, 这个数就称为"完数"。例如 6=1+2+3. 编程找出 1000 以内的所有完数。

125. 输入一个字符串, 输出该字符串中字符的所有组合。

例如：输入字符串"1,2,3"，则输出为 1,2,3,12,13,23,123(组合数，不考虑顺序，所以 12 和 21 是等价的)

126. 执行以下代码后, i 和 n 的值为

```
1. int i=10;
2. int n=i++%5
```

- A. 10, 0
- B. 10, 1
- C. 11, 0
- B. 11, 1

127. 执行以下代码段后, x 的值为

```
1. int x=10;
2. x+=x-=x-x;
```

- A. 10
- B. 20
- C. 30
- D. 40

128. 对于一个非空字符串, 判断其是否可以有一个子字符串重复多次组成, 字符串只包含小写字母且长度不超过 10000

示例 1:

- 1. 输入 "abab"
- 2. 输出 True
- 3. 样例解释: 输入可由 "ab" 重复两次组成

示例 2:

- 1. 输入 "abcabcabc"
- 2. 输出 True
- 3. 样例解释: 输入可由 "abc" 重复三次组成

示例 3:

- 1. 输入 "aba"
- 2. 输出 False



## 第二章 函数

1. 通过代码实现如下转换：

二进制转换成十进制：v = "0b1111011"

十进制转换成二进制：v = 18

八进制转换成十进制：v = "011"

十进制转换成八进制：v = 30

十六进制转换成十进制：v = "0x12"

十进制转换成十六进制：v = 87

2. Python 递归的最大层数？
3. 列举常见的内置函数？
4. filter、map、reduce 的作用？
5. 什么是闭包？
6. 简述 生成器、迭代器、装饰器以及应用场景？
7. 使用生成器编写 fib 函数, 函数声明为 fib(max), 输入一个参数 max 值, 使得该函数可以这样调用。

```
1. for i in range(0,100):
2. print fib(1000)
```

并产生如下结果(斐波那契数列), 1, 1, 2, 3, 5, 8, 13, 21...

8. 一行代码, 通过 filter 和 lambda 函数输出以下列表索引为基数对应的元素。

```
list_a=[12,213,22,2,2,2,22,2,2,32]
```

9. 写一个 base62encode 函数, 62 进制。

即: 0123456789AB...Zab...z (10 个数字+26 个大写字母+26 个小写字母)。

```
base62encode(1)=1
```

```
base62encode(61) = z
```

```
base62encode(62)=10
```

10. 请实现一个装饰器, 限制该函数被调用的频率, 如 10 秒一次
11. 请实现一个装饰器, 通过一次调用使函数重复执行 5 次。
12. python 一行 print 出 1~100 偶数的列表, (列表推导式, filter 均可)
13. 解释生成器与函数的不同, 并实现和简单使用 generator.

14. 列表推导式和生成器表达式 `[i % 2 for i in range(10)]` 和 `(i % 2 for i in range(10))` 输出结果分别是什么？
15. `map(str,[1,2,3,4,5,6,7,8,9])` 输出是什么？
16. python 中定义函数时如何书写可变参数和关键字参数？
17. Python3.5 中 `enumerate` 的意思是什么？
18. 说说 Python 中的装饰器,迭代器的用法:描述下 `dict` 的 `item` 方法与 `iteritems` 方法的不同
19. 是否使用过 `functools` 中的函数？其作用是什么？
20. 如何判断一个值是函数还是方法？
21. 请编写一个函数实现将 IP 地址转换成一个整数。

如 10.3.9.12 转换规则为:

|    |          |
|----|----------|
| 10 | 00001010 |
| 3  | 00000011 |
| 9  | 00001001 |
| 12 | 00001100 |

再将以上二进制拼接起来计算十进制结果: 00001010 00000011 00001001 00001100 = ?

22. `lambda` 表达式格式以及应用场景？
23. `pass` 的作用？
24. `*arg` 和 `**kwarg` 作用？
25. 如何在函数中设置一个全局变量？
26. 请写出打印结果：

```

1. # 例 1
2. def func(a,b=[]):
3. b.append(a)
4. print(b)
5. func(1)
6. func(1)
7. func(1)
8. func(1)
9.
10. # 例 2
11. def func(a,b={}):
12. b[a] = 'v'
13. print(b)
14. func(1)
15. func(2)

```

27. 求结果：lambda

```

1. def num():
2. return [lambda x:i*x for i in range(4)]
3. print([m(2) for m in num()])

```

28. 简述 yield 和 yield from 关键字。

29. 有 processFunc 变量,初始化为 processFunc = collapse and (lambda s:" ".join(s.split())) or (lambda s:s)

调用上下文如下

```

1. collapse = True
2. processFunc = collapse and (lambda s:" ".join(s.split())) or (lambda s:s)
3. print processFunc("i\tam\ntest\tobject !")
4.
5. collapse = False
6. processFunc = collapse and (lambda s:" ".join(s.split())) or (lambda s:s)
7. print processFunc("i\tam\ntest\tobject !")

```

以上代码会在控制台输出什么?

30. 请给出下面代码的输出结果

```

1. a = 1
2. def fun(a):
3. a = 2
4.
5. fun(a)
6. print a
7.
8. a = []
9. def fun(a):
10. a.append(1)
11.
12. fun(a)
13. print a

```

31. 全局变量和局部变量的区别, 如何给 function 里面的一个全局变量赋值

32. 什么是 lambda 函数, 下面这段代码的输出是什么

```

1. nums = range(2,20)
2. for i in nums:
3. nums = filter(lambda x:x==i or x % i, nums)
4. nums

```

33. 指出下面程序存在的问题

```

1. def Lastlindextem(src, index):
2. '''请返回传入 src 使用空格或者"\ "切分后的倒数第 index 个子串'''
3. return src.split("\ ")[-index]

```

34. 有一个数组[3,4,1,2,5,6,6,5,4,3,3] 请写一个函数, 找出该数组中没有重复的数的总和.(上面数据的么有重复的总和为 1+2=3)

35. 求打印结果

```

1. arr = [1,2,3]
2. def bar():
3. arr+=[5]
4.
5. bar()
6. print arr

```

- A. error
- B. [5]

- C. [1,2,3]
- D. [1,2,3,5]

36. 请写一个函数,计算出如下几个字母代表的数字

AB-CD=EF  
EF+GH = PPP

37. 请给出下面代码片段的输出

```
1. def say_hi(func):
2. def wrapper(*args,**kwargs):
3. print("HI")
4. ret = func(*args,**kwargs)
5. print("BYE")
6. return ret
7. return wrapper
8.
9. def say_yo(func):
10. def wrapper(*args,**kwargs):
11. print("YO")
12. return func(*args,**kwargs)
13. return wrapper
14.
15. @say_hi
16. @say_yo
17. def func():
18. print("ROCK & ROLL")
19.
20. func()
```

38. 请简述标准库中 `functools.wraps` 的作用

39. 请给出下面代码片段的输出

```
1. def test():
2. try:
3. raise ValueError("something wrong")
4. except ValueError as e:
5. print("Error occurred")
6. return
7. finally:
8. print("Done")
9.
10. test()
```

40. 下面的函数,那些会输出 1,2,3 三个数字

```
1. for i in range(3):
2. print i
3. alist = [0,1,2]
4. for i in alist:
5. print i+1
6. i = 1
7. while i<3:
8. print i
9. i+=1
10.
11. for i in range(3):
12. print i+1
```

41. 以下函数需要在其中引用一个全局变量 k, 请填写语句

```
1. def fun():
2. _____
3. k = k+1
```

42. 请把以下函数转化为 python lambda 匿名函数

```
1. def add(x,y):
2. return x+y
```

43. 阅读以下代码, 并写出程序的输出结果

```

1. my_dict = {"a":0,"b":1}
2.
3. def func(d):
4. d["a"]=1
5. return d
6.
7. func(my_dict)
8. my_dict["c"]=2
9. print my_dict

```

#### 44. 填空题

```

1. # 有函数定义如下
2. def calc(a,b,c,d=1,e=2):
3. return (a+b)*(c-d)+e
4.
5. # 请分别写出以下标号代码的输出结果，如果出错请写出 Error
6. print calc(1,2,3,4,5) # ____
7. print calc(1,2,3) # ____
8. print calc(1,2) # ____
9. print calc(1,2,3,e=4) # ____
10. print calc(e=4, c=5, a=2,b=3) # ____
11. print calc(1,2,3, d=5,4) # ____

```

#### 45. def(a, b=[] )这种写法有什么陷阱？

#### 46. 函数

```

1. def add_end(l=[]):
2. l.append("end")
3. return l
4.
5. add_end() # 输出什么
6. add_end() # 再次调用输出什么？为什么

```

#### 47. 函数参数 \*args,\*\*kwargs 的作用是什么

```

1. def func(a,b,c=0,*args,**kwargs):
2. pass

```

48. 可变参数定义 `*args,**kwargs` 的区别是什么?并且写出下边代码的输入内容

```
1. def foo(*args,**kwargs):
2. print("args=",args)
3. print("kwargs=",kwargs)
4. print("-----")
5.
6. if __name__ == '__main__':
7. foo(1,2,3,4)
8. foo(a=1,b=2,c=3)
9. foo(1,2,3,4,a=1,b=2,c=3)
10. foo("a",1,None,a=1,b="2",c=3)
```

49. 请写出 log 实现(主要功能时打印函数名)

```
1. @log
2. def now():
3. print "2013-12-25"
4.
5. now()
6. 输出
7. call now()
8. 2013-12-25
```

50. Python 如何定义一个函数

- A. class <name>(<Type> arg1, <type> arg2, ...)
- B. function <name>(arg1,arg2,...)
- C. def <name>(arg1, arg2,...)
- D. def <name>(<type> arg1, <type> arg2...)

51. 选择代码运行结果



```
1. country_counter = {}
2.
3. def addone(country):
4. if country in country_counter:
5. country_counter[country] += 1
6. else:
7. country_counter[country] = 1
8.
9. addone("China")
10. addone("Japan")
11. addone("china")
12. print len(country_counter)
```

- A. 0
- B. 1
- C. 2
- D. 3
- E. 4

52. 选择输出结果

```
1. def doff(arg1,*args):
2. print type(args)
3.
4. doff("applea","bananas","cherry")
```

- A. str
- B. int
- C. tuple
- D. list
- E. dict

53. 下面程序的输出结果是

```
1. d = lambda p:p*2
2. t = lambda p:p*3
3.
4. x = 2
5. x = d(x)
6. x = t(x)
7. x = d(x)
8. print x
```

54. 什么是 lambda 表达式?

55. 以下代码输出是什么,请给出答案并解释

```
1. def multipliers():
2. return [lambda x:x*i for i in range(4)]
3.
4. print([m(2) for m in multipliers()])
```

请修改 `multipliers` 的定义来产生期望的结果

56. 有  $0 < x \leq 10$ ,  $10 < x \leq 20$ ,  $20 < x \leq 30$ , ...,  $190 < x \leq 200$ ,  $200 < x$  这样的 21 个区间分别对应 1-21 二十一个级别, 请编写一个函数 `level(x)` 根据输入数值返回对应级别。

57. 写函数

有一个数据结构如下所示, 请编写一个函数从该结构数据中返回由指定的字段和对应的值组成的字典。如果指定字段不存在, 则跳过该字段。

```
1. data:{
2. "time":"2016-08-05T13:13:05",
3. "some_id":"ID1234",
4. "grp1":{"fld1":1, "fld2":2,},
5. "xxx2":{"fld3":0, "fld4":0.4,},
6. "fld6":11,
7. "fld7": 7,
8. "fld46":8
9. }
10.
11. fields:由"|"连接的以 fld 开头的字符串, 如 fld2|fld7|fld29
12.
13. def select(data,fields):
14. return result
```

58. 补全代码

若要将  $N$  个 `task` 分配给  $N$  个 `worker` 同时去完成, 每个 `worker` 分别都可以承担这  $N$  个 `task`, 但费用不同. 下面的程序用回溯法计算总费用最小的一种工作分配方案, 在该方案中, 为每个 `worker` 分配 1 个 `task`. 程序中,  $N$  个 `task` 从 0 开始顺序编号,  $N$  个 `worker` 也从 0 开始顺序编号, 主要的变量说明如下:

- `ci`: 将任务  $i$  分配给 `worker j` 的费用
- `task[i]`: 值为 0 表示 `task i` 未分配, 值为  $j$  表示 `task, i` 分配给 `worker j`
- `worker[k]` 值为 0 表示未分配 `task`, 值为 1 表示 `worker k` 已分配 `task`;
- `mincost`: 最小总费用

## 程序

写个函数接收一个文件夹名称作为参数, 显示文件夹中文件的路径, 以及其中包含文件夹中文件的路径。

```

1. N=8
2. mincosr = 65535
3. worker = []
4. task = []
5. temp = []
6. c = []
7. def plan(k, cost):
8. global mincosr
9. if __ (1) __ and cost<mincosr:
10. mincosr = cost
11. for i in xrange(N):
12. temp[i] = task[i]
13. else:
14. for i in xrange(N):
15. if worker[i] ==0 and __ (2) __:
16. worker[i] = 1
17. task[k] = __ (3) __
18. plan(__ (4) __, cost+c[k][i])
19. __ (5) __
20. task[k] = 0
21. def main():
22. for i in xrange(N):
23. worker.append(0)
24. task.append(0)
25. temp.append(0)
26. c.append(0)
27. for j in xrange(N):
28. print "请输入 worker"+str(i)+"完成 task" + str(j)+"的花费"
29. input_value = input()
30. c[i].append(int(input_value))
31. plan(0,0)
32. print('\n 最小费用: '+str(mincosr))
33. for i in xrange(N):
34. print "Task"+str(i)+"is assigned to Worker" + str(temp[i])
35. if __name__ == "__main__":
36. main()

```



## 第三章 模块

1. 列举常用的模块。
2. 如何安装第三方模块？
3. re 的 match 和 search 区别？
4. 什么是正则的贪婪匹配？或 正则匹配中的贪婪模式与非贪婪模式的区别？
5. 如何生成一个随机数？
6. 如何使用 python 删除一个文件？
7. logging 模块的作用？以及应用场景？
8. json 序列化时，可以处理的数据类型有哪些？如何定制支持 datetime 类型？
9. json 序列化时，默认遇到中文会转换成 unicode，如果想要保留中文怎么办？
10. 写代码实现查看一个目录下的所有文件。
11. 用 Python 匹配 HTML tag 的时候,<.>和<.>有什么区别？
12. 如何判断一个邮箱合法
13. 请写出以字母或下划线开始, 以数字结束的正则表达式
14. 下面那些是 Python 开发网络应用的框架
  1. Django
  2. Yii
  3. Struts
  4. Rails
  5. Uliweb
  6. CodeIgniter
  7. gevent
  8. Flask
  9. web2py
  10. Pylons
  11. Tornado
  12. Twisted
  13. TurboGears
15. 写 Python 爬虫都用到了那些模块, 分别是做什么用的？
16. sys.path.append("/root/mods")的作用？
17. 列举出 Python 中比较熟知的爬虫框架

18. 输入某年某日,判断这是这一年的第几天?(可以用 Python 的内置模块)
19. 使用过 Python 那些第三方组件?

## 第四章 面向对象

1. 简述面向对象的三大特性。
2. 什么是鸭子模型？
3. `super` 的作用？
4. `mro` 是什么？
5. 什么是 `c3` 算法？
6. 列举面向对象中带双下划线的特殊方法。
7. 双下划线和单下划线的区别？
8. 实例变量和类变量的区别？
9. 静态方法和类方法区别？
10. `isinstance` 和 `type` 的作用？
11. 有用过 `with statement`（语句）吗？它的好处是什么？
12. 下列数据结构中,哪一种是不可迭代的
  - A. `dict`
  - B. `object`
  - C. `set`
  - D. `str`
13. 实现一个 `Singleton` 单例类, 要求遵循基本语言编程规范（用尽量多的方式）。
14. 请描述 `with` 的用法, 如果自己的类需要支持 `with` 语句, 应该如何书写？
15. `python` 中如何判断一个对象是否可调用? 那些对象可以是可调用对象? 如何定义一个类, 使其对象本身就是可调用对象？
16. 请实现一个栈。
17. 关于 `Python` 类的继承不正确的说法是?(多选)
  - A. `Python` 类无法继承
  - B. 可以继承, 无法执行父类的构造函数
  - C. 可以有多个父类
  - D. 只能有一个父类
18. 实现一个 `hashtable` 类, 对外暴露的有 `add` 和 `get` 方法, 满足以下测试代码

```

1. def test():
2. import uuid
3. name = {"name", "web", "python"}
4. ht = HashTable()
5. for key in names:
6. value = uuid.uuid4()
7. ht.add(key,value)
8. print("add 元素",key,value)
9.
10. for key in names:
11. v = ht.get(key)
12. print("get 元素",key, v)

```

19. 请用两个队列来实现一个栈(给出伪代码即可)

20. 已知如下链表类, 请实现单链表逆置

```

1. class Node:
2. def __init__(self, value, next):
3. self.value = value
4. self.next = next

```

21. 类的加载顺序(类中有继承有构造有静态)?

22. 参考下面代码片段

```

1. class Context:
2. pass
3.
4. with Content() as ctx:
5. ctx.do_something()

```

请在 **Context** 类下添加代码完成该类的实现

23. 以下代码输出是什么? 请给出答案并解释。



```

1. class Parent(object):
2. x = 1
3.
4. class Child1(Parent):
5. pass
6.
7. class Child2(Parent):
8. pass
9.
10. print Parent.x, Child1.x, Child2.x
11.
12. Child1.x = 2
13. print Parent.x, Child1.x, Child2.x
14.
15. Parent.x = 3
16. print Parent.x, Child1.x, Child2.x

```

eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

24. 函数 `del_node(self,data)`的功能: 在根节点指针为 `root` 的二叉树(又称二叉排序树)上排除数值为 `K` 的节点,若删除成功,返回 `0`,否则返回`-1`, 概述节点的定义类型为

```
1. class Node(object):
2. def __init__(self,data):
3. self.data = data # 节点的数值
4. self.left_child = Node # 指向左右子树的指针
5. self.right_child = Node
6.
7. def set_data(self,data):
8. self.data = data
```

25. 请给出下面代码片段的输出，请简述上面代码需要改进的地方？

```
1. class Singleton:
2. _instance = None
3. def __new__(cls, *args, **kwargs)
4. print("New")
5. if cls._instance is None:
6. print("Create")
7. cls._instance = super().__new__(cls,*args, **kwargs)
8. return cls._instance
9.
10. def __init__(self):
11. print("Initalize")
12. self.prop = None
13.
14. s1 = Singleton()
15. s2 = singleton()
```

26. 请简单解释 Python 中的 static method(静态方法)和 class method(类方法), 并将以下代码填写完整。

```

1. class A(object):
2. def foo(self,x)
3. print 'executing foo(%s, %s)'%(self,x)
4.
5. @classmethod
6. def class_foo(cls,x):
7. print 'executing class_foo(%s, %s)'%(cls,x)
8.
9. @staticmethod
10. def static_foo(x):
11. print 'executing static_foo(%s)'%(x)
12.
13. a= A()
14. # 调用 foo 函数,参数传入 1
15. _____
16. # 调用 class_foo 函数,参数传入 1
17. _____
18. # 调用 static_foo 函数,参数传入 1
19. _____

```

27. 已知一个订单对象（tradeOrder）有如下字段：

| 字段英文名    | 中文名    | 字段类型               | 取值                         |
|----------|--------|--------------------|----------------------------|
| Id       | 主键     | Long               | 123456789                  |
| Name     | 姓名     | String             | 张三                         |
| Items    | 商品列表集合 | List<商品><br>(关联商品) | 查找商品对象，一个订单有两个商品。商品字段任意取值。 |
| IsMember | 是否是会员  | Boolean            | True                       |

|              |       |            |      |
|--------------|-------|------------|------|
| CouponAmount | 优惠券金额 | BigDecimal | Null |
|--------------|-------|------------|------|

商品对象

| 字段英文名称 | 中文名  | 字段类型   | 取值        |
|--------|------|--------|-----------|
| Id     | 主键   | Long   | 987654321 |
| Name   | 商品名称 | String | 手机        |

问题：若将订单对象转成 JSON 格式，请书写出转换后的 JSON 字符串。

## 28. 写代码(栈与队列)

编程实现一个先进先出的队列类, 能指定初始化时的队列大小, 以及 enqueue, dequeue, isEmpty, isfull 四种方法

使用方法如下

```

1. s = Queue(2) # 初始化一个大小为 2 的队列
2. s.is_empty() # 初始化后, 队列为空, 返回 True
3. s.enqueue(1) # 将 1 加入队列
4. s.enqueue(2) # 将 2 加入队列
5. s.isfull() # 加入了两个元素, 队列已满, 返回 True
6. s.dequeue() # 移除一个元素, 返回 1
7. s.dequeue() # 移除一个元素, 返回 2
8. s.is_empty() # 队列已经为空, 返回 True

```

## 第五章 网络和并发编程

1. python 的底层网络交互模块有哪些？
2. 简述 OSI 七层协议。
3. 什么是 C/S 和 B/S 架构？
4. 简述 TCP 三次握手、四次挥手的流程。
5. 什么是 arp 协议？
6. TCP 和 UDP 的区别？为何基于 tcp 协议的通信比基于 udp 协议的通信更可靠？
7. 什么是局域网和广域网？
8. 什么是 socket？简述基于 tcp 协议的套接字通信流程。
9. 什么是粘包？socket 中造成粘包的原因是什么？哪些情况会发生粘包现象？
10. IO 多路复用的作用？
11. 什么是防火墙以及作用？
12. select、poll、epoll 模型的区别？
13. 简述 进程、线程、协程的区别 以及应用场景？
14. 什么是 GIL 锁？
15. Python 中如何使用线程池和进程池？
16. threading.local 的作用？
17. 进程之间如何进行通信？
18. 什么是并发和并行？
19. 解释什么是异步非阻塞？
20. 路由器和交换机的区别？
21. 什么是域名解析？
22. 如何修改本地 hosts 文件？
23. 生产者消费者模型应用场景？
24. 什么是 cdn？

25. 程序从 Flag A 执行到 Flag B 的时间大致为多少秒

```
1. import threading
2. import time
3. def _wait():
4. time.sleep(60)
5. #FlagA
6. t = threading.Thread(target=_wait, daemon=False)
7. t.start()
8. #FlagB
```

26. 有 A.txt 和 B.txt 两个文件, 使用多进程和进程池的方式分别读取这两个文件

27. 以下那些是常见的 TCP Flags?(多选)

- A. SYN
- B. RST
- C. ACK
- D. URG

28. 下面关于网络七层和四层的描述, 那条是错误的?

- A. SNMP 工作在四层
- B. 四层是指网络的传输层, 主要包括 IP 和端口信息
- C. 七层是指网络的应用层(协议层), 比如 http 协议就工作在七层
- D. 四层主要应用于 TCP 和 UDP 的代理, 七层主要应用于 HTTP 等协议的代理

29. traceroute 一般使用的是那种网络层协议

- A. VRRP
- B. UDP
- C. ARP
- D. ICMP

30. iptables 知识考察, 根据要求写出防火墙规则?

- A. 屏蔽 192.168.1.5 访问本机 dns 服务端口
- B. 允许 10.1.1.0/24 访问本机的 udp 8888 9999 端口

31. 业务服务器 192.168.1.2 访问 192.168.1.3 数据接口, 无法正常返回数据, 请根据以上信息写出排查思路。

32. 请实现一个简单的 socket 编程, 要求

- 1. 实现 server 端的功能即可
- 2. 遵循基本语言编程规范

33. 谈一下对于多线程编程的理解,对于 CPU 密集型怎样使用多线程,说说线程池,线程锁的用法,有没有用过 multiprocessing 或 concurrent.future?
34. 关于守护线程的说法,正确的是
- A. 所有非守护线程终止,即使存在守护线程,进程运行终止
  - B. 所有守护线程终止,即使存在非守护线程,进程运行终止
  - C. 只要有守护线程或者非守护线程其中之一存在,进程就不会终止
  - D. 只要所有的守护线程和非守护线程中终止运行之后,进程才会终止
35. TCP 协议在每次建立或者拆除连接时,都要在收发双方之间交换()报文
- A.一个 B.两个
  - C.三个 D.四个
36. 描述多进程开发中 join 与 daemon 的区别
37. 请简述 GIL 对 Python 性能的影响
38. 曾经在哪里使用过:线程、进程、协程?
39. 请使用 yield 实现一个协程?
40. 请使用 python 内置 async 语法实现一个协程?
41. 简述线程死锁是如何造成的? 如何避免?
42. asynio 是什么?
43. gevent 模块是什么?
44. 什么是 twisted 框架?
45. 什么是 LVS?
46. 什么是 Nginx?
47. 什么是 keepalived?
48. 什么是 haproxy?
49. 什么是负载均衡?
50. 什么是 rpc 及应用场景?
51. 什么是反向代理和反向代理?
52. csv 文件【erotic.csv】中共存在 271 万多条数据,请获取其中的 subscription\_id,并使用线程池为每 100 条数据创建一个线程去处理(打印或通过爬虫去提交到某处),erotic.csv 文件格式为:

```
"subscription_id","erotic","num"
"UCURGHWSDe7S-v1ufCAq9Rfw","5","1"
```



## 第六章 数据库和缓存

1. 列举常见的关系型数据库和非关系型都有那些？
2. MySQL 常见数据库引擎及区别？
3. 简述事务及其特性？
4. 简述触发器、函数、视图、存储过程？
5. MySQL 索引种类
6. 索引在什么情况下遵循最左前缀的规则？
7. MySQL 常见的函数？
8. 列举 创建索引但是无法命中索引的情况。
9. 数据库导入导出命令（结构+数据）？
10. 你了解那些数据库优化方案？
11. char 和 varchar 的区别？
12. 简述 MySQL 的执行计划的作用及使用方法？
13. 1000w 条数据，使用 limit offset 分页时，为什么越往后翻越慢？如何解决？
14. 什么是索引合并？
15. 什么是覆盖索引？
16. 简述数据库读写分离？
17. 简述数据库分库分表？（水平、垂直）
18. 数据库锁的作用？
19. where 子句中有 a,b,c 三个查询条件, 创建一个组合索引 abc(a,b,c)，以下哪种会命中索引
  - (a)
  - (b)
  - (c)
  - (a,b)
  - (b,c)
  - (a,c)
  - (a,b,c)

20. mysql 下面那些查询不会使用索引,between, like "c%" , not in, not exists, !=, <, <=, =, >, >=,in

21. mysql 中 varchar 与 char 的区别以及 varchar(50)中的 50 代表的含义

22. 请简述项目中优化 sql 语句执行效率的方法

23. 从 delete 语句中省略 where 子句,将产生什么后果?

- A. delete 语句将失败因为没有记录可删除
- B. delete 语句将从表中删除所有的记录
- C. delete 语句将提示用户进入删除的标准
- D. delete 语句将失败,因为语法错误

24. 叙述 mysql 半同步复制原理

25. sql 查询

存在的表有

- 1. products(商品表) columns 为 id, name, price
- 2. orders(商城订单表) columns 为 id, reservation\_id, product\_id, quantity(购买数量)
- 3. reservations(酒店订单表) columns 为 id,user\_id, price, created

需要查询的:

- 1. 各个商品的售卖情况, 需要字段 商品名 购买总量 商品收入
- 2. 所有用户在 2018-01-01 至 2018-02-01 下单次数, 下单金额, 商城下单次数, 商城下单金额
- 3. 历月下单用户数: 下单一次用户数, 下单两次用户数, 下单三次及以上用户数

26. 考虑如下表结构, 写出建表语句

| ID(自增主键) | NAME(非空) | Balance(非空) |
|----------|----------|-------------|
| 1        | A        | 19.50       |
| 2        | A        | 20.50       |
| 3        | A        | 100.00      |

27. 假设学生 Students 和教师 Teachers 关系模型如下所示

- 1. Student:(学号,姓名,性别,类别,身份证号)
- 2. Teacher:(教师号,姓名,性别,身份证号,工资)

其中,学生关系中的类别分别为"本科生"和"研究生两类", 性别分为"男"和"女"两类.

查询研究生教师平均工资(显示为平均工资), 最高工资与最低工资之间的差值(显示为差值)的 SQL 语句

```
select (1) as 平均工资, (2) as 差值 from Students,Teacher where (3);
```

查询工资少于 10000 元的女研究生教师的身份证号和姓名的 SQL 语句(非嵌套查询方式);

```
select 身份证号,姓名 from Students where (4) (5)
```

```
select 身份证号,姓名 from Teachers where (6)
```

28. mysql 中怎么创建索引?
29. 请简述 sql 注入的攻击原理及如何在代码层面防止 sql 注入?
30. 使用 Python 实现将数据库的 student 表中提取的数据写入 db.txt?
31. 简述 left join 和 right join 的区别?
32. 索引有什么作用,有那些分类,有什么好处和坏处?
33. 写 sql 语句

TableA

| ID | NAME | KECHEN<br>G | FENSHU |
|----|------|-------------|--------|
| 1  | 张三   | 语文          | 81     |
| 2  | 张三   | 数学          | 75     |
| 3  | 李四   | 语文          | 76     |
| 4  | 李四   | 数学          | 90     |
| 5  | 王五   | 语文          | 81     |
| 6  | 王五   | 数学          | 100    |
| 7  | 王五   | 英语          | 90     |

tableB

| ID | NAME |
|----|------|
| 1  | 张三   |
| 2  | 李四   |
| 3  | 王五   |
| 4  | 赵六   |

查询：

1. 查询出每门课程都大于 80 分的学生姓名
  2. 查询出语文成绩最大的学生姓名
  3. 查询没有成绩的学生姓名
34. 试列出至少三种目前流行的大型关系型数据库的名称

试列出至少三种目前流行的大型关系型数据库的名称  
其中您最熟悉的是  
什么时候开始使用

### 35. 数据库查询

有表 List，共有字段 a,b,c。类型都是整数，表中有如下几条记录

| 1 | a | b  | c |
|---|---|----|---|
| 2 | 2 | 7  | 9 |
| 3 | 5 | 6  | 4 |
| 4 | 3 | 11 | 9 |

现在对该表依次完成一下操作

1. 查询出 b 和 c 列的值，要求按 b 列的升序排列
  2. 写入一条新的记录，值为(7,9,8)
  3. 查询 c 列,要求消除重复的值,按降序排列
36. 用一条 sql 语句查询出每门课程都大于 80 分的学生姓名. 表 score 如下：

| name | kecheng | fenshu |
|------|---------|--------|
| 张三   | 语文      | 81     |
| 张三   | 数学      | 75     |
| 李四   | 语文      | 76     |

37. 设计表, 关系如下: 教师, 班级, 学生, 科室. 科室与教师为一对多关系, 教师与班级为多对多关系, 班级与学生为一对多关系, 科室中需体现层级关系

1. 写出各张表的逻辑字段
  2. 根据上述关系表
    1. 查询教师 id=1 的学生数
    2. 查询科室 id=3 的下级部门数
    3. 查询所带学生最多的教师的 id
38. 有 staff 表, 字段为主键 Sid, 姓名 Sname, 性别 Sex(值为"男"或"女"), 课程表 Course, 字段为主键 Cid, 课程名称 Cname, 关系表 SC\_Relation, 字段为 Student 表主键 Sid 和 Course 表主键 Cid, 组成联合主键, 请用 SQL 查询语句写出查询所有选"计算机"课程的男士的姓名
39. 什么是 MySQL 慢日志?

## 40. 数据库题（要求写出逻辑字段）

1. 写出建表语句完成如下操作，列名自由定义(id 自增):  
新建 A, B, C 三张表;  
A 表自关联;  
A、B 表为 1:n 表;  
B、C 表为 m:n 表
2. 写出插入语句完成如下操作:  
在 C 表中插入记录 c1, 并使其关联 B 表中 id 为 1 和 2 的两条记录。
3. 写出删除语句完成如下操作:  
删除 A 表中的记录 a1(id=1), 并级联删除 A、B、C 表中的其他相关记录。
4. 写出查询语句完成如下操作(3 条 SQL):  
A 表中存在记录 a2(id=2), 分别查询 A、B、C 表中与 a2 相关联的记录数据。

## 41. 在对 name 做了唯一索引前提下, 简述以下区别:

```
1. select * from tb where name = 'Oldboy-Wupeiqi'
2. select * from tb where name = 'Oldboy-Wupeiqi' limit 1
```

## 42. redis 和 memcached 的区别?

## 43. 如何高效的找到 redis 中所有以 oldboy 开头的 key?

## 44. 什么是一致性哈希?

## 45. redis 是单进程单线程的吗?

## 46. redis 中数据库默认是多少个 db 及作用?

## 47. 如果 redis 中的某个列表中的数据量非常大, 如果实现循环显示每一个值?

## 48. redis 如何实现主从复制? 以及数据同步机制?

## 49. redis 中的 sentinel 的作用?

## 50. 如何实现 redis 集群?

## 51. redis 中默认有多少个哈希槽?

## 52. 简述 redis 的有哪几种持久化策略及比较?

## 53. 列举 redis 支持的过期策略。

## 54. MySQL 里有 2000w 数据, redis 中只存 20w 的数据, 如何保证 redis 中都是热点数据?

55. 写代码，基于 redis 的列表实现 先进先出、后进先出队列、优先级队列。
56. 如何基于 redis 实现消息队列？
57. 如何基于 redis 实现发布和订阅？
58. 什么是 codis？
59. 什么是 twemproxy？
60. redis 如何实现事务。
61. redis 中的 watch 的命令的作用？
62. 简述 redis 分布式锁和 redlock 的实现机制。
63. 请设计一个商城商品计数器的实现方案？
64. 了解过 Hbase、DB2、SQLServer、Access 吗？

## 第七章 前端

1. JavaScript(或 jQuery)如何选择一个 id 为 main 的容器
2. JavaScript(或 jQuery)如何选择一个 class 为 menu 的容器
3. 简述什么是浏览器时间流
4. 用 css 如何隐藏一个元素
5. 一行 css 实现 padding 上下左右分别为 1px, 2px, 3px, 4px
6. 前后端分离的基本原理。
7. 前后端分离的通信数据安全
8. 给 ul 设置样式为:背景色黑色,给 ul 下的 li 设置样式为: 宽度 30px, 背景红色
9. 用 bootstrap 写一个响应式栅格, 一个页面分左右两栏, 大屏情况下分 6/6, 小屏情况下分为 12/12 (大屏: 屏幕 $\geq$ 992px, 小屏;992px $\geq$ 屏幕 $\geq$ 768px)
10. 写一个正则表达式获取 HTML 源码中的编码, 如下的编码是;utf-8 怎么通过 Python 的热模块获得?

```
1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4. <meta charset="utf-8">
5. <title>404</title>
6. </head>
```

11. 如何创建响应式布局?
12. 你曾经使用过哪些前端框架?
13. 什么是 ajax 请求? 并使用 jQuery 和 XMLHttpRequest 对象实现一个 ajax 请求。
14. 如何在前端实现轮询?
15. 如何在前端实现长轮询?
16. vuex 的作用?
17. vue 中的路由的拦截器的作用?
18. axios 的作用?

19. 列举 vue 的常见指令。
20. 简述 jsonp 及其原理？
21. 简述 cors 及其原理？
22. 看 javascript 代码写结果：

```
1. var name = '武沛齐';
2. function func(){
3. var name = 'alex';
4. function inner(){
5. console.log(name);
6. }
7. return inner;
8. }
9. var ret = func();
10. ret()
```

23. 看 javascript 代码写结果：

```
1. function main(){
2. if(1==1){
3. var name = "武沛齐";
4. }
5. console.log(name);
```

24. 看 javascript 代码写结果：

```
1. var name = "武沛齐";
2. function func(){
3. var name = "alex";
4. function inner(){
5. var name = "老男孩";
6. console.log(name);
7. }
8. return inner();
9. }
10. func();
```

25. 看 javascript 代码写结果：



```
1. function func(){
2. console.log(name);
3. var name = "武沛齐";
```

26. 看 javascript 代码写结果:

```
1. var name = "武沛齐";
2. function Foo(){
3. this.name = "alex";
4. this.func = function(){
5. console.log(this.name);
6. }
7. }
8. var obj = new Foo();
9. obj.func();
```

27. 看 javascript 代码写结果:

```
1. var name = "武沛齐";
2. var info = {
3. name: 'alex',
4. func:function(){
5. console.log(this.name);
6. (function(){
7. console.log(this.name);
8. })()
9. }
10. }
11. info.func();
```

28. 看 javascript 代码写结果:

```
1. var name = "武沛齐";
2. var info = {
3. name: 'alex',
4. func:function(){
5. console.log(this.name);
6. var that = this;
7. (function(){
8. console.log(that.name);
9. })()
10. }
11. }
12. info.func();
```

## 第八章 django

1. 简述 http 协议及常用请求头。
2. 列举常见的请求方法。
3. 列举常见的状态码。
4. 简述 websocket 协议及实现原理。
5. django 中如何实现 websocket?
6. Python web 开发中, 跨域问题的解决思路是?
7. 请简述 http 缓存机制。
8. 谈谈你所知道的 Python web 框架。
9. Http 和 Https 的区别?
10. django、flask、tornado 框架的比较?
11. 什么是 wsgi?
12. 列举 django 的内置组件?
13. 简述 django 下的(内建的)缓存机制
14. django 中 model 的 SlugField 类型字段有什么用途
15. django 中想要验证表单提交是否格式正确需要用到 form 中的那个方法
  - A. form.save()
  - B. form.save(commit=False)
  - C. form.verify()
  - D. form.is\_valid()
16. django 常见的线上部署方式有哪几种?
17. django 对数据查询结果排序怎么做, 降序怎么做?
18. 下面关于 http 协议中的 get 和 post 方式的区别, 那些是错误的?(多选)
  - A. 他们都可以被收藏, 以及缓存
  - B. get 请求参数放在 url 中
  - C. get 只用于查询请求, 不能用于数据请求
  - D. get 不应该处理敏感数据的请求
19. django 中使用 memcached 作为缓存的具体方法? 优缺点说明?
20. django 的 orm 中如何查询 id 不等于 5 的元素?

21. 使用 Django 中 model filter 条件过滤方法,把下边 sql 语句转化成 python 代码

```
1. select * from company where title like "%abc%" or mecount>999
2. order by createtime desc;
```

22. 从输入 <http://www.baidu.com/>到页面返回,中间都是发生了什么?
23. django 请求的生命周期?
24. django 中如何在 model 保存前做一定的固定操作,比如写一句日志?
25. 简述 django 中间件及其应用场景?
26. 简述 django FBV 和 CBV?
27. 如何给 django CBV 的函数设置添加装饰器?
28. django 如何连接多个数据库并实现读写分离?
29. 列举 django orm 中你了解的所有方法?
30. django 中的 F 的作用?
31. django 中的 Q 的作用?
32. django 中如何执行原生 SQL?
33. only 和 defer 的区别?
34. *selectrelated* 和 *prefetchrelated* 的区别?
35. django 中 filter 和 exclude 的区别
36. django 中 values 和 values\_list 的区别?
37. 如何使用 django orm 批量创建数据?
38. django 的 Form 和 ModelForm 的作用?
39. django 的 Form 组件中,如果字段中包含 choices 参数,请使用两种方式实现数据源实时更新。
40. django 的 Model 中的 ForeignKey 字段中的 on\_delete 参数有什么作用?
41. django 中 csrf 的实现机制?
42. django 如何实现 websocket?
43. 基于 django 使用 ajax 发送 post 请求时,有哪种方法携带 csrf token?

44. django 缓存如何设置?
45. django 的缓存能使用 redis 吗? 如果可以的话, 如何配置?
46. django 路由系统中 name 的作用?
47. django 的模板中 filter、simpletag、inclusiontag 的区别?
48. django-debug-toolbar 的作用?
49. django 中如何实现单元测试?
50. 解释 orm 中 db first 和 code first 的含义?
51. django 中如何根据数据库表生成 model 类?
52. 使用 orm 和原生 sql 的优缺点?
53. 简述 MVC 和 MTV
54. django 的 contenttype 组件的作用?

## 第九章 Flask

1. 请手写一个 flask 的 Hello World。
2. Flask 框架的优势？
3. Flask 框架依赖组件？
4. Flask 蓝图的作用？
5. 列举使用过的 Flask 第三方组件？
6. 简述 Flask 上下文管理流程？
7. Flask 中的 g 的作用？
8. 如何编写 flask 的离线脚本？
9. Flask 中上下文管理主要涉及到了那些相关的类？并描述类主要作用？
10. 为什么要 Flask 把 Local 对象中的值 stack 维护成一个列表？
11. Flask 中多 app 应用如何编写？
12. 在 Flask 中实现 WebSocket 需要什么组件？
13. wtforms 组件的作用？
14. Flask 框架默认 session 处理机制？
15. 解释 Flask 框架中的 Local 对象和 threadinglocal 对象的区别？
16. SQLAlchemy 中的 session 和 scoped\_session 的区别？
17. SQLAlchemy 如何执行原生 SQL？
18. ORM 的实现原理？
19. DBUtils 模块的作用？
20. 以下 SQLAlchemy 的字段是否正确？如果不正确请更正。
21. SQLAchemy 中如何为表设置引擎和字符编码？
22. SQLAchemy 中如何设置联合唯一索引？

## 第十章 tornado

1. tornado 中的 `gen.coroutine` 的作用？
2. 简述 tornado 框架特点及应用场景。
3. tornado 框架中 `Future` 对象的作用？
4. tornado 框架中如何编写 `webSocket` 程序？
5. tornado 中静态文件是如何处理的？
6. tornado 操作 MySQL 使用的模块？
7. tornado 操作 redis 使用的模块？

## 第十一章 api

1. 什么是 webservice?
2. 什么是 rpc?
3. 谈谈你对 restfull 规范的认识?
4. 什么是接口的幂等性?
5. 为什么要使用 django rest framework 框架?
6. django rest framework 框架中都有那些组件?
7. 使用 django rest framework 框架编写视图时都继承过哪些类?
8. django rest framework 框架如何对 Queryset 进行序列化?
9. 简述 django rest framework 框架的认证流程。
10. django rest framework 如何实现的用户访问频率控制? (匿名用户和注册用户)



## 第十二章 git

1. 你在公司如何做的协同开发？
2. git 常见命令。
3. 简述以下 git 中 stash 命令作用以及相关其他命令。
4. git 中 merge 和 rebase 命令 的区别。
5. 公司如何基于 git 做的协同开发？
6. 如何基于 git 实现代码 review？
7. git 如何实现 v1.0 、 v2.0 等版本的管理？
8. 什么是 gitlab？
9. github 和 gitlab 的区别？
10. 如何为 github 上牛逼的开源项目贡献代码？
11. git 中 .gitignore 文件的作用？

## 第十三章 爬虫

1. 写出在网络爬取过程中, 遇到防爬问题的解决办法。
2. 如何提高爬虫的效率?
3. 你的爬虫 爬取的数据量有多少?
4. 列举您使用过的 python 网络爬虫所用到的模块。
5. 简述 requests 模块的作用及基本使用?
6. 简述 beautifulsoup 模块的作用及基本使用?
7. 简述 seleninu 模块的作用及基本使用?
8. 简述 scrapy 框架中各组件的工作流程?
9. 在 scrapy 框架中如何设置代理 (两种方法)?
10. scrapy 框架中如何实现大文件的下载?
11. scrapy 中如何实现限速?
12. scrapy 中如何实现暂定爬虫?
13. scrapy 中如何进行自定制命令?
14. scrapy 中如何实现的记录爬虫的深度?
15. scrapy 中的 pipelines 工作原理?
16. scrapy 的 pipelines 如何丢弃一个 item 对象?
17. 简述 scrapy 中爬虫中间件和下载中间件的作用?
18. scrapy-redis 组件的作用?
19. scrapy-redis 组件中如何实现的任务的去重?
20. scrapy-redis 的调度器如何实现任务的深度优先和广度优先?

## 第十四章 算法和数据结构

1. 顺序表、链表的区别及应用场景。
2. 哈希树的构造及应用场景。
3. B Tree 和 B+ Tree 的区别?
4. 什么是中序遍历?
5. 具有三个节点的二叉树有几种形态:
  - A. 1
  - B. 3
  - C. 5
  - D. 7
6. 无向图  $G$  中的边的集合  $E=[(a,b), (a,e),(a,c),(b,e),(e,d),(d,f),(f,c)]$ , 则从顶点  $a$  出发进行深度优先遍历可以得到一种顶点序列为
  - A. aedfcb
  - B. acfebd
  - C. aebcfd
  - D. aedfbc
7. 一颗具有  $n$  个节点的平衡二叉树, 其平均查找长度为
  - A.  $O(1)$
  - B.  $O(\log_2 n)$
  - C.  $O(n \log_2 n)$
  - D.  $O(n^2)$
8. 以下序列中不是二叉堆的是
  - A. 100, 86, 48, 73, 35, 39, 42, 57, 66
  - B. 12, 70, 33, 65, 92, 41, 40, 81, 75, 99
  - C. 103, 97, 56, 38, 89, 23, 45, 10, 37, 52, 6
  - D. 7, 32, 10, 53, 90, 27, 41, 70, 61, 82
9. 快速排序按排序思想分类属于
  - A. 基数排序
  - B. 选择排序
  - C. 插入排序
  - D. 交换排序
10. 奇偶交换排序

如下所述:

第一趟对所有奇数  $i$ , 将  $a[i]$  和  $a[i+1]$  进行比较;

第二趟对所有偶数  $i$ , 将  $a[i]$  和  $a[i+1]$  进行比较, 若  $a[i] > a[i+1]$ , 则两

者交换;

第三趟对奇数  $i$ , 第四趟对偶数  $i$ , 以此类推, 直至整个序列有序为止. 若有初始序列为逆序, 规模为 7 的有序序列, 欲通过奇偶交换排序获得正序序列, 则排序过程中所需的数据交换次数为多少?

- A. 6
- B. 20
- C. 21
- D. 28

11. 对长度为  $N$  的线性表进行顺序查找, 在最坏情况下所需要的比较次数为

- A.  $N+1$
- B.  $N$
- C.  $(N+1)/2$
- D.  $N/2$

12. 一直青蛙可以跳上一级台阶, 也可以跳上两级台阶, 求该青蛙跳上一个十级台阶共有多少中跳法.

- A. 15
- B. 89
- C. 144
- D. 512

13. 合并两个有序的数组, 数组都是非递减的, 合并后的数组依然有序

```
1. class Solution:
2. def merge(self, nums1, m, nums2, n)
3. ...
4. :type nums1: list[int]
5. :type m: int
6. :type nums2: list[int]
7. :type n1: int
8. :rtype nums: list[int]
```

14. 反转数字, 例子:

输入 123, 输出 321; 输入 -123, 输出 -321; 输入 1032100, 输出 12301

```
1. class Solution:
2. def reversr(self, x)
3. ...
4. :type x: int
5. :rtype :int
6. ...
```

15. 实现一个二分查找
16. 采用递归的方式用 javascript 写一下快速排序
17. 一个数组, 找到和为  $n$  的所有数对. 如[1,7,3,5,6,2,9,5,4,8]  $n=11$ , 问 数对 (7,4),(2,9),(5,6)...效率尽可能高
18. 简单描述下快速排序的原理。
19. 12 个大小一样的球, 其中一个的重量与其他的不一樣, 如何只称三次找出这只球
20. Given a string  $s$ , find the longest palindromic substring in  $s$ , you may assume that the maximum length of  $s$  is 1000;

Example:

Input: "babad"

Output: "bab"

Note: "aba" is also a valid answer

Example:

Input: "cbdd"

Output: "bb"

21. Divide two integers without using multiplication, division and mod operator. If it is overflow, return MAX\_INT;

```

1. For example, given: ["eat", "tea", "tan", "ate", "nat", "bat"]
2. return :
3. [
4. ["ate", "eat", "tea"],
5. ["nat", "tan"],
6. ["bat"]
7.]
8.
9. Note: All inputs will be in lower-case

```

22. 括号匹配

Given a string containing just the characters '(', ')', '{', '}', '[', and ']', determine if the input string is valid.

The brackets must close in the correct order, "()" and "[]{}" are all valid but "[(" and "(]" are not

23. 给定一个 int list  $a$ , 满足  $a[i+1] \geq a[i]$ , 满足  $a[i+1] \geq a[i]$ , 给定 int  $key$ , 找出 list  $a$  中第一个大于等于  $key$  元素的 index, 无满足要求的元素则返回 -1

函数定义：

```
1. def findindex(int_list, int_key):
```

24. 假设在  $n$  进制下,  $567 \times 456 = 150216$  成立, 请问  $n$  的值是

- A. 9
- B. 12
- C. 14
- D. 18

25. 给一个链表, 将其中的节点两两交换后, 返回链表的头结点

实例:

给出  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$

你的程序应该返回这样一个链表:  $2 \rightarrow 1 \rightarrow 4 \rightarrow 3$

注意: 不能修改链表的节点的值

26. 给定一个整数数组, 返回两个数字的索引, 使得他们相加和一个目标值相等. 可以假设每个输入都有且只有一个解 例如 数组 `nums = [2,7,11,15]`, 目标值是 9 因为 `nums[0]+nums[1] = 2+7=9` 所以返回 `[0,1]`

27. we have a list of number

```
l = [1,2,4,8,16,32,64,128,256,512,1024,32768,65536,4294967296]
we want to make a dictionary with the number of digits as the key
and list of numbers the value:
```

```
{1:[1,2,4,8],2:[16,32,64],3:[128,256,512],4:[1024],5:[32768],10:[4294967296]}
```

28.

for a given pair of numbers (row x column), create an array. for example, for (4,5) pair, we want to create the following 4x5 matrix:

```
[[0,0,0,0,0],[0,1,2,3,4],[0,2,4,6,8],[0,3,6,9,12]]
```

29. we want to define a function with a generator which can iterate the numbers, which are divisible by 7 within range(n)

30. 布尔运算

已知  $x=43$ ,  $ch="A"$ ,  $y=1$ , 则表达式  $(x>=y \text{ and } ch<'B' \text{ and } y)$  的值是

- 1. 0
- 2. 1
- 3. 出错
- 4. True

31. 编程实现斐波那契数列(注:使用递归)
32. 请用自己的算法, 按升序合并如下两个 list, 并去除重复元素

```
1. list1=[2,3,8,4,9,5,6]
2. list2=[5,6,10,17,11,2]
```

33. 现有编号分别为 A,B,C,D,E 的 5 个盒子, 某方法每次调用都输出盒子的一个编号, 输出这五个盒子的概率分别是 10%, 20%, 25%, 15%, 30%, 请实现该方法.

34. 填充代码

```
1. import random
2.
3. def get_card(config):
4. ...
5. ...
6.
7. card_config = [['card1',10],['card2',30],['card',15]]
8. card_name = get_card(card_config)
9. print card_name
```

现需要编写一个抽取卡牌的函数, 按照给定的权重分布来随机抽取, 输入的配置示例为[['card1',10],['card2',30],['card',15]], 其中字符串为卡牌名, 数字为权重, 返回'card1'的概率是  $10/(10+30+15)$ , 请完成代码中省略的部分

35. 在数组中找到具有最大和的连续子数组(至少包含一个数字)

例如, 给定数组[-2,1,-3,4,-1,1,1,-5,4]

连续子阵列[4,-1,2,1]具有最大的 sum=6

36. 算法是值

- A. 数学的计算公式
- B. 程序设计语言的语句序列
- C. 对问题的精准描述
- D. 解决问题的精准步骤

37. 斐波那契数列

1,2,3,5,8,13

求出 400 万以内的最大的斐波那契数, 并求出他是第几个

## 38. 写代码(数桃子)

海滩上有一堆桃子, 五只猴子来分, 第一只猴子把这堆桃子平均分成五份, 多了一个, 这只猴子把多的一个扔到了海里, 拿走了一份, 第二只猴子把剩下的盒子了一起, 有平均分成五份, 又多了一个, 同样把多的一个扔到了海里, 拿走了一份, 第三只, 第四只, 第五只都是这样做的, 问海滩上原来最少有多少桃子

## 39. 写程序

上机编程实现一个小程序, 给出一个罗马数字, 先判断是否是罗马数字, 如果是, 转换为阿拉伯数字, 不是的话报错. 本题只考虑 3999 以内的数

I(1) V (5) X(10) L(50) C(100) D(500) M(1000)

计数规则

1. 若干相同数字连写表示的数是这些罗马数字的和, 如 III=3
2. 小数字在大数字前面表示的数使用大数字减去小数字, 如 IV=4
3. 小数字在大数字后面表示的数是大数字加上小数字, 如 VI=6

组合规则:

1. 基本数字 I, X, C 中的任何一个, 自身连用构成数目, 或者放在大数的右边连用构成数目, 都不能超过三个; 放在大数的左边只能用一个
2. 不能把基本数字 V, L, D 中的任何一个作为小数放在大数的左边采用相减的方式构成数目; 放在大数的右边采用相加的方式构成数目, 只能使用一个
3. V 和 X 左边的小数字只能用 I
4. L 和 C 左边的小数字只能用 X
5. D 和 M 左边的小数字只能用 C

## 40. 请列举出三种常用的排序算法

## 41. 用 4,9,2,7 四个数字, 用加减乘除, 和每个数字(必须且只能用一次), 使表达式结果为 24

## 42. 一个数组的中位数定义为: 数组中元素排序后在中间位置的数, 如果数组长度为偶数, 则为中间两个数的平均数.

请写出两个函数, 分别实现:

- \* 向数组插入新元素
- \* 找出插入元素后新数组中的中位数

提示: 插入新元素的方法影响找中位数的复杂度, 我们希望找减少求中位数的复杂度



43. 有一个容量为  $N$  的背包,和  $M$  个物品,这些物品的体积可以用一个数组  $[a_1, a_2, a_3 \dots a_m]$  表示,假设背包的容量不足以装下所有的物品,请编程实现:找出一个最佳的背包方案,使得背包的空间利用率最大
44. 在一个二位坐标体系中有多个任意位置的点,请编程找出其中多有的边缘点.边缘点的定义为在该点的右上象限内没有其他的点
45. 语言不限,实现以下问题
1. 链表的冒泡排序
  2. 树的顺序遍历
  3. 顺序表

## 第十五章 Linux

1. 下面的 linux 命令中, 那个不能显示出文件的内容
  - A. tac
  - B. more
  - C. head
  - D. man
2. 默认情况下管理员创建了一个用户, 就会在()目录下创建一个用户主目录
  - A. /usr
  - B. /home
  - C. /root
  - D. /etc
3. 你使用命令"vi /etc/inittab"查看该文件的内容, 你不小心改动了一些内容, 为了防止系统出问题, 你不想保存所修改的内容, 你应该如何操作
  - A. 在末行模式下, 键入:q!
  - B. 在末行模式下, 键入:x!
  - C. 在末行模式下, 键入:wq
  - D. 在末行模式下, 键入"ESC"键直接退出 vi
4. 用"rm -i", 系统会提示什么来让你确认
  - A. 命令行的每个选择
  - B. 是否真的删除
  - C. 是否有写的权限
  - D. 文件的位置
5. 在 CentOS7.2 中, 用一句话将所有的 test.py 进程全部杀死
6. 在 CentOS7.2 中, 如何查看程序执行所消耗的 CPU, 内存等硬件资源
7. 写一个 Bash Shell 脚本来得到当前的日期,时间, 用户名和当前的工作目录
8. 如何查看当前登录用户
9. 如何定位占用端口 8080 的服务
10. 如何切换用户
11. 查找/tmp/path 下的以 A 开头的文件
12. 如有两台机器 a/b, a 的 ip 地址包括 45.32.12.222, 10.10.121.22 b 的 ip 地址包括 45.32.12.226,10.10.121.69,两台机器的 ssh 监听端口为 11111

请写出远程登录机器 a 的命令

请写出从 a 远程登录 b 的命令

请说明从 b 机器访问 qq.com 时的详细过程, 以及到 qq.com 记录到的 ip

13. 计划任务---如何让 abc.sh 每周一执行一次 若执行失败的原因是什么
14. 如何查看发往本机 8080 端口的流量
15. 现有一文件文件内容为 ip 地址及路径, 例如 192.168.31.9 /tmp/destpath, 现需要将本机的 hello.txt 文件发送到文件记录的机器相应的目录中, 编程实现(shell)
16. 在 linux 中, 如何批量的删除多个 Python 进程
17. 下面那个函数能够在 linux 下创建一个子进程
  - A. os.popen
  - B. os.fork
  - C. os.system
  - D. os.link
18. 文件内容实例如下

```
192.168.0.20--[0B/Sep/2015:20:05:30+0800] "GET /android/login/?method=get_server HTTP/1.1" -200 22268 "-" "-" "-" 0.055 0.139
```

请写出命令行下(shell 默认 bash), 提取请求时间(\$request\_time) 大于 0.1 秒的请求(\$request), 并写入 access\_long\_time.log 文件中的命令
19. 你最熟悉的 unix 环境是?
20. unix 下查询环境变量的命令是
21. 查询脚本定时任务的命令是
22. Apache 服务器默认的接听连接端口号是
  - A. 1024
  - B. 800
  - C. 80
  - D. 8
23. 建立一个新文件可以使用的命令为
  - A. chmod
  - B. more
  - C. cp
  - D. touch
24. /etc/ethX 表示

- A. 系统会送接口
  - B. 以太网接口设备
  - C. 令牌环网设备
  - D. PPP 设备
25. 显示用户的 ID, 以及所属组的 ID, 要使用的命令是
- A. su
  - B. who
  - C. id
  - D. man
26. 显示用户的 ID, 以及所属组的 ID, 要使用的命令是
- A. su
  - B. who
  - C. id
  - D. man
27. 为了修改文件 test 的许可模式, 使其文件属性具有读写和运行的权限, 组和其他用户可以读和运行, 可以采用
- A. chmod 755 test
  - B. chmod 700 test
  - C. chmod +rwx test
  - D. chmod g-w test
28. 取 ls -l 输出结果的第 5 列的值的正确写法是
- A. ls -l |awk "{print\$5}"
  - B. ls -l |awk '{print\$5}'
  - C. ls -l |awk {print\$5}
  - D. ls -l |awk 'print\$5'
29. 什么命令解压缩 tar.gz 文件
- A. tar -czcf filename.tar.gz
  - B. tar -zxvf filename.tar.gz
  - C. tar -tzvf filename.tar.gz
  - D. tar -dzvf filename.tar.gz
30. 以下哪个命令是 vi 编辑器中执行存盘退出的
- A. q
  - B. zz
  - C. :q!
  - D. :wq
31. 简述 saltstack、ansible、fabric、puppet 工具的作用?
32. uwsgi 和 cgi 的区别?

- 33. supervisor 的作用？
- 34. 解释 PV、UV 的含义？
- 35. 解释 QPS 的含义？

## 第十六章 设计题

### 1. 设计一个办公室摄像头的 web 后台管理系统

假设功能如下：

1. 可以 开启关闭重启每个摄像头
2. 可以调整摄像头的方向
3. 可以设置摄像头每天的工作时间
4. 后台可以查看每个摄像头的历史录像，并且可以删除，设置保存最大保存时间
5. 后台可以随时查看实时监控

要求：

1. 设计系统架构，考虑全系统无单点，画出设计图，标出所需选用的所有技术和所有组件及关系
  2. 设计系统数据结构，设计一套基于 mysql 的数据库表
  3. API 设计，写出符合以上功能的所有用户 API 及主要的内部工程逻辑
  4. 设计用户和权限系统，考虑只有两种角色即可，一种是普通用户只有功能 5，也就是只能看实时录像，另一个是管理员可以拥有 1-5 的全部功能权限
- ### 2. 编写一个脚本, 5 分钟检查一次日志, 发现有暴力 ssh 破解现象的, 提取此类 ip 地址, 并去重, 并按降序排序。

要求：

1. 同一个 ip 暴力破解超过 10 次，自动屏蔽 IP 地址
2. 指定办公室 ip 地址(192.168.100.100)为可信任 ip 地址，不受屏蔽规则限制，以下为日志格式

日志样板如下：

```
May 4 03:43:07 tz-monitor sshd[14003]: Failed password for root
from 124.232.135.84 port 25251 ssh2
```

```
May 4 03:43:07 tz-monitor sshd[14082]: Invalid user postgres fro
m 124.232.135.84
```

3. 一个文本文件, 大约有一万行, 每行一个词, 要求统计出其中最频繁的前 10 个词, 请给出设计思路。
4. 请为公司设计一个并发处理 key-value 引擎, 要求每条请求的数据小于 16k, 数据总量为 1T, QPS 为 50。

要求：

1. 请给出该系统需要配备多少资源，服务其数量，服务器内存大小及硬盘空间等
2. 要求系统平滑可扩展
3. 尽可能的降低系统复杂度

5. 请设计秒杀系统,大并发下如何不会超卖?
6. 如果有一个订单系统,包含订单信息,商品信息,价格信息,并且要记录那些状态,在设计系统时,你会提供那些建议?
7. 现在要开发一款游戏,根据游戏的分数做出这款游戏的用户排名,用户信息可以通过第三方接口获取,用户的好友信息可以通过第三方获取.游戏排名分为世界排名和好友排名..(参考例子跳一跳)请简要设计数据库模型。
8. 写代码

有一个 3G 大小的文件,文件每行一个 `string`,内容为酒店的 `id` 和一个图片的名字,使用 `"\t"` 分割示例, `ht_1023134+"\t"+hiadwqerscnsdjkfhwe.jpg`

表示的是一个酒店包含的一张图片,统计含有图片数量为[20,无穷大]的酒店的 `id`,含有图片数量范围[10,20]的酒店 `id`,含有图片数量为[5,10]的酒店的 `id`,图片数量为[0,5]的酒店的 `id`,并将结果输出到文件中

文件格式:

```
1. 0-5+"\t"+id1+"\t"+id2
2.
3. 5-10+"\t"+id1+"\t"+id2
4.
5. 10-20+"\t"+id1+"\t"+id2
6.
7. 20-无穷大+"\t"+id1+"\t"+id2
```

9. 现在小明一家过一座桥,过桥时是黑夜,所以必须有灯.现在小明过桥要 1 秒,小明的弟弟要三秒,小明的爸爸要 6 秒,小明的妈妈要 8 秒,小明的爷爷要 12 秒,而过桥的时间依过桥最慢者而定,而且等灯在点然后 36 秒就会熄灭,问小明一家如何过桥。

## 第十七章 客观题

1. 请列举最近关注的一些技术?
2. 请列举你认为不错的一些技术书籍和你最近再看的书籍(不限于技术)?
3. 请列举你阅读过源码的一些项目?
4. 请给出你对这份笔试题的看法?
5. 作为项目负责人的你,你应该要做那些除编码外的工作来更快的推动项目开发?
6. 请列举经常访问的计数网站或博客
7. 对您最有影响的或是您认为最有价值的软件方面的几本书。
8. 在软件开放方面,您最擅长的是
  1. 管理
  2. 需求分析
  3. 系统分析
  4. 设计
  5. 代码
  6. 测试
  7. 软件工程
9. 在中国东北有这样两个存款,赵村所有的村民都是白天祭祀祖先,而李庄的人都是晚上祭祀祖先,能肯定的没人在白天和晚上都祭祀祖先的.我么也知道李明是晚上祭祀祖先的人,那么以下哪个选项是正确的
  - A. 李明是赵村的人
  - B. 李明不是赵村的人
  - C. 李明是李庄的人
  - D. 李明不是李庄的人
  - E. 李明既不是李庄的人也不是赵庄的人
10. 相传在古时候某国的国民都分别居住在两个怪城里,一座真城,一座假城,真城的人各个都说真话,假城的人各个都说假话,一位知道情况的外国游客来到其中一个城市,他只问了一个问题就知道自己来到的是哪个城了,下面那句话最恰当
  - A. 你是真城人吗
  - B. 你是假城人吗
  - C. 你是说真话的人吗
  - D. 你是说假话的人吗
  - E. 你是这个城市的人吗



11. 设'并非无奸不商'为真,则下面哪一个选项一定为真
- A. 所有商人都是奸商
  - B. 所有商人都不是奸商
  - C. 并非所有商人不是奸商
  - D. 并非所有商人是奸商
  - E. 有的商人不是奸商
12. 小赵比小钱个子高;小孙不如小李个子高;;小李个子不如小周高;小钱和小周个子正好一样高.如果上面这些陈述都是对的,name 下列哪个选项是对的
- A. 小孙比小周个子高
  - B. 小孙比小赵个子高
  - C. 小钱比小孙个子矮
  - D. 小赵比小李个子高
  - E. 小钱比小李个子矮

## 第十八章 其他知识点

1. 什么是 rpc?
2. 简述 RabbitMQ、Kafka、ZeroMQ 的区别?
3. RabbitMQ 如果在消费者获取任务后未处理完前就挂掉时，保证数据不丢失?
4. RabbitMQ 如何对消息做持久化?
5. RabbitMQ 如何控制消息被消费的顺序?
6. 以下 RabbitMQ 的 exchange type 分别代表什么意思？如：fanout、direct、topic。
7. 简述 celery 是什么以及应用场景?
8. 简述 celery 运行机制。
9. celery 如何实现定时任务?
10. 简述 celery 多任务结构目录?
11. celery 中装饰器 @apptask 和 @shared\_task 的区别?
12. 谈谈你们公司的上线流程?
13. 公司用的什么做的 bug 管理?
14. numpy 和 pandas 模块的作用?
15. opencv 是什么?
16. 设计一个可抗高并发

## 第十九章 100+套企业面试真题

上述章节中的面试题是作者对将近 100+ 套面试真题整理并抄录所得，获取更多资源可以做如下操作：

- 100 套企业面试真题、新书《Python 企业面试题锦集(含答案)》、最新面试题更新，请微信扫码关注【Python 之路】公众号获取。



- 技术交流 QQ 群：737021730

## 第二十章 附赠

### 1. 人生没有白走的路，每一步都算数

看了一位新同学的自我介绍，看完后让人沉思，有多少人也跟他一样，不是不想努力，而是一直找不到方向，只能在底层兜兜转转消磨殆尽美好青春年华。

他叫车融，在深圳工作，我去深圳分校时，他正好去咨询，他跟我聊过很多，感觉是个很有想法的人，今天看完他写的这个东西，感觉有些沉重，但人生的路，每一步都算数，过往的错配才让我们意识到有规划的人生是多么重要。我们老男孩和路飞学城，承载着一群有志向的人改变命运的希望和期待，一定要努力做好，不忘初心，不要被短期利益诱惑，要对得起这份信任。

—————以下为他的正文—————

大家好，我是一位来自东北的 29 岁老男孩，现居深圳。说实话，在这个年纪选择转行确实是下了很大决心，因为横向去看一下同龄的同学，大多都已成家，且决意在本行业中一直做下去，人生基本定型稳定。都说 20-30 岁是人的黄金年龄段，因为在这个年龄段会有美好的校园青春，也会经历初涉社会的迷茫，会有独自在异乡生活的经历，也会积累各种各样的工作经验，最后在临近 30 岁终于确定自己愿意奋斗终生的事业，运气好的话也会确定那个陪伴自己余生的人。这是绝大多数人的 20-30 岁的成长路径，但是于我自身而言，我这 10 年可用“看似平坦正常，实则混沌迷茫”来概括，即我一直在被时间推着裹挟着往前走，却从未静下心来好好规划自己的人生。从未真正掌控自己的人生。

我这一代东北人（具体说是吉林省长春市），普遍都是 7 岁上小学，正常初高中读完高考是 19 岁上大学。我第一年因为贪玩，没有考取理想学校，选择复读后在 20 岁那年考上了华南理工大学。我至今还记得，当初抱着高考志愿指南选专业时候的场景，华工在吉林省总共录取的人不多，但录取分数并没有高到变态（毕竟在吉林省知名度有限），我比较了一下历年专业分数线，分数最高的建筑专业我肯定是不能报的，其他的专业其实录取线梯度不算大，我都可以去尝试一下。但是那个年代，没有什么靠谱的职业咨询平台，也没有知乎，985 吧之类的论坛，我只能通过新浪教育，新浪高考这类门户网站去看未来专业的预测，后来的事实证明，这些未来专业预测的文章都非常不靠谱，比如我清晰地记得十大就业难的专业中，法学，英语，计算机名列三甲，其分析的理由是中国要是个大学就都有这三个专业，所以这些专业招生泡沫太大，以后人才供给一定大于需求，就业一定是困难的，这个基于“人才供需平衡”的理论乍一看还挺有道理，所以我就顺着看这位专家对于十大前景好专业的分析，我清晰地记得有环保类的专业，当时的分析说国家一直大力发展制造业，对于环境污染非常严重，所以未来一定会重视环境治理，会催生环保产业蓬勃发展，未来人才缺口巨大。我当时觉得这篇文章写的太好了，就报了环境科学这个专业，被录取后还沉浸在巨大的喜悦中。但是进入大学校园后，与师兄师姐各种有了接触后，发现这个专业的平均薪酬并不高，尤其是本科毕业出去找工作，因此很多人选择读研再去就业。我在大三时决定跨考经济类的研究生，所以那时上专业课或者选修课时都会啃各种目标院校的参考教材，当时觉得准备得挺充分的，但是最终还是因为志愿录取线过高未能录取。24 岁那年，我只能灰溜溜地回了老家，在父母帮忙下进了一个市政工程设计院，不过当时是没有编制的，因为进那个单位必须要有研究生学历，所以就边工作边准备考研，在 2014 年考上了哈工程的研究生。读研期间收获可用一句话概括，谈了一段恋爱，发了一篇专业 Top SCI，收获了自学--实践的方法。关于自学这一块儿的收获，我可以简单展开说说，因为半脱产备考的原因，我第二次考研也算失利了，我第一志愿是哈工大的市政工程专业，但是最后总分过了录取线，专业课单科没有过线，所以没法进入复试，

就调剂到哈工程读化学专业了，相当于又被迫地转了一次专业，在研一下学期正式入驻课题组后，我被分了一个具体的研究方向，可以说我当时整个状态都是懵逼的，因为我们导师很忙，带的学生又多，他不可能对很细节的一些东西对每个学生去指导，只能大方向去把握我们的研究进度，而那时我连怎么搜英文文献都不会，更别提如何找 idea 独立进行科研创作了，当时学院好像还有不成文的规定，就是每个硕士毕业生要发一篇 SCI 才能答辩，所以当时每天都慌得一 B，感觉什么都不会，很怕毕不了业。还好课题组有很多师兄师姐，他们会教我们最基本的一些入门操作，包括如何了解学术方向的全貌，如何追踪动态之类的，后来我发现很多东西总问师兄师姐太麻烦了，毕竟他们平时也很忙，所以我就自己去百度，去小木虫论坛自己搜索各种问题，也就是从那时起，我养成了一个习惯，能百度谷歌到的答案，绝不麻烦别人，这个习惯养成后，极大地提升了自己网络搜索的能力，到后来我发现，很多人问的问题我以前就搜过，我就顺嘴或者顺手就帮人解决了，有的人觉得我懂得多，其实我也是一点一点自己去搜索--实践--消化的，我觉得读研期间这个算是我唯一的收获，这会让我后来与师兄师姐或者导师的交流更有效率，也更有质量。就这样，靠着万能的互联网和线下的高质量交流，我从一个科研小白开始，啃文献，找 idea，动手做实验，这个实践过程也是被不断优化的，比如我一开始读 SCI 文章很慢，可能几天才能读一篇，后来就上小木虫找了一些科研达人，还有科研不错的师兄去讨教经验，我亲身实践后进步飞快，到最后一天都能高效读八九篇文献，虽然这段经历对于我职业发展没有太多帮助，但是这种互联网+线下高质量提问+实践+优化+进步的这套正反馈机制，让我重新形成了对自学的理解，也重新树立了步入新领域的一个信心，觉得其实没有什么太难的，只要方法对，愿意投入时间和精力，就算从零入门，也一定会有很多收获。一转眼，研究生也毕业了，面临找工作，当时有两个选择，一个就是回老家进原来那个单位按部就班过日子，再一个就是去外地闯一闯。我后来综合多方意见，还是选择南下深圳去闯一闯试一试，正好当时的女友也拿到了深研院的博士 offer。在 2017 年 4 月，我毕业拎着大包小包去到了深圳中兴报到，正式开启了职场生涯。因为我放弃了专业方向的择业，所以在中兴做的是业务工作，具体的话就是做供应链中心某一个物料的采购业务。这份工作可以说虽简单但非常熬人，辛苦，简单的话就是说几乎没有任何技术门槛，所以后来我们私下也经常议论为什么这么个岗位还非要招研究生，真的是很浪费，辛苦的话就是你每天要打好多电话，要跟公司的计划部不断汇报交期，跟各种供应商去催货，对于新项目还要跟研发的同事和供应商的项目经理不断沟通何时能量产，总之就是非常忙，每天要发几百封邮件，一个月通话要 2000 多分钟。当时每天忙得连轴转，只有在临睡的时候才有机会想想这份工作到底有什么意义，做 3 年和做 5 年会有多大区别，我真的喜欢或胜任这份工作吗等等，很多问题没有深想的时候，就累得直接睡过去了，第二天又急急忙忙起床赶地铁，周而复始，被忙碌的工作和生活推着走，战略上是懒惰迷茫的，战术上倒是异常勤奋忙碌，所以实际的收获其实很有限。除了职业方面的挫败感，当时感情方面也出了问题，我没能很好地在个人生活，工作和爱情方面处理好吧，导致最终这段感情也是无疾而终，正好在她开学的时候，我们也正式分手了。在分手之后，工作依然是忙碌的，去年的十月份，十一月份，因为供应商交不上的货太多，还专门出差去东莞，一住就是一个星期，天天去现场盯着生产，出货，我当时也有一个感受，现在中国做制造业真的太难了，从经理到一线工人，每个人都很辛苦，但是每个人都抱怨赚不到太多钱，这种劳动密集型的工厂也许未来都会被无人自动化车间取代吧。就这样断断续续干到了今年的四月份，我终于决定辞职了，因为深感这份工作附加值真的太有限，又非常忙，导致我没有时间去思考未来，我很怕这几年就这样稀里糊涂地忙过去了，所以跟领导提请辞职，想换一份工作。辞职流程走完后，先歇了三四天，狠狠地睡了两天觉，因为这份工作给我带来了太多疲惫，频繁的出差驻厂，手机 24 小时开机待命，半夜协调收货催货，这些对采购来说都是家常便饭。在短暂的休养生息过后，我开始频繁地投递简历，包括环境方向，化学方向的小公司，大公司，销售岗位，研发岗位，投了很多简历，也面试了几家，但是给的起薪都太低了，无奈只能放弃。后来为了平衡收支，就仓促地选择了一份外贸的工作，因为收入差不多，但是至少不用接触各种化学药剂伤身体，就先签了。从 5 月 4 日到入职新公司到现在，因为新工作的强度远不如第一份，所以我有很多时间去复盘 20 岁至今的 9 年，



复盘就分三部曲，做错了什么，做对了什么，未来该怎么做。过去 9 年坦白讲，做对的事情不多，极少，唯一做对的就是离开了东北老家，因为可以预见的一件事是，未来中国一定是区域发展极不平衡的格局，加剧的马太效应会让本就发达的省份、地区越来越发达，包括人才的规模聚集，资本的规模聚集，这些都会让本就发达的地区经济不断发展，然后带动辐射周边地区，而我的家乡东北，却存在一个无解的困局，人才方面留不住本地人，引不进外地人，资本更是，“投资不过山海关”“关门打狗”等负面形象早已传遍全国，因此东北注定是一座孤岛，会被这个时代抛弃。说到做错的事，就是看得不够远，读研，第一份工作的选择过于按部就班，导致荒废了挺多不少时间，年纪一把工作经验却零星跳跃，当然了，也挺对不起当初追随我来深圳的前女友，没有好好维系这段感情。

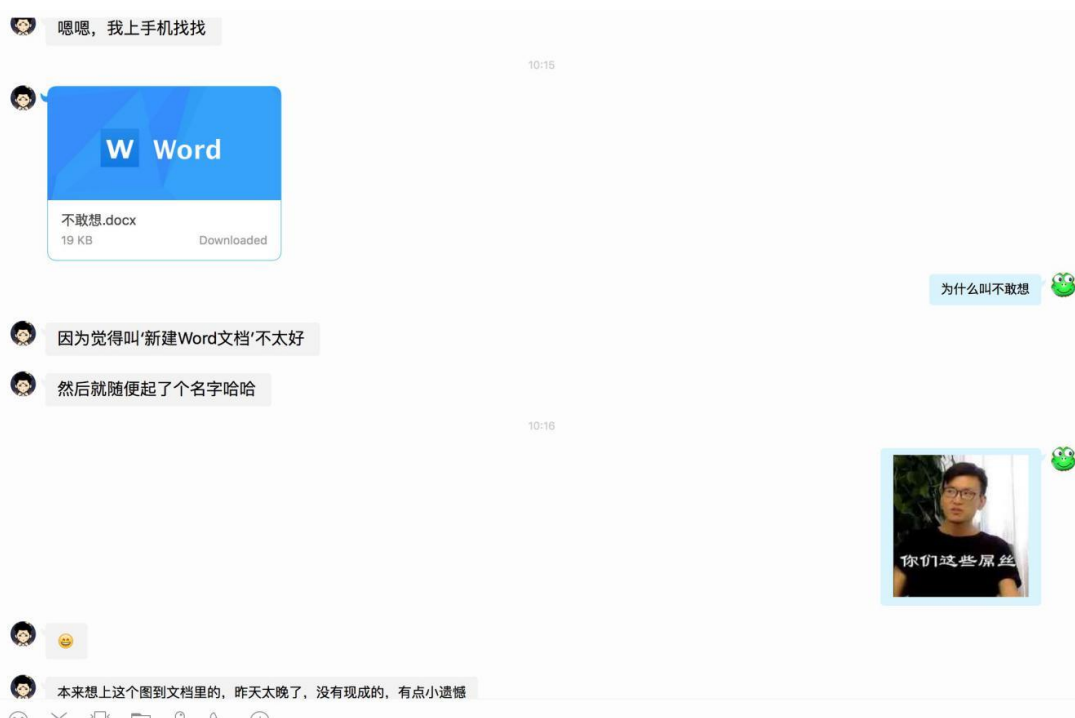
复盘了这九年的得失，接下来就是要考虑未来怎么做了，因为留给我选择的时间和空间并不多了，我们都知道一句话，选择大于努力，行业势能和行业红利可以决定一个人未来的成长曲线，我们肯定要选择曲线导数尽可能大的行业，经过较为仔细的思考和查阅各种资料，我决意果断忽略沉没成本（大学，研究生期间的专业培养，过往工作经验），高龄转行做程序员。在决定转行后，接下来面临的的就是学哪门语言（C++，Python，还是 Java？），学习途径（自学 or 报班？），在多大的学习时间才能达到一个胜任工作的水平？首先是语言的选择，关于这方面我也是搜了不少信息，综合比较后决定选择 Python，原因主要是 Python 对于转行人士来说，是相对友好易学的语言，尤其与 C++ 这种编译类语言相比；其次，Python 在科学运算，例如函数调用方面支持非常好，因此在人工智能诸多领域，它都有广泛的应用空间；另外，Python 支持多种操作系统，兼容性非常好，且随着 CPU 运算速度的不断更新，其运行效率相比编译类语言较低的劣势也会被不断弥补，因此 Python 一定是属于未来的高级语言；确定了学习语言，那么下面就是选择如何去学了，我结合自身实际情况，决定还是选择报培训班去学习，虽然现在免费的公开课，学习资料很多，但是于我而言时间紧迫，且这方面的培训已经非常正规，产业化，我应该用钱去换时间，通过一个系统化的，科学的培训尽快转行，如果能尽快成功转行，那培训成本并不算高，因为钱可以赚，但是时间一去不返，早点入行意味着早日积累工作经验，身价也能早日进入成长模式；在综合比较各种机构后，我决定选择路飞学城的网络课程，因为我每天除了工作外有很多业余时间，我可以有固定的时间和精力投入学习，其次就是导师 1V1 跟踪这种模式吸引了我，我觉得这会让我保持一个良好的学习态度，且得到比较好的答疑反馈，还有一层缘分就是路飞学城的线下地点就在我住处附近，我曾有幸跟创始人 Alex 亲密交流，我觉得他们是一个充满热情，专业且真诚的团队，他们也非常理解我的处境，对于我的一些隐忧和疑问都给予了鼓励和解惑，也是让我对转行充满了信心。我其实深知一句话，“师傅领进门，修行在个人”，我下决心把这个培训机构当做一个入门平台，就像读研时期培养的那个正反馈一样，只不过这次的路径变成了：视频课的知识摄入+动手实践写代码+遇到问题尽量思考或尽可能地去 Stack Overflow 搜索答案+高质量地与导师互动+每日学习心得（代码经验）的固化（博客形式），我接下来要做的就是逐渐建立这个反馈，然后强化反馈的每个组成部分，在这样的自学方式下，我有信心在通过培训后找到一份合适的工作，成功转行。最后说句鸡汤吧，当我纠结于是否高龄转行时，脑海中浮现出电影《盗梦空间》的一句经典台词：So do you want to take a leap of faith or become an old man, filled with regret, waiting to die alone? 为了几十年后的自己不成为那个充满悔恨，碌碌无为的老人，我决心改变，我也知道这条路我并不是踽踽独行，因为有老师背后的支持，有学员们的陪伴，当然还有那颗坚定不服老的心。我期待这段学习之旅，也更期待未来的精彩。

—————END—————

我问他，能不能引用到我的 blog 上，他说没问题并且等学完后，转行成功，还会再写一篇，好样的，祝他 6 个月后转行成功。

## 2. 不敢想

今天跟大家分享一篇毕业的 Pythonn2 期脱产班老学员感想，写的非常有意思，所以分享上来。这哥们之前是做客服的，一个月好像四五千，后来学完了转行做爬虫开发：



下面是正文

今天和武 sir 聊天，武 sir 突然提出让我写一下在老男孩学习 Python 的感想。感觉有些突然，然后就瞎想：为什么让我写呢？难道是因为我能代表广大屌丝群体？不对，不可能，我最帅。

说起学习 Python 的感想，我更想说一下我整个转行过程和一些感悟。跑题就跑题吧，反正我最帅。我 90 年出生，11 年参加工作，16 年下半年开始学习的 python。从 11 年到 16 年上半年，我一共做了四份工作，这几份工作无一例外都是比较乏味、也没有高工资的支撑。互联网行业之所以吸引我，最重要的一点就是较高的工资水平。

我来北京工作是在 2014 年 10 月左右，当时刚好有一个朋友学习完了 iOS 开发，找了一份 11K 的工作，我来北京之前的工作都是 3K 左右，当时就震惊了，马上就滋生出了学习 iOS 的想法，但是那个朋友随即就泼了一盆冷水给我，引用一下她的原话：你没有接触过这一行，我学成了是因为我在大学里学习过 Java，就算是我学习起来也是比较吃力，吧啦吧啦……。听完之后我心里也没底，毕竟没有接触过，学习也是一笔不小的投资。心里想：不如就此作罢，待安定下来再图之。没想到因为这个想法，险些大事休矣。一个疏忽，去做了客服，一年下来也能剩下来个 2W 来块钱。当时还有个安慰自己的想法：虽然工资不高，但是消费相对较低（公司有食堂，有餐补，有集体宿舍），况且工作也比较轻松，这么一对比感觉也还行啊。就这么浑浑噩噩的混了接近两年。后来我突然有了一个想法（也是现实存在的）：如果别人每个月把该花的钱都花了，然后剩下的钱都比我整月工资多的时候，我还能自欺欺人吗？大家都已经知道答案了，没错：我最帅。

这个时候我的另一个朋友（高中同学，在 2016 年的时候差不多 10 年兄弟了，一直没有断开联系），在老男孩 Python 全栈一期学习到了后半阶段。告诉我二期要开了，有没有兴趣。我既然已经面对现实了，当然不会错过这次的机会了。当时就决定来老男孩搞起（这一年我周岁 26）。

我决定学的时候从来没有考虑过别家教学机构，我也不知道有哪些。当时抱着试试看的心情到老男孩试听了一会课。当时没有见到向往已久的金角大王 Alex，试听的是武 sir 的课。课上只见武 sir 激情洋溢，唾沫横飞，前排学员顿感天降甘露，欲仙欲死。偶尔停顿半刻，尤觉余音绕梁，不绝于耳。让人生出无限瞎想。课间的时候由我十年的兄弟介绍，我和武 sir 进行了会晤，之后亲切的握了握手，顿时一股咸湿的温暖传递到我的掌心。回去之后我就向招生的老师问了一下：我学习的时候也是这个咸湿，哦不这个亲切的武 sir 讲课吗？

Python 全栈二期是 2016 年 8 月 15 日开班的，开班第一天，终于见到了传说中的金角大王，金角大王果然名不虚传，上来就是一段引人深思的鸡汤，中间夹杂着 sowhateveranyway 之类我等学渣听不懂的东西。听得我云里雾里，顿时觉得整个世界都在向好的地方发展，而我就是这个世界里最帅的。

说一说两位大王的 Python 吧。虽然我学的时候 Python 全栈才是第二期，但是周末班已经进行了十几期，所以教学经验在当时还是很充分的。也和班里其他同学聊过，他们有去其他学校试听的，给我说的原话是：那什么什么学校教的 Python 什么\*\*玩意啊。我们班里当时有很多从外地过来上课的，有深圳、广东、上海、南京、成都等等。这些城市互联网发展的也还不错，培训机构应该也会有，但是为什么来了北京呢？大家应该已经有答案了，没错，我最帅。

接下来说说讲师，在二期讲过课的一共有 5 位（如果比这个多，可能是我睡着忘记了），金角大王 Alex、银角大王武 sir、黑山老妖苑日天、海峰老师以及小岳岳老师（估计你们见不到了）。海峰老师给我们讲的网络基础的课程，时间很短，一天好像是，小岳岳老师大概讲了一个周左右的 Linux 课。重点说一下前三位老师。首先 Alex，对吧，sowhateveranyway。好，下一位。好好说正事：Alex 是一个我感觉精神层次比较高的一个人，用他自己的话说应该是有格局。他的一些想法反正当时我很难理解，索性就不去理解，反正那啥。然后感觉他的眼界也是比较高的那种。不管这个了，说讲课的事情。Alex 的讲课风格适合周末班，周末班的学生什么特点：基本上都会有计算机基础，懂运维，想转开发或者想转自动化运维的。所以 Alex 的快节奏他们可以轻松跟上会很自然的去学习知识，但是对于全栈来说，更多的是我这样的 0 基础渣渣，所以去跟他的节奏会感觉非常吃力。再后来我们就更希望武 sir 和苑昊来给我们讲课，虽然 Alex 确实能力很强，但是对于全栈来说他还是不太合适。然后 Alex 不仅是一个 Python 讲师，更是一个鸡汤大师，每当我们学习的精疲力竭的时候，能打上一管鸡血还是蛮好的。且有些鸡汤不只是励志，还让人沉思，比如他通过旅行者一号飞行器引发出人生意义的讨论，通过美国之行讲中美政治体制差异，通过《毛泽东传》这本书讲老毛对中国政治进程的影响，通过日本之行讲日本人不给别人添麻烦的理念等，都对当时只是长的帅却事实很屌丝的我的价值观产生了深刻的影响，现在我们 2 期的盗版视频在网上很多，各位有兴趣可以下载下来听听。

然后来说武 sir，武 sir 是个控场高手，也是个讲课的能手，武 sir 上课完全不接受任何人任何事打断他，然后所有人一直跟着他的思路走。记得第一次给我们正式上课，讲的面相对象，类。上课各种理解各种明白各种懂，然后做面相对象的作业的时候忽然发现，这不对啊，为什么我没有思路啊？当时就觉得这个咸湿，哦不亲切的老师难道就这样的水平吗？后来给我们上课上的多了，发觉这个老师确实行，



只是第一次上课的时候讲的是面向对象，对于 0 开发基础的人来说还是难以理解的。总体来说武 sir 的课更容易让人集中注意力，讲课条理性很强，对知识点的剖析也比较透彻。

最后一个是苑昊老师，苑昊老师也是一个很负责的老师，虽然他的经验没有 Alex 和武 sir 那么丰富，不过对于我们小白来说刚刚好，当时我们的基础课程绝大部分都是他讲的，会讲的很细，可能是他刚踩过我们即将踩的坑吧。苑昊老师更随和一点，所以课堂上经常会因为对知识点的理解变得很热闹，有时候还会吵起来。其实也不是什么坏事，可以加深对知识点的理解。现在也很庆幸当时的基础课程是苑昊讲的。所以我们那期的 Python 基础掌握的还都挺好，不信可以随时找我切磋，我搭理你一句算我输。

想想那段日子还挺怀念的，那样的上课气氛，那样的老师，那样的同学。

后来，为了表达我们对几位老师的怀念，我们毕业后全班同学，还每人出了 10 块钱，送了个最大的锦旗给学校，中间最大那个就是我们的，力压其它班哈哈。



好，接下来说一些好的、有用的话吧：

- 1、首先分析我 14 年的时候那个事情，当时对自己没有信心，导致接下来的近两年几乎虚度。当时就不应该害怕什么，既然想改变那么就尽早下定决心，然后努力。如果你确定了一个想法，那么最好从现在开始，拖得越久越会后悔，甚至拖到你放弃这个想法。
- 2、说说我开始学习 Python 不久的时候，当时班里也有几个 0 基础的，有时候他们会担心学不好怎么办、找不到工作怎么办。有可能是我比较浪吧，我学习的时候从来没有想过学不会和找不到工作，我当时想的是能不能找到 15K 的工作。既然我们已经开始学了，已经做出选择了，就不要再担心犹豫，与其担心犹豫倒不如全身心投入到学习中，学不会怎么办？加班学，直到学会。找不到工作怎么办？你学会了怎么会找不到工作。PS：我学习 Python 是有着破釜沉舟的决心的。
- 3、学习的过程会比较长，中间疲惫是肯定的，我也上课偷懒去看漫画、走神、困的睁不开眼。但是这些情况一定要是短暂的，当你有这些情况的时候你就要思考了，这个状态是不对的，你要摆脱这个状态，找身边的任何人，老师、同学、朋友、父母等等和他们聊你的困境，寻求帮助。相信，总有一款适合你。
- 4、坚持到最后，不是混日子混到结束，而是尽力的跟上课程的进度。坚持整个 Python 学习过程都让自己尽量保持在线状态。
- 5、再说一下年龄的问题，我之前也想过我都二十好几了，现在学不会晚吗？有一次读了一段鸡汤，说有一个人，他多少多少岁的时候学会了这个，多少多少岁的时候又学会了那个。反正就是活到老学到老的那种。希望你们也知道鸡汤里的一句话：你想做一件事的时候，什么时候都不晚。



现在全栈开班好像都是大班，所以你们有一些人学习起来可能更要努力一些，人多就容易有人不被监控到，所以希望你们更要克制自己投入学习中去。当然你们也有优势，全栈前两期可能给你们充当了小白鼠，所以你们享受到了更多的教学经验。然后，你们中可能更多的都是比较年轻的，刚从大学毕业或者是毕业没有多久的。单从这一点上就可以让我狠狠的羡慕了。

好了就说这么多，谁管你们看了之后有什么想法，反正我最帅。

2017/9/25 晚

看到有同学在评论区问写这个文章 的老学员的工资情况， 那 我说一下吧， 他 3 月工作， 在一家公司做爬虫， 下面是 offer...

11"

【此处有敏感信息】

保密申明：机密级

尊敬的 [REDACTED] 先生/女士：您好！

感谢您选择 [REDACTED]！我们非常期待您的加盟，有关您的工作及待遇事宜预期如下：

|                                      |                                                                                                                                                                                                                                       |                                                  |
|--------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------|
| 职位名称                                 | 数据抓取工程师                                                                                                                                                                                                                               | 工作地点：北京                                          |
| 入职部门                                 | 研发部                                                                                                                                                                                                                                   | <input type="checkbox"/> 待签订劳动合同时最终确认            |
| 汇报上级                                 | <span style="background-color: red; color: black;">[REDACTED]</span>                                                                                                                                                                  | <input type="checkbox"/> 待签订劳动合同时最终确认            |
| 入职日期                                 | 2017年4月10日                                                                                                                                                                                                                            | <input checked="" type="checkbox"/> 待签订劳动合同时最终确认 |
| 首期合同期限                               | 3年，从正式入职当日起计算。                                                                                                                                                                                                                        |                                                  |
| 试用期限                                 | 试用期限为3个月，从正式入职当日起计算。                                                                                                                                                                                                                  |                                                  |
| 岗位职责                                 | 您的岗位职责包括但不限于以下：<br>1. 负责分布式环境下爬虫系统的设计和开发。<br>2. 负责公司相关数据的清洗和处理。<br>3. 负责相关技术文档的编写。                                                                                                                                                    |                                                  |
| 薪酬                                   | 试用期第一个月收入：固定薪酬（税前）10400元/月<br>试用期第二个月起及转正后收入为月工资（税前）13000元，其中包含固定薪酬（税前）10400元/月，岗位津贴（税前）2600元/月                                                                                                                                       |                                                  |
| 福利                                   | <ul style="list-style-type: none"> <li>法定福利：享受国家法定医疗、养老、生育、工伤、失业及住房公积金；</li> <li>公司福利：年度体检；</li> <li>年假：员工入职后，可以享受带薪年假。具体年假制度见公司规定。</li> <li>补助：车补（300元/月）、饭补（440元/月）、电话补助（100元/月）</li> <li>团队活动：体育活动、聚餐、员工生日庆祝、年度公司级外出活动。</li> </ul> |                                                  |
| 按国家规定，公司将从您的收入中代扣代缴个人所得税及五险一金个人负担部分。 |                                                                                                                                                                                                                                       |                                                  |

PY脱产2期，纯0基础，之前在联通做客服，4K，一个大男人，谁愿意做一辈子客服呢？学了6个月，刚拿到13K的OFFER，他拿到OFFER后通过语音跟我汇报，我能听出他声音中的喜悦，讲真，最让我觉得自己工作有意义的事情，就是有机会帮助这些之前散落在各行业底层但不甘心的年轻人进入这个行业，13K在北京不算什么，但这是一个好的起点！我替他高兴！

### 3. 给一位做技术迷茫的同学回信

昨天收到一封网上看我视频的 同学的邮件，我想其他人可能也遇到这样的问题，与是与大家分享下。

邮件内容

Dear Alex 老师:

很高兴能成为你的学员（网络班学员），不算第一次接触编程，但确实还是属于比较低的水平；之前属于“自学型”的，结果你肯定猜到了，当然是没有结果了。

之所以以这种私信的形式与你交流，不是全听了你伟大的“鸡汤”而对你表示崇高的敬意和崇拜，而是想与你真正的交流下，不求你指点我的人生，不过很乐意看到你的所想所感，因为自我认为自己是 **loser!**

首先说一下我的情况吧，我是一个快接近三十岁的 IT 男，无天赋，其貌也不扬，算属于踏实工作的那种；毕业已经 7 年了，出来算是干了 IT 这块（专业不是 IT），不怕你笑话，那时所谓的 IT 就是在电脑城组装电脑；一直摸爬滚打，换了好几份工作，干过售后，干过售前，从开始的三线存储厂家，到现在的主网网络产品的总代公司，拿过华为的 HCIE；自己还算属于对技术比较热衷，不排斥对技术的厌倦。可一路走来，感觉自己一无所有，没有房子，没有车子，还没结婚，钱袋子里更是剩的不多，就像你说的感觉——一直属于穷人的认知层次。其实我自己也在苦苦的思想挣扎中，想改变自己的认识。

在公司里工作，我还算是兢兢业业的，对交代的工作都能完成。不过这也是我最恐惧的，混个五六年可能还能混出一个高职位，不过看到那些已经五六年的老员工，我真心不想成为那个样子，说真的我怕那种生活，死板而又无聊，说句高逼格的话，我向往的是：阳光和自由！我愿意走出舒适区，趁着我现在还有点冲劲。

就拿学你的课程来说，当然不是心血来潮，或者钱多了没地方花；自己也是考虑过的，结合了时代发展的趋势，提示自己的价值考虑，多维度的发展，提升自己的技术壁垒，还有一个小小的愿望，就是圆我大学时的梦吧（大学那会参加编程比赛输了，那时我可伤心了好几天）。当时反对的声音也是有，摆着眼前的当然是什么年龄大，什么不务正业，什么不好好工作等等。每当这时我还有句高逼格的回答：我 60 岁壮年，我还年轻。说归说，内心其实还是有这般或者那般的纠结，不过抛开所有的困惑，就当圆自己一个当初的梦吧！

你的见识和格局肯定比我高，当然您也不愿自称“高人”，不过您有特斯拉，我没有，哈哈；别人有过人之处的地方，虚心学习有何不可。目前自我其实对自己的前途比较迷茫，不知往哪里走。我想积极的改变自己，不想自己陷于“穷人的思维”，可自己的格局确实太浅，虽然自己也在慢慢的改变，不过处于我人生的这个阶段，有点所谓的“前后夹击”，着实让我困惑难免。

不知不觉跟您说了一些内心的话，希望您原谅一个 loser 给您传播的负能量，我知道自己路还得自己走！

至此！您的学生（石 xx）于 2017 年 09 月 10 号

## 我的回复

heyman,

我仔细看了你的来信，本想故意放几天再回复，那样显的沉思熟虑些，可我忍不住了。

其实有很多人，如你一样，活在莫名其妙的迷茫、焦虑中，这种迷茫与工资收入关系并不是太紧密，不是所谓 loser 的专属，我有些朋友，一个月工资挣好几万，一年收入大几十万或数百万，但他们除了物质上滋润些，依然内心忐忑、迷茫。包括我自己，之前也偶尔这样，买了很多很多人觉得好像很 nb 的 TESLA 也并没改变这些。所以，你迷茫这事，本质上与钱关系不大。

你迷茫或许是因为你没有找到自己真正喜欢做的事情，这句话太俗套了，接下来就要告诉你 follow your heart, do what ever you want 之类的了。容我讲个事情。

我的健身教练，他身材很好，跑马拉松，全程下来 4 小时，过中极为痛苦，我说那还跑个毛线，他说感觉自己每跑完一次，人生就得到了一次升华，我说扯蛋，他说真的，他说跑完后，那种爽，比搞姑娘要爽 100 倍，搞的我竟然有点心动了。

我看到一篇文章，讲一个老板，有钱了，开始玩起海钓，钓一种叫石斑鱼的东西，这鱼，不太容易长大，说是能长 2 公斤就算非常大了，只能在特定海域才能发现大的，且跟气候等有很大关系，这位老板，为了钓一条大的石斑鱼，花大量时间研究它的习性、生活海域、气候相关知识，经常跑到公海上呆好几天在船上。只为这么一条我们平常人都不太 care 的事情。

我今年去硅谷，走访 google,fb,airbnb 等公司，看到很多公司的程序员，四五十岁，还在写代码，一件事情干个几十年，乐于其中，而我们中国的程序员，写了几年代码，刚熟络起来，就想着做管理、想着创业、想着转行，焦躁的不行，生怕自己的才识被埋没，好像他做的现在的工作，都只是下一步的铺路石，他从来没有真心热爱过、上心过这个工作，心中喜欢的、期待的一直是下一个 job。

通过几个小片断，我想告诉你的是，很多让人很有成就感、感到真正满足的事情，都是需要下大功夫、沉得住气熬出来的，浅显的快乐会迅速的来，亦会迅速的走。所以，对于你的建议，若你真是喜欢编程，请一头扎进去，用代码实现你之前一个个的对于技术的想法，别管他以后能给你带来多少的工资。做有趣的事情，比挣钱更让人快乐。

我当年学英语，并不是为了挣多少钱，只因为我就是喜欢自己能流利说英语的样子。我当时自学编程，并不是为了拿它挣高工资，或为了今天的创业，只是因为，我一直想独立开发一套 IT 自动化的工具，当时想，如果能做出来，会觉得自己非常 cool。

若可以，我希望你也能与当时的我一样。



## 4. 白领如何才能买得起特斯拉

2014 年提的这个问题，当时在中关村上班，公司对面新东方楼下停着一辆白色的 tesla,每次中午去吃饭路过，都要去瞅一眼，觉得这车太 nb,但离自己太遥远，当时闲的蛋疼跑到知乎来随手提了这个问题，但从来没敢想过能实现，那时刚买第一辆车不久，不到 20 万，当时朋友推荐买帕萨特、蒙迪欧、阿特兹，都不敢买，因为基本都要 20 多万，当时看了看大众 cc，都觉得离我非常远，至少 tesla，呵呵，打死也不敢想。2016 年回来自问自答，两年后，终于实现了小目标，两年前不敢想，只能意淫的东西，现在才发现，梦想还是要有的，其实一不小心就实现了。



-----更新于 2016 年 10.7 日-----

这两天一直在写代码，没上知乎，刚才随手打开，发现新消息提醒 99+，我还觉得应该是知乎又有问题了，没想到点开一看，我擦我擦我擦，1900 多个赞，此时的我，真的是这个样子。。。



这个问题从发布到现在 2 年多了呀，妈的，之前一共有 5 个答案，都说讽刺说让我去做鸭子\去搬砖之类的，我只是前天又闲的蛋疼自己做了个总结呀，随便回答了下上面那段话，结果被这么多人发现了，是不是知首有推广机制呀。。。我去吃了饭，整理了下忐忑的心情，看了看大家的评论，决定再随便扯几句。首先，我是个张扬的人，直接点讲，是个爱装 13 的人，所以评论里讲我是在装逼，其实我一点都不生气哈，反正我又不是干装。。。至少是自己实现了个小目标嘛，你们就当我是浮夸吧。2014 年，我在中关村上班的公司，就是汽车之家，我是在技术部门，做 Python 开发喽，屌丝程序员一个喽，当时的工资其实还不到 3 万喽，差一点(10.8 日早更新，看评论讲我 14 年就三万起点高。。。呵呵，哥哥们，我 07 年就工作啦了呀，刚工作时才 2500 呀，最后每月剩 1500 给家里，那时穷的叮当响，住群居宿舍，为省钱一天只花三块钱吃 2 顿饭，早上一张煎饼，中午一个馅饼，晚上不吃饭，有 2 次贫血的晕过去，我也穷过好么。。。)。所在的公司就是搞车的，公司老板李想当时就买了辆 Tesla 呀，好像 CEO 秦致也有一辆，当时 Tesla 刚进中国不久，大街上跑的 tesla 非常少，绝比是土豪玩具，我们对面楼就是新东方，当时停了一辆白色的，我在 10 层，我的工位就在窗户边上，往下一看就是这辆车，每天真心都要盯着看几眼，前面讲过，当时从来不觉得这车跟自己有关系，也不敢想自己以后有机会买上这车，毕竟，一个技术屌丝，工资又不高，不吃不喝干 3 年也就勉强买上，但是买上了也觉得不符合自己的身份呀。就像一个真是特别漂亮的女神，你只敢看看或意淫下，拥头的事想都不敢想。当时我的小领导开的是 bmw3 系，就觉得挺 nb 了，部门里有个新来的同事，开的是 A6，我操，就惊呆了，真心觉得人家 nb，给他记名叫 A6 哥，心想，什么时候，我能达到这个 level 呢。。。有的同学讲，你一月小 3 万，A6 便宜的才 30 多万，你可以买的起呀，没错，是买的起，但不敢买，说白了，就是消费观还没达到那个水平。

**其实人的痛苦很多时候确实就是因为想要的太多，而自己的能力又有限，**

我当时就是这个样子，想要的东西跟自己的现实落差较大。不过，虽然觉得不可能实现，自己心里还是希望有朝一日，能搞上一辆 Tesla 的。当然靠给人打工的话，感觉希望不大，我之所以还抱有希望，是因为当时已经在兼职创业，但真心不是做微商哈。从 2013 年，就利用周末时间在一家小 IT 培训机构教 Python，讲了 1 年多课，得到老板认可，终于拿到了股份成了合伙人，由于当时学校非常小，2013 年我刚加入时，就一间教室，租的还是特别偏远的昌平沙河一个破写字楼里那种商住 2 用的 1 居室，我第一次去到这个学校谈合作时，一下地铁，荒的不得了，冬天，特冷，还要走路 15 分钟，才到，一个非常破的小破楼，当时生怕自己进了传销组织，大家可自行补脑。当时就是抱着挣钱外快的想法开始讲课，一周讲一天，由于是第一次系统性的讲 python 课程，好多知识细节自己也需要不断学习，基本上除了上班之外，就是备课，压力挺大的。由于我在 2012 年左右短暂的转行去华尔街英语做过半年销售，对于课程销售\宣传还是学了些东西的，所以我不光负责讲课，还帮公司做一些公开课宣传什么的，做了一年左右，逐渐在市场开始有了些声音，就这样慢慢的，我就开始全权负责这个项目了，13 年这个项目才做了不到 50 万，14 年做到 80 多万好像，到后面我从一周讲一天变成了周六日同时带 2 个班，周一周五还要上班，几乎等于全年无休吧，有那么一年时间，几乎要抑郁了，因为已全然没有了自己的生活，butanyway,就这样坚持了 2 年，由于自己讲课风格还是比较随意的，并且经常给学生讲讲段子呀\鸡汤呀，比较受学生喜欢，所以吸引的人越来越多，同时所在的学校在业内知名度也开始高了起来，2015 年一开春，学校招生量开始多了起来，同时开的班在 4 个左右，一个班的人数也达到了四五十人。到了 2015 年 4 月份，我已经想离职了，因为实在是觉得，在公司天天写代码，再写十年，我也与 Tesla 无缘呀哈，加上当时学校的创始人一直希望我全职跟他一起搞，之前讲真是不太敢，毕竟 13 和 14 年时，并没信心觉得学校以后能做大，当时自己还是打工心态，觉得一下子离开收入稳定的大公司，不安全感就上来了。不过有天早上，我从天通苑做地铁去公司上班，地铁里听到有 2 个人在我身后讨论 12306 订票技术的问题，从讨论的细节来看断定他们是程序员，突然间就觉得，妈的，全中国这么多程序员，我们公司就好几百个，多我一个不多，少我一个不少，我他妈还恋着这个职位，有毛线意义呀，妈的，不干了，然后，就到了公司，直接找到了老板，辞职了。。。呵呵。辞职后，就全职搞培训喽，去年夏天，学校从 3 间教室变成了 6 间，教室面积达到了 500 多平，全年收入到了 600 万左右，今年至 8 月份，收入已经突破了 1 千万，全年应该 1500 问题不大，同时教室现在应该有 1 千多平吧。做到这些，是在公司没花一分钱做广告的前提下噢，基本都是口碑传播，哈哈，还是挺开心的，不过从 7 月份开始，也开始投点广告了，毕竟，想做大，全靠口碑还是太慢了。目前公司刚在融资，10 月份应该差不多钱进来，纯线下模式发展太慢，还是希望把在线教育搞起来，去年开始试水在线，今年在线收入应该能到 500w 左右吧，自己搞出了一套随到随学，1 对 1 辅导的模式，目前看效果还算不错，明年收入有信心应该在 3000-4000w 左右吧。扯远了，anyway,虽然心里一直想要的是 tesla，但总觉得离多比较远，于是在 2015 年，我搞了个分步走计划，想先努力买一辆牧马人，当时觉得这个红色 4 门的也是帅爆了，天天意淫着买了这车到三里屯去把妹然后车震。。。呵呵，就在去年，自己还是这么天真。。。



不过，今年年初，我去美国度假，发现大街上好多人看这个 jeep,然后我就想，果然我品味不错，美国人也都喜欢这车，知道这车是美国产的，就随着网上查了上美国官网的价格，查完后，彻底无语了，在美国才 4 万美元左右，也就是 25 万 rmb 呀，妈的到了国内要 100% 关税，到 50 多万。当时就觉得，妈的花 50 多万买个 20 来万的车，实在不值，然后，就基本不考虑了。。。虽然到现在依然挺喜欢这个外观，但，爱过，就放她走吧。。。于是目标又转成了这个，心想，今年先搞个野马玩玩吧，再不开跑车，再过几年，就算有了钱，开这个也不合适了吧。毕竟岁月是杀猪刀呀。。。



虽然决定先搞野马了，但还是忍不住每过几天还是要跑到网上看看 tesla 论坛，就像已经有媳妇了但还是忘不了曾经那个没能追上的姑娘一样，得不到的永远在骚动嘛。好像三四月份时吧，Tesla 发布了 Model3，讲真，我并不觉得 model3 好看，且起不到装逼效果哈哈，在看 model3 发布会时随手就约了个 modelS 的试驾，心想，反正试试又不花钱哈哈，结果去了后，发现了 tesla 一个大阴谋，本来我只是喜欢 tesla 的外形和大屏幕，对车的加速本并无感，结果销售带我试了个驾，我那个去，我那个去，试的是最高配的 p90d,我操，3s 百公里加速，直接血脉喷张，激动的不得了，跟我一起去试驾的 2 个哥们也激动的不行，都觉得这车太 nb 了，然后我就开始给自己各种灌鸡汤，什么现在就要幸福啦，什么人生苦短啦，什么三里屯的妹子在等你啦，还等什么呢，于是当场就订了。。。去他妈的，老子就是现在就要幸福。翻出当时的朋友圈。。。





那句话说的真是对，很多东西，只有亲自体验过后才知道你是真的需要他。。。很多人，一生也没被生活温柔的对待过，一生也没见过更好的世界，一辈子浑浑噩噩，就这样混过去了，我才不要做这样的人，oops,一不小心鸡汤又上来了。。so ,就这样喽。。。噢，装逼要彻底，这车不是贷款买的，全款哈，因为我的工资发放不稳定，银行不给批贷款，所以只能全款啦，全下来 90 万左右，还好啦，也就是 5 环上一个卫生间的价位。8 月份，取了车。。。开了到现在 2 个月，感觉非常满意，不是为 tesla 做广告，我确实就是非常喜欢这车哈，另外，开上这车后，还发现一个问题，就是确实心态不一样了，之前觉得大街上开 bmw,audi,奔驰的都是有钱人，现在，哈哈。。。哈哈。。。不过有一个事情比较遗憾就是，提车到现在，由于工作忙，也没去成三里屯。。。当然，必然，也没有撩到妹子。。。哈哈最后最后，推荐大家去试驾特斯拉，你会明白为啥这么多人喜欢他的。我的偶像是埃隆马斯克，他之前也是程序员奥，以后我还要努力做他的火星乘客，特斯拉真的只是个小目标。毕竟我们要去的是星辰大海!PS. 特斯拉是不是应该付我点广告费哈哈最后最后再废话一句，人与人的差距真的多是在 20-30 岁间拉开的，我并不算身边朋友里混的最好的，2010 年我在金融公司才挣一万二，当时做我旁边写代码的同事，转了技术金融，做量化，去年他们 6 个人帮公司挣了一个多亿，今年奖金 600 多万，另一个同事奖金拿了 1000 万呵呵，确实我也惊呆了，公司名不说了，做金融的都知道，反正是国内顶级投行了，不过这就是事实，真的有好多牛逼的人，逼得自己不努力都不行，不努力人家以后就不跟你玩啦，我体会过穷的滋味，穷人的选择权特别少，只能跟一大堆穷人争夺最次的资源，年轻时穷点没问题，因为年轻可以掩饰很多问题，20 来岁你没钱，没人笑话你，但到三十多了，你还是穷，那干很多事情你都没底气，整个人也不会很有自信，到时候那种酸溜溜无奈的感觉，只有体会了你才能明白，所以给 20 多岁的朋友建议，趁年少抓紧努力吧，折腾折腾说不定就成了。

## 5. 关于认知、隔绝多维度发展的感触

写在 go 开发 6.3 号开班之际

昨天开管理会议，我得到 go 开发的报名人数非常少，让我非常诧异，go 未来大火的趋势这么明显，本以为轻轻松松开班的事，但招生结果却出乎意料。由此我得出一些感悟，想跟大家分享，相信听进去的人，会对你的人生发展有帮助。

现在的做运维的、写代码的技术人员如此之多，多数人都希望能不断往上爬，为什么往上爬？除了带来直接的收入增长之外，其它的一些东西不知各位有没有深思考，就是你越往上走，你的机会越多，这个社会永远是 2/8 原则的，20% 的人掌握着社会 80% 的资源，你越往上走，就发展好多机会越是唾手可得，这个我自己有切身体会。现在经常有人跑来问我有没有兴趣搞几十万甚至上百万的软件项目。我也没间接，也不想花时间搞这小项目。但是妈蛋的，放在 5 年前，我各种想着找挣外快的机会，但就是找不到。我现在的段位还不够高，所以只能碰到一些小机会。但我接触过一些段位高的人，身家过亿的基金公司高管、企业创始人、还有特别 nb 的 03 年就投资过阿里巴巴投资人。发现他们的机会真是太多了，动不动就是几亿的项目啥的。这些人不缺钱，所以人家也能特别淡定的看很多机会。

有的同学讲，他对挣太多的钱不感兴趣，够花就行，搞这么累干嘛？哈哈，我的哥，这只能代表你只看到了表面，其实真正体会过的人会明白，爬到高段位后，更多的钱和机会都只是附加带来的东西。真正让人着迷的是你在高段位所收获的格局和认识。

我估计我跟别人都讲过不下 100 次认知的事情了，今天还是要再说。2008 年我尝试创业搞快餐外卖，09 年尝试搞团购，2010 年尝试自己搞培训，2011 年我还动过搞类似滴滴打车的产品，特别是培训项目，我一度注册了公司、域名、网站、传单等，最后当然是失败了。那时我觉得一切的设想都是如此完美，坚信找到了所有的理由来验证自己的想法是正确的，竟然他妈的会失败。。。现在想想，当时真是 sb 一个，如此的 too young too simple too fucking naive. 为什么几年前觉得无比正确的事情，现在自己都觉得 sb? 所以请各位记住一句话，“你现在的认为的无比正确的事情有可能只是假象”，前些天，跟一基金公司高管吃饭，他 40 多岁了，他是公司 2 把手，他表弟是公司创始人，去年他们公司利润 5 亿+，注意是利润奥！他之前也做技术的，后来抓住一个机会，起飞了，讨论到认知的问题，他说了一句话，让我印象深刻，他说，他现在看身边的很多人，就像看装在盒子里的几只蚂蚁，忙忙碌碌爬来爬去，你能预测到这些蚂蚁的结局，且你还不要尝试去帮他，因为认知差距大，你说的话他不一定听的进去，甚至觉得你忽悠他，或对他有所图（真他妈可笑，人家段位甩你几十条街，能从你个穷屌丝身上图啥？括号里的是我的评论哈）。听了他的话，我觉得真是太对了。我现在看我们很多同学，就真的像高中生看小学生，我无心抬高自己，但是事实的差距就在那里，你们天天想的是工资怎么从 15k 到 20，到 30k，刘耀混的好些，才 22 岁，就 1 年内工资从 5k 涨到 22k，但不也就是 3 年后想拿到 50k 么？我在想什么？我在想怎么把老男孩做大做强，干掉达内，百亿市值上市，成为受人尊敬的企业家，做影响更多年轻人生活的事情，天天想的、做的事情不一样，结局必然不同。

我看着咱们的一些同学，真是活生生的把一手好牌打烂了，本可以 3 年架构师、5 年 CTO 的主，结果却在那 sb 一样荒废青春，还洋洋得意。我记得我跟很多同学说过，如果你迷茫，你找我，我免费给你指路，你从老男孩毕业后，咱们就没有利益关系了，但我为什么还要跟各位在这扯蛋？我是个爱装 b 的人，大家也知道，但现在装 b 开 tesla，泡北大姑娘、买海外别墅都真的不能让我兴奋起来了，让我真正开心并燃起来的的是能看着我带过的人一路从屌丝逆袭成高富帅。所以我纯粹希望各位混的好，这样我以后可以吹 nb 说我的学生是哪个公司的开发总监、cto、甚至他始人。结果呢，我跟 n 多人说你去学英语，你会受益终身，别在乎这几万块，你的几万块不过是别人一个月的生活费，别短视，别陷入穷人思维的 deadloop，但有几个人去了？

要想往上爬，必须不断武装自己，别天天跟身边和你 level 一样的人混，这样你当然不会有危机感，也找不到攀比对象，我去硅谷看到的工程师，一水的斯坦福、mit、卡梅隆等，就算之前我在中金的时候，身边也多是清华北大，在这样的人群中，你才会感知到差距。你想往上爬，这些人不想么？你身边的屌丝多数是没机会的，你以后竞争者是那些各种吊炸天名校、身背各种干死你装备的人，你怎么跟他们拼？？？？呵呵

我看到了 go 的趋势，我觉得各位会 python，再会 go，下一步就离架构师的路越近了，当你拿到一年 50 万的时候，100 万就容易太多。所以我开了这个课程，讲真，现在 python 的班同期有五六个在并行讲课，每个班都上百万的收入，这个 go，6 千来块的学费，招到 50 人也才 30 万，我要是想多挣钱，直接多开 py 好了，废这毛线劲在这扣这几十万有毛线意思。但我个人对老男孩的定位是，希望做每个学生终身的职业生涯辅导伙伴，我

希望能在各位不断往上爬的时候能帮你加更强的装备，让你轻易把你未来职业路上的对手、竞争者干掉。只是帮各位帮工资从 3 千涨到 1 万，从一万涨到 2 万真是太他妈没意思了。

现在会 python 开发的运维不算什么优势了，老男孩还在开足马力突突突的培养你的竞争者，当你会的技能大家都会时，你就要思考了，怎么能甩开 80% 大部队，成为那 20% 的人，所以一定要有多维度发展的思路，做到别人会的你会，别人不会的你会，然后你就 nb，你就有更多机会，你就能更快的往上走，越往上走，你就发现机会就越他妈多，每抓住一个机会，你就更进一层，掌握更多资源，不出几年，你就把其它人甩 n 条街了。

我之前在群里发课程表，很多人说，学了公司也用不上，没啥意义，或者 py 还没学好呢，学更多也没用，我想说，这还是穷人思维，很多时候，我们学了好多东西，可能一辈子也用不上，但正是这些东西，让你逐渐变成一个更优秀的人。中国的原子弹你见什么时候用过？但有了这东西就是 nb，就是比别的国家硬气，就是可以和平发展好几十年，就是能在联合国稳当的做常任理事国，可以一票否决权。好多浅显的道理，不需要过多智商就能明白，只需脑子转个弯，换个思路就通了。

一口气说了这么多，我不希望你看完之后觉得，我是在忽悠你报 go 课程，那就完全跟我的初衷背道而驰，这个班有你没你都会正常开，我只是借这个 go 开班事情，说一下我近来的感触，各位当鸡汤喝掉就好了，因为无论说啥，终究还是有一大片人永远叫不醒。



——于 2017 年 5.25 日清晨

## 6. 自学和报班你想不明白，必然是智商不够

昨天有人在我的技术群里又讨论起了学 python 是自学还是报培训的话题，事端引起是一个小白网友说自己没什么基础，想学一下 python，然后有人说这东西简单，建议他自己买本书跟着学学就行了，也有人说想快速提升还是报班快，双方就这样各执一词，一会就吵起来了，坚持自学的说培训学校都是骗钱的，说那些推荐报班人都是培训机构的托等等。最后谈话变成了人身攻击，管理员不得不禁言双方才算平息，那个刚开始提问题的小白也悄悄的退出了群，目测是相信那个推荐他自学的网友的话，因为力荐他自学对于这位网友来讲并不会有什么利益所得，所以看上去似更不像是骗人的，而推荐他报班学习的那个网友，说他是培训机构的托这个逻辑也很能令人信服，要不然你推荐培训机构干嘛，肯定是你推荐一个人有提成嘛。。。

这样的论战在各个技术群里经常发生，相信多数人都见怪不怪了，但我总是非常看得很多说话不负责。任随意指点别人人生且不考虑后果的人，这样的人信口雌黄一下，可能就带一个小白走入了一条弯路。所以今天我就给大家一些我对培训和自学的看法，希望能真正帮助众多技术小白做出最适合自己的学习技术的方式，呵呵。

首先我个人是搞培训的，应该算是国内最早一批搞 python 培训且到现在活的还算不错的人。估计很多人一看我的身份，就觉得这注定是一篇软文，看我的话也开始变的漫不经心了，因为在他心目中已不知何时被埋下了“培训机构都垃圾”的观念。追之缘由，也有可能是“伤之深，恨之切”吧，so 有句话我坚信，如果一个人首先不认可你，那你无论表达什么观点，在他看来都是错的，都是别有用心。所以没关系，我也懒的去叫醒这些装睡的人，呵呵。

先说我的观点(以下所有论点都以学习 python 这个语言来为讨论对象):

1. 如果你是0基础小白，那“自学约等于浪费时间”这个公式对你来讲在大多数情况下都成立，且此公式适用于几乎所有没考进985、211的人，别跟我扯淡装b说你天资聪慧，我不care,反正你有的是时间浪费，那就以身试水后才知道水深吧。
2. 如果你是有1-2年其它语言（shell,perl,php除外，因为会这三种语言的人群里面产出最多只是掌握编程语言的皮毛而已的伪程序猿，然后就说自己是会开发，呵呵，你会毛线。）开发经验的人，如果现在想学python,我建议你自学，因为语言在多数情况下都是相通的，差异更多是语法和一些特定功能而已，so此时上万的学费去报班学习，确实看似不值，虽然如果你报班，依然会比你自学这门语言学的更系统些，but相信多数这样的人群还是会选择自学。因为觉得自己都会开发了，再花钱去学一门其它语言，在其它同行眼中可能会被看做能力很菜的表现。

python 我是自学的，学 python 之前我是做运维的，只会写 shell,也学过一点 java\c++,但真心只是皮毛而已，由于自己想搞些自动化工具但 shell 太局限，java 和 c++对我想做的东西来讲又感觉有些重，于是发现了 python,

2009 我刚开始接触 python 的时候，市场上哪有做 python 培训的呀，连中文技术资料也几乎只能找到一本叫《python 简明教程》的在线基础文档，当年我只能对着这个文档去写代码，结果是文档看了一遍又一遍，知识点看懂了，但一写代码就又会，完全没思路，没思路就再回去看，看完还是没思路，很多时候觉得自己非常笨，明明感觉有些东西不难，但自己就是没办法自如的运用。我到现在都记得非常清晰的事情是有一次我在写通过 socket 传文件的程序，发现程序有时能接收完完整的文件，有时不能，自己搞了近一周也搞不明白，网上也找不到相关的解决方案（或者是因为我不知道该如何去搜），甚至都开始怀疑这是不是 socket 本身的 bug,最后终于忍不住像一个搞 c++开发的同事请教，他当时不跟我一个办公楼，然后远程连上来，拿我的代码改了几行，不到 10 分钟，问题就搞定了。当时我瞬间陷入了对高手的无比崇拜和对自己的愚笨无比逼视的冰火两重天之中。这样的事情后来也是经常发生，个中痛楚，我相信只有经历过的人才有体会。

不过我还是跌跌撞撞自己学会了 python,但只有我自己深知，自学的过程是多么的缓慢和漫长，遇到问题找不到原因，又没人可问时的无助。这样的处境我再也不想再经历。

so 对于同是小白的你，我相信你应该也不比当年的我聪明到哪里去，如果你做好了愿意受自学这种折磨的准备，我依然还不建议你自学，因为即使你是甘愿找虐小受型人才，但你依然经不起时间这把杀猪刀的考验。为什么？因为你要明白，这个社会是有竞争的，你的同行不会等着你前进，你的行业不会等着你前进，你走慢了，你落伍了，你出局，就是这么简单。

曾经有一个做运维的，14 年听过我的公开课，今年也来参加公开课，但从来都没报名学习，由于经常在群里@我问一些很基础的问题，搞的我很烦，因为这些基础的东西在我的课程里都有细讲，我就

问他，你为什么没报班学习，他说没钱，我说好吧，这真是个好借口，我又问，你自己学了 2 年了，还在问我基础的问题，你不觉得你在浪费时间么？他说也知道，我问那你为什么还继续这样做呢？他就开始笑笑打马虎眼，我得知他是 93 年的之后，就问他，对于现在的你来讲，你觉得什么对你来讲是最重要的，他想了想说，钱，我当时真是笑的不行，不知他是故意气我还是本来就这么想的，我说你 1 个月只能挣 1 万多点，你再努力挣你的能力和水平目前也就只能支持你挣这些，结果你认为对你来讲最重要的钱，却还不够我一个月的生活费，你觉得这代表着什么？代表着你拼了一个月的青春，却只换回了别人的生活费。想想也是挺可悲的。他语塞，或者觉得自尊受了伤害，but I don't care, 因为我知道自己再也不想跟他有交集，我说，你以后不要再找我了，我帮不了你，你的问题不在技术上，你这种人注定在底层混。

## 论传统教育与培训的区别

我女朋友毕业于中国最好的大学，现在拿的国家全额奖学金在美国留学，有一次我们在远程 xxoo 完之后突然聊起培训和教育的区别，她说传统教育其实是金字塔式的精英教育，由于教育资源是有限的，特别 nb 的大学就那么十来所，还不错的大学也就几十所，他们一年能消化的学生最多也就几十万，每年参加高考的好几百万，能进大学的 70% 多，能进好大学的 20% 都不大，除此之外还有一大票没考上大学的，因此你会发现，学习越好的，越能享受最顶端的教育资源，很多人讲说大学没用，大学学不到什么东西，呵呵，那是因为你没进入好的大学。又有很多人讲说学历没什么毛用，上了大学又怎样，不照样还是有很多人找不到工作，呵呵，我真不想与这种返智言论讨论，降低我的逼格，想说，说学历没什么用更多是那些本来就有学历的人一种自谦表达，是为了安慰那些教育背景不好的人，让他们觉得自己也有机会而已，你当真，你就输了。

so 我们会发现，有 n 多人其实是没有办法挤到金字塔上层的，甚至有 n 多人连金字塔都下层都排不进去，连大学都没得上，那这些在塔底或塔外的人如果想往上爬怎么办，草莽的靠卖羊肉串也能发财的时代已经过去，这年头要创业拉投资，人家投资人也可看你他创始团队的教育背景，至少学习好代表你智商不差，出身名校能把事做成的机率至少从统计学角度上来讲更大一些。因此那些被传统教育刷下来的人想往上爬怎么办，肯定要学习。

这时候培训的作用就体现出来了，当然我不说这些人除了参加培训就没出路了，但是培训机构其实是相当于给了这些传统教育的 loser 们一个 second chance, 给了他们一个跟这些从传统教育金字塔中上层训练出来的相对精英的人群一个竞争的机会，比教育背景拼不过，但比实战技术能力上至少不差，总得占一样吧，要不然公司要你干毛线。

## 谬论之“找个人带比培训要高效”

有人讲，对于编程这东西，找个公司或高手带一下，一年就能学到的东西肯定比培训机构多多了，其实我想讲，说这话的人连基本的社会规则都搞不明白，公司凭什么带你？高手凭什么带你？公司是福利院么？人家找你过去是要让你干活的，不是让你去学习的，人家之所以愿意培养你是因为你目前至少能干干活，至少你不是小白，至少你能写代码来，而不是连面向对象知识都搞不明白的小白，如果培养一下能帮他们创造更大的效益，每个老板心里都是有衡量天平的，哪个 sb 老板愿意找个小白来培养呢？如果一个公司连纯小白都愿意培养了，那这个行业的人才得是多么多么的稀缺呀。

另外再说找高手带，高手凭什么带你？高手脑子进水了？你是长的好看还是特别有钱？高手之所以是高手，是因为人家花了大量的时间不断的学习和提高，把时间浪费在你身上能给他带来什么呢？是能图你的财还是能图你的色？

所以奉劝那些依然抱这种想法的人，不要再做白日梦了，你是什么样的人，基本你的社交圈就是什么样的人，你所认识的所谓的牛人，之所以愿意接教你，只不过是你们俩人之间至少还有一个东西是可以等价交换的，就是钱，你花钱买人家的时间。

## 谬论之“培训机构都是骗人的”

利益相关，我必须反驳这个论点，市面上确实充斥着大量的很烂的培训机构，我发现一个特点，那些很烂的机构，多数情况下都有一个特点就是创始团队不是技术出身，他们对于技术本身没有并没什么情怀，对于课程体系的技术深入程度无感，他们更多注重的业绩，基本上是处于只要给钱就收的状况，甚至有些机构连初中都没读完的人也收，背景这么差的人怎么能在短期内培训出来呢？这样的机构很多，有这样的同行，我也表示无奈。但请不要一棒子打死，这对那些依然在这个行业里努力踏实耕耘的培训从业者是不公平的，这个行业里依然有一些好的机构，圈里人自己互相打听一下应该就知道，我就不做广告了。

## 其实本质的区别还是“时间成本和坚持”

无论参加培训或自学，你都要投入很多的精力和时间去做这个事情，很多人来我们这报班，说自己可能时间不多，经常加班，可能不能保证都能来上课，我会直接讲，如果你不拿出充分时间来做这个事情，那你肯定学不好，你要考虑好，你若学，就做好接下来一段时间内，把这个事情在你生活中的优先级提到很高，我们

的课程内容很多，如果只是吊儿郎当，那你学不好。

人都是懒惰的，学习是反人性且痛苦的事情，谁不想天天出去泡妹子、跟朋友耍，不到不得已，谁愿意坐下来安静的学习呢？坚持是一个很难的事情，如果干一个事情只需要花几天，我相信大多数人能都坚持，但当这个成功回报周期变的很长，能坚持的就越来越少了。如健身、学英语、学编程这些都应该算回报周期比较长的，健身教练、培训讲师的作用就是帮你把这个回报周期尽量缩短，并且在你想偷懒的时候督促你继续前进，我最近刚花 1 万多办了私人教练会员，发现塑形效果真的很明显，我之前断断续续自己练了 5 年，结果还是只有一块腹肌。跟着教练不到 2 个月身体就开始显形了，这就是区别。

另外再说英语，我 07 年就花 3 万多报了算是中国最贵的成人口语华尔街去学英语，当时我刚毕业，一个月收入 2500，学费还要贷款，当时所有身边的人都觉得我疯了，花好几万去英语，简直不能理解，当时的钱还很值钱，宇宙中心五道口的房价才 1 万多，我 3 万多快够买个厕所了，其实报华尔街之前，我还在一个便宜的地方报了名准备学，还交了 500 报名费，那里学费 1 万多，没过几天去了华尔街发现学费 3 万多，当然环境等各方面更高大上很多，虽然我不懂，但我知道 1 万和 3 万的学费定价肯定是有原因的，三思后还是报了华尔街，因为我坚信一句话，一定要买你能承受范围内最贵的东西，买了后基本不会后悔，教育投资更是这样。后来我花了两三年在那学英语，泡了一大票华尔街英语的漂亮姑娘，解锁了各种新姿势，什么优衣库视频什么的我只是笑而不语，甚至现在的女朋友也是在那认识的，然后还去了好几家高大上的外企，有机会跟我比

nb 很多人一起工作，长见识，学知识，这些东西都是我当年报名学习时没考虑到，这些帮我拓宽了眼界，让我认识到更大世界的机会是不能用几万块培训费来衡量的，那些当年劝我不要花钱去报培训的朋友们，到现在还蜷缩在一些不知名的公司里挣一两万的死工资混日子，我却已看不上年薪百万的职位，身边已经有几个朋友年收入 500w+ 了，这才是我要看齐的对象。

## 你迷茫只是因为你太 low

很多人跟我讲他很迷茫，不知道接下来该怎么走，感觉都能看到接下来 5 年之后自己是什么样子，问这些问题的有的是刚进入这个行业的新手，有的是工作了五六年的技术老手，其实这两类人都有一个共同特点，就是他现有的知识和能力储备使他的职业生涯发展在接下来几年没有了想象空间。对于这样的同学，我想让你认真的问自己，是不是该充电了？是不是该读个 MBA 了？是不是该报个编程班了？是不是该学一门新语言了？是不是该交个新女朋友了？不要把自己的认识限制在你现有的知识储备怪圈里。其实当你在海上航行迷失方向，首先你不应该原地打转耗青春，也不应加足马力朝一个方向硬冲，因为走错了方向，可能就离陆地越来越远，你此时应该做的是养精蓄锐，学习一切能让你辨识方向的新知识，解锁海上航行的更多高大上新姿势，然后你就会发现，其实不是你迷了方向，只不过是之前的望远镜望的不够远而已，因为陆地一直在那里。

这篇文章看完后，可能又会引来一些揶揄的评论，but I don't give a damn about how they feel，因为有些人的认识就是那个样子，我不屑与这些人讨论。

## 7. 普通运维就是秋后的蚂蚱

我虽是一个 IT 屌丝，但特别喜欢关注整个大行业的动态，干 IT 运维相关工作到现在也快 8 年了，企业对运维人员的专业能力要求确实提高了不少，现在再去面个运维工程师的职位都要求会个开发语言啥的，这在 2007 是不可想象的，那个时候，运维和开发是泾渭分明，两个完全不同的工种，大多数运维人员只需要会配置常用服务、简单的调优、再会写个 SHELL 脚本处理一些日常事务(比如监控应用、数据备份等)的话就能算是一个合格的运维工程师了(虽然现在依然有很多运维人员也还只是这种水平)。以上这些要求其实也算蛮简单的，那些年市场上充斥青鸟毕业生，稍微带一下就能干上面讲的这些事情。所以在那时，干运维还是蛮好混的，只不过工资确实是比开发人员要低一个档次的，当然行业地位也是，好多企业运维部门算是最边缘的一个技术部门了，在人们眼中，与运维最相近的字眼就是网管，我当时特别不喜欢别人说我是网管，总感觉是在网吧修机器的。

擦，一转眼快 8 年过去了，我习惯每年最后一天写篇文章，之前都是在北京过元旦，一个人孤独的写下一年来的酸甜，今天这篇文章却写在纽约，身边还陪着美好的姑娘，估计好多年前自己英语还很烂



时也想不到今天就会站在美帝的土地上，就像刚做运维时，想不到自己日后会走向开发这条路。不过我想我这也算是顺应了行业的大趋势吧，因为风来了，相信已入行的运维伙伴们都已看到，普通的运维人员已是秋后的蚂蚱，蹦跶不了几天了，他们已经走在了被淘汰的路上，IT 自动化必将砸掉大多数不思进取的运维人员的饭碗，寿终正寝只是时间问题，

一些人可能觉得我是危言耸听，说我夸大现实，运维行业正需要大把的人手，对此，我只能说这是蚂蚱的垂死挣扎。我给各位讲个真实的故事，希望能给这些人以些许触动。

我来自山东的一个小农村，小时候，整个村会经常停电，然后就会看到过一会就有个电工爬到电线杆上去修了，有时候修不好的话，还要请县城里供电局的更专业的电工来修，那时候停个电是家常便饭，几乎每个月都会发生数次，长大后我才想明白为啥经常停电，因为那时候电闸都是旧的，不好用，一家一不小心短路了，整个村都会跟着断电。那时每个村都会配一个电工，村里固定每月会发给他三四百块，这基本上算是个兼职，毕竟不会天天断电，所以这个职业在那时看来还是不错的，并且技术要求也不高，好多人还是争着想干这个活的，但饭碗毕竟只有一个，所以其它人只能眼馋了，每个村的电工自己当然也很 HAPPY，认为自己吃上了铁饭碗，可是好景不长，好像是我上初中的时候，村里统一换了新的电闸，好多线路也都换成了新的，果真停电率迅速由每月数次降低到了好几个月一次，我们家的小卖部终于不用担心冰柜里的冰棍会经常在没电时化掉了。我们村的电工突然闲了下来，好长时间都不再见他爬电线杆了。没过多久，村支部就觉得每个月花几百块养个闲差不值，但又不能没有这个职位，村领导们于是想了个好办法，就是好几个相邻的村共用一个电工，于是再停电时，我就看到了外村不认识的人在爬电线杆了，因为我们村的电工不知何故不让他干了，我想可能是技术不精或没送礼吧。每个月损失几百块，他当然不爽，在他下岗后，一个月内村里连续断了好几次电，查出来原来是有人故意把两条电线搭在一起导致短路，村领导很生气，估计都知道是谁干的，但是不能明说，只能在村广播里说，已经发现有人故意破坏电线，并且已经掌握了一些证据，如果此事再发生，就让公安局来抓人了，从那以后，用电才正常起来。慢慢的，一年都停不了一两次电了，后来，几个村都不再养电工了，而是变成了供电所给每个镇配上配个兼职电工，一个电工管十几个村。再后来我就到外地上学了，不予关注这个事情了。

回到我们自己，请各位想想，我们自己会不会也是那个电工呢？呵呵，别想了，运维人员其实就是那个电工，多年前，每个公司都养一些运维人员，维护自己公司小机房里的几十台服务器，慢慢到现在，小公司都不设置机房了，甚至都不买服务器了，业务直接放在云上，也基本不需要考虑断电问题，因为云后面有一大群专业的运维人员帮这些小公司维护着他们的线上业务，所以好多小公司连运维人员都省了，好吧，这些原本只能在小公司工作的运维工程师就逐渐涌向大公司，大公司都是几百上千、甚至数万台的服务器，一般都会有数十个甚至上百个的运维人员，以维护和运营他们庞杂和繁多的各项 IT 应用，但大公司真的需要这么运维人员吗？在这方面国外做的是更先进一些，之前看到文章说 facebook 一个运维人员要管理 2 万台服务器，我不确定数据是否真实，但我确信如果自动化做的好，并且管理得当的话，这个数字是肯定可以实现的，也将最终会实现。想一下，等这一天到来时，每个公司那一大堆闲下来的运维人员的出路在哪里呢？这个问题到底怎么破？我觉得其实很简单，就是去做顺应行业大趋势的事情，那这个行业的大趋势是什么呢？其实就是以下两点：

1. 资源集中化：IT 运维终将变成整个行业的基础服务，就像水和电一样，你要用电不代表你需要自己买一台发电机，早晚有一天，绝大部分的 IT 业务会运行在数十家巨型的 IT 基础服务提供商身上（他们的前身就是今天在跑马圈地的运维服务提供商），这样将会大大降低整个行业的 IT 运营成本。
2. 高度自动化：我坚信，只要时机合适，绝大部分现在运维人员干的时候都是可以完全自动化起来的，因为人不如机器可靠，更不如机器高效。现在所谓的自动化不过都是半自动，是在人的监督和调动下的半自动，就像自动挡汽车还是要人来开一样，但无论你信不信，不需要人开的汽车终将普遍起来。

总之，建议所有做运维的小伙伴们，不管你是新手还是老手，。

## 8. 哥们别逗了，写个脚本真不叫运维自动化

好久没写文章了，最近要来刷下存在感，近两年，运维自动化被炒的火的不行，行业趋势不可挡，现在企业招运维工程师都要求会一门开发语言。我们公司也不例外，由于刚上市，一下子有钱了，开始招兵买马瞎折腾，因此最近我也面试了不下十来个求职者，本成想可以很容易招到几个不错的小伙，结果却令我很失望，今天贴几个面试者例子上来，跟大家吐槽下：

面试 A 君：

应聘职位：高级系统工程师，工资要 18K

此君简历写的不错，在 360 干过几年，简历上写的是维护公司的 360 网盘平台，管理着 2000 多台机器，写了很多自动化工具。然后我就让他跟我聊聊他做的自动化工具，哥们娓娓道来跟我讲他写的那

些脚本（自动部署、批量命令、日志分析），从他的表情中感觉他好像觉得他做过的这些东西很牛 B（其实都是一堆

SHELL+PYTHON 拼凑起来的粗糙工具，需求稍一改变就不能满足，脚本的扩展性极差），他说他现在的工作基本上 80%都能通过脚本完成，说完后直视我，貌似是等待着我的认可，出于尊重，我只好说那确实不错。接下来我就拿他写过的批量并发执行命令的脚本跟他深入聊了下，他说这个脚本是 Python 多线程并发的，1000 台机器执行一次批量命令 1 分钟就能全部完成，我于是让他给我讲讲多线程与多进程的区别，在什么时候用线程或进程更合适，结果哥们说了很多废话也没有讲明白。然后我又让他用 PYTHON 多线程给我写个简单的生产者消费者模型，哥们愣是不知道这是啥东西？那我又问，如果你不知道生产者消费者模型是什么，那你的并发异步处理是怎么做的？哥们语塞说没在这方面做过深入研究，我于是又问了几个其它方面的问题，比如他的 Nginx 日志是如何分析的？自动部署如何跟 git 结合的？监控报警接口是如何优化才降低误报率的？但回答的都不理想，然后，就没有然后了....。在这里我只能说，我不是想打击他，如果你只是写了几个脚本就声称自己就是自动化大拿了，未免有点牵强。所以他被 PASS 掉了，因为我觉得把他招过来，真的不会给我们公司的自动化水平提高多少！

面试 B 君：

应聘职位：运维自动化开发工程师，工资不能低于 16K

此君简历上说他擅长 PHP\Python 开发，在原公司做过运维自动化平台。我很感兴趣，让他讲一下他做的东西，主要是监控平台和 CMDB 资产数据库，让他着重讲了一下他的监控架构，他说他的项目主要是主动监控方式，就是监控服务器每隔一两分钟去轮巡一次所有的机器的 SNMP 接口，把各机器的基本系统性能信息抓回来，再通过 RRDTOOL 出图。他们有 500 多台机器散落在 3 个不同地区的机房，我问他你们这种做主动监控轮巡一次得多少时间？他说 1 分钟左右，我说如果轮巡过程中，如果有几台机器连不上怎么办？他说他的轮巡是并发的，连不上的不会影响其它机器的监控，我说但是你的线程池的线程个数是有限的，有几个线程因为机器连不上，那就会产生阻塞直至超时，但在超时之前这几个线程是不会再空闲出来的，因此肯定会导致整个轮巡时间的加长呀。哥们想了想说，确实存在这样的问题。然后我问他有没有考虑过用被动监控方式，就是让客户端自动汇报数据呢？他说当时他们也是这样想过，但是一想到要在所有的机器上装个客户端就觉得会增加复杂度，并且维护和管理不容易。我说采用被动模式确实会增加点复杂度，但会给你带来很多好处呀，监控客户端自动给你的服务器汇报数据会大大减少你主监控服务器的负载，并且可监控的东西要多的多呀，你还可以自定义插件，自动升级，并且还可以把监控客户端当做它用，比如自动化部署、任务下发等。可能是出于尊重，哥们假装同意了我的看法。

然后我又问他们的项目是几个人开发的，他说算他在内有 3 个人一块做，我说那你们之前是如何协作的，比如接口互相是如何调用和约定的？他说基本上是每个人写自己的接口，互相之间约定好如何调用。我问那你们有没有遵循什么标准？比如是统一用 httpapi 还是其它的？接口格式是统一用 json 还是用 XML 或其它？哥们说他们有的用 JSON、有的用 XML。我说你们没有用 restful 标准吗？哥们表示没听过。。。。OH，好吧！如果大家开发时都不遵循一定的标准和规范，我真心不知道他们的系统以后如何扩展，不知道这个哥们离职后谁还能看懂他的代码？不知道这种拿一堆随心所欲写出来的脚本来拼凑起来的所谓的系统能满足多少实际需求？

面试 C 君：

应聘职位：运维工程师工资要求 11K

哥们刚工作不到 2 年就要 11K，真比我当时工作了 2 年挣的多多了(即使去掉通货膨胀影响)，但如果技术好那也没有问题的。结果问了一堆基础问题都答不好，再要这些钱是否有点自不量力了？问他 LVS，结果只是配置过，让他讲几种负载调度原理也讲不明白，问他平时怎么管理大量机器，他说用 SaltStack 或自己批量写脚本，我问那你用脚本批量管理机器，是通过什么方式呢？他说是用 SSH 批量密钥登录，我说那你给我讲讲

RSA 密钥认证原理吧，他接下说的就是怎么通过 ssh-keygen 命令，怎么把公钥拷到客户端机器上等等怎么实现密钥认证的过程。我打断他说我想听的是原理，不是认证方法，为什么需要把 public\_key 拷到客户机上而不是 private\_key？为什么 COPY 过去后就登录不需要再输入密码了？结果哥们一点都说不出来，接下来问的一些其它问题也是这样，只知其然，不知其所以然，最后我说，你这种情况我们给不了 11K，如果低点你愿意考虑吗？哥们说不太会考虑，那。。。好吧，只能打发他走了。

其它的一些面试者情况也好不到哪里去，一路十多个面下来，让我很失望，本以为过了这么多年以后，整个行业的从业素质会提高很多。但结果却还是老样子，所以大家可以想想业内大家都在炒的运维自动化到底实际有多水？如果从业人员技术水平都这样，还谈个妹自动化呢？自动化真的是写写脚本，再拿个开源软件拼拼凑凑下就完了吗？在我看来这撑死了只能叫辅助运维，不叫自动化，自动化应该是真正的开始让机器帮你监测问题\发现问题\处理问题\解决问题\自我修复\自我维护\自带干粮，

各模块之间尽量低耦合\可扩展\可插拔。应该是真正能帮企业降低 IT 运营成本，使运营成本可视化\可测量\可对比，应该是真正能减轻运维人员的工作量而不是又制造一堆新的问题，应该是切合企业真正的实际需求做出来一些好用的工具和平台，而不是搞一些花里胡哨却最后扔在那里没人用的花架子（我自己在这方面就掉进过大坑，之前主导做的一个开源软件就是个失败案例）。

总之最后给我的感觉是，会开发的不真正的懂业务逻辑，开发出来的东西没人用，会运维的开发水平又太烂，写出来的代码烂到哭。想找个真正合格的运维自动化人才真是不容易。

我近期一共只面了 10 多个，确实不能代表全行业水平，有些真正技术牛人估计我也没有碰到，但是 10 多个样本里面一个合适的都找不到，我觉得这也能差不多从侧面说明一个行业的整体平均水平了，这些求职者水平不高，却又想要高工资，我能说这是眼高手低、好高骛远的表现么？打铁还需自身用，如果真想要高工资，请先踏实点把自身技术水平提高，如果真想做架构师，那请把架构技术和思想学好，如果真要做自动化运维，请先把至少一门开发语言学好、学透，不只是会写个脚本就完了，脚本只是脚本，那不是自动化，So，哥们别再逗了！

希望此文能给业内小伙伴选择职业路径时提供帮助！

**P.S:** 我也不是什么大牛，只不过做好几年运维，做了几年自动化开发，大多数运维人员走过的路我都走过，上面写的这几案例在我自己求职时也发生过，我也是一个从月薪 2500 一路走过来的屌丝而已，我写这篇文章只是为了表达对这个行业的低质、粗糙、落后的现状吐槽一下，我也是这个行业里的一员，所以所有的问题也有我的一份，SO 评论里骂我装逼的人我就不解释了，我这人就这风格，我写的其它文章比这篇可激进多了，So 如果某些看官不喜欢，那就 let it be 吧。

补充：其实我上面面的这几个人，我都是可以把他们留下的，只是没有办法给他们期待的工资而已，论运维水平，A 君要 18K，但我觉得他不值这些呀，我最多只能给到 15K 以下，因为如果我校招一个 211 学校计算机毕业的研究生也就给 11-13 就可以了，再培养一年的话，相信运维+开发水平都会比 A 君强很多！

B 君也是，一个连开发模式和标准都没搞明白的半调子程序员我能给他 10K+ 就不错了，我现在手下的一个刚本科毕业的学生写出来的东西都比他强，有人说我对求职者要求太高，没有人什么都精通，我当然不要他什么都精通了呀，但是在我看来是一些基础性的、通用性的知识，如果你都不了解的话，在我看来只能算做不合格，不能说我要要求高。

## 下面附上我认为一些很好的问题，我回答了下：

问：我的问题不涉及技术原理，就是要问下运维工程师的工资怎么定的，一个熟练使用工具或者说自己写的脚本的人，起码是对运维这块有过多年经验的人，为什么不值 18k，这类人的定位完全可以定位于一个可以干活的人，试问一个可以干活的人，怎么就不能值 18k。如果招人都执着于原理，能干出活的人少，执着于原理何用，我绝对不是说懂原理的人不会干活。that's all

回复花花工资 hgz:

我很想认真的回答下这个问题，但一会我又要去面试一个人，所以只能简要回答了，我做过 5 年运维，2 年开发到现在，你说的，在我看来，这个要 18K 的人就跟我最后 2 年做运维的时候情形是一样的，就是感觉自己什么都懂了，其实是什么都大概懂了，就像一个汽车修理工一样，干了好多年了，感觉什么车都会修了，车子出了问题很快就能修好。但是如果这个厂商把车造出来时就有技术缺陷的话，这个他是不会去关注的。因为他只会修车，再深入的比如说发动机的详细原理他是不知道并不关注的，他只能靠听声音来确定是不是发动机出了问题。我觉得现在好多工作了多年的运维人员就是这种情况，工作多年也只是变成了熟练的修车工人，而自动化则是尝试改变修车行业的事情，熟练的修车工人不去多关注的话，那我不看好他接下来的发展前景的，但一旦说到自动化，他如果只是自己闭门造车的话，可能造出来的又只是一个三蹦子，那怎么办呢？在我看来，如果做运维时间久了，迷茫了，就应该想想如何提高自动化水平，多跟开发人员多学学开发模式，不要只满足于只是把各种开源软件和脚本拼凑起来就完了，做运维的如果只止步于运维，不去多想想如果做点创造性的东西的话，那你也只能是一个运维。

问：是不是有些太矫情，运维最最重要的就是解决各种问题，原理，呵呵...



答：我看到很多类似的评论，觉得运维最重要的是能干活，能解决各种问题，那我们来讨论下什么叫能干活？什么叫能解决各种问题？把把各种服务配置好、业务运行稳定不出问题算不算能干活？当然算！但是

是花了多少的成本和代价让业务运行稳定了？这个问题你自己有没有衡量过？我买一堆 F5，一堆豪华服务器让厂商把东西都给我配置好，出了稳定找厂商帮解决，这算不算能干活？一个优秀的运维人员能把负责的整个业务以最小的代价和成本管理的井井有条并努力尝试整个公司的 IT 管理环境，一个普通的运维会认为他也做到了这一点，但是是否真的做到了呢？谁知道呢？毕竟他自己相信了。

再说能解决各种问题，什么叫能解决各种问题？我相信大多数的运维人员所谓的解决问题就是处理各种故障，但不怎么涉及优化，尤其是业务级的、代码级的优化，运维通常只关注 CPU、内存、负载是否过高，但业务非常慢怎么办？加 CDN？加负载？加完后如果还是慢怎么办？如果确实是因为代码写的不好导致慢怎么办？想过没有？当然你可以说这他妈哪是运维的问题，这是开发写的代码烂导致的问题，但是如果你能告诉开发是他的哪块代码的调用可能导致了业务变慢的话，是否能更凸显你的价值呢？当然如果只去解决一些常规问题也无可厚非呀，但那就不要老想着拿太高的工资了吧！

附赠在群里看网友讨论看到的一段不错的話：

---北京-菜菜左

建议搞运维的还是要吧计算机体系的基础知识打牢，万丈高楼平地起，基础很重要。看似没用只是因为你没学，所以你不知道学了是啥样。别人都说不重要只是因为大家都说不重要，也反映了运维行业的浮躁。不跟风不盲从，不迷信权威不吹水忽悠，走自己的路，不忘初心。这些话可能反对的人很多，就当我吹水吧。

问：文人相轻，做技术的也难免落入这个俗套，总是提一些别人回答不出的问题来显示自己的牛逼，忘了自己当初是个啥熊样，一个人的技术水平就凭你简单的问问就能做出答案，我服了你。

答：你自己想想，我问的那些问题是非常规问题么？这是基础问题，任何一个正经干过几年运维或运维开发都需要知道的问题！他答不上来不是我为难他，是他本身技术功底不扎实，一个人的技术水平不能通过面试面出来的话？那还要面试干个蛋用？

问：你怎么就只看到人家缺点，就没有看到人家优点。你做的那软件也那么渣渣，你都不好意思拿出来讲。做为面试者你只见不得人家比你优秀。面了这么多人不想想是否自己是否有问题！

答：这个看官提到我做的软件是个渣渣，这个我承认呀，我就是做过一个失败的开源软件呀，我在博文里也说了呀，所以我才吸取了教训以后不能只做表面功夫呀，但是我面试的这些人多跟我之前一样呀，拼拼凑凑搞个花架子出来了就叫做自动化，看着好看但解决不了什么实际问题。写个脚本就说自己提高了企业生产力。我为什么要招这样的人？你可能觉得没有一个人一开始就很牛 B，需要培养，我觉得没问题呀，可以培养呀，但既然是让我培养你，就请你不要想着即拿比你实际能力高很多的工资，又不想让我揭穿你了吧。

问：作为招聘者，你并没有从一个管理者的角度出发，认为自己只是一个干活的，不是管理者。而且你面试了十几个，竟是没发现一个合适的。好好想想是不是自己的问题？

答：我跟你保证，我没问题！面试十几个都没有合适的就代表我有问题了？为什么不能是从业人员普遍水平不高呢？你这简直就是千夫所指，无疾而终的调调！

问：其实楼主写这篇文章说白了还是装 13 用的，自己都搞不清楚什么是自动化，真正的自动化是什么含义，什么样的环境适合什么样的技术，张口就自动化，上万台机器，显得很牛 B 似的，说白了，哄哄那些刚毕业的小弟还能凑乎，懂行的人一看这就是在给自己吹吹牛 B 而已。技术是在环境和场景下出来的，一个公司什么样的规模基本决定了什么样的技术实力，不是每天喊喊口号你就能涨起来的，之前也看过楼主的开源软件，号称自动化工具，说白了也就是个脚本集成工具而已，再加点页面展示，这有何难，你的就叫自动化了？别人的就不是了？只能说太你吗自恋了

答：你这种评论我平时都懒的回，今天心情大好，回一下！

问：有要求可以，不过你认为要求一个运维人员，即熟练安装配置管理开源，集群管理工具，又熟练说出各种基础原理，还要熟练编写监控脚本，却只值 18K？开玩笑~现在一个熟练的快递员都是 10K 的时代。一个工作 3 年的金融工作者都是 20K 的时代。

你的要求不算高，但起码工资给高些，会有人来的。

答：一个工作 3 年的金融从业者 20K，这个确实不少，但哥们，我自己曾在牛 B 的金融公司干过，并且我可以告诉你，他们很多的时候拿的比这个多多了，但是人家都是名校毕业，都是高知呀，做运维的如果连开发的东西都不会一些，只会各种照文档配置配置还敢要 18K？在我看来这就是一个熟练工，一你没有名校学历，而你做的事情也没人家有技术含量，还敢要 18K，脑子是不是烧坏了？我发现现在有很多的做运维的，看着整个行业工资不断上涨，也跟着不自量力起来，我听过 N 多什么一跳槽就翻倍，就 20 多 K 了，就年薪 40 万了，自己也跟着开始漫天要价。你要价可以呀，但你得真有两把刷子，你要了 18K，就请对的住这 18K 的工资。

## 9. 为什么很多 IT 公司不喜欢进过培训机构的人呢

这几天在知乎看到一个问题“为什么很多 IT 公司不喜欢进过培训机构的人呢？”

身为老男孩的教学负责人，这样的问题必然会引起我的关注，

花时间看了各路人士的回答，

我去了个去，尽是对培训机构排山倒海的谩骂声，

负面评价不胜枚举，

抛开幸存者偏差，这些回答也能反应一些问题：大部分培训班都差强人意，靠谱的比例很低。我回答的冲动来自于看到很多学员都是省吃俭用攒点钱想改换跑道，谁能想便宜了这些不靠谱的培训班，一万块钱吃煎饼果子都管饱好几年了。

1 天前

三期学员,来作证,这个都是实情,建议想报名的同学们注意了,随便网上找的公开课的内容都不这个家强,被坑了9000块含泪来提醒你们,心都在滴血啊

1

10 小时前

哥们讲的太好了，我是四期的学员，真是感同身受，辛辛苦苦省吃俭用攒了一万块钱，本来期望着能物有所值学到真正有用的东西以后找到大数据工作，没想到这么坑。就是咽不下这口气，可以说这跟骗子没啥两样。希望我们这些学员团结起来，一起揭露，的真实面目。





林二  
多学多做

106 人赞同了该回答

不要培训，不要培训，不要培训，重要的事情说三遍！

我跟你一样是二本大学，去年刚毕业。专业是电子商务。临近大四的时候很心慌，不知道自己将来该做什么，能做什么。本身对摄影有爱好，学了ps，在搜教程的时候知道了网页设计这个工作，好奇心驱使继续了解。在百度搜相关资料，有幸收到了知乎关于web开发的相关问题，在进行了相对当时对这个岗位的认识较为全面的了解之后，决定自己要从事程序开发这个职业方向。接下来就是



李马刀  
by sense or by sensibility

7 人赞同了该回答

因为培训机构大部分是糊弄学员和企业，而学员呢，本身素质就良莠不齐。

学会了一种语言，不等于会写程序——因为有些人本身的逻辑思维就有缺陷，并不适合搞这行，简单的问题，却写得一团乱麻；会写程序，不等于就会搞开发——因为非科班出身，基础不扎实，工作中用到的基础东西，往往有欠缺。

看完久久不能释怀，心里堵的慌，

又委屈又气愤又无奈，

委屈的是自己辛苦为之奋斗的事业在别人眼里成了骗人的行当，

气愤的是太多无良同行们为了短期利益把行业做臭了，

无奈的是连少量的认真做事情的人也要跟着背负骂名。

冷静下来后，我想写一篇回答为培训机构正名，但想来想去，

觉得无论我如何辩解还是有人在认真做教育等 blablabla 类的言辞，

在那些被培训机构出来的学生通过包装假简历坑了的公司面前，

在那些被无良培训机构通过招转(招聘名义转招生)骗了的学生面前，

我的辩解显的毫无意义，

我为培训机构的正名正好把我自己变成了那些无良机构的同谋，

这个行业就是被糟蹋啦，都他妈 2018 年了，IT 培训行业还到处存在着卑劣的坑蒙拐骗，

就应该被唾弃，操。

讲真，我并不是通过这个问题才得知培训机构名声不好的，

毕竟干这行不是一天两天了，行业中的各种作案手段自是门儿清，

早就知道某行业老大一直大量通过招转招生，

早就知道某知名大机构的每个毕业生出来都包装成 2 年工作经验，

早就知道某人尽皆知的品牌现在已沦落到连小学没毕业的人都敢招，

只不过之前觉得坚持 做好自己就行了，别家机构这样做下去，口碑越来越差，肯定长久不了。

然而，好几年过去，那些机构依然屹立不倒，且业务有越做越好之势。

究其原因，还是广告打的响+销售人员忽悠能力强吧。

1 年给百度 1 个亿(真的，请自行去看某上市培训机构的财报)，就能忽悠到足够多的小白用户。

不管什么人，交钱就招，什么机构这么 NB 可以在四五个月把一个只有初中学历的人培养成一个合格的程序员呢？但不管，交钱就招，培训机构通过忽悠+过度承诺不设门槛疯狂招人，教学质量又控制不好，导致学生毕业后能力根本达不到企业要求，那怎么办，没关系，大不了毕业时包装简历嘛。

这样的故事上演的多了之后，就出现了连锁反应，

培训机构骗学生——》学生毕业后骗公司——》公司被骗的多了，开始直接过滤掉培训机构毕业的学员——》导致更多学员找不到工作，只能认倒霉，到网上去骂培训机构——》负面消息过多，导致机构招生成本上涨——》只能投更多广告招生，增加成本——》羊毛出在羊身上，导致学费上涨，且教研成本压缩——》导致招不到优秀教师——》导致教学质量下降——》导致毕业生质量更差——》更多的企业不想要培训机构出来的人——》死局。

没错，这最后会变成一个死局，并且会形成劣币驱逐良币效果。

做为外行的你知道么？现在 IT 培训行业平均获客成本已经到了 3000-6000 了！！！！

接近平均学费(15000-20000)的 3 分之一啦。注意这只是获客成本，还不包括招生人员的提成等。

销售费用这么高真的只是因为竞争激烈么？

扯蛋，明明就是因为整个行业都在被那些垃圾无良同行们承担着巨大的信任成本。

继续这样下去，获客成本涨到 1 万块是迟早的事，相当于 50%的费用全给了百度啊，呵呵。

试问，如果这样，即使想认真做教育的机构，哪还有更多的钱做教学研究？哪还有更多的钱招优秀讲师？

哪还有更多钱提高教学的硬软件环境？等着国家拨款么？呵呵，不可能的，这辈子都不可能的。

想到这个灰暗的前景，让我感到深深的焦虑。

其实还有一件事，更让我如鲠在喉，

前些天，因为公司要做新的宣传册，你懂的嘛，毕业生就是我们的产品，

希望找一些老学员给个照片，再写点感言，这样显的多真实+有吸引力。

很自然找到一些平时跟我联系比较多的老学员，结果呢，

有几个学生很为难，不愿意给，说怕泄漏隐私，我说，放心，不会写真名的，只写 xx 同学，还是不愿意，

最后才说出来，说怕让别人知道自己是培训出来的。



上面 2 个同学，一个在小米，一个在汽车之家。

我突然间好难过，真的。

难过的不是他们连这点小忙不帮，我难过的是，我们如此含辛茹苦培养出来的好学生，

出去后，竟然不敢跟别人承认自己是培训过的。

我难过的是，我们一起度过的那段艰苦、充实、激情的岁月竟只能埋在他们心里不能对别人讲。

我难过的是，那些努力学习、提升自己，只不过借助培训班让自己成长的更快些的初衷却造就了别人眼中的低人一等。

这是个多么操蛋又不得不接受的现实呀！唉！

这些事让我觉得瞬间失去了干这份职业的荣誉感，甚至开始质疑我当初从企业辞职出来做讲师是不是正确的选择？

本来是师者，所以传道受业解惑也，

本来是一直秉承“教育不是灌输，而是点燃火焰”，

本来是一直坚信“优秀的老师教授知识只是基本，还要对学生的人生观价值观产生积极正向的影响”

本来是一直跟带的新老师讲，“真正 NB 的老师，一定是活成了学生心中想活成的样子。”

我们的讲师天天如此拼命工作，到头来却发现，所从事的行当竟变成了过街老鼠，人人喊打。



我们的学员如此努力的学习，最后出去找工作竟然要靠隐瞒培训经历+包装简历方能得到面试机会。

最后被逼的，我们也要教给学生如何包装简历，因为不这样，可能连面试机会都拿不到，或者只能接受 5 千的起薪，即使你的技术水平比那些科班毕业的学的好还要好。不好意思，包装 2 年就可以十几 K 一个月，不包装就工资 5 千，只因你是培训出来的，你就处在鄙视链的底层。

清晰记得上次在毕业班，教他们包装简历，边讲边觉得自己的形象在崩塌，你老男孩的价值观呢？不是说 Do the right thing 么？这是 right thing 么？为人师表却教别人如何耍诈，呵呵，真是可笑呀。

我当然会自我安慰啦，比如对自己说，你可以坚持让学生不包装，你心中是正义了，但代价是学生需要接受比其它机构毕业生低一倍的工资，凭什么呢？凭什么你自己的正义坚持却让学员买单？所以，别自个在那端着了，至少少包装点吧，包装个半年，这样子也不太过分。

呵呵，于是乎，我们也变成了同谋。

“当正义长久得不到伸张，当受害者得不到应有的同情，荒谬也就变成真理。”

直到去年 5 月，我去硅谷参加 Google 的 IO 大会，认识一个 Uber 的工程师，聊起我的职业，他说他老婆就是在美国一家培训机构培训了几个月，现在 TripAdvisor（相当于国内的大众点评）做开发，并且他自己在 Uber 的小 team 里也有 2 个是培训出来的，发现技术能力还挺强的，我问，你知道他们是培训出来的？他说对呀，面试时直接说了，我问那你们不会区别对待么？他说不会呀，只要具备实际能力，为什么要在意人家是培训出来的呢，本职工作干到半路想转行做 IT 的人多了去了，这反而是勇敢尝试新事物的表现呀。

这个事让我思考，明明是帮人改变、提升的好事，在国内怎么就变成这个样子了呢？

那这个游戏还玩着有什么劲呢？

难过、愤慨、无奈、想骂人。  
妈的，不能再这样下去了。  
要么改变，要么滚蛋，  
如果没人要出来做点什么，那我来做。  
经过很多天的深思及和小伙伴们的讨论，我们决定在目前老男孩教育的基础上，推出一个新的品牌。  
这个品牌要通过行动为培训行业正名，  
证明给社会这个行业还是有希望，  
证明给无良同行其实不用靠坑蒙拐骗也能招到学生，  
证明给学员你们的努力不应该受人质疑，  
证明给企业培训出来的学生是可以用的，  
证明给自己，我们依然在做一件很崇高伟大的事情。  
这个新品牌，我们给他起了名字，叫“骑士计划”。

“骑士”——骑士(Knight、Cavalier)是欧洲中世纪时受过正式的军事训练的骑兵，后来演变成为一种荣誉称号用于表示一个社会阶层。

在骑士文学中，骑士往往是勇敢、忠诚的象征，  
每一位骑士都以骑士精神作为守则，是英雄的化身。

我们会对入学门槛进行严格的设置，  
我们会派出老男孩最牛逼的讲师任教，  
我们会对教学过程严格的把控，  
总之，一切按我们现在能做到的最高标准来执行。  
我们希望，每个从“骑士计划”毕业的学员，都能在简历上豪不畏惧的写下“骑士计划”，  
我们希望，每个从“骑士计划”毕业的学员，除了技术之外，都能收获面对未来生活的自信、坚毅和勇敢。

当然这件事并不容易，这是一场革命，  
一场通过自身进化而推动整体行业变革的革命。  
一场把劣币驱逐出这个行业的革命，  
毛主席说：  
“革命不是请客吃饭，不是做文章，不是绘画绣花，不能那样雅致，  
那样从容不迫，文质彬彬，那样温良恭俭让。  
革命是暴动，是一个阶级推翻一个阶级的暴烈的行动。”  
我意识到这将会是一场漫长的战役，  
即使胜利了也许都不会有任何的勋章，  
也不会有任何人为你骄傲。  
所以我能做的就是坚持作战，  
直到环顾四周，我不再是孤身一人时。  
再见面时，我愿我能以骑士之名，赋予你勇气。