

适用于 YF408 创新团队的 Python 学习路线

一、学习基本路线

408 的各个项目中，无非都具有软硬结合的特点，这其中就离不开前端、后端等概念，具体可参考这篇文章：[到底什么是前端、后端、后台啊？ - 知乎 \(zhihu.com\)](https://www.zhihu.com/question/20462016/answer/1000000000)。

而之所以 python 会成为 408 实验室的首选语言，莫过于他的几大特点：

- 1) 语法简单，学习成本低，有了 C 的基础和逻辑能力，你可以很快地上手 python。
- 2) 由于第一点而使 python 的开发效率极高。
- 3) 近几年 python 是热度数一数二的编程语言，学习 python 可以帮助你提高自己的竞争力。

另外，由于 python 的第三方库/packages)众多，方向多而杂，因此本文仅仅介绍适合之后各位接触项目后可能会用到的方向，其余方向学弟学妹们可以自行学习。

1.1 基础语法—必修课

要求：学弟学妹需要掌握 python 的基础语法，从最基本的 python 安装和环境配置，到基本程序的编写。

请独立的完成下文中给出的任务，作为自己是否掌握这部分的知识的标准
包括以下检查点：

1) python 安装：

- 1 Python 语言的特点
- 2 Python 开发环境的建立
- 3 Python 程序的编写与运行机制
- 4 标识符与关键字
- 5 数据类型及变量
- 6 运算符和表达式

学会安装编译器，配置编程环境。

了解解释型语言的运行机制。

掌握类比并区分 python 语言与 C 语言的区别、特点。

掌握 Python 语言的基础语法。

任务—练习 1：

- 1: 在两个不同的环境下分别安装不同的包
- 2: 编写一个简单的 a+b 程序

2) python 数据结构：

- 1 条件控制与循环语句
- 2 字符串的基本操作
- 3 列表和元组的基本操作
- 4 字典和集合的基本操作
- 5 函数的定义与使用
- 6 数据结构应用

学习使用 `if-else` 控制语句。

学习使用 `for`、`while` 循环语句、`break`、`continue` 等程序流程控制语句。

掌握字符串、列表、元组、字典和集合的定义与基本操作。

掌握函数的声明与调用，及其基本操作。

了解数据结构的部分应用。

任务一练习 2:

自定义函数，设置固定次数的登陆

1.由键盘输入密码；

2.若密码正确则屏幕显示: “Login success!”

3.若密码错误则显示: “Wrong password or invalid input”，并显示剩余输入机会次数;

4.共 3 次机会，用完则屏幕显示: “Your account has been suspended”，并退出程序。

3) Python 面向对象编程:

1 对象和类

2 创建类

3 属性与方法

4 继承

5 面向对象案例

掌握面向对象程序设计基本思想。

掌握类的设计、构造方法、类的成员变量和成员方法的定义。

理解实例对象的创建与使用。

掌握 `self`、`super` 关键字的使用。

学会使用覆盖方法与 `super` 创建子类。

任务一练习 3:

学生有下面几个属性：姓名、年龄，考试分数包括：语文、数学、英语。

比如定义 2 个同学：

姓名：李明，年龄 25，考试分数：语文 75，数学 80，英语 90

姓名：张强，年龄 24，考试分数：语文 89，数学 81，英语 80

1) 给学生添加一门 `Python` 课程成绩，李明 60 分，张强 80 分

2) 把张强的数学成绩由 81 改成 89 分

3) 删除李明的年龄数据

4) 对张强同学的课程分数按照从低到高顺序输出

5) 外部删除学生所在的城市属性，不存在返回字符串 `beijing`

4) 模块与文件操作

1 模块与包

2 一些深受欢迎的模块

3 文件的基本方法

4 迭代文件内容

理解模块与包的机制。

了解一些热门模块。
简单理解什么是文件以及简单的文件操作办法。

任务一练习 4:

自定义函数将敏感词过滤后的文本写入指定位置的 txt 文档中

- 1.自定义函数 `texcreat(name,text)`,将 `text` 写入 “`name.txt`” 文件中
- 2.自定义函数 `Repalce`,将某些敏感词(比如 “暴力”)替换成 “和平”
- 3.自定义函数将敏感词过滤后的文本写入指定 txt 文档中。

掌握到这里,学弟学妹们就可以去尝试着写一写简单的程序,也可以尝试着将自己 C 语言课上的复杂程序用 python 编写,你也许会对 python 的编写效率叹为观止。

1.2 (可以跑几万个 for 循环试试电脑风扇坏了没) 进阶语法—选修课

可以自己选择学习正则表达式等知识。

1.3 学习途径&资源共享

1.3.1 资源

Pycharm 破解版安装包以及其他学习软件已发送至 408 群文件。

1.3.2 途径

给出较为不错的学习 python 的途径,仅供参考,有更好的请补充。

- 1) [Python 快速入门 - 泰迪云课堂 - 专注于大数据培训及人工智能培训 - Powered By EduSoho \(tipdm.org\)](#) (主要参考)
- 2) [Python3 教程 | 菜鸟教程 \(runoob.com\)](#)
- 3) [Python 基础教程, Python 入门教程 \(非常详细\) \(biancheng.net\)](#)
- 4) [Python 教程 — Python 3.9.10 文档](#)

二、学习检查点: pyqt5

2.1 前言

在各位学弟学妹掌握 python 的基础语法知识后,剩下的路就要靠大家自己走了,各位在学习钻研项目的时候,一定要注重自己的信息检索能力,哪些网站比较好,哪些网站逐渐变成坑(比如某 dn),这些需要大家自己花时间去了解。师傅领进门,修行靠个人。408 的学长学姐们其实也挺忙的,毕竟要毕业了,出国留学、考研以及就业的都有,因此我们无法事无巨细的教给大家每一个知识点,也没办法准时的回复大家的每一个问题,另外,学长能力也有限呜呜呜,面试的时候也发现很多比我自己厉害的学弟学妹,不过这不影响什么,因为本来 408 的平台就是供大家一块学习的,如此才能长足进步。

2.2 Pyqt5

在此设立学习检查点,过了此检查点的同学可以进入自己的项目去做生产力。

大家请自主学习 pyqt5 相关知识,学习包括但不限于以下内容

- 1) 学会用 qt designer 绘制程序图形界面,包括基本组件的使用,布局的设置、对象属性设置等。
- 2) 学会用 python 处理 GUI 的后端(此后端非彼后端,也可能真的就是那个后端),在组件上绑定函数,如何不改变 GUI 代码而编写自己“后端”逻辑等。

3) 如何打包自己的程序。

最后，就请大家做一个属于自己的程序吧，能实现的功能不限，也不需要到和数据库交互或者通信的地步，但需要有以下要求：

1) 有个基本的美观，上中下哪是哪请尽量清楚。

2) 实现基本的交互功能，例如简单点的话是一个计算器？输入运算语句，点击按钮出来结果；高级点的话，也可以是一个翻译小软件，点击按钮翻译一段话（这个可能相当有难度，仅当个例子，因为还要调用翻译 API，请求数据什么的），

3) 程序没有强烈影响使用的 bug

4) 程序已经打包（可执行程序 `exe` 文件）
