在实训第一次课中，杨老师从语言结构、语言语法要素、编程思想、标准库、应用库和框架六个方面介绍了计算机语言，着重强调了要在实践运用中学习掌握一门语言，不必等待完全了解理论再实践，边学边用、活学活用的方法既能提高效率又能加深理解。在课后自行完成环境装配的过程中，我对VSCode、git、python这三个工具以及路径等相关知识有了进一步了解，通过有所选择的采纳网络信息成功安装，但对背后的部分原理依旧模糊不清，有待在后续学习中更全面完备地认识计算机系统。

第二次课正式开始python语言的学习，入门讲授的是python语言结构，新语言对工程目录的管理，包路径、模块等新概念的引入，import、\_init\_.py的使用，其中着重解释路径查找与包路径知识。为了更方便学生理解，老师穿插了命令行、环境变量等知识的补充介绍。其中命令行中的help指令功能强大，可以切实有效地提供指令内涵辅助理解调用，命令行提供的脚本编程以及其中的bat型、powershell型的区别与联系，在VSCode中选择command prompt shell。而环境变量（PATH）在计算机系统中有广泛深刻的作用，操作系统必须依赖环境变量工作，从使用set查找、定义环境变量，使用echo查看变量取值，再到环境变量的分类（临时、用户、全局），最后提供查找程序执行的方式。在查找中可以在命令行使用where命令。

之后开始讲解python语言结构与基本语法，文件的命名规则（扩展名py）、编码方式（默认utf-8）。字对应空格、换行、tab及其他特殊字符；句对应换行符、续行符，是计算机执行最小单位，可依照用途分为申请空间的数据语句和用于计算与流程控制以及调用的功能语句，其中数据语句放置自由而功能语句必须在函数中。段对应缩行操作，该操作与已学的C/C++语言有明显区别，需要非常细致。字句段的依次形成有赖语法规则，语法又可分为量、运算符、控制执行流程三要素。

短短两次课的学习让我对高度灵活的python语言有了较为全面的认识，计算机语言虽多，但普遍有较多相通之处，知识的迁移与对比为进一步了解不同语言的优劣势提供了途径，有助于在具体问题分析中选择较优方案。

第二周借助矩形菱形的打印介绍了计算机计算过程的流程控制以及面向过程与面向对象两种代码组织方式。

其中面向过程的编程使用整体命名法，段头使用“def 名字：”，段体使用相同的缩行做区分，缩行相似于大括号的作用，需要非常细心区分。面向对象的编程则将数据与过程分隔，”def \_init\_(self,参数):”作为构造器采用“self.数据=值”定义数据，之后是处理过程。调用时需要先定义对象初始化数据，之后再调用对象的相应过程处理数据。

之后老师就作业提交方式介绍了git工具的三种应用场景以及项目工程代码管理方法。第一个应用场景是第一次创建仓库初始化，提交master公共文件。第一次创建工作目录并在命令行中使用cd切换到工作路径后再使用git init初始化，第一次使用时需设置个人名字与邮件。第二个应用场景是建立分支并提交代码到分支，建立与删除服务器代理，拉取仓库中的代码，查看、创建、切换分支时需非常细心，最好每次操作都通过git branch查看当前分支。提交时使用三个固定连续步骤即可。第三个场景是后期分支代码合并，克隆服务器仓库并拉去分支代码，之后在主分支中“git master 分支名”实现合并到master，最后用同方法提交到仓库。