一、Python语言简介：

Python 是一个高层次的结合了解释性、编译性、互动性和面向对象的脚本语言。

Python 的设计具有很强的可读性，相比其他语言经常使用英文关键字，其他语言的一些标点符号，它具有比其他语言更有特色语法结构。

* **Python 是一种解释型语言：** 这意味着开发过程中没有了编译这个环节。类似于PHP和Perl语言。
* **Python 是交互式语言：** 这意味着，你可以在一个Python提示符，直接互动执行写程序。
* **Python 是面向对象语言:** 这意味着Python支持面向对象的风格或代码封装在对象的编程技术。
* **Python 是初学者的语言：**Python 对初级程序员而言，是一种简单易懂且实用的语言，它支持广泛的应用程序开发，从简单的文字处理到 WWW 浏览器再到游戏。

Python 特点

**1.易于学习：**Python有相对较少的关键字，结构简单，和一个明确定义的语法，学习起来更加简单。

**2.易于阅读：**Python代码定义的更清晰。

**3.易于维护：**Python的成功在于它的源代码是相当容易维护的。

**4.一个广泛的标准库：**Python的最大的优势之一是丰富的库，跨平台的，在UNIX，Windows和Macintosh兼容很好。

**5.互动模式：**互动模式的支持，您可以从终端输入执行代码并获得结果的语言，互动的测试和调试代码片断。

**6.可移植：**基于其开放源代码的特性，Python已经被移植（也就是使其工作）到许多平台。

**7.可扩展：**如果你需要一段运行很快的关键代码，或者是想要编写一些不愿开放的算法，你可以使用C或C++完成那部分程序，然后从你的Python程序中调用。

**8.数据库：**Python提供所有主要的商业数据库的接口。

**9.GUI编程：**Python支持GUI可以创建和移植到许多系统调用。

**10.可嵌入:** 你可以将Python嵌入到C/C++程序，让你的程序的用户获得"脚本化"的能力。

**Python的缺点：**

Python的唯一缺点是与C和C++相比执行的效率还不够快，因为Python没有将代码编译成底层的二进制代码，一些Python程序比像C语言这样的完全编译语言慢一些。

但上面提到的Python具有嵌入性的特征，对于大型程序，完全可以采用多语言混编策略，对于需要较快运行的模块，例如图像处理，则可以用C语言编程，对性能要求不是很高的地方则可以用Python编程，当需要他图像处理的时候Python程序把代码发送至Python解释器中内部已经编译的C代码，这样综合开发效率和性能综合起来是最高的。例如作为一个Python的数值计算扩展，NumPy将Python变为一个高效并简单的易用的数值计算编程工具。

二、个人项目（学生信息管理系统）注释分析：

def functionList(): 功能性注释： 定义功能菜单，用户在菜单中选择需要的功能

def sexInputDebug(sexInput): 功能性注释： 检查性别输入是否正确

def ageInputDebug(ageInput): 功能性注释：检查年龄输入是否正确

def IDInputDebug(IDInput): 功能性注释：检查学号输入是否正确

def nameListFunction(): 功能性注释：显示单个学生姓名信息

def findNameLocation(studentname): 功能性注释：通过名字找到学生位置

def listFunction(): 功能性注释：定义显示现有学生信息函数

def addFunction(): 功能性注释：定义增加学生函数

def delFunction(): 功能性注释： 定义删除学生的函数

def modifiFunction(): 功能性注释： 定义修改学生的函数

def searchFunction(): 功能性注释： 定义搜索学生的函数

nameSearch = input("-------------请输入要查找的名字：")

print("")

nameList = nameListFunction()

if nameSearch in nameList:

print("-----------------%s在学生管理系统中-------------------" % nameSearch)

print("")

j = findNameLocation(nameSearch)

print("姓名：%s--性别：%s--年龄：%s--学号：%s--备注：%s--" % (

studentList[j]["name"], studentList[j]["sex"], studentList[j]["age"], studentList[j]["studenID"],

studentList[j]["extra"]))

print("")

else:

print("----------------%s不在学生管理系统中-----------------" % nameSearch)

print("")

解释型注释：默认学生信息系统内容

解释型注释： 函数主体

print("-" \* 11 + "欢迎来到学生管理系统" + "-" \* 11)

print("")

print("")

functionList()

while True: 功能性注释：进入循环，根据序号选择操作

userInput = input("----------------请输入您要选择的功能序号：")

print("")

if userInput == "1": 功能性注释：显示现有学生和返回

listFunction()

functionList2()

continue

elif userInput == "2": 功能性注释：使用增加函数和返回

addFunction()

functionList2()

continue

elif userInput == "3": 功能性注释：使用删除函数和返回

delFunction()

functionList2()

continue

elif userInput == "4": 功能性注释：使用修改函数和返回

modifiFunction()

functionList2()

continue

elif userInput == "5": 功能性注释：使用搜索函数和返回

searchFunction()

functionList2()

continue

elif userInput == "6": 功能性注释：返回功能列表

functionList()

continue

elif userInput == "7": 功能性注释：退出

break

else:

print("----------输入有误，请重新输入！----------")

按照课题内容进行修改：

按照课上内容，此代码编码风格符合规范，项目说明文档、使用文档、函数方法的注释齐全，变量与方法的命名都具有较好的语义，且抽象程度合理。

但是稍微有些欠缺的是在函数内部对某些语句的功能进行描述的注释较少，可以在函数内部一些较难的语句后面加上一些功能性注释，使得程序的可读性更高。