

数据科学导论第一周作业

一、实验目的

根据实验要求，本次实验旨在实现**小型学生宿舍管理程序**，包含以下3个功能：

1. 支持按学号查询学生
2. 支持录入新的学生信息
3. 支持显示所有学生的信息

二、实验内容与方法

2.1 核心数据结构

在设计各种函数来解决这道问题之前，我们需要先定义，学生包含哪些信息

由题目可知，需要包含：

1. 学号
2. 姓名
3. 性别
4. 宿舍房间号

5. 联系电话

这五个信息

因为本实验是做一个简单的版本的宿舍管理系统，所以此处暂不引入数据库，但是，数据库的思想仍然可以引入进来

包括，根据题意，要按照学生的学号进行查询学生的信息，由此可知，学生的学号应当是唯一的，所以可以把它设置为主键，也就对应着字典里的key。

而对于字典的值，则为包含学生完整信息的子字典（如 `{"姓名": "张三", "性别": "男", ...}`）

2.2 平台界面

程序通过**循环菜单**持续运行，用户可反复执行“查询、录入、显示”操作，直到输入 `0` 或按 `Ctrl+C` 才退出，符合实际场景中“多次操作、持续管理”的需求。

```
=====
学生宿舍管理程序
=====
可以提供3个功能：
1. 按学号查询某一位学生的具体信息
2. 录入新的学生信息
3. 显示现有的所有学生的信息
0. 退出程序
=====
```

2.3 显示所有学生信息

(`display_all_students` 方法)

该功能不仅满足“展示所有学生信息”的基础要求，还额外做了**体验优化**：

包括 **格式对齐&宿舍统计**

格式对齐表现在：通过 `:<10` 等格式控制，让“学号、姓名、性别”等字段对齐显示，更易阅读；
宿舍统计表现在：额外统计并打印“各宿舍当前人数”，帮助管理员直观判断宿舍饱和情况，避免后续在安排宿舍的时候，导致宿舍超过4人的情况发生。

请输入功能编号 (1/2/3, 输入0退出): 3

=== 所有学生信息 ===

共有 6 名学生:

学号	姓名	性别	宿舍房间号	联系电话
2024001	张三	男	A101	13800138001
2024002	李四	女	B202	13800138002
2024003	王五	男	A101	13800138003
2024004	刘洋	女	B202	13800138004
2024005	罗琦	女	B202	13800138005
2024006	赵凯	男	A202	13800138006

各宿舍人数统计:

宿舍 A101: 2 人

宿舍 A202: 1 人

宿舍 B202: 3 人

如上图所示，系统清晰列出 6 名初始学生的详细信息，并统计出 A101 有 2 人、B202 有 3 人、A202 有1人的情况

2.4 录入新学生信息 (add_new_student 方法)

该功能重点体现“鲁棒性”(对非法输入的校验与处理)，包含多层校验逻辑：

1. **学号校验**：非空 + 唯一性（若学号已存在，拒绝录入并提示）；
2. **性别校验**：强制只能输入“男 / 女”，通过循环反复校验直到输入合法；

3. **宿舍超额校验**：非空 + 人数限制（调用 `count_room_students` 统计当前宿舍人数，若 ≥ 4 人则提示“换宿舍”）；
4. **手机号校验**：强制为“11 位数字”，不符合则重新输入。

运行示例（下面展示“合法录入”、“宿舍超额校验”与“手机号检验”场景）：

case 1: 合法录入

```
-----  
请输入功能编号（1/2/3，输入0退出）： 2
```

```
=== 录入新的学生信息 ===
```

```
请输入学号： 2024007
```

```
请输入姓名： 童话
```

```
请输入性别(男/女)： 女
```

```
请输入宿舍房间号： B202
```

```
宿舍 B202 当前有 3 人，可以入住。
```

```
请输入联系电话： 13800138007
```

```
学生 童话 的信息录入成功！
```

```
宿舍 B202 现在有 4 人。
```

我们可以看到，当我们往里面插入一位新的同学“童话”的时候，系统会自动告诉我们，当前是否能够插入这位同学，比如“宿舍B202当前有3人，可以入住”

case 2:宿舍超额检验

```
-----  
请输入功能编号 (1/2/3, 输入0退出): 2
```

```
=== 录入新的学生信息 ===
```

```
请输入学号: 2024008
```

```
请输入姓名: 莫言
```

```
请输入性别(男/女): 女
```

```
请输入宿舍房间号: B202
```

```
宿舍 B202 已满员 (4人), 请选择其他宿舍!
```

```
请输入宿舍房间号:
```

那如果我们在再插入一位同学“莫言”会怎么样呢？

如上图所示，我们会发现，这个时候已经不再支持再次插入了

“宿舍B202已满员（4人），请选择其他宿舍”

这个系统并不会直接崩溃，而是会提示，请输入下一个宿舍的房间号

此时，我们需要安排这位同学去另外一个宿舍

case 3:手机号检验

=== 录入新的学生信息 ===

请输入学号：2024008

请输入姓名：莫言

请输入性别(男/女)：女

请输入宿舍房间号：B202

宿舍 B202 已满员（4人），请选择其他宿舍！

请输入宿舍房间号：B101

宿舍 B101 当前有 0 人，可以入住。

请输入联系电话：138001380091

请输入有效的手机号码（11位数字）！

请输入联系电话：

现在，我们不妨把她安排入住宿舍B101，显示可以成功入住。

在输入手机号的过程中，如果我们输入12位手机号，此时会发生什么呢？

如上图所示，系统会检测出这样的错误并指出

“请输入有效的手机号码（11位数字）”

请输入功能编号（1/2/3，输入0退出）： 2

=== 录入新的学生信息 ===

请输入学号： 2024008

请输入姓名： 莫言

请输入性别(男/女)： 女

请输入宿舍房间号： B202

宿舍 B202 已满员（4人），请选择其他宿舍！

请输入宿舍房间号： B101

宿舍 B101 当前有 0 人，可以入住。

请输入联系电话： 138001380091

请输入有效的手机号码（11位数字）！

请输入联系电话： 13800138009

学生 莫言 的信息录入成功！

宿舍 B101 现在有 1 人。

请输入功能编号（1/2/3，输入0退出）：

如上图所示，如果输入的手机号不是11位的情况，此时就会报错，我们再次输入一个符合要求的手机号，比如11位的，可以发现，系统显示录入成功！

2.5 按学号查询学生信息

（ search_student_by_id 方法）

流程：用户输入学号 → 校验“学号非空” → 校验“学号是否存在于系统” → 存在则打印详细信息，否则提示“未找到”。

```
-----  
请输入功能编号（1/2/3，输入0退出）： 1  
  
=== 按学号查询学生信息 ===  
请输入学号： 2024008  
  
学生信息如下：  
-----  
学号： 2024008  
姓名： 莫言  
性别： 女  
宿舍房间号： B101  
联系电话： 13800138009  
-----  
  
-----  
请输入功能编号（1/2/3，输入0退出）：
```

如上图所示

我们不妨就以被我们用来检测各种鲁棒性的“莫言”同学为例，查询其学号，核对各个信息是否准确；如上图所示，此学号所匹配的学生信息准确。

三、总结

结合实验标准，本程序在“程序功能”与“文档分析”各维度的表现如下：

1. 代码简洁无冗余：

代码通过 类将功能模块化（查询、录入、显示分别为独立方法），且重复逻辑（如“宿舍人数统计”）被抽取为 `count_room_students` 方法，结构清晰、无冗余代码。

2. 系统功能完善和鲁棒：

三大核心功能完整实现，且每个功能都包含**多层校验逻辑**：

查询功能：学号“非空 + 存在性”双重校验；

录入功能：学号唯一性、性别合法性、宿舍人数限制、手机号格式等四重校验；

显示功能：不仅展示原始数据，还额外提供“宿舍人数统计”，功能更贴合实际管理需求。

3. 出错应对与回退：

程序通过 `try...except` 捕获两类异常：

(1)主动中断（`KeyboardInterrupt`）：提示“程序被用户中断，正在退出...”，优雅处理；

(2)未知错误（`Exception as e`）：打印具体错误原因（如 `print(f"程序运行出现错误：{e}")`），确保出错时不崩溃、可追溯。

4. 界面完善：

- (1) 菜单界面：启动时清晰打印“功能列表（1/2/3/0）”，引导用户操作；
- (2) 过程提示：执行每个功能前，打印模块标题（如 `=== 按学号查询学生信息 ===` ），增强指引；
- (3) 数据展示：学生信息“格式对齐”，宿舍人数“额外统计”，界面可读性与实用性强。