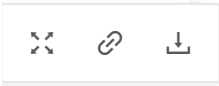
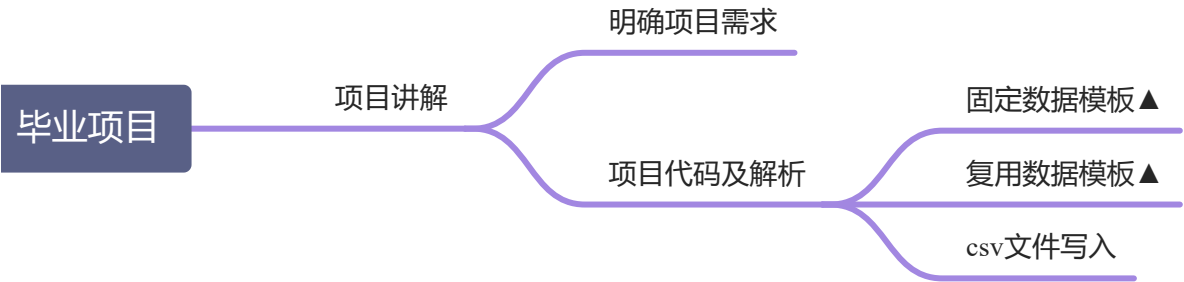


# 第19课 毕业项目

## 一、课程结构导图





注：▲为重点知识点。

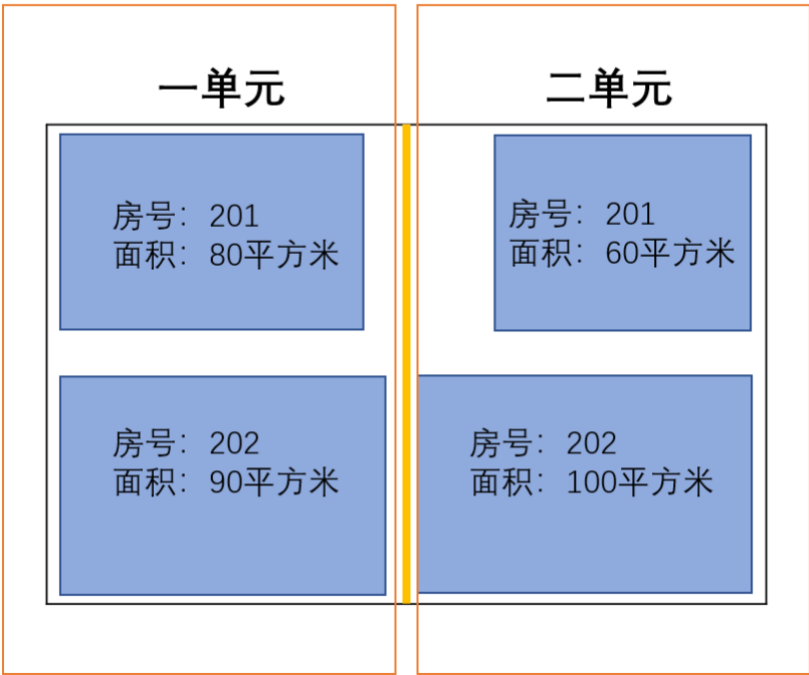
## 二、项目讲解

### 2.1 明确项目需求

- 需求：
- 1. 对住宅小区的名字，小区的地址、建筑年份、所有栋楼号和单元号里的每个楼层的每一户房间的朝向和面积的信息进行记录下来。
  - 2. 例如-富安花园（小区）1栋 1单元 201户；朝向：南北朝向；面积：90平方米。

图解：

某一栋楼的俯视图



解题思路：

- 1. 同一个单元中，同一层楼的不同房间号，户型（朝向和面积）可能会不同，例如图解中，一单元中的201房间和202房间的户型就不相同。
- 2. 同一个单元中，每层楼尾号相同的房间，户型都是相同的，例如图解中，假设一单元有4层楼，那么201房间和301房间，401房间的户型都是一样的。
- 3. 流程图中，主要展示的是一个单元中所有房间信息记录流程，当记录完一单元的起始楼层的所有房间信息后，一单元中的其它楼层，只需要在初始楼层所有房间信息的基础上，用循环代码对楼层数进行递增，得出其他楼层的房间信息。
- 4. 重复一单元记录的操作，可对二单元的所有房间信息进行记录。

2.2 项目代码及解析

2.2.1 固定数据模板

目的：记录起始楼层的所有房间信息。

关键代码：

```
1 start_floor = input('请输入起始楼层：')
2 start_floor_rooms = {}
3 floor_last_number = []
4
5 # 定义循环控制量
6 room_loop = True
7 while room_loop:
8     last_number = input('请输入起始楼层户室的尾号：(如01, 02) ')
9     floor_last_number.append(last_number)
10    #将尾号用append()添加列表里，如floor_last_number = ['01','02']
11
12    room_number = int(start_floor + last_number)
13    #房间号为room_number,由楼层start_floor和尾号last_number组成,如301
14
15    direction = int(input('请输入 %d 的朝向(南北朝向输入1, 东西朝向输入2): ' % room_number ))
16    area = int(input('请输入 %d 的面积, 单位 m² : ' % room_number))
17    # 输入朝向和面积
18
19    start_floor_rooms[room_number] = [direction,area]
20    # 房间号为键，朝向和面积组成的列表为值，添加到字典里，如start_floor_rooms = {301:[1,70]}
21
22    continued= input('是否需要输入下一个尾号？按 n 停止输入，按其他任意键继续：')
23    if continued == 'n':
24        room_loop = False
25    else:
26        room_loop = True
27 print(start_floor_rooms)
```

代码解析：

1. 变量名及其含义：

变量名	含义
start_floor	初始楼层（例如：2层）
start_floor_rooms	初始楼层房间信息（例如：{201: [2, 32]}）
floor_last_number	房间尾号（例如：02, 03）

room_number	房间号 (例如: 202, 203)
direction	朝向 (例如: 1-南北)
area	面积 (例如: 60平方米)

2. room\_loop变量是控制while循环的‘开关’，当room\_loop变量值为True时，while进入死循环，当room\_loop变量值为False时，while循环结束。
3. 第19行是字典的键值对添加语法：字典名[键名]=键值。

## 2.2.2 复用数据模板

**目的：**在起始楼层所有房间信息的基础上，用循环代码对楼层数进行递增，得出其它楼层的房间信息。

**关键代码：**

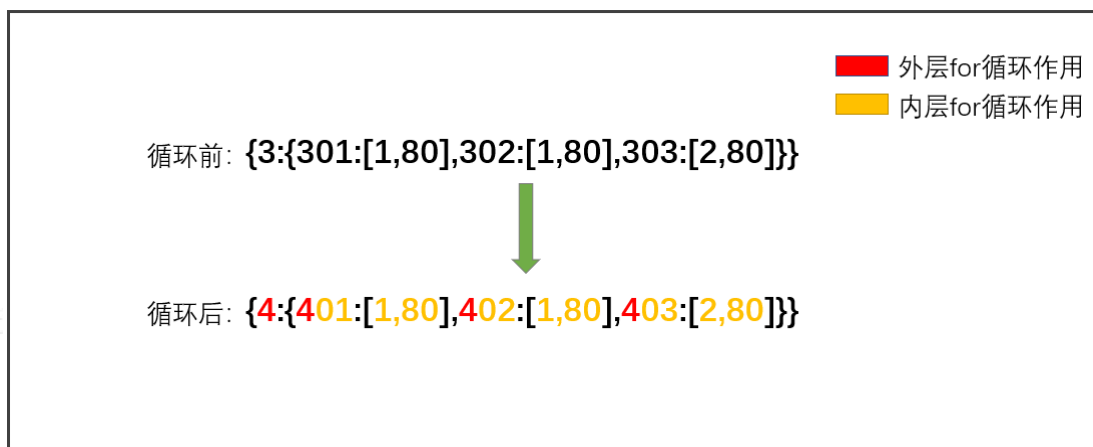
```

1 start_floor = '3'
2 end_floor = '7'
3 floor_last_number = ['01','02','03']
4 start_floor_rooms = {301:[1,80],302:[1,80],303:[2,90]}
5 #初始楼层的模版数据
6 unit_rooms = {}
7 #新建一个存放所有楼层的字典
8 unit_rooms[int(start_floor)] = start_floor_rooms
9
10 for floor in range(int(start_floor) + 1, int(end_floor) + 1):
11     floor_rooms = {}
12     for i in range(len(start_floor_rooms)):
13         number = str(floor) + floor_last_number[i]
14         info = start_floor_rooms[int(start_floor) + floor_last_number[i]]
15         # 依次取出字典start_floor_rooms键对应的值，即面积和朝向组成的列表
16
17         floor_rooms[int(number)] = info
18         #给字典floor_rooms添加键值对， floor_rooms = {401:[1,80]}
19     unit_rooms[floor] = floor_rooms
20 print(unit_rooms)

```

**代码解析：**

1. 若第11行代码在内层for循环中，第19行的代码中的unit\_room变量则添加的是每层楼最后一个房间号信息。
2. 代码主要由两个for循环的嵌套组成（下图所示），外层for循环，即for floor in range(int(start\_floor) + 1, int(end\_floor) + 1)，作用是对楼层数进行+1；内层for循环，即for i in range(len(start\_floor\_rooms))，作用是将楼层3所有房间的房间尾号，朝向，面积，复用到楼层4。



## 2.2.3 csv文件写入

**目的1：** 固定csv文档的表头内容。

**关键代码1：**

```
1 import csv
2 with open('assets.csv', 'a', newline='', encoding='utf-8') as csvfile:
3     writer = csv.writer(csvfile, dialect='excel')
4     header = ['小区名称', '地址', '建筑时间', '楼栋', '单元', '门牌', '朝向', '面积']
5     writer.writerow(header)
```

**注意点：**

1. 当windows系统打开assets.csv，若出现乱码时，则须在第2行代码中，encoding='utf-8'改为encoding='utf-8-sig'。
2. 当Mac系统打开assets.csv，出现乱码时，须在第2行代码中，在encoding='utf-8'改encoding='gbk'，因为Mac电脑打开excel是默认用GBK编码处理中文的，而Python默认是UTF-8编码。

3. `newline=""`的作用是使csv文档不出现换行现象。因为csv库在的写入操作时，会加入换行符的写入，当`newline=""`时，则不会将换行符写入到文档中。

**目的2：** 提取房间号，朝向和面积。

**关键代码2：**

```
1 unit_rooms = {3:{301:[1,80],302:[1,80],303:[2,90],304:[2,90]},
2               4:{401:[1,80],402:[1,80],403:[2,90],404:[2,90]},
3               5:{501:[1,80],502:[1,80],503:[2,90],504:[2,90]}
4               }
5
6 for sub_dict in unit_rooms.values():
7     for room,info in sub_dict.items():
8         dire = ['', '南北', '东西']
9         print('户室号: %d 朝向: %s 面积: %d' % (room,dire[info[0]],info[1]))
```

**代码解析：**

1. 第6行代码，`for sub_dict in unit_rooms.values()`，`value()`函数的作用是取出字典中的键值。例如`sub_dict={301:[1,80],302:[1,80],303:[2,90],304:[2,90]}`。
2. 第7行代码，`for room,info in sub_dict.items()`，`items()`函数的作用是取出字典中的键名和键值。`room`存放的是键名，`info`存放的是键值。例如`room=301`，`info=[1,80]`。
3. `dire[info[0]]`，是先通过索引值，取出`info`的值，再通过`info`的值，作为新的索引值，取出 `dire`列表的元素，将1-南北，2-东西关联起来。例如`info=[1,80]`，`info[0]=1`，`dire[1]=南北`。