

## 实验 列表与元组

学号	180809335	指导教师	黄俊莲
姓名	施嘉伟	日期	2020.9.28

### 实验 1 用列表完成刮刮乐游戏（20 分）

#### 1. 实验要求

假设现在有一张刮刮乐，该卡片上面共有 8 个刮奖区，每个刮奖区对应的兑奖信息分别为“谢谢惠顾”、“一等奖”、“二等奖”、“三等奖”中任意一个，大家只能刮开其中一个区域。

本实验要求编写程序，实现模拟刮刮乐刮奖的过程。

#### 2. 运行截图

```
本次抽奖有： ['谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '一等奖', '二等奖', '三等奖']

请输入刮去位置（1-8）： 1
二等奖
```

```
本次抽奖有： ['谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '一等奖', '二等奖', '三等奖']

请输入刮去位置（1-8）： 1
三等奖
```

```
本次抽奖有： ['谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '一等奖', '二等奖', '三等奖']

请输入刮去位置（1-8）： 8
谢谢惠顾
```

### 3. 学生截图

```
/usr/local/bin/python3.8 /Users/fushengyuanwuhui/PycharmProjects/experiment5/text1.py
本次抽奖有: ['谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '一等奖', '二等奖', '三等奖']
请刮去位置(1-8): 8
谢谢惠顾

Process finished with exit code 0
```

```
/usr/local/bin/python3.8 /Users/fushengyuanwuhui/PycharmProjects/experiment5/text1.py
本次抽奖有: ['谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '一等奖', '二等奖', '三等奖']
请刮去位置(1-8): 1
二等奖

Process finished with exit code 0
```

```
/usr/local/bin/python3.8 /Users/fushengyuanwuhui/PycharmProjects/experiment5/text1.py
本次抽奖有: ['谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '一等奖', '二等奖', '三等奖']
请刮去位置(1-8): 5
谢谢惠顾

Process finished with exit code 0
```

### 4. 学生源码

```
import random

a = ['谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '谢谢惠顾', '一等奖', '二等奖', '三等奖']

print(f"本次抽奖有: {a}")

input("请刮去位置(1-8): ")

b = random.randint(1, 8)-1

print(a.__getitem__(b))
```



## 实验 2 用列表完成商品价格排序（20 分）

### 1. 实验要求

假设现在某平台共有 10 件商品，每件商品对应的价格如下表所示，请按**价格升序**、**价格降序**、**商品名称升序**、**商品名称降序**后排序并输出。

表 1 商品信息表

序号	商品名称	价格
1	联想(Lenovo)小新 Pro13 高性能轻薄本	5299.00
2	联想(Lenovo)小新 Air14 性能版轻薄本	4999.00
3	联想(Lenovo)小新 15 2020 性能版轻薄本	5199.00
4	荣耀笔记本电脑 MagicBook	3499.00
5	华为笔记本电脑 MateBook	4899.00
6	宏碁(Acer)暗影骑士·擎	5899.00
7	戴尔 DELL 灵越 5000	4199.00
8	华硕(ASUS) VivoBook15s	4499.00
9	惠普（HP）战 66	4599.00
10	Apple MacBook Air 13.3	5499.00

### 2. 学生截图

```
F:\Python\Python37\python.exe F:/Python_file/experiment5/text2.py
[('宏碁(Acer)暗影骑士·擎', 5899), ('Apple MacBook Air 13.3', 5499), ('联想(Lenovo)小新Pro13高性能轻薄本', 5299), ('联想(Lenovo)小新15 2020性能版轻薄本', 5199), ('联想(Lenovo)小新Air14性能版轻薄本', 4999), ('华为笔记本电脑 MateBook', 4899), ('惠普（HP）战66', 4599), ('华硕(ASUS) VivoBook15s', 4499), ('戴尔DELL灵越5000', 4199), ('荣耀笔记本电脑MagicBook', 3499)]

Process finished with exit code 0
```

### 3. 学生源码

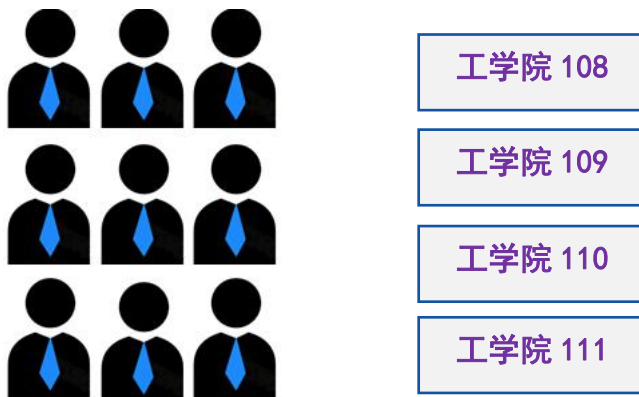
```
d = {"联想(Lenovo)小新 Pro13 高性能轻薄本": 5299,  
     "联想(Lenovo)小新 Air14 性能版轻薄本": 4999,  
     "联想(Lenovo)小新 15 2020 性能版轻薄本": 5199,  
     "荣耀笔记本电脑 MagicBook": 3499,  
     "华为笔记本电脑 MateBook": 4899,  
     "宏碁(Acer)暗影骑士·擎": 5899,  
     "戴尔 DELL 灵越 5000": 4199,  
     "华硕(ASUS) VivoBook15s": 4499,  
     "惠普(HP) 战 66": 4599,  
     "Apple MacBook Air 13.3": 5499}  
  
print(sorted(d.items(), key=lambda item: item[1],  
             reverse=True))
```

## 实验 3 使用元组完成办公室的随机分配（20 分）

### 1. 实验要求

要求编写程序，实现将 9 名教师随机分配到办公室的功能。

某学校新招聘了 9 名教师，已知该学校有 4 个空闲办公室且工位充足，现需要随机安排这 9 名教师的工位。



### 2.1 运行截图（9 个老师分配到 4 个办公室）

```
text4 text3_2
F:\Python\Python37\python.exe F:/Python_file/experiment5/text3_2.py
111 109 110
108 111 108
110 110 110
Process finished with exit code 0
```

### 3.1 学生源码（9 个老师分配到 4 个办公室）

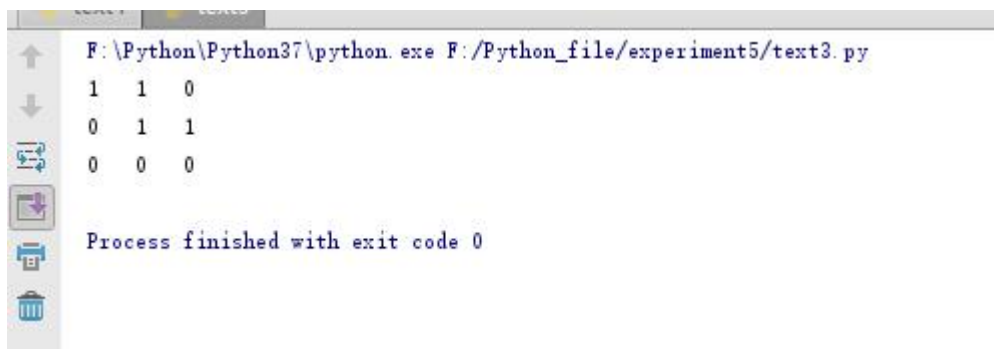
```
import random
```

```
matrix = [[0 for i in range(3)] for i in range(3)]
```

```
for i in range(3):  
    for j in range(3):  
        matrix[i][j] = random.randint(108, 111)
```

```
for i in range(3):  
    for j in range(3):  
        print(matrix[i][j], end="\t")  
    print()
```

## 2.2 运行截图（4 个老师分配到 9 个办公室）



## 3.2 学生源码（4 个老师分配到 9 个办公室）

```
import random  
  
k = 4  
  
matrix = [[0 for i in range(3)] for i in range(3)]  
  
flag = True
```

```
while (k > 0):
    for i in range(3):
        for j in range(3):
            b = random.randint(-4, 3)
            if (b >= 0 and matrix[i][j] != 1):
                matrix[i][j] = 1
                k -= 1
                if (k == 0):
                    flag = False
                    break
            else:
                matrix[i][j] = 0
        if not flag:
            break
    for i in range(3):
        for j in range(3):
            print(matrix[i][j], end="\t")
    print()
```



## 实验 4 使用列表完成好友管理系统 (30 分)

### 1. 实验要求

如今的社交软件层出不穷，虽然功能千变万化，但都具有好友管理系统的基本功能，包括添加好友、删除好友、备注好友、展示好友等。

本实验要求编写程序，模拟实现如上所述的好友管理系统。

### 2. 运行截图

```
=====欢迎使用好友管理系统=====
```

```
1: 添加好友
```

```
2: 删除好友
```

```
3: 备注好友
```

```
4: 展示好友
```

```
5: 退出
```

```
请输入您的选择: 1
```

```
请输入要添加好友姓名: 张三丰
```

```
添加成功!
```

```
请输入您的选择: 1
```

```
请输入要添加好友姓名: 李小丽
```

```
添加成功!
```

```
请输入您的选择: 4
```

```
我的好友列表是: ['张三丰', '李小丽']
```

```
请输入您的选择: 3
```

```
请输入要修改的好友姓名: 张三丰
```

```
请输入备注名: 小丰
```

```
备注成功
```

```
请输入要删除好友姓名: 小丰
```

```
删除成功!
```

```
请输入您的选择: 2
```

```
请输入要删除好友姓名: 小丽
```

```
对不起, 此好友不存在!
```

### 3. 学生截图

```
F:\Python\Python37\python.exe F:/Python_file/experiment5/text4.py
```

```
=====
欢迎使用好友管理系统
=====
```

1. 添加好友
2. 删除好友
3. 备注好友
4. 展示好友
5. 退出

```
请输入您的选择: 1
```

```
请输入要添加的好友姓名: 施嘉伟
```

```
添加成功!
```

```
=====
欢迎使用好友管理系统
=====
```

1. 添加好友
2. 删除好友
3. 备注好友
4. 展示好友
5. 退出

```
请输入您的选择: 4
```

```
我的好友列表是['施嘉伟']
```

```
=====
欢迎使用好友管理系统
=====
```

1. 添加好友
2. 删除好友
3. 备注好友
4. 展示好友
5. 退出

```
请输入您的选择: 2
```

```
请输入要删除好友姓名: 施嘉伟
```

```
删除成功
```

请输入您的选择: 2

请输入要删除好友姓名: 施嘉伟

删除成功

---

欢迎使用好友管理系统

1. 添加好友
2. 删除好友
3. 备注好友
4. 展示好友
5. 退出

请输入您的选择: 1

请输入要添加的好友姓名: 王

添加成功!

---

欢迎使用好友管理系统

1. 添加好友
2. 删除好友
3. 备注好友
4. 展示好友
5. 退出

请输入您的选择: 3

请输入要修改的好友姓名: 王

请输入备注名: 李

备注成功

---

欢迎使用好友管理系统

1. 添加好友
2. 删除好友
3. 备注好友
4. 展示好友
5. 退出

```
请输入您的选择: 3
请输入要修改的好友姓名: 王
请输入备注名: 李
备注成功
```

```
=====欢迎使用好友管理系统=====
```

- ```
1. 添加好友
2. 删除好友
3. 备注好友
4. 展示好友
5. 退出
```

```
请输入您的选择: 4
我的好友列表是['李']
```

```
=====欢迎使用好友管理系统=====
```

- ```
1. 添加好友
2. 删除好友
3. 备注好友
4. 展示好友
5. 退出
```

```
请输入您的选择: 5
您已退出
```

```
Process finished with exit code 0
|
```

## 4. 参考源码

```
friend = []

while (True):

    print("""
```

=====欢迎使用好友管理系统=====

1. 添加好友
  2. 删除好友
  3. 备注好友
  4. 展示好友
  5. 退出
- 

```
""")

i = int(input("请输入您的选择："))

if (i == 1):

    name=input("请输入要添加的好友姓名：")

    friend.append(name)

    print("添加成功！")

elif (i == 2):

    name=input("请输入要删除好友姓名：")

    if(friend.count(name)==0):

        print("对不起，此好友不存在")

    else:

        friend.remove(name)

        print("删除成功")

elif (i == 3):
```

```
name=input("请输入要修改的好友姓名：")
name2=input("请输入备注名：")
friend.remove(name)
friend.append(name2)
print("备注成功")
elif (i == 4):
    print(f"我的好友列表是{friend}")
elif (i == 5):
    print("您已退出")
    exit()
```

## 本次实验心得 (10 分)

通过本次实验，我学会了列表和元组的使用方法和基本内置函数，学会了数组的运用，并且进一步巩固了循环结构和选择结构。