

综合应用 —— 名片管理系统

目标

综合应用已经学习过的知识点：

1. 变量
2. 流程控制
3. 函数
4. 模块

开发 名片管理系统

一、系统需求

1. 程序启动，显示名片管理系统欢迎界面，并显示功能菜单

欢迎使用【名片管理系统】V1.0

1. 新建名片
2. 显示全部
3. 查询名片
0. 退出系统

2. 用户用数字选择不同的功能
3. 根据功能选择，执行不同的功能
4. 用户名片需要记录用户的 姓名、电话、QQ、邮件
5. 如果查询到指定的名片，用户可以选择 修改 或者 删除 名片

二、开发步骤

1. 框架搭建
2. 新增名片
3. 显示所有名片

4. 查询名片
5. 查询成功后修改、删除名片
6. 让 Python 程序能够直接运行

2.1 编写主运行循环

1. 新建 cards_main.py 保存 主程序功能代码

1.1 程序的入口

1.2 每一次启动名片管理系统都通过 main 这个文件启动

2. 新建 cards_tools.py 保存 所有名片功能函数

将对名片的 新增、查询、修改、删除 等功能封装在不同的函数中

在 cards_main 中添加一个 无限循环

```
while True:

    # TODO 显示系统菜单

    action = input("请选择操作功能: ")

    print("您选择的操作是: %s" % action)

    # 根据用户输入决定后续的操作

    if action in ["1", "2", "3"]:

        pass

    elif action == "0":

        print("欢迎再次使用【名片管理系统】")

        break

    else:

        print("输入错误, 请重新输入")
```

补充知识点:

TODO 注释

在 # 后跟上 TODO, 用于标记需要去做的工作

```
# TODO(作者/邮件) 显示系统菜单
```

2.2 在 cards_tools 中增加四个新函数

```
def show_menu():  
  
    """显示菜单"""  
  
    pass  
  
def new_card():  
  
    """新建名片"""  
  
    print("功能：新建名片")  
  
def show_all():  
  
    """显示全部"""  
  
    print("功能：显示全部")  
  
def search_card():  
  
    """搜索名片"""  
  
    print("功能：搜索名片")
```

2.3 导入模块

在 cards_main.py 中使用 import 导入 cards_tools 模块

```
import cards_tools
```

修改 while 循环的代码如下：

```
import cards_tools  
  
while True:  
  
    cards_tools.show_menu()  
  
    action = input("请选择操作功能：")  
  
    print("您选择的操作是：%s" % action)  
  
    # 根据用户输入决定后续的操作  
  
    if action in ["1", "2", "3"]:
```

```
    if action == "1":

        cards_tools.new_card()

    elif action == "2":

        cards_tools.show_all()

    elif action == "3":

        cards_tools.search_card()

elif action == "0":

    print("欢迎再次使用【名片管理系统】")

    break

else:

    print("输入错误，请重新输入：")
```

2.4 完成 show_menu 函数

```
def show_menu():

    """显示菜单"""

    print("*" * 50)

    print("欢迎使用【菜单管理系统】V1.0".center(42))

    print("")

    print("1. 新建名片".center(42))

    print("2. 显示全部".center(42))

    print("3. 查询名片".center(42))

    print("")

    print("0. 退出系统".center(42))

    print("*" * 50)
```

2.5 保存名片数据的结构

程序就是用来处理数据的，而**变量**就是用来存储数据的

使用 **字典** 记录 每一张名片 的详细信息

使用 **列表** 统一记录所有的 **名片字典**

定义名片列表变量

在 `cards_tools` 文件的顶部增加一个 列表变量

```
# 所有名片记录的列表
```

```
card_list = []
```

注意

所有名片相关操作，都需要使用这个列表，所以应该 **定义在程序的顶部**

程序刚运行时，没有数据，所以是 空列表

2.6 新增名片功能

分析：

1. 提示用户依次输入名片信息
2. 将名片信息保存到一个字典
3. 将字典添加到名片列表
4. 提示名片添加完成

```
def new_card():
```

```
    """新建名片"""
```

```
    print("-" * 50)
```

```
    print("功能：新建名片")
```

```
    # 1. 提示用户输入名片信息
```

```
    name = input("请输入姓名：")
```

```
    phone = input("请输入电话：")
```

```
    qq = input("请输入 QQ 号码：")
```

```
    email = input("请输入邮箱：")
```

```

# 2. 将用户信息保存到一个字典

card_dict = {"name": name,

             "phone": phone,

             "qq": qq,

             "email": email}

# 3. 将用户字典添加到名片列表

card_list.append(card_dict)

print(card_list)

# 4. 提示添加成功信息

print("成功添加 %s 的名片" % card_dict["name"])

```

2.7 显示所有名片

分析：

循环遍历名片列表，顺序显示每一个字典的信息

```

def show_all():

    """显示全部"""

    print("-" * 50)

    print("功能：显示全部")

    # 打印表头

# 1. 判断是否有名片记录

if len(card_list) == 0:

    print("提示：没有任何名片记录")

    return

for name in ["姓名", "电话", "QQ", "邮箱"]:

    print(name, end="\t\t")

print("")

```

```

# 打印分隔线

print("=" * 50)

for card_dict in card_list:

    print("%s\t\t%s\t\t%s\t\t%s" % (card_dict["name"],

                                    card_dict["phone"],

                                    card_dict["qq"],

                                    card_dict["email"]))

```

2.8 查询名片

分析：

1. 提示用户要搜索的姓名
2. 根据用户输入的姓名遍历列表
3. 搜索到指定的名片后，再执行后续的操作

```

def search_card():

    """搜索名片"""

    print("-" * 50)

    print("功能：搜索名片")

    # 1. 提示要搜索的姓名

    find_name = input("请输入要搜索的姓名：")

    # 2. 遍历字典

    for card_dict in card_list:

        if card_dict["name"] == find_name:

            print("姓名\t\t\t\t\t电话\t\t\t\t\tQQ\t\t\t\t\t邮箱")

            print("-" * 40)

            print("%s\t\t\t\t%s\t\t\t\t%s\t\t\t\t%s" % (

                card_dict["name"],

```

```

        card_dict["phone"],

        card_dict["qq"],

        card_dict["email"]))

    print("-" * 40)

    # TODO 针对找到的字典进行后续操作: 修改/删除

    break

else:

    print("没有找到 %s" % find_name)

```

增加名片操作函数: **修改/删除/返回主菜单**

```

def deal_card(find_dict):

    """操作搜索到的名片字典

    :param find_dict:找到的名片字典

    """

    print(find_dict)

    action_str = input("请选择要执行的操作 ")

    """[1] 修改 [2] 删除 [0] 返回上级菜单"""

    if action == "1":

        print("修改")

    elif action == "2":

        print("删除")

```

修改:

```

if action == "1":

    find_dict["name"] = input("请输入姓名: ")

    find_dict["phone"] = input("请输入电话: ")

    find_dict["qq"] = input("请输入 QQ: ")

```



```
find_dict["email"] = input("请输入邮件: ")

print("%s 的名片修改成功" % find_dict["name"])
```

删除:

```
elif action == "2":

    card_list.remove(find_dict)

    print("删除成功")
```