Python程序设计

亲例: 通讯簿程序



张华 WHU

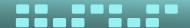
COMPUTER PROGRAMMING

通讯簿程序

■ 需求分析和功能设计

*通讯簿程序对若干联系人信息进行管理,包括以下功能:

- (1)显示通讯录清单。如果通信录字典中存在用户信息,则显示通讯录清单,包括姓 名和电话号码;如果通信录字典中不存在任何用户信息,则提示"通讯录为空"。↓
- (2)查询联系人。提示用户输入姓名 name,在通信录字典中查询该键,如果存在,输出联系人信息;如果不存在,提示是否新建联系人。₽
- (3)插入新的联系人。提示用户输入姓名 name,在通信录字典中查询该键,如果存在, 提示是否更新联系人信息;如果不存在,提示输入电话号码,并插入字典键-值对。。
- (4) 删除已有联系人。提示用户输入姓名 name,在通信录字典中查询该键,如果不存在,输出"联系人不存在"的提示信息;如果存在,从通信录字典中删除键-值对,并输出信息。↓



■ 需求分析和数据结构设计

- *通讯簿包括若干联系人的信息,每个联系人包含姓名和手机号。
- *用字典来保存通讯簿的数据,键是姓名,值是手机号。
- *用文件来持久保存通讯簿数据,并采用JSON格式文件。

JSON

- * JSON(JavaScript Object Notation, JS 对象简谱) 是一种轻量级的数据交换格式。
 - →采用完全独立于编程语言的文本格式来存储和表示数据。易于人阅读和编写。同时也易于机器解析和生成。

* 例如:

- > {"a": 1, "b": [1, 2, 3]}
- > {"name": "John Doe", "age": 18, "address": {"country" : "china", "zip-code": "10000"}}





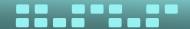
- * Python标准库模块json包含将Python对象编码为JSON格式字符串(简称JSON)和将JSON解码到Python对象的函数:
 - → dumps(obj): 返回把obj对象序列化为JSON字符串。→
 - → dump(obj, fp): 把obj对象序列化为JSON字符串写入到文件fp。→
 - → loads(s): 返回把JSON字符串s反序列化后的对象。→
 - → load(fp):返回把从文件fp中读取JSON字符串反序列化后的对象。

```
>>>-import-json-/
>>>-data=-[{'a':-'A',-'b':-(2,-4),-'c':-3.0}]-/
>>>-str_json=-json.dumps(data)-/
>>>-str_json--#输出: '[{"a":-"A",-"c":-3.0,-"b":-[2,-4]}]-/
>>>-data1=-json.loads(str_json)-/
>>>-data1----#输出: [{'a':-'A',-'c':-3.0,-'b':-[2,-4]}]-/
```



■ 流程设计

- *通讯簿程序启动后,从JSON文件中读出全部数据。
- * 进入信息管理循环
 - ▶显示功能菜单
 - ▶提示用户输入要执行的功能
 - ▶根据用户选择的功能执行相应的业务处理
 - 显示清单
 - 查询
 - 增加
 - 删除
 - > 如果选择退出,则保存全部数据到文件



```
``'`简易通讯簿程序"""
import os, json
ab = {} #通讯簿保存在字典中name: tel
#从JSON文件中读取通信录
if os.path.exists("addressbook.json"):
   with open(r'addressbook.json', 'r', encoding='utf-8') as f:
       ab = ison.load(f)
while True:
   print("|---欢迎使用通讯簿程序---|")
   print("|---1: 显示通讯簿清单 ---|")
   print("|---2: 查询联系人资料 ---|")
   print("|---3: 插入新的联系人 ---|")
   print("|---4: 删除已有联系人 ---|")
   print("|---0: 退出 -----
   choice = input('请选择功能菜单(0-4):')
```

```
if choice == '1': # 1: 显示通讯簿清单
   if(len(ab) == 0):
       print("通讯簿为空")
   else:
       for k, v in ab.items():
           print("姓名={},联系电话={}".format(k, v))
elif choice == \2': # 2: 查询联系人资料
   name = input("请输入联系人姓名: ")
   if(name not in ab):
       ask = input("联系人不存在,是否增加用户资料(Y/N)")
       if ask in ["Y", "y"]:
           tel = input("请输入用户联系电话: ")
           ab[name] = tel
   else:
       print("联系人信息: {} ".format(name, ab[name]))
```

COMPUTER PROGRAMMING

通讯簿程序

```
elif choice == '3': # 3: 插入新的联系人
        name=input("请输入联系人姓名: ")
        if(name in ab):
            print("已存在联系人: {} {}".format(name, ab[name]))
            ask = input("是否修改用户资料(Y/N)")
        if ask in ["Y", "y"]:
            tel = input("请输入用户联系电话: ")
            dict[name] = tel

else:
        tel = input("请输入用户联系电话: ")
        ab[name] = tel
```

```
elif choice == '4': # 4: 删除已有联系人
    name = input("请输入联系人姓名: ")
    if(name not in ab):
        print("联系人不存在: {}".format(name))
    else:
        tel = ab.pop(name)
        print("删除联系人: {} {}".format(name, tel))

elif choice == '0': # 保存到JSON文件并退出循环
    with open(r'addressbook.json', 'w', encoding='utf-8') as f:
        json.dump(ab, f)
    break
```