



CK0211 - Fundamentos de Programação: Registros e Arquivos

Emanuele Santos

Bibliografia: Ascencio, Cap. 11

Objetivos

- Aprender a ler registros de arquivos e salvar registros para arquivos

Introdução

- Os exemplos de arquivos que vimos até agora foram bem simples e homogêneos:
 - Cada linha do arquivo continha apenas um tipo de dado e todas as linhas do arquivo eram do mesmo tipo
 - Exemplo: arquivo de notas
 - Objetivo: armazenar dados mais complexos em arquivos e poder lê-los depois

notas.txt

60
100
76
50
95
75
63
...

Programa FUPfones

- Suponha que você gostaria de fazer um programa de agenda de telefones
- A agenda seria formada apenas por nome do contato e um número de telefone
- Suponha que os nomes não se repitam

Funcionalidades do FUPfones

***** FUPfones *****

*** Existem: 0 contatos ***

1. Inserir um contato
2. Consultar telefone
3. Remover um contato
4. Listar toda a agenda
5. Ler contatos de arquivo
6. Salvar agenda
7. Finalizar

Digite a opção desejada:

Implementação

- Opções de estruturas de dados para armazenar a agenda
 - Lista de dicionários
 - **Dicionário cujos campos são os próprios nomes dos contatos e os valores são os números de telefone (mapa)**

Inserindo contatos

- Função para inserir um contato de cada vez na agenda:

```
agenda = {}

def insere_contato(agenda):
    nome = input("Digite o nome do contato: ")
    fone = input("Digite o número do telefone: ")
    agenda[nome] = fone
```

Inserindo contatos (execução)

```
***** FUPfones *****
*** Existem: 0 contatos ***
*****
```

1. Inserir um contato
2. Consultar telefone
3. Remover um contato
4. Listar toda a agenda
5. Ler contatos de arquivo
6. Salvar agenda
7. Finalizar

Digite a opção desejada: 1

Inserindo contato

Digite o nome do contato: mary

Digite o número do telefone: 1234

Inserindo contatos (execução)

```
***** FUPfones *****  
*** Existem: 1 contatos ***  
*****
```

1. Inserir um contato
2. Consultar telefone
3. Remover um contato
4. Listar toda a agenda
5. Ler contatos de arquivo
6. Salvar agenda
7. Finalizar

Digite a opção desejada: 1

Inserindo contato

Digite o nome do contato: peter

Digite o número do telefone: 4456

Consultando a agenda (função)

- Como a agenda está usando um dicionário, podemos acessar o telefone diretamente, sem precisar percorrer a agenda inteira:

```
def busca_contato(agenda):  
    nome = input("Digite o nome do contato: ")  
    if nome in agenda:  
        return agenda[nome]  
    else:  
        return None
```

Consultando a agenda (programa principal)

- O programa principal irá chamar a função busca_contato(agenda) e só vai mostrar o telefone se ele for válido. Caso contrário, mostra mensagem de erro:

```
...
elif opcao == 2:
    print("\nConsultando um telefone\n")
    telefone = busca_telefone(agenda)
    if telefone is not None:
        print("Telefone: %s" % telefone)
    else:
        print("Contato não existe.")
```

Consultando contato (execução)

```
***** FUPfones *****  
*** Existem: 2 contatos ***  
*****  
1. Inserir um contato  
2. Consultar telefone  
3. Remover um contato  
4. Listar toda a agenda  
5. Ler contatos de arquivo  
6. Salvar agenda  
7. Finalizar  
Digite a opção desejada: 2
```

Consultando um telefone

```
Digite o nome do contato: mary  
Telefone: 1234
```

Consultando contato (execução)

```
***** FUPfones *****  
*** Existem: 2 contatos ***  
*****
```

1. Inserir um contato
2. Consultar telefone
3. Remover um contato
4. Listar toda a agenda
5. Ler contatos de arquivo
6. Salvar agenda
7. Finalizar

Digite a opção desejada: 2

Consultando um telefone

Digite o nome do contato: mark
Contato não existe.

Listando toda a agenda (função 1)

- Lembre-se que dicionários não são como listas no que diz respeito à ordem dos elementos. Conceitualmente dicionários não possuem ordem. Os elementos serão mostrados em qualquer ordem

```
def lista.todos(agenda):
    print("-----")
    print("Nome           - Telefone")
    print("-----")

    for nome in agenda:
        print("%-11s - %s" % (nome, agenda[nome]))
    print("-----")
```

Listando toda a agenda (função 2)

- Podemos ordenar os campos da agenda usando a função sorted:

```
def lista.todos(agenda):
    print("-----")
    print("Nome           - Telefone")
    print("-----")

    for nome in sorted(agenda):
        print("%-11s - %s" % (nome, agenda[nome]))
    print("-----")
```

Listando toda a agenda (execução)

```
***** FUPfones *****
*** Existem: 2 contatos ***
*****
1. Inserir um contato
2. Consultar telefone
3. Remover um contato
4. Listar toda a agenda
5. Ler contatos de arquivo
6. Salvar agenda
7. Finalizar
Digite a opção desejada: 4
```

Agenda de telefones:

Nome	- Telefone
------	------------

mary	- 1234
peter	- 4456

Removendo um contato (função)

- Para remover um campo de um dicionário, usamos a função **del**:

```
def remove_contato(agenda):
    nome = input("Digite o nome do contato: ")
    if nome in agenda:
        del agenda[nome]
        print("%s foi removido(a) da agenda." % nome)
    else:
        print("%s não foi encontrado(a) na agenda." % nome)
```

Removendo um contato (execução)

```
***** FUPfones *****
*** Existem: 2 contatos ***
*****
1. Inserir um contato
2. Consultar telefone
3. Remover um contato
4. Listar toda a agenda
5. Ler contatos de arquivo
6. Salvar agenda
7. Finalizar
Digite a opção desejada: 3
```

Removendo contato

```
Digite o nome do contato: mary
mary foi removido(a) da agenda.
```

Removendo um contato (execução)

```
***** FUPfones *****
*** Existem: 1 contatos ***
*****
1. Inserir um contato
2. Consultar telefone
3. Remover um contato
4. Listar toda a agenda
5. Ler contatos de arquivo
6. Salvar agenda
7. Finalizar
Digite a opção desejada: 3
```

Removendo contato

```
Digite o nome do contato: mark
mark não foi encontrado(a) na agenda.
```

Salvando a agenda (função)

- Vamos guardar cada contato (nome e telefone) em uma linha
- Usamos um caractere separador para separar cada elemento do registro na linha do arquivo
 - Normalmente é uma vírgula, ponto-e-vírgula, tab, etc

```
def salva_agenda(agenda):  
    nome_arquivo = input("Digite o nome do arquivo: ")  
    arquivo = open(nome_arquivo, "w")  
    for nome in agenda:  
        arquivo.write("%s,%s\n"%(nome, agenda[nome]))  
    arquivo.close()  
    print("Agenda gravada com sucesso.")
```

Salvando a agenda (execução)

```
***** FUPfones *****  
*** Existem: 8 contatos ***  
*****  
1. Inserir um contato  
2. Consultar telefone  
3. Remover um contato  
4. Listar toda a agenda  
5. Ler contatos de arquivo  
6. Salvar agenda  
7. Finalizar  
Digite a opção desejada: 6  
  
Salvando a agenda:  
Digite o nome do arquivo: telefones.txt  
Agenda gravada com sucesso.
```

telefones.txt

```
ringo,6543  
elvis,7723  
paul,4897  
mary,1234  
john,1475  
mark,2003  
george,1029  
peter,4456
```

Carregando a agenda (função)

- Vamos ler cada contato (nome e telefone) em uma linha, separados por vírgula
- Usamos a função **split(sep)** para transformar uma string em uma lista, quebrando os elementos na string **sep**; **sep** é ignorada e não é inserida na lista
- Para remover os caracteres em branco e quebras de linha do começo e do final de uma string, usamos a função **strip()**
 - São removidos brancos, tabs e \n

Carregando a agenda (função)

```
def carrega_agenda(agenda):
    nome_arquivo = input("Digite o nome do arquivo: ")
    arquivo = open(nome_arquivo, "r")
    for linha in arquivo:
        lista = linha.split(",")
        nome = lista[0].strip()
        tel = lista[1].strip()
        agenda[nome] = tel
    arquivo.close()
    print("Agenda carregada com sucesso.")
```

Carregando a agenda (execução)

```
***** FUPfones *****  
*** Existem: 0 contatos ***  
*****  
1. Inserir um contato  
2. Consultar telefone  
3. Remover um contato  
4. Listar toda a agenda  
5. Ler contatos de arquivo  
6. Salvar agenda  
7. Finalizar  
Digite a opção desejada: 5
```

Lendo contatos de arquivo

```
Digite o nome do arquivo: telefones.txt  
Agenda carregada com sucesso.
```

Carregando a agenda (execução)

```
***** FUPfones *****
*** Existem: 8 contatos ***
*****
1. Inserir um contato
2. Consultar telefone
3. Remover um contato
4. Listar toda a agenda
5. Ler contatos de arquivo
6. Salvar agenda
7. Finalizar
Digite a opção desejada: 4
```

Agenda de telefones:

Nome	-	Telefone
elvis	-	7723
george	-	1029
john	-	1475
mark	-	2003
mary	-	1234
paul	-	4897
peter	-	4456
ringo	-	6543

Exercício 1

- Ler um arquivo de notas (notas2.txt) (nomes e notas separados por tab). Mostre a lista de alunos aprovados e a lista de alunos reprovados (ambas em ordem alfabética)
- Para ser aprovado a média deve ser maior ou igual a 7.0

notas2.txt

mary	9.0
peter	7.5
mark	6.3
john	8.0
paul	10.0
ringo	7.4
george	6.0

Exercício 2

- Ler um arquivo de notas (notas3.txt) (nome, nota1 e nota2 separados por tab, com uma linha cabeçalho) Mostre a lista de alunos aprovados e a lista de alunos reprovados (ambas em ordem alfabética). Para ser aprovado, a média das notas deve ser superior ou igual a 7.0

notas3.txt

nome	nota1	nota2
mary	9.0	8.5
peter	7.5	6.8
mark	6.3	6.5
john	8.0	9.0
paul	10.0	9.0
ringo	7.4	10.0
george	6.0	8.0

Para a próxima aula...

filmes.csv

```
Título,Ano,Gênero
Ratatouille,2007,Animação
Spotlight,2015,Drama
Her,2013,Ficção Científica
Moulin Rouge!,2001,Musical
WALL-E,2008,Animação
...
...
```

avaliacoes.csv

```
Título,Estrelas
Ratatouille,5
Spotlight,5
Her,4
Moulin Rouge!,3
WALL-E,4
Ratatouille,4
Her,3
...
...
```