

ContatinhosApp - uma aplicação completa

Após concluir seus estudos de toda a matéria do curso introdutório de programação em Python da UFRJ, você está pronto para criar uma aplicação completa, com interface textual, para o aplicativo mais interessante de todos os tempos deste curso: o contatinhosApp.

Já temos as funcionalidades de manipulação da agenda de contatinhos implementadas em exercícios anteriores ao longo do curso. Vamos agora organizar estas funções em um módulo e criar, em outro arquivo, a aplicação principal e sua interface.

Vamos trabalhar com dois arquivos de código - um módulo com as funções de manipulação da agenda, e outro para a função main e demais funções de interface que sejam necessárias.

Sua aplicação deve permitir que o usuário utilize uma interface textual para realizar as manipulações da agenda de contatinhos implementadas pelas funções desenvolvidas ao longo do curso.

- **Organizando o código em dois arquivos.**

Crie em seu computador uma pasta para salvar os arquivos deste trabalho. Coloque as funções já feitas para implementar as funcionalidades de manipulação da agenda de contatinhos em um arquivo chamado *contatinhosbib.py* (parte 1 do trabalho), e salve este arquivo na pasta criada.

Agora, crie na mesma pasta, um arquivo novo no IDLE. Uma sugestão de nome para este arquivo é *ContatinhosApp.py*. Neste arquivo, importe todas as funções do módulo correspondente à parte 1 do trabalho. (veja exemplo de código abaixo).

Crie uma lista de contatos vazia.

Crie uma função main() para executar seu aplicativo. Dentro dessa função main, escreva uma mensagem de cabeçalho do aplicativo.

```
from contatinhosbib import *

contatos=[] #lista inicialmente vazia

def main():
    'Função principal do aplicativo ContatinhosApp'
    print("ContatinhosApp - organize seus contatinhos")
    print("-----")
```

A função main, além de ter a lista “contatos” que servirá para armazenar toda a agenda do usuário, também será a responsável por iniciar o aplicativo, mostrando a interface inicial:

um menu de opções que o usuário vai utilizar para escolher o que deseja fazer. A exibição deste menu na tela pode ser feita diretamente dentro da função main, mas você também tem a opção de criar uma outra função para fazer isso (uma função chamada menu, por exemplo). Fica a seu critério. O que deve ser exibido na tela quando a aplicação é iniciada:

```
ContatinhosApp - organize seus contatinhos
```

```
-----
```

```
Escolha sua opção
```

```
1- Criar novo contato
```

```
2- Buscar um contato
```

```
3- Atualizar um contato
```

```
4- Excluir um contato
```

```
5- Identificar quem ligou
```

```
6- Aglutinar dois contatos de uma mesma pessoa
```

```
7- Para sair
```

```
Digite o número correspondente a opção:
```

As opções 1 a 6, ao serem escolhidas, vão necessitar de interfaces específicas para permitir que o usuário passe as informações de entrada para o seu funcionamento. Vamos ver abaixo exemplos de interfaces para cada uma delas.

- **Contatinho novo!**

A função de criar contato deverá ter a seguinte interface:

ContatinhosApp - organize seus contatinhos

Escolha sua opção

- 1- Criar novo contato
- 2- Buscar um contato
- 3- Atualizar um contato
- 4- Excluir um contato
- 5- Identificar quem ligou
- 6- Aglutinar dois contatos de uma mesma pessoa
- 7- Para sair

Digite o número correspondente a opção: 1

PARA CRIAR UM CONTATO

- > Digite o nome: Rita Lobo
 - > Digite o telefone, se tiver: 21888777
 - > Digite o email, se tiver: ritalob@minha.net
 - > Digite o instagram, se tiver: @ritalolob
- Contatinho novo inserido na agenda.

Escolha sua opção

- 1- Criar novo contato
- 2- Buscar um contato
- 3- Atualizar um contato
- 4- Excluir um contato
- 5- Identificar quem ligou
- 6- Aglutinar dois contatos de uma mesma pessoa
- 7- Para sair

Digite o número correspondente a opção: |

- As informações em azul fazem parte da interface. As informações em preto foram digitadas pelo usuário.
- Quando o usuário termina o fornecimento de informações, o aplicativo faz a criação do contato (utilizando a função para criar contato que já está no módulo feito na parte 1 do trabalho).
- Lembre que essa função retorna uma lista com as informações do contatinho. Então, para colocar estas informações na agenda, lembre-se de acrescentar a lista criada pela função de criar contato à variável chamada contatos, que foi criada na função main.
- Depois que cada funcionalidade é finalizada, deve ser exibido na tela novamente o menu de opções, para que o usuário possa continuar usando a aplicação.

- **Buscar um contatinho** (pra mandar um “oi sumide!”)

Vamos considerar agora a opção de “buscar um contato”. Uma interface possível seria a seguinte:

Digite o número correspondente a opção: 2

PARA BUSCAR UM CONTATO

Digite o nome (ou parte do nome) do contato:ri

Contatos encontrados:

0- Rita Lobo, Tel: 21888777, Email: ritalob@uol.com.br, Insta: @ritalolob

Note que, antes de chamar a função de buscar contato que está no nosso módulo, temos que pedir para o usuário digitar uma parte do nome do contato que ele deseja buscar. Note também que a parte do nome que o usuário digitou pode ser parte do nome de mais de um contato, e nesse caso, a função de buscar deveria retornar todos os contatos cujo nome tem essa parte. Veja abaixo um exemplo:

```
Digite o número correspondente a opção: 2
PARA BUSCAR UM CONTATO
Digite o nome (ou parte do nome) do contato:a
Contatos encontrados:
0- Rita Lobo, Tel: 21888777, Email: ritalob@uol.com.br, Insta: @ritalolob
1- Roberto Carlos, Tel: 21777888, Email: , Insta: @robertoRoca
```

Os contatos retornados pela função de busca são mostrados um em cada linha, e antes dos dados do contatinho, é mostrado também um número que representa o índice do contatinho na lista de contatos. Essa informação será útil mais pra frente.

- **Atualizar as informações de um contatinho** *(já tinha o insta, agora conseguiu o zap <3)*

Para a atualizar alguma informação de um contato, será necessário, primeiramente, fazer uma busca para saber qual contatinho o usuário deseja atualizar. O resultado dessa busca (feita usando a função de busca de contatinhos definida anteriormente) pode ser zero, um ou mais contatos, e deve ser exibido para que o usuário possa escolher o contatinho que deseja atualizar. Aqui, o usuário deve indicar sua escolha fornecendo o índice numérico que é mostrado na lista de contatos exibida após a busca (lembra que foi dito que essa informação seria útil?).

Depois de identificada a pessoa, deve-se perguntar qual das informações dessa pessoa se deseja atualizar, para só então fazer a atualização (use a função para atualizar contato do módulo). Será necessário criar vários passos de interface, conforme ilustrado no exemplo abaixo:

```
Digite o número correspondente a opção: 3

PARA ATUALIZAR UM CONTATO
Digite o nome (ou parte do nome) do contato:be
Contatos encontrados:
1- Roberto Carlos, Tel: 21777888, Email: , Insta: @robertoRoca
Digite o índice do contatinho: 1
-> Qual informação você quer atualizar?
1- nome
2- telefone
3- email
4- instagram
5- Cancelar
Digite o número correspondente a opção: 3
-> Qual a nova informação? robcarlos95@ufrj.br
Contato atualizado
```

Atenção: Como um contatinho pode ter vários telefones, caso o usuário escolha atualizar o telefone, será necessário mostrar a lista de telefones armazenados e perguntar qual deles a pessoa deseja alterar.

```
Digite o número correspondente a opção: 3

PARA ATUALIZAR UM CONTATO
Digite o nome (ou parte do nome) do contatinho: rita
Contatos encontrados:
0- Rita Lobo, Tel: 21888777,21999777, Email: ritalob@uol.com.br, Insta: @ritalolob
Digite o índice do contatinho: 0
-> Qual informação você quer atualizar?
1- nome
2- telefone
3- email
4- instagram
5- Cancelar
Digite o número correspondente a opção: 2
Telefones deste contato:
0- 21888777
1- 21999777
Digite o índice do telefone: 0
-> Qual o novo telefone?219999999
Atualizado: ['Rita Lobo', ['219999999', '21999777'], 'ritalob@uol.com.br', '@ritalolob']
```

Caso a pessoa forneça um novo telefone vazio, o telefone escolhido deve ser excluído (e não alterado). Consulte as funções já existentes no módulo (parte 1 do trabalho) para identificar o que pode ser usado para executar essas tarefas.

```
Digite o número correspondente a opção: 3

PARA ATUALIZAR UM CONTATO
Digite o nome (ou parte do nome) do contatinho: ri
Contatos encontrados:
0- Rita Lobo, Tel: 219999999,21999777, Email: ritalob@uol.com.br, Insta: @ritalolob
Digite o índice do contatinho: 0
-> Qual informação você quer atualizar?
1- nome
2- telefone
3- email
4- instagram
5- Cancelar
Digite o número correspondente a opção: 2
Telefones deste contato:
0- 219999999
1- 21999777
Digite o índice do telefone: 0
-> Qual o novo telefone?
['Rita Lobo', ['21999777'], 'ritalob@uol.com.br', '@ritalolob']
```

- **Excluir um contatinho** (*é importante se livrar dos tóxicos*)

Para realizar a opção de excluir um contato, será necessário perguntar qual contato se deseja excluir. Faça uma interface semelhante à mostrada anteriormente para a opção

“atualizar contato”, que realize uma busca por parte do nome de um contato e pergunte, dentre os contatinhos retornados, qual se deseja excluir.

```
Digite o número correspondente a opção: 4
PARA EXCLUIR UM CONTATO
Digite o nome (ou parte do nome) do contatinho: be
Contatos encontrados:
1- Roberto Carlos, Tel: 21777888, Email: , Insta: @robertoRoca
Digite o índice do contatinho: 1
Contatinho excluído
```

- **Quem ligou?** *(vamos valorizar quem aparece!)*

Aqui, o usuário irá fornecer um número de telefone. Devem ser mostradas as informações do contato que ligou.

```
Digite o número correspondente a opção: 5
PARA BUSCAR UM TELEFONE
Telefone:21888777
['Rita Lobo', ['21888777'], 'ritalob@uol.com.br', '@ritalolob']
```

Caso o número não seja encontrado na agenda de contatinhos, deve ser mostrada a string “desconhecido”.

```
Digite o número correspondente a opção: 5
PARA BUSCAR UM TELEFONE
Telefone:21999000
Desconhecido
```

- **Aglutinar dois contatos** *(achou que era contatinho novo, mas já estava na agenda!)*

Em algum momento, o usuário percebeu que tinha salvo informações referentes a uma mesma pessoa como dois contatinhos separados. Agora ele pode usar a funcionalidade de aglutinar dois contatinhos. Para isso, novamente teremos primeiro que usar a função de busca para permitir que o usuário escolha quais contatinhos quer aglutinar.

Abaixo, uma sugestão de interface:

```
Digite o número correspondente a opção: 6
PARA AGLUTINAR DOIS CONTATOS
Digite o nome (ou parte do nome) do contatinho: isa
Contatos encontrados:
2- Isadora, Tel: 21808077, Email: , Insta:
3- Isa, Tel: Email: , Insta: @isadorada
Digite o índice do primeiro contatinho: 2
Digite o índice do segundo contatinho: 3
Aglutinado: ['Isadora', ['21808077'], '', '@isadorada']
```

- **Sair**

Simplesmente encerre a aplicação.

Observações importantes:

- Seu aplicativo deve funcionar mesmo que a agenda de contatos esteja vazia! Certifique-se que não ocorre um erro inesperado nesta situação, para cada uma das opções que o usuário pode escolher. Tome providências para que o usuário não consiga fazer seu aplicativo dar problema, mesmo que ele se esforce.
- Garanta a boa organização do seu código e a separação entre funcionalidades construídas e interface:
 - Nenhum *print* ou *input* deve aparecer no arquivo de módulo *contatinhosbib.py*
 - Apenas funções que estejam no arquivo *contatinhosbib* devem alterar a lista de contatos
- A funcionalidade de atualizar um contatinho é a mais difícil de ser feita (só avisando!)