

Computação I - Python

Laboratório 6

Seguindo com nossas boas práticas, para cada um dos exercícios a seguir:

- antes de começar a escrever código, faça o estudo do problema e o planejamento de sua solução.
- lembre de botar a **documentação**:

```
'''Calcula e retorna a divisão de a por b;  
int, int -> float'''
```

- escolha **nomes elucidativos** para suas funções e parâmetros;
- pense em **dados de teste** relevantes para testar sua função.
- recorra ao **teste de mesa** quando estiver com dificuldade para entender algum erro de funcionamento ou resultado inadequado de sua função.
- para fazer a entrega desta atividade prática, escreva suas funções no **IDLE**.

Vamos lá!

1. Um aplicativo de agenda de telefones chamado *contatinhosApp* está sendo desenvolvido em Python por uma equipe. Uma parte da equipe está fazendo uma linda interface pra ele, de forma que o usuário final vai interagir diretamente com a interface, e essa interface passará informações que o usuário fornece para as funções de operação, ou seja, as funções que implementarão as funcionalidades oferecidas pelo *contatinhosApp*. As funções de operação vão retornar informações para as funções de interface, de forma que as informações serão formatadas, posicionadas na tela e exibidas lindamente para o usuário pela interface. Você é um dos programadores da equipe de serviço do *contatinhosApp*, e deve desenvolver funções que implementam algumas das funcionalidades do aplicativo. Para isso, é importante levar em conta a modelagem de dados que já foi projetada para o *contatinhosApp*:
 - As informações de cada um dos contatos é armazenada em uma lista.
 - A única informação obrigatória nesse aplicativo para que um contato seja cadastrado é o nome. Além destas, a agenda comporta também as seguintes informações: um ou mais números de telefone, email e instagram.
 - Um detalhe: O número de telefone (ou telefones) de um contato é armazenado em uma lista, pois cada contato poder ter mais de um telefone.
 - A lista com as informações de cada contato segue o seguinte padrão:
 - Exemplo dos dados de um contato no formato do modelo de dados do *contatinhosApp*: ['Bruno Campos', ['2199112233', '2133992211'], 'brunoc91@emailquente.com.br', '@brunocampos91'].

Foram atribuídas a você a implementação de duas funções de serviço: criar novo contato e atualizar informações de um contato. A seguir, sua lista tarefas:

<i>índice na lista</i>	<i>informação</i>
0	nome
1	lista de telefones
2	email
3	instagram

- (a) Criar um contato novo: sua função deve criar e retornar uma lista com as informações do contato. Sua função terá 4 parâmetros: nome, telefone, email e instagram de um contato (as informações que não são obrigatórias tem como default a string vazia). No momento de criação de um novo contato, no máximo um telefone pode ser fornecido, mas atenção: na lista com as informações do contato que sua função irá retornar, o telefone fornecido tem que estar dentro de uma lista, conforme especificado no modelo de dados.
- (b) Atualizar um contato. Isso significa adicionar ou modificar informações de um contato existente. Atenção! Só é permitido atualizar **uma** informação do contato a cada chamada da função. Será passado como entrada a lista com as informações atuais do contato, o índice da informação que se deseja atualizar (conforme a tabela do modelo de dados), e a informação nova. Sua função deve realizar a atualização, da seguinte forma:
- nome, email e instagram: simplesmente salva a nova informação, desprezando a informação anterior.
 - telefone: caso este telefone já se encontre na lista de telefones do contato, ele deve ser removido. Caso não se encontre, deve ser incluído.
 - Caso seja passado um índice que não corresponda a nenhuma informação presente no modelo de dados, nenhuma atualização será feita.

Como sua função alterou diretamente a lista de contatinhos de entrada, sua função deve retornar apenas um valor booleano, indicando se a alteração foi feita ou não.