
Objetivos:

Utilizar boas práticas na resolução de problemas que envolvem conceitos de funções, listas, tuplas, dicionários e módulos.

Tarefa:

- Baixar o arquivo **ac01.py**.
- Inserir os nomes dos alunos do grupo no comentário inicial do arquivo.
- Implementar as funções definidas no arquivo de acordo com os comentários e descrições apresentadas abaixo (não esqueça que as funções devem retornar um resultado, utilizando o return).

Atenção:

- Não serão aceitos trabalhos entregues em atraso.
- Os nomes das funções contidas no arquivo **ac01.py** não devem ser alterados. Caso algum nome seja alterado, a função será considerada incorreta.
- Caso seja identificada a cópia de trabalhos entre os grupos, a nota será zerada.
- O arquivo **ac01_teste.py** pode ser utilizado para verificar a implementação das funções, porém outros testes poderão ser realizados na correção dessa atividade

Forma de Entrega:

- Anexar no Classroom o arquivo **ac01.py** com a implementação das funções solicitadas.
- Pode ser enviado por apenas um integrante do grupo (mas não esqueça de colocar o nome de todos os integrantes do grupo no início do arquivo).

Descrição das Funções:

1. Escreva uma função com o nome **pertence**, que recebe como argumentos de entrada uma tupla e um item e retorna **True**, se o item estiver armazenado na tupla, e **False**, caso contrário.

Exemplos:

```
tupla = (2, 3, 4)
resultado = pertence(tupla, 3)
print(resultado)                                # True

tupla = (2, 3, 4)
resultado = pertence(tupla, 1)
print(resultado)                                # False
```

2. Escreva uma função chamada **substituir** que recebe como argumentos de entrada uma lista e dois itens (**velho** e **novo**) e retorna uma lista onde todas as ocorrências do item **velho** são substituídas pelo item **novo**.

Exemplos:

```
lista = [1, 2, 3, 2, 4]
resultado = substituir(lista, 2, 99)
print(resultado)                                # [1, 99, 3, 99, 4]
```

3. Escreva uma função chamada **posicoes_lista** que recebe como argumentos de entrada uma lista e um item, e retorna uma lista contendo todas os índices em que o item aparece na lista.

```
lista = ['a', 2, 'b', 'a', 'a']
resultado = posicoes_lista(lista, 'a')
print(resultado) # [0, 3, 4]
```

```
alunos = {'Augusto': [4.5, 7.0, 6.0, 3.0],
          'Denise': [9.0, 8.5],
          'Ana Paula': [3.5, 1.0, 6.5],
          'Marcelo': [9.0, 10.0, 7.0, 7.0]}
resultado = aprovados(alunos)
print(resultado)                                # ['Denise', 'Marcelo']
```

```
alunos = {'Augusto': [4.5, 7.0, 6.0, 3.0],
          'Denise': [9.0, 8.5],
          'Ana Paula': [3.5, 1.0, 6.5],
          'Marcelo': [9.0, 10.0, 7.0, 7.0]}
resultado = incluir_nota(alunos, 'Denise', 10.0)
print(resultado)
# {'Augusto': [4.5, 7.0, 6.0, 3.0],
#  'Denise': [9.0, 8.5, 10.0],
#  'Ana Paula': [3.5, 1.0, 6.5],
#  'Marcelo': [9.0, 10.0, 7.0, 7.0]}
```

[illegible]