

INSTITUTO SANTOS DUMONT, ISD
INSTITUTO INTERNACIONAL DE NEUROCIÊNCIAS EDMOND E LILY SAFRA,
IIN-ELS

PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROENGENHARIA

**COMPONENTE CURRICULAR: FUNDAMENTOS DE PROGRAMAÇÃO E
DESENVOLVIMENTO DE PROJETOS APLICADOS À NEUROENGENHARIA**

DOCENTE: PROF. Dr. ANDRÉ FELIPE OLIVEIRA DE AZEVEDO DANTAS

DISCENTE: YAGO DANIEL SOUTO

LISTA 05

QUESTÕES OBJETIVAS:

Objetivo 1: Definir tipos de dados compostos como Tupla, Lista, Dicionário

Questão 1: Tipos de dados podem ser compostos por pequenas partes, exemplo: uma string é composta de strings menores que contém um único caractere. Conforme as características de tipos de dados compostos como Tupla, Lista e Dicionário, julgar como verdadeira ou falsa as características referentes a cada uma delas.

- a) Listas: é uma coleção ordenada;
- b) Tuplas: são sequências de valores;
- c) Dicionário: estrutura de dados que implementa mapeamentos entre uma chave (key) e algum conteúdo (value);
- d) Os valores de uma tupla, ao contrário de uma lista, não são imutáveis;
- e) Tuplas usam parênteses enquanto listas usam colchetes.

Objetivo 2: Usar as estruturas básicas de programação com Listas, Tuplas e Dicionários

Questão 2: São sintaxes de criação de uma Lista, Tupla e Dicionário:

- a) Lista = [] ou Lista = [elemento1, elemento2, elemento3, ...];
- b) Tupla = (1, 2, 3, ...);
- c) Dicionário:

```
agenda = {}  
print(agenda)  
agenda["Teresa"]=[65443322]  
print(agenda)
```

QUESTÃO CONTEXTUALIZADA

QUESTÃO 01:

```
#INSTITUTO INTERNACIONAL DE NEUROCIÊNCIAS EDMOND E LILY SAFRA, IIN-ELS
#INSTITUTO SANTOS DUMONT
#PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROENGENHARIA
#COMPONENTE CURRICULAR: FUND. DE PROGRAMAÇÃO E DESENVOLVIMENTO DE PROJETOS APLICADOS À NEUROENGENHARIA
#DOCENTE: PROF. Dr. ANDRÉ FELIPE OLIVEIRA DE AZEVEDO DANTAS
#DISCENTE: YAGO DANIEL SOUTO

# a. Realize o download dos dados de forma manual e crie uma lista e uma tupla com as informações disponíveis no documento csv (coloque pelo menos 1 linha por estado e 10 regiões de saúde diferentes, algo próximo de umas 40 linhas).

#Dados atualizados de 15.09.2021
#HIST_PAINEL_COVIDBR_2021_Parte2_15set2021

#Listas com estados escolhidos
estados1 = ['AL', 'PB', 'RN', 'CE', 'PE', 'BA', 'MA', 'SE', 'PI', 'RJ']
populacaoTCU20191 = [3337357, 4018127, 3506853, 9132078, 9557071, 14873064, 7075181, 2298696, 3273227, 17264943]
casosAcumulado1 = [237350, 436645, 366532, 935273, 614430, 1227412, 353095, 277747, 318125, 1160185]
casosNovos1 = [63, 16, 115, 761, 289, 471, 348, 61, 59, 384]
obitosAcumulado1 = [6151, 9253, 7309, 24144, 19574, 26689, 10120, 6003, 6980, 64295]
obitosNovos1 = [5, 3, 0, 12,14, 12, 10, 0, 0, 218]

#Tuplas com estados escolhidos
estados2 = ('AL', 'PB', 'RN', 'CE', 'PE', 'BA', 'MA', 'SE', 'PI', 'RJ')
populacaoTCU20192 = (3337357, 4018127, 3506853, 9132078, 9557071, 14873064, 7075181, 2298696, 3273227, 17264943)
casosAcumulados2 = (237350, 436645, 366532, 935273, 614430, 1227412, 353095, 277747, 318125, 1160185)
casosNovos2 = (63, 16, 115, 761, 289, 471, 348, 61, 59, 384)
obitosAcumulados2 = (6151, 9253, 7309, 24144, 19574, 26689, 10120, 6003, 6980, 64295)
obitosNovos2 = (5, 3, 0, 12,14, 12, 10, 0, 0, 218)

HIST_PAINEL_COVIDBR_2021_Parte2_15_1 = [estados1, populacaoTCU20191, casosAcumulado1, casosNovos1, obitosAcumulado1, obitosNovos1]
HIST_PAINEL_COVIDBR_2021_Parte2_15_2 = (estados2, populacaoTCU20192, casosAcumulados2, casosNovos2, obitosAcumulados2, obitosNovos2)

#b. Mande printar na tela o número de casos acumulados para o estado do rio de janeiro
```

```

#tanto para a tupla quanto para a lista.

if 'RJ' in HIST_PAINEL_COVIDBR_2021_Parte2_15_1 [0]:
    print ('Número de casos acumulados no estado do Rio de Janeiro: ', HIST_PA
INEL_COVIDBR_2021_Parte2_15_1[2][9])
if 'RJ' in HIST_PAINEL_COVIDBR_2021_Parte2_15_2 [0]:
    print ('Número de casos acumulados no estado do Rio de Janeiro: ', HIST_PA
INEL_COVIDBR_2021_Parte2_15_2[2][9])

#c. Apresente na tela todos os óbitos acumulados mostrando os casos apenas par
a o
#caso dos estados (sem mostrar regiões de saúde, etc..).
#Lista
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][0], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][0])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][1], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][1])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][2], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][2])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][3], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][3])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][4], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][4])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][5], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][5])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][6], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][6])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][7], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][7])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][8], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][8])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0][9], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_1 [4][9])
#Tupla
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][0], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_2 [4][0])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][1], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_2 [4][1])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][2], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_2 [4][2])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][3], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_2 [4][3])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][4], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_2 [4][4])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][5], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_2 [4][5])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][6], HIST_PAINEL_COVIDBR_2021_Pa
rte2_15_2 [4][6])

```

```

print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][7], HIST_PAINEL_COVIDBR_2021_Parte2_15_2 [4][7])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][8], HIST_PAINEL_COVIDBR_2021_Parte2_15_2 [4][8])
print (HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0][9], HIST_PAINEL_COVIDBR_2021_Parte2_15_2 [4][9])

#d. Assuma que os dados de óbitos novos para o estado da paraíba estejam errados em 10 unidades
#para menos. Sobrescreva a informação tanto na lista quanto na tupla, corrigindo os dados.
#e. As duas operações foram possíveis (lista e tupla)? Justifique.
#####-----
#####
#Resposta: A alteração da lista foi possível, no entanto, a alteração da tupla é impossível por ser imutável.
#####-----
#####
#Lista
if 'PB' in HIST_PAINEL_COVIDBR_2021_Parte2_15_1[0]:
    HIST_PAINEL_COVIDBR_2021_Parte2_15_1[5][1] += 10
    print ('As correções foram realizadas nos dados de óbitos novos na Paraíba, segue os dados: ', HIST_PAINEL_COVIDBR_2021_Parte2_15_1[5][1])
    print ('Óbitos novos na Paraíba atualizados: ', HIST_PAINEL_COVIDBR_2021_Parte2_15_1[5])

#Tupla
#if 'PB' in HIST_PAINEL_COVIDBR_2021_Parte2_15_2[0]:
    #HIST_PAINEL_COVIDBR_2021_Parte2_15_2[5][1] += 10
    #print ('As correções foram realizadas nos dados de óbitos novos na Paraíba, segue os dados: ', HIST_PAINEL_COVIDBR_2021_Parte2_15_2[5][1])
    #print ('Óbitos novos na Paraíba atualizados: ', HIST_PAINEL_COVIDBR_2021_Parte2_15_2[5])

#f. Crie uma nova lista com apenas dados de 1 estado e todos os municípios e adicione essa lista nova a lista já existente (append ou insert).
#g. Remova da lista os dados das regiões de saúde.
#h. Verifique se a soma dos dados dos municípios na data de 18/08/2020 é igual ao dado da lista, mostrando na tela apenas se for verdadeiro.
#i. Retorne o tamanho total da lista.
#Rio Grande do Norte (14 municípios selecionados)
municipioRN = ['Acari', 'Alexandria', 'Caraúbas', 'Caicó', 'Cruzeta', 'Currais Novos', 'Jardim do Seridó', 'Macaíba', 'Marcelino Vieira', 'Martins', 'Mossoró', 'Natal', 'Parelhas', 'Pau dos Ferros']
RegiaoSaudeRN = ['4ª REGIAO DE SAUDE - CAICO', '6ª REGIAO DE SAUDE - PAU DOS FERROS', '2ª REGIAO DE SAUDE - MOSSORO', '4ª REGIAO DE SAUDE - CAICO', '4ª REGIAO DE SAUDE - CAICO', '4ª REGIAO DE SAUDE - CAICO', '7ª REGIAO DE SAUDE - METROPOLITANA', '6ª REGIAO DE SAUDE - PAU DOS FERROS', '6ª REGIAO DE SAUDE - PAU DOS FERROS', '2ª REGIAO DE SAUDE - MOSSORO', '7ª

```

```

REGIAO DE SAUDE - METROPOLITANA', '4ª REGIAO DE SAUDE - CAICO', '6ª REGIAO DE
SAUDE - PAU DOS FERROS']
populacaoTCU2019RN = [11136, 13577, 20493, 67952, 7998, 44786, 12396, 80792, 8
347, 8725, 297378, 884122, 21477, 30394]
casosAcumuladoRN = [1120, 1900, 3096, 12332, 1178, 3953, 1776, 5800, 1012, 549
, 29998, 99810, 2724, 5054]
casosNovos = [0, 0, 0, 0, 0, 0, 3, 0, 0, 15, 31, 0, 0]
obitosAcumuladosRN = [19, 32, 44, 152, 12, 85, 23, 171, 13, 8, 581, 2686, 29,
38]
obitosNovosRN = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

HIST_PAINEL_COVIDBR_2021_Parte2_15_RN = [municipioRN, RegiaoSaudeRN, populacao
TCU2019RN, casosAcumuladoRN, casosNovos, obitosAcumuladosRN, obitosNovosRN]

HIST_PAINEL_COVIDBR_2021_Parte2_15_RN.append(HIST_PAINEL_COVIDBR_2021_Parte2_1
5_RN)
print(HIST_PAINEL_COVIDBR_2021_Parte2_15_RN)

HIST_PAINEL_COVIDBR_2021_Parte2_15_RN.remove(RegiaoSaudeRN)
print(HIST_PAINEL_COVIDBR_2021_Parte2_15_RN, HIST_PAINEL_COVIDBR_2021_Parte2_1
5_1)

somaRN = [sum(HIST_PAINEL_COVIDBR_2021_Parte2_15_RN[1]), sum(HIST_PAINEL_COVID
BR_2021_Parte2_15_RN[2]),sum(HIST_PAINEL_COVIDBR_2021_Parte2_15_RN[3]), sum(HI
ST_PAINEL_COVIDBR_2021_Parte2_15_RN[4]), sum(HIST_PAINEL_COVIDBR_2021_Parte2_1
5_RN[5])]

if (somaRN[0] == HIST_PAINEL_COVIDBR_2021_Parte2_15_1[1][0]):
    if(somaRN[1] == HIST_PAINEL_COVIDBR_2021_Parte2_15_1[2][0]):
        if(somaRN[2] == HIST_PAINEL_COVIDBR_2021_Parte2_15_1[3][0]):
            if(somaRN[3] == HIST_PAINEL_COVIDBR_2021_Parte2_15_1[4][0]):
                if(somaRN[4] == HIST_PAINEL_COVIDBR_2021_Parte2_15_1[5][0]):
                    print("As somas são iguais.")

print ("O tamanho da lista é dado por: ", len(dadosCOVID))

#j. Verifique qual é o maior valor numérico de óbitos novos e o menor valor nu
mérico de óbitos novos.

print (HIST_PAINEL_COVIDBR_2021_Parte2_15_1[5])
print ('O menor valor numérico de óbitos foi: ', min(HIST_PAINEL_COVIDBR_2021_
Parte2_15_1[5]), ' pessoas.')
print ('O maior valor numérico de óbitos foi: ', max(HIST_PAINEL_COVIDBR_2021_
Parte2_15_1[5]), ' pessoas.')

```

