

Aula 16 – Tuplas

72. Crie um programa que tenha uma tupla totalmente preenchida com uma contagem por extenso, de zero até vinte. Seu programa deverá ler um número pelo teclado (entre 0 e 20) e mostrá-lo por extenso.
73. Crie uma tupla preenchida com os 20 primeiros colocados da tabela do Campeonato Brasileiro de futebol, na ordem de colocação. Depois mostre:
- Apenas os 5 primeiros colocados;
 - Os últimos 4 colocados da tabela;
 - Uma lista com os times em ordem alfabética;
 - Em que posição na tabela está o time Chapecoense.
74. Crie um programa que vai gerar cinco números aleatórios e colocar em uma tupla. Depois disso, mostre a listagem dos números gerados e também indique o menor e o maior valor que estão na tupla.
75. Desenvolva um programa que leia quatro valores pelo teclado e guarde-os em uma tupla. No final, mostre:
- Quantas vezes apareceu o valor 9;
 - Em que posição foi digitado o primeiro valor 3;
 - Quais foram os números pares.
76. Crie um programa que tenha uma tupla única com nomes de produtos e seus respectivos preços, na sequência. No final, mostre uma listagem de preços, organizando os dados em forma tabular.
77. Crie um programa que tenha uma tupla com várias palavras (não usar acentos). Depois disso, você deve mostrar, para cada palavra, quais são as suas vogais.

Aula 17 – Listas (parte 1)

78. Faça um programa que leia 5 valores numéricos e guarde-os em uma lista. No final, mostre qual foi o maior e o menor valor digitado e as suas respectivas posições na lista.
79. Crie um programa onde o usuário possa digitar vários valores numéricos e cadastre-os em uma lista. Caso o número já exista lá dentro, ele não será adicionado. No final, serão exibidos todos os valores únicos digitados, em ordem crescente.
80. Crie um programa onde o usuário possa digitar cinco valores numéricos e cadastre-os em uma lista, já na posição correta de inserção (sem usar o `sort()`). No final, mostre a lista ordenada na tela.
81. Crie um programa que vai ler vários números e colocar em uma lista. Depois disso, mostre:
- Quantos números foram digitados;
 - A lista dos valores, ordenada de forma decrescente;
 - Se o valor 5 foi digitado e está ou não na lista.

82. Crie um programa que vai ler vários números e colocar em uma lista. Depois disso, crie duas listas extras que vão conter apenas os valores pares e os valores ímpares digitados, respectivamente. Ao final, mostre o conteúdo das três listas geradas.
83. Crie um programa onde o usuário digite uma expressão qualquer que use parênteses. Seu aplicativo deverá analisar se a expressão passada está com os parênteses abertos e fechados na ordem correcta.

Aula 18 – Listas (parte 2)

84. Faça um programa que leia nome e peso de várias pessoas, guardando tudo em uma lista. No final, mostre:
 - Quantas pessoas foram cadastradas;
 - Uma listagem com as pessoas mais pesadas;
 - Uma listagem com as pessoas mais leves.
85. Crie um programa onde o usuário possa digitar sete valores numéricos e cadastre-os em uma lista única que mantenha separados os valores pares e ímpares. No final, mostre os valores pares e ímpares em ordem crescente.
86. Crie um programa que crie uma matriz de dimensão 3x3 e preencha com valores lidos pelo teclado. No final, mostre a matriz na tela, com a formatação correcta.
87. Aprimore o desafio anterior, mostrando no final:
 - A soma de todos os valores pares digitados;
 - A soma dos valores da terceira coluna;
 - O maior valor da segunda linha.
88. Faça um programa que ajude um jogador da *MEGA SENA* a criar palpites. O programa vai perguntar quantos jogos serão gerados e vai sortear 6 números entre 1 e 60 para cada jogo, cadastrando tudo em uma lista composta.
89. Crie um programa que leia nome e duas notas de vários alunos e guarde tudo em uma lista composta. No final, mostre um boletim contendo a média de cada um e permita que o usuário possa mostrar as notas de cada aluno individualmente.

Aula 19 – Dicionários

90. Faça um programa que leia nome e média de um aluno, guardando também a situação em um dicionário. No final, mostre o conteúdo da estrutura na tela.
91. Crie um programa onde 4 jogadores joguem um dado e tenham resultados aleatórios. Guarde esses resultados em um dicionário. No final, coloque esse dicionário em ordem, sabendo que o vencedor tirou o maior número no dado.

92. Crie um programa que leia nome, ano de nascimento e carteira de trabalho e cadastre-os (com idade) em um dicionário. Se por acaso a CTPS for diferente de zero, o dicionário receberá também o ano de contratação e o salário. Calcule e acrescente, além da idade, com quantos anos a pessoa vai se aposentar.
93. Crie um programa que gerencie o aproveitamento de um jogador de futebol. O programa vai ler o nome do jogador e quantas partidas ele jogou. Depois vai ler a quantidade de golos feitos em cada partida. No final, tudo isso será guardado em um dicionário, incluindo o total de golos feitos durante o campeonato.
94. Crie um programa que leia nome, sexo e idade de várias pessoas, guardando os dados de cada pessoa em um dicionário e todos os dicionários em uma lista. No final, mostre:
- Quantas pessoas foram cadastradas;
 - A média de idade do grupo;
 - Uma lista com todas as mulheres;
 - Uma lista com todas as pessoas com idade acima da média.
95. Aprimore o desafio 93 para que ele funcione com vários jogadores, incluindo um sistema de visualização de detalhes do aproveitamento de cada jogador.

Aula 20 – Funções (parte 1)

96. Faça um programa que tenha uma função chamada *área()*, que receba as dimensões de um terreno rectangular (largura e comprimento) e mostre a área do terreno.
97. Faça um programa que tenha uma função chamada *escreva()*, que receba um texto qualquer como parâmetro e mostre uma mensagem com tamanho adaptável.

Ex:

```
escreva('Olá, mundo!')
```

Saída:

```
=====
Olá, mundo!
=====
```

98. Faça um programa que tenha uma função chamada *contador()*, que receba três parâmetros: início, fim e passo e realize a contagem. Seu programa tem que realizar três contagens através da função criada:
- a) De 1 até 10, de 1 em 1;
 - b) De 10 até 0, de 2 em 2;
 - c) Uma contagem personalizada.
99. Faça um programa que tenha uma função chamada *maior()*, que receba vários parâmetros com valores inteiros. Seu programa tem que analisar todos os valores e dizer qual deles é o maior.

100. Faça um programa que tenha uma lista chamada *números* e duas funções chamadas *sorteia()* e *somaPar()*. A primeira função vai sortear 5 números e vai colocá-los dentro da lista e a segunda função vai mostrar a soma entre todos os valores pares sorteados pela função anterior.

Aula 21 – Funções (parte 2)

101. Crie um programa que tenha uma função chamada *voto()* que vai receber como parâmetro o ano de nascimento de uma pessoa, retornando um valor literal indicando se uma pessoa tem voto NEGADO, OPCIONAL ou OBRIGATÓRIO nas eleições.
102. Crie um programa que tenha uma função *factorial()* que receba dois parâmetros: o primeiro que indique o número a calcular e o outro chamado *show*, que será um valor lógico (opcional) indicando se será mostrado ou não na tela o processo de cálculo do factorial.
103. Faça um programa que tenha uma função chamada *ficha()*, que receba dois parâmetros opcionais: o nome de um jogador e quantos golos ele marcou. O programa deverá ser capaz de mostrar a ficha do jogador, mesmo que algum dado não tenha sido informado correctamente.
104. Crie um programa que tenha a função *leiaInt()*, que vai funcionar de forma semelhante à função *input()* do Python, só que fazendo a validação para aceitar apenas valor numérico.

Ex:

```
n = leiaInt('Digite um n')
```

105. Faça um programa que tenha uma função *notas()* que pode receber várias notas de alunos e vai retornar um dicionário com as seguintes informações:
- Quantidade de notas
 - A maior nota
 - A menor nota
 - A média da turma
 - A situação (opcional)

Adicione também as *docstrings* da função.

106. Faça um mini-sistema que utilize o *InteractiveHelp* do Python. O usuário vai digitar o comando e o manual vai aparecer. Quando o usuário digitar a palavra 'FIM', o programa se encerrará.

OBS: use cores.