Revisão

Matéria de Hoje

Matéria Da Aula de Hoje

- Laço de repetição for como contador
- Variáveis coleção – listas e tuplas
- Usando laço de repetição com listas e tuplas
- Funções e métodos de listas
- Indexação de listas

Laço de repetição for como contador

Criando Funções

Laço de repetição for como contador

- O laço de repetição for é um laço quantitativo.
- Diferente do while, que funciona enquanto True, o for ele funciona uma determinada quantidade de vezes e depois para.

Laço de repetição for como contador

- Para trabalharmos com o for como contador, devemos escolher a quantidade de vezes que ele deve funcionar usando a função range( )

Laço de repetição for como contador

- A sintaxe do for é a seguinte:

  for i in range(10):

  ….<código a ser repetido>

Laço de repetição for como contador

- A sintaxe do for é a seguinte:

  for i in range(10):

  ….<código a ser repetido>

- O i é uma variável quase padrão do for, mas pode ser trocada por qualquer outra variável.

Laço de repetição for como contador

- A sintaxe do for é a seguinte:

  for i in range(10):

  ….<código a ser repetido>

- O range() deve receber um número inteiro. Este número é a quantidade de vezes que o for irá repetir o código

Exemplo1:

Laço de repetição for como contador

- Nós podemos também usar a variável i para mostrar um número a cada ciclo.
- O número irá começar do 0 (zero) e vai até um numero antes do número escolhido

Exemplo2:

Exercício 8

Variáveis coleção – listas e tuplas

Variáveis coleção – listas e tuplas

- As variáveis tipo coleção são variáveis que podem guardar vários dados de forma ordenada.
- Não há limite de tamanho podendo crescer conforme a necessidade.

Variáveis coleção – listas e tuplas

- No Python, 2 variáveis tipo coleções são muito usadas.
- Tuplas e Listas

Variáveis coleção – listas e tuplas

- As Tuplas e Listas são muito semelhantes.
- A diferença é que a lista pode ser alterada e a tupla é imutável (não pode ser alterada)

Variáveis coleção – listas e tuplas

- Tuplas são criadas usando os parênteses ( )
  - tupla = (1, 2.5, "Abacate", True)
- Após criada ela não pode ser mais mudada
- A forma de acessar os dados da tupla e trabalhar com ela é semelhante às listas.

Variáveis coleção – listas e tuplas

- Listas são criadas usando colchetes [ ]
  - lista = [1, 2.5, "Abacate", True]
- Após criada ela pode ser modificada, adicionando mais dados, apagando, mudando valor...
- Por este dinamismo, a lista é muito usada para guardar dados evitando de criar várias variáveis.

Exemplo3

Usando laço de repetição com listas e tuplas

Usando laço de repetição com listas e tuplas

- Uma forma de poder trabalhar com os dados da lista é usar o for.
- A união do for com a lista pode ajudar muito na programação poupando de criar inúmeras variáveis e repetir o código.

Usando laço de repetição com listas e tuplas

- Podemos trabalhar com 2 formas:
- 1ª Usando o for e lista para receber vários dados
- 2ª Usando o for para acessar dados da lista de forma sistemática.

Usando laço de repetição com listas e tuplas

- 1ª Usando o for e lista para receber vários dados
  - O princípio desta técnica é usar um loop para repetir o um código que recebe o valor do teclado e armazena em uma lista.

Exemplo4

Usando laço de repetição com listas e tuplas

- 2ª Usando o for para acessar dados de uma lista.
  - O princípio é usar o for para percorrer uma lista.
  - Ao percorrer esta lista, os dados serão resgatados um a um até a lista acabar.
  - Com estes dados resgatados, podemos fazer o que quisermos.

Usando laço de repetição com listas e tuplas

- 2ª Usando o for para acessar dados de uma lista.
  - Existe 2 formas de percorrer uma lista.
    - No for diretamente ou usando a indexação
    - Vamos ver usando o for diretamente.

Exemplo5

Exercício 09

Funções e métodos de listas

Funções e métodos de listas

- Para trabalharmos com listas de forma a tirar bom proveito dela, usamos funções e métodos para nos auxiliarmos.

Funções e métodos de listas

- Vamos ver 4 funções que podem ser usadas nas listas
- sun() - Retorna a soma de todos os elementos da lista
- min() - Retorna o menor valor da lista
- max() - Retorna o maior valor da lista
- len() - Retorna o número de elementos de uma lista.

Exemplo6

Funções e métodos de listas

- Vamos ver os métodos mais importantes da lista.

Funções e métodos de listas

- .append() - Adiciona elemento no final da lista
- .insert() - Adiciona um elemento em uma determinada posição
- .pop() - Remove o ultimo elemento ou de uma determinada posição e retorna para o código
- .remove() - Remove o elemento pelo valor dele
- .sort() - Organiza a lista do menor para o maior número
- .reverse() - Inverte a lista

Exemplo7

Indexação de listas

Indexação de listas

- Uma funcionalidade das listas, string e tuplas é a indexação.
- A indexação é o o uso de índices para poder fatiar ou até isolar um elemento.

Indexação de listas

- Primeiro precisamos saber como a lista é estruturada.

Indexação de listas

- A lista possui índices que marca a posição do dado inserido

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice	0	1	2	3

- É de se notar que a lista começa pela posição/índice 0 e vai crescendo de um em um

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice	0	1	2	3

- Para acessarmos uma posição devemos chamar a lista com colchetes e dentro deles colocar o índice do objeto que queremos recuperar

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice 0 1 2 3

- Para recuperar a palavra peixe

  lista[1]

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice 0 1 2 3

- Para recuperar a palavra peixe

  lista[1]

Exemplo8

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice 0 1 2 3

- Se queremos fatiar a lista devemos sinalizar a posição inicial e a posição final+1 dentro do colchetes separado por :
- lista[inicio : fim+1]

Exemplo9

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice 0 1 2 3

- Podemos até pegar de dois em dois
- lista[inicio : fim+1 : incremento]

Exemplo10

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice 0 1 2 3

- Note como os números do índice acabam aumentando de de 1 em um

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice 0 1 2 3

- E que para chamar um elemento da lista basta colocar um número dentro dos parênteses.
- lista[0]
- lista[1]
- lista[2] ...

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice	0	1	2	3

- Sabe quem pode gerar números sequenciais em um loop?

Indexação de listas

lista = ["Água", "Peixe", "Golfinho", "Tubarão"]

índice 0 1 2 3

- O nosso for!
- Se adicionarmos o len(lista) dentro do range() o for irá girar a quantidade de vezes necessário para poder resgatar item por item da nossa lista.

Exemplo11

Exercício 10