

INTRODUÇÃO A PROGRAMACÃO

PROF. DEMÉTRIUS DE CASTRO

PROF2303@IESP.EDU.BR

83 9 87730383

WWW.DEMETRIUSDECASTRO.COM.BR

PYTHON

LISTAS E TUPLAS

LISTAS

Para criar (declarar) uma lista com elementos, deve-se usar colchetes [] e adicionar os itens dentro deles e separados por vírgula.

Ex.:

```
frutas = ["Uva", "Maça", "Melão", "Caqui", "Pera", "Jaca"]
```

```
programadores = ["Thiago", "Raissa", "Demetrius"]
```

```
animais = ["Cachorro", "Gato", "Cavalo"]
```

PYTHON

LISTAS E TUPLAS

LISTAS

Como podemos chamar os valores de uma Lista?

Usando o comando `print()`

Ex.:

Para exibir o valor da posição 3 da lista de frutas, fazemos:

```
print(frutas[2])
```

Para exibir o nome do programados na primeira posição da lista de programadores:

```
print(programadores[0])
```

PYTHON

LISTAS E TUPLAS

LISTAS

E para imprimir todos os valores de uma lista:

```
print(programadores).
```

Também é possível obter informação de uma lista (tamanho, tipo), usando comando específicos.

Ex.:

Exibir a quantidade (tamanho) da lista:

```
print(len(programadores))
```

PYTHON

LISTAS E TUPLAS

LISTAS

Podemos alterar valores em uma lista de maneira bem simples, bastando informar qual o índice do valor a ser alterado.

Ex.:

Para alterar o nome da fruta da posição 2 (Melão) para melancia.

```
frutas[2] = "Melancia"
```

PYTHON

LISTAS E TUPLAS

LISTAS

Uma lista pode ter um número infinito de valores, e estes valores podem ser adicionados com o decorrer da execução do código.

Para adicionar novos itens a uma lista, podemos usar 2 comando, o “.append()” e o “.insert()”. O primeiro adiciona um novo valor a última posição da lista, já o segundo, você pode informar a posição para que aquele valor seja adicionado.

Ex.:

Adicionando uma nova fruta ao final da lista

```
fruta.append(“abacaxi”)
```

PYTHON

LISTAS E TUPLAS

LISTAS

Já o comando `.insert()`, recebe 2 parâmetros, sendo a posição e o valor a ser inserido e os demais valores existentes, assumem a posição seguinte em cadeia.

Ex.:

Vamos adicionar o abacate ao índice 2 da nossa lista

```
frutas.insert(2, "Abacate")
```

PYTHON

LISTAS E TUPLAS

LISTAS

Assim como podemos adicionar, podemos remover valores da nossa lista.

Da mesma forma que a inserção, a remoção também pode ser feita de duas formas, pelo `.remove()`, que remove o item pela seu valor.

Ex.:

```
frutas.remove("Caqui")
```


PYTHON

LISTAS E TUPLAS

LISTAS

Como podemos excluir um valor pelo seu índice usando o comando `.pop()`

Ex.:

```
fruta.pop(1)
```

Outra vantagem de uma lista, é que podemos ter diferentes tipos de dados na mesma lista.

Ex.:

```
frutas = {"Uva", 10, 3.50}
```

Nome, quantidade e valor (string, int e double)

PYTHON

LISTAS E TUPLAS

FAÇA

Crie uma lista vazia.

Adicione cinco valores (“Lenovo”, “Acer”, “Samsung”, “Azus”, “Dell”).

Remova o valor da posição 3.

Adicione o valor “Positivo” na última posição

Adicione o valor “Apple” na terceira posição

Imprima o resultado

PYTHON

LISTAS E TUPLAS

TUPLA

É uma estrutura de dados semelhantes a lista, porém, ela tem a característica de ser imutável, ou seja, após ser criada ela não pode ser alterada.

O comportamento de uma tupla é semelhante a lista, quanto ao chamado de seus valores.

As principais diferenças entre lista e tupla, é que, não usamos os [] e sai () para sua declaração ,e caso seja declarada uma tupla com apenas um valor, deve-se usar a vírgula após esse valor.

Ex.:

```
fruta = ("Uva",)
```

PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

Funções min() e max()

As funções min e max são utilizadas para exibir os valores mínimo e máximo de uma lista ou tupla, independente de em qual posição estão esses dados.

Ex.:

```
lista = [1, 5, 2, 9]
```

```
print(min(lista))
```

```
print(max(lista))
```

Quais valores serão impressos?

PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

Funções min() e max()

As funções min e max são utilizadas para exibir os valores mínimo e máximo de uma lista ou tupla, independente de em qual posição estão esses dados.

Ex.:

```
lista = [1, 5, 2, 9]
```

```
print(min(lista))
```

```
print(max(lista))
```

Quais valores serão impressos?

1 e 9

PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

Funções min() e max()

Da mesma forma funciona listas com dados do tipo String, onde os valores mínimo e máximo serão os valores de ordem alfabética.

Ex.:

```
lista = ["Maria", "Ana", "João", "Pedro"]
```

```
print(min(lista))
```

```
print(max(lista))
```

Quais valores serão impressos?

PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

Funções min() e max()

Da mesma forma funciona listas com dados do tipo String, onde os valores mínimo e máximo serão os valores de ordem alfabética.

Ex.:

```
lista = ["Maria", "Ana", "João", "Pedro"]
```

```
print(min(lista))
```

```
print(max(lista))
```

Quais valores serão impressos?

Ana e Pedro

PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

Funções min() e max()

Sabemos que listas aceitam valores de tipos diferentes, inteiro, string, float, na mesma lista. Neste caso, qual o retorno das funções min e max no exemplo?

```
lista = ["Maria", 1, "João", 3]
```


PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

Funções `min()` e `max()`

Sabemos que listas aceitam valores de tipos diferentes, inteiro, string, float, na mesma lista. Neste caso, qual o retorno das funções `min` e `max` no exemplo?

```
lista = ["Maria", 1, "João", 3]
```

ERRO DE EXECUÇÃO

PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

Função sum()

A função sum() é bastante útil quando trabalhamos com soma de valores. Ela realiza a soma de todos os valores de uma lista apenas com a sua chamada.

Ex.:

```
lista = [1, 5, 6, 2]
```

```
print(sum(lista))
```

resultado: 14

PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

Função len()

A função len() é bastante utilizada quando precisamos trabalhar com condicionais e laços de repetição. Ela retorna o tamanho da lista (quantidade de valores da lista).

Ex.:

```
lista = [1, 5, 6, 2]  
print(len(lista))  
resultado: 4
```

Exemplo de uso:

```
for i <= len(lista):  
    print(i)
```

PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

Função sorted()

A função sorted() serve para ordenar uma lista, seja ela por ordem alfabética, como ordem crescente numérica.

Ex.:

```
lista = [1, 5, 6, 2]
```

```
print(sorted(lista))
```

```
resultado: [1, 2, 5, 6]
```

PYTHON DICIONÁRIOS

São coleções desordenadas de itens. Enquanto outras coleções, como listas e tuplas, são indexadas por uma faixa de números, os dicionários são indexados por chaves, que podem ser de qualquer tipo imutável. Cada par chave-valor em um dicionário é separado por vírgulas e todo o conjunto é colocado entre chaves {}.

Ex.:

```
dicionario = {'nome': 'Alice', 'idade': 25}
```

Acessando os valores:

```
print(dicionario['nome'])
```

Saída: Alice

PYTHON DICIONÁRIOS

Com o uso de dicionários, podemos atribuir várias chaves/valor, por exemplo, podemos ter vários dados de um cliente em um único dicionário.

Ex.:

```
clientes = {  
    1001:{'nome': 'Demetrius', 'idade': 37},  
    1002:{'nome': 'Alice', 'idade': 25}  
}
```

Podemos acessar esses dados de algumas formas diferentes, seja todos os dados do dicionário, os dados apenas de um cliente, ou até um dado específico de um cliente.

PYTHON DICIONÁRIOS

Acessando todos os dados:

```
print(clientes)
```

Saída:

```
{1001: {'nome': 'Demetrius', 'idade': 37}, 1002: {'nome': 'Alice', 'idade': 25}}
```

Acessar os dados de apenas um cliente:

```
Print(clientes[1001])
```

Saída:

```
{'nome': 'Demetrius', 'idade': 37}
```

PYTHON DICIONÁRIOS

Acessando apenas um dado de um cliente:

```
Print(cliente[1001]['nome'])
```

Saída:

Demetrius

PYTHON

FUNÇÕES PARA LISTAS E TUPLAS

FAÇA

Crie uma lista com os valores (23, 32, 64, 18, 9, 51, 67, 90)

Os ordene em ordem crescente e imprima

Os ordene em ordem decrescente e imprima

Imprima o valor da soma de todos os valores

Imprima o tamanho da lista

Imprima o último valor

PYTHON DICIONÁRIOS

Exercícios

Crie um programa que tenha dois dicionários sendo eles, feminino e masculino.

Receba os dados de clientes (nome, sexo e idade), até que o usuário não queira mais receber dados (pelo menos 10 clientes)

Se for do sexo feminino, inclua os dados no dicionário feminino. O mesmo para o masculino.

Por fim, informe todos os dados cadastrados para cada sexo.