Curso de Python

Termos

Print: comando para aparecer na tela

ex: Print(‘Ola Mundo!)

Variáveis só precisam colocar o =

ex: a = 1

Input: serve para receber algo do usuário, voce precisa colocar o tipo que ira receber do usuário:

ex: n1 = int(input(‘Digite um número’)

Pode se criar scrips para deixar uma programação pré definida

Para descobrir o tipo de caractere pasta usar o Type

ex: type(n.isalnum())

O isalnum()método retornará True se todos os caracteres forem alfanuméricos, ou seja, letra do alfabeto (az) e números (0-9).

Exemplo de caracteres que não são alfanuméricos: (espaço)!#%&? etc.

Números:

Int:Número inteiro

Float:numero com virgula

Bool: e valor verdadeiro e valor falso

Str: Palavra

.format para mostrar o que esta entre chaves

Operações aritméticas:

+ (adição)

- (subtração)

\* (multiplicação)

/ (divisão)

\*\*(potência, elevado ao numero)

// (divisão inteira)

% (resto da divisão)

== (igual)

Ordem de precedência

1° ( )

2° \*\*

3° \* / // %

4° + -

Para juntar ‘prints’ na mesma linha basta colocar end=’’

 E para quebrar uma linha \n pode se usar em qualquer lugar do código

Import: E para importar bibliotecas para usar no código

From: e para escolher uma importação em especifico

Math: e a biblioteca para usar matemática

Funcionalidades da biblioteca **math**:

Ceil : faz arredondamento para cima

Floor: faz arredondamento para baixo

Trunc: ele elimina o que vem depois da virgula

Pow: potencia

Sqrt: para calcular a raiz quadrada

Factorial: para calcular o fatorial

Hypot calcular o valor da hipotenusa

Cos,sin,tan: para calcular o cosseno seno e tangente, e para calcular precisa colocar o math.radians( ) para calcular em radiano

Random: gera números aleatórios,

\033[style; text;back m , serve para colocar cores no python

|  |  |  |
| --- | --- | --- |
| style | Text(cor da letra) | Back(cor de fundo) |
| 0 = nada | 30 = branco | 40 = branco |
| 1 = negrito | 31 = vermelho | 41 = vermelho |
| 4 = sublinhado | 32 = verde | 42 = verde |
| 7 = inverte as condiçoes | 33 = amarelo | 43 = amarelo |
|  | 34 = azul | 44 = azul |
|  | 35 = roxo | 45 = roxo |
|  | 36 = azul claro | 46 = azul claro |
|  | 37 = cinça | 47 = cinza |

**Manipulando cadeias de texto**

**SEMPRE COMEÇA EM 0**

**[**x**:**x**]**  fatiamento de string usa para pegar uma lista e terminar uma antes dela, aonde começa e aonde termina um antes

**[**x**:]** ele pega ou do inicio até o final (quando você coloca aonde ele vai terminar) ou ele coloca aonde você quer q ele pegue ate o final.

 Analise

**Len( XXX ):**é para saber o tanto de caracteres que existem.

xx**.count**(x): serve para contar o tanto de caracteres que existe na frase

frase = Curso em Video Python

ex. frase.count(‘o’)

3

xx**.find**(‘xxx’)Serve para mostrar aonde encontrou o que ele pediu

aonde não existe ele vai te mostrar -1(significa que não tem)

**in**:e para saber se existe algo na frase

ex.’Curso’ in frase

true

e o not in e ao contrario

Para printar as aspas “” basta usar o \” que no print ira aparecer as aspas.

Transformação

xxx.**replace**(xxx.o que você quer)-: serve para substituir a palavra que você quer pela palavra que você quer.

xxx**.upper()**: ele vai transformar o que você quer em letras maiúsculas

xxx**.lower()**: ele vai transformar o que você quer em letras minúsculas

xxx**.capitalize():** ele vai pegar toda a sua frase e transformar para minúscula , **menos a primeira letra**

xxx**.title()** : ele pega palavra por palavra e coloca com a primeira letra maiúscula.

xxx**.strip()**: serve para remover os espaços antes e depois da sua frase

xxx**.rstrip()**: remove todos os espaços em brancos da direita(right)

xxx**.lstrip()**: remove todos os espaços em branco da esquerda(left)

xxx**.split()**: serve para dividir onde existem os espaços em branco.

‘X’**.join(**xxx**):** serve para adicionar (juntar) as palavras que estão separadas com o caracter que a pessoa escolher.

Condição

If e els

**If**xxx.xxx**():**

**Xxxxxx**

**Xxxxxx**

**Else:**

**Xxxxx**

**Xxxxx**

*Condições aninhadas*

If xxx

Elif

Xxx

Else

Xxx

Estruturas de repetição

For

Range é para fazer contagens

para fazer um laço com for precisa de uma estrutura para “parar” o programa ex

for c in range(0,3):

xxxx

ele vai fazer isso de 0 ate 3

Na contagem regressiva você coloca -1  no final

Ex:

for c in range(0, 10, -1)

ele não considera o ultimo número.

voce pode utilizar tambem a contagem por passo ex

for c in ragne(1, 10, 2)

ele vai contar de 1 até 10 pulando de 2 em 2

While pode ser referido como um ‘enquanto’

Se usa quando não se sabe o limite q vai ser usado, então deve ter uma condição.

regras, precisa ter uma

inicialização de variavel de controle

condição( teste logico)

e os comandos dentro do while

ex:

i = 0

while i < 10:

  print(i)

  i +=1

print("Fim do Loop")

Quando é para parar um laço basta colocar o break em qualquer lugar

ex:

i = 0

while i < 10:

  print(i)

  i +=1

  if (i>=5):

    break

print("Fim do Loop")

Variável composta é aquela que pode ser armazenada mais de uma opção e é possível ter variável composta com tuplas, listas ou dicionários

Tuplas: as tuplas são imutáveis  ou seja, não pode ser mudada de jeito nenhum se é declarada ela fica armazenada pra sempre

Para aplicar as tuplas, basta criar a variável e colocar entre parênteses () o que a variável vai receber

A tupla pode ser apagada mas não mudada

Para apagar algo basta colocar  del(xxx)

Coisas que pode fazer dentro de tupla

Pode mandar printar algo em especifico em uma tupla, utilizando comandos de texto

pode se mostrar a quantidade de uma tupla utilizando o ‘len’

pode se utilizar as estruturas de repetições

Se você quiser mostrar a variável em ordem alfabética basta colocar sorted dps do print

—---------------------------------------------------------------------------------------------------------------------------

LISTAS

Para fazer listas basta colocar eles entre colchetes [ ].elas podem ser modificadas.

para adicionar itens na lista precisa se user o comando append(xxx)

ex:

lanche = [hambúrguer, suco, pizza, picolé]

lanche.append(cookie)

aqui ele vai adicionar o cookie no FINAL da lista

se você quiser adicionar algo em um local específico basta usar o insert

ex

lanche.insert(0,cachorro quente)

ele vai adicionar o cachorro quente na posição 0 e vai “empurrar” o restante da lista

para apagar elementos na lista basta usar o del ou pop(normalmente ele é utilizado para eliminar o último elemento mas você pode especificar utilizando o número da posição entre parênteses () ), ou .remove( você pode especificar algo que você vai remover da lista)

ex:

del lanche[3]

ou

lanche.pop(3) neste caso se voce não colocar nada dentro do parenteses ele vai remover o ultimo elemento da lista

ou

lanche.remove(‘pizza’)

nesses 3 casos vai remover a pizza

o recomendado é utilizar o if porque caso não exista ele vai dar erro

ex:

if ‘ELEMENTO’ in ‘nome da lista’:

xxx.remove(‘elemento’)

Você pode criar listas com range ex:

valores = list(range(4,11)) ele vai criar uma lista do 4 até o 10 lembrando q o ultimo elemento é eliminado

se quiser colocar os elementos em ordem basta usar o nome\_da\_variavel..sort()

se quiser em ordem ao contrario, decrescente basta usar o reverse=True

nome\_da\_variavel.sort(reverse=True)

para saber o tamanho de uma lista basta usar o len(nome da lista)

você pode fazer cópias de uma lista basta usar

nome da variável = list(nome da lista que vai ser copiada)

Voce também pode juntar listas, basta usar o + e criar outra variável ex:

nome da variável = lista um  + lista 2

o python faz uma ligação  entre listas então se for mudar algo em uma lista ela automaticamente vai ser mudada na outra, para resolver isso basta fazer uma cópia usando o : fatiamento ex

a = [2, 3, 4, 7]

b = a[ : ]

b[2] = 8

result

Lista A: [2, 3, 4, 7]

Lista B: [2, 3, 8 ,7]