Python3

# Números:

No Python vamos ter números de tipo inteiro e ponto flutuantes, ou seja, int e float, a regra serve como as regras normais de matemática, multiplicação depois adição!

## Variáveis

Variáveis são espaço na memória que vai guardar algum tipo de informação!

Não pode começar com número, não pode ter espaços, e não pode utilizar caracteres proibidos, eles são:

“.<>/?|\()!@#$%^&\*~-+

## Códigos aprendidos na aula:

- **type()** – utilizado para saber o tipo do objeto – ex: **type(1)** = int , **type(3.1)** = float

# Strings:

Tipo de objeto em python que vai armazenar texto, temos dois tipos para declarar uma string, uma com aspas simples e outra com aspas duplas.

ex: ‘olá’ e “olá”

o Python entende uma string como uma sequência de caracteres, junto com virgulas, pontos e espaços!

O porque o python aceita aspas duplas e simples:

vamos se dizer que estamos escrevendo em inglês, e quero dizer que sou um professor, eu escreveria I’m a teacher, se eu fizesse isso a string iria iniciar no M e não teria o fim, então nesses casos, utilizamos aspas duplas, e ficaria assim:

ex: “ I’m a teacher”

Mesma coisa serve se eu quisesse fazer uma citação, por exemplo: ‘A seguir uma “citação”’

Note que coloquei aspas duplas em “citação” e iniciei com aspas simples no ‘A e fechei na “citação”’

## Indexação:

Vamos já começar com um exemplo, vamos dizer que criamos uma var a que recebe Olá mundo:

a = Olá mundo

a indexação é como fosse uma arrey, a string e sequencial ou seja começa do 0, o O é 0, I é 1, á é 2, espaço é 3, m é 4, e assim por diante.

Agora vamos indexar!

Se eu quiser puxar tudo posso fazer de duas maneiras

Utilizando colchetes vazios  
ex: a[]

E utilizando os dois pontos  
ex: a[:]

O que isso quer dizer, ele quer pegar tudo antes dos dois pontos e tudo depois.

Com isso posso pegar os valores que eu quiser também, por exemplo se eu quiser pegar o O faço assim – a[0] = O

Se eu quiser pegar a palavra olá, posso fazer assim – a[0:2] = olá

Se eu quiser pegar a ultima letra poderia contar todos os valores até o último ou utilizar números negativos, como por exemplo: a[-1] = o

Ou a[-2] = d

Ou podemos fazer assim, não queremos pegar o último “o”, podemos fazer assim: a[:-2] = Olá mund

## Indexação com espaçamentos:

Com espaçamentos temos dois dois-pontos: a[::]

O segundo significa o espaçamento, de padrão ele é 1: a[::1]

Onde ele vai pegar valores de 1 por 1, que ficaria assim: a[::1] = Olá mundo

Posso pegar de 2 em 2, que ficaria assim: a[::2] = O uo

De 3 em 3 – a[::3]= Omo

E assim por diante!

Posso fazer também invertido com números negativos, ex –1 que seria de trás para frente!

## Propriedades de Strings:

Strings são imutáveis, não posso ficar alterando a mesma, por exemplo eu digo que a é igual a b e depois dizer que é c, isso gera um erro, ex: a=’b’ – depois digo a=’c’ , isso gera Error!

Mas strings são concatenáveis, posso junta-las

ex: a = ‘b’

a = a + “ “ +a

resultado seria: b b

se eu fizer uma string vezes um número

ex:

a =’b’

a \* 10

result: bbbbbbbbb

ele pega 10 vezes o b!

String são objetos que tem umas funções que são especificamente delas, como fazemos para acessa-las? Colocamos o ponto depois da string, ex:

str = ‘Olá Mundo’

str. *Aqui consigo acessar diversas funções que tem dentro dela, como por exemplo o lower(), onde deixa tudo em caixa baixa*

str.lower()

result: olá mundo

## Códigos aprendidos na aula:

- **print ()** = ele é um comendo utilizado para printar , **ex: print(**Olá mundo) = ira printar Olá mundo.  
posso utilizar dentro do pint o contra barra e n “\n” assim posso quebrar a linha  
*ex: print(olá \n Mundo):  
Olá  
 Mundo  
se eu colocar duas ficaria assim:  
ex: print(olá \n\n Mundo)  
Olá  
  
Mundo*ficaria com 2 espaços  
OBS: podemos usar também o \t que seria como fosse um tab , o espaçamento seria maior!

- **len()** = ele pega o tamanho total da minha string e me da o valor  
ex**: len(olá mundo) = 9**  
ele soma todas as letras, números, virgulas e espaçamentos