**AULAS #06 TIPOS PRIMITIVOS E SAÍDAS DE DADOS**

O python trabalha principalmente com 6 tipos primitivos

Int = inteiro

float = números reais com vírgulas

bool = booleano (True or False)

str = tipo string (frases e letras)

**Um novo print**

print(‘a soma vale{ }’.format(variavel))

utilizando o format podemos trabalhar de forma muito mais efetiva

**Descobrindo o tipo de uma variável**

Podemos fazer isso utilizando o comando *type(variavel)*

**Conversao de tipos**

Existem as funções

int(variavel)

float(variavel)

str(variavel)

bool(variavel)

Essas funções servem para converter as variaveis para os respectivos tipos primitivos

**Método is**

Podemos formar condições com o método is aplicado a uma variável, dentre ele podemos citar:

isupper(), isalnum(),isdigit(),islower(),isnumeric(),isspace() e etc.

variavel.isnumeric()

O retorno será verdadeiro ou falso (True or False)

**Formatando casas decimais**

Para deixar um número com duas casas decimais utilizamos dentro a notação {:.2f}, por exemplo:

print('A raiz cúbica do número é {:.2f}'.format(n5))

**Utilizando o end**

Ao final do print() podemos utilizar o end=’ ’ para não quebrar a linha

print('A raiz cúbica do número é {:.2f}'.format(n5), end=’ ’)

Em qualquer lugar do print() podemos utilizar o \n para quebrar a linha

print('A raiz cúbica do número é \n {:.2f}'.format(n5), \n)

A saída será:

A raiz cúbica do número é

4.99

**AULA #8 Utilizando Módulos**

**Biblioteca math**

Utilizando a biblioteca *math* podemos utilizar ceil(arredonda pra cima), floor(arredonda pra baixo), trunc(quebra o número após a virgulo), pow(faz a potência), sqrt(faz a raiz quadrada), factorial(faz o fatorial) e muitos outros elementos.

Podemos fazer de duas formas:

import math

importamos a biblioteca math

from math import sqrt

nesse exemplo importamos somente a funcionalidade de raiz quadrada

from math impor sqrt, pow

nesse exemplo importamos as funcionalidades de raiz quadrada e potência

**Biblioteca random**

Podemos importar o *random* para gerar numeros aleatórios

*random.random()* gera um número aleatório entre 0 e 1

*random.rendint(x , y)* gera um número aleatório inteiro entre x e y

**AULA #9 MANIPULANDO TEXTOS**

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

frase(9:21) sempre o último número é ignorado logo aqui vamos de 9 até 20

frase(9:21:2) vai do 9 ao 20 pulando de 2 em 2

frase(:5) vai do 0 ao 4

frase(15:) vai do 15 até o fim

frase(9::3) vai do 9 até o fim pulando de 3 em 3

**MÉTODOS IMPORTANTES**

*len(frase)* retorna o tamanho da frase (nesse caso 21 contando com o 0)

*frase.count(‘a’)* retorna quantas vezes ‘a’ minúsculo aparece na frase

*frase.count(‘a’,0,13)* faz a contagem do 0 ao 12

*frase.find(‘Curso’)* retorna a posição que a palavra “curso” começa, nesse caso é 0

Quando a string não existe, find() nos retorna o valor (-1)

*frase.replace(“Curso”,”Android”)* muda o nome *Curso* para *Android*

*frase.upper()* deixa tudo em UPPERCASE

*frase.lowe()* deixa tudo em lowercase

*frase.capitalize()* deixa tudo como minúsculo e somente a primeira letra maiúscula

*frase.title()* deixa toda palavra em capitalize

*frase.strip()* Remove os espaços desnecessários no início e no final da string

*rstrip* faz o mesmo só com a direita e *lstrip* com a esquerda

*frase.split()* transforma a frase em um vetor onde cada posição é formada por uma palavra

*frase.aplit(“#”)* divide os objetos do vetor por #

ex: nomes = “banana#melao#pera”

nomes.split(“#”) vai retornar [‘banana’ , ’melao’ , ’pera’]

para realizar a junção de elementos separados podemos utilizar o elemento.*join.frase()*

ex: [‘banana’ , ’melao’ , ’pera’]

“-”.join.frase() retorna banana-melao-pera

OBS:PARA IMPRIMIR TEXTOS NO TERMINAL USAMOS print(“”” texto gigante “””)