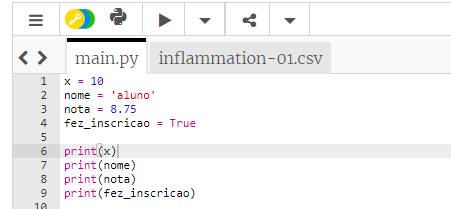
PYTHON

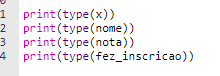
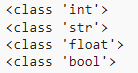
AULA 1

VARIAVEIS E TIPOS BASICOS DE DADOS

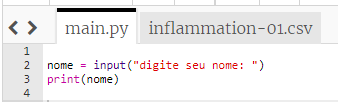
Dentro do python, não precisamos estipular o tipo de dado que uma variável vai receber. Lembrando que variável é composta por nome e a seu conteúdo, podendo ser número, número flutuante, palavras e resultado booleanos (falsos ou verdadeiros). Então no python, o próprio sistema consegue identificar, qual o tipo do conteúdo dentro da variável.



Na imagem acima, podemos notar que existem quatro variáveis, X, NOME, NOTA e FEZ\_ISCRICAO. Note que elas não têm definido o tipo das variáveis, pois dentro do sistema, ele mesmo consegue identificar. Para isso, basta a gente dar um PRINT(TYPE), estipulando a variável para identificar o seu tipo de conteúdo:



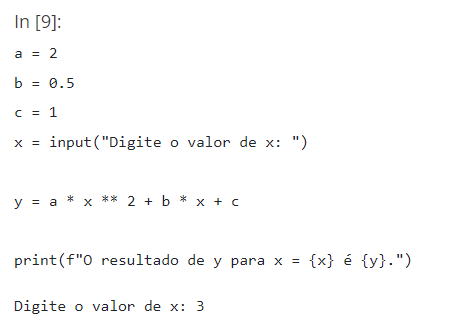
Outra questão é que, como se trata de uma linguagem orientada a objeto, tudo já vem com a tag CLASS, pois se trata de um objeto, ou melhor, de uma classe com atributo e métodos.

Dentro do phyton, o comando que fica responsável em o pedir para o usuário escrever dentro do sistema é o comando INPUT. Vamos pedir para que o usuário digite o nome dele e depois que o mesmo seja impresso na tela:  


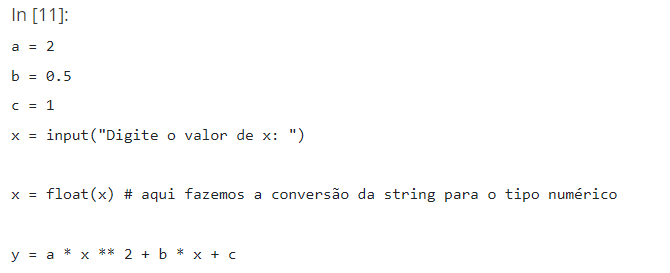
Simples e fácil.

OPERAÇÕES MATEMATICAS EM PYTHON.

Quando pedimos para o usuário digitar um numero para ser armazenado dentro de uma variável, o mesmo volta para o sistema com o tipo de STRING. Para mudarmos isso, repare no exemplo a seguir:

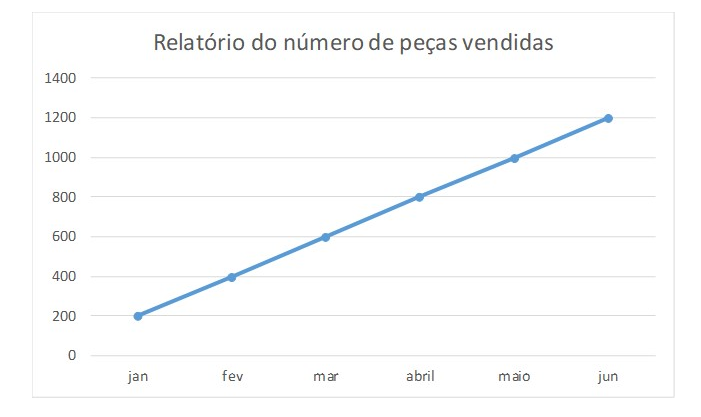
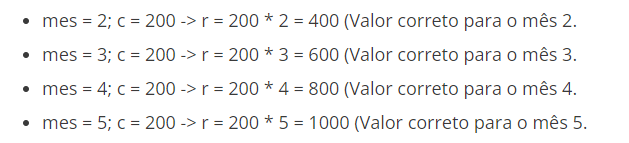


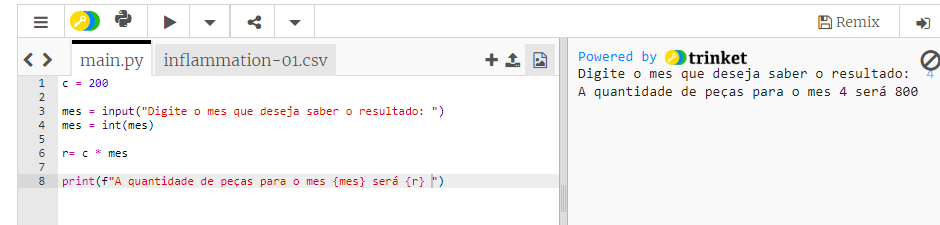
Estamos pedindo para que o usuário digite o valor X, só que, quando formos executar esse sistema, ele vai dar um erro, alegando que alguma variável, esta salvando informações que não servem para ela, que no caso, seria a variável X. Então, para que o sistema altere a variável para string, temos que fazer a seguinte mudança:



Temos que falar para o sistema que o X, assim que ele receber o valor digiteado pelo usuário, ela vai ser modificada para Float, deixando de ser String.

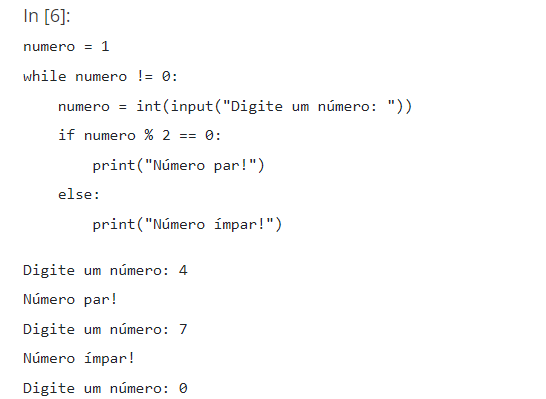
De resto, as operações padrões de matemáticas, continuam as mesmas.

EXERCICIO 1



ESTRUTURAS DE REPEPITÇÃO WHILE E FOR

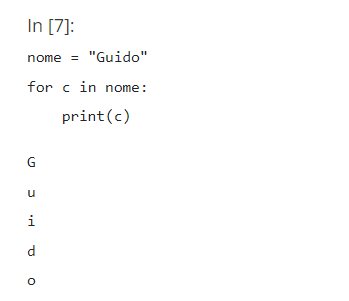
While (enquanto) seria basicamente igual as outras linguagens de programação, vamos dar algumas ordens para o sistema enquanto o usuário estiver digitando o valor diferente do que foi programada para parar. Então “enquanto x < 10, faça”, esse é o jeito de lermos um comando de WHILE:



No exemplo a cima, começamos primeiro com uma variável e o valor dela “numero = 1”. Depois, escrevemos a sequencia do comando WHILE “while número != 0”, que significa, “enquanto a variável numero foi diferente de zero, faça:”. Depois, em seguida dos dois pontos, damos início ao código sobre o while, que na linha, esta assim “numero = int(input(“Digite um número: “)) “a variável numero vai receber um número inteiro(int), digitado pelo usuário(input)”. Na quarta linha, damos a condição IF, “se a sobra da divisão (%) entre o numero salvo dentro da variável e dois, for igual (==) a zero, faça (:)”. Na quinta linha, estamos pedindo para que o sistema, se a condição for verdadeira, imprima na tela a frase “numero par”. E em seguida a condição inversa, se caso o valor da sobra da divisão for diferente de zero.

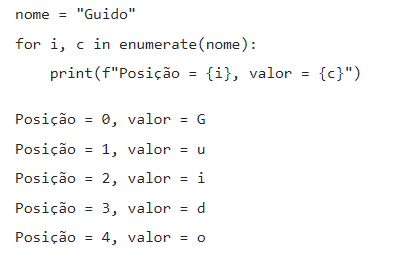
COMANDO FOR

O comando for dentro do python, server para mostrar alguma que está dentro de uma variável, mas de um jeito diferente. Por exemplo, com o comando FOR, da para sequenciar as letras de uma palavra:



Na imagem, estamos declarando uma variável do tipo string, contendo a palavra GUIDO. Em seguida, damos o comando FOR dizendo para ele utilizar uma variável de controle chamada C e ir acrescentando as letras da palavra nome dentro da variável de controle. Depois, pedimos para que ele imprima na tela a sequencia salva dentro da variável de controle.

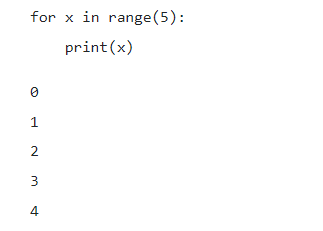
Dentro da função FOR, podemos usar a função ENUMERATE(), que server para numerar a sequencia dentro de uma variável de controle:



Na imagem acima, utilizamos o mesmo texto dentro da variável nome. Então declaramos a variável nome recebendo o texto GUIDO. Depois, dizemos para o sistema criar uma variável de controle, variável i, para enumerar a sequência e outra para guardar as letras em sequência, variável c, e em seguida, usamos a função ENUMERATE com um parâmetro de entrada, que nesse caso, é a própria variável NOME em questão, pois é dela que ele vai enumerar e salvar dentro das variáveis estipuladas. Depois, pedimos para ele printar essas duas variáveis na tela, com a função F concatenando as variáveis com o texto.

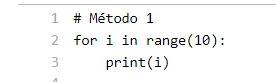
CONTROLE DE REPETIÇÃO COM RANGE, BREAK E CONTINUE.

Ainda no comando FOR, temos outras funções utilizadas junta a ele. Por exemplo, suponhamos que queiramos que o sistema apresente uma sequencia de numero de um determinado valor a outro. Então, podemos utilizar a função RANGE. Que seria basicamente a mesma coisa que o ENUMERATE, com um parâmetro de entrada:

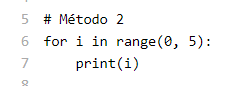


No comando acima, estamos dizendo para que o sistema utilize a variável de controle X com a função RANGE de duração até com 5 números e depois, pedimos para que ele imprima essa variável ATÉ o número 4. Lembre-se que o parâmetro de entrada é de 5 números, como não estipulamos nenhuma variável antes, com qualquer valor de início, ele vai começar a impressão no numero 0 e assim por diante, até chegar no quinto número.

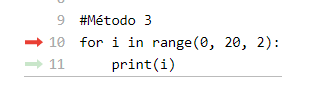
Podemos usar a função de RANGE de 3 formas:



No primeiro método, seria basicamente igual a o exemplo numero 1, com o parâmetro de entrada valendo 5. Então ele vai imprimir cinco números, do zero ao quatro.



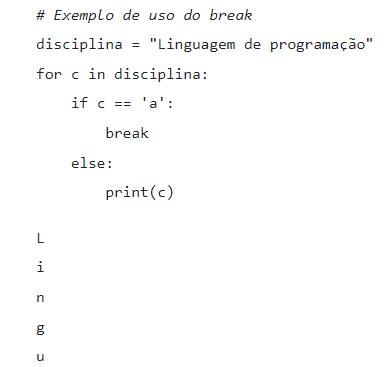
No segundo método, dentro do parâmetro de entrada, note que tem dois valores entre uma virgula, isso quer dizer que estamos falando para o sistema que na função RANGE, queremos que ele COMECE a contagem a partir do número 0. Então o resultado vai ser normal, igual os valores do primeiro método.



No terceiro método, a função seria basicamente a mesma, se não fosse a questão do incremento. Isso quer dizer que, o sistema vai começar do numero zero, vai até o vinte começando do zero, só que agora, ele vai de dois em dois, então vai ser, zero, dois, quatro, e assim por diante.

FUNÇÃO BREAK

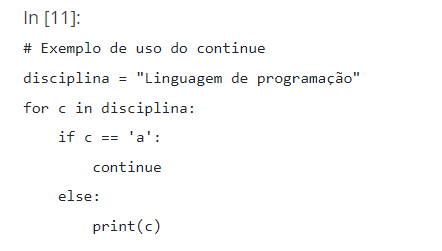
A função break, seria uma função para quebrar uma repetição quando a mesma chegar em uma determinada palavra ou número. Por exemplo, assim que ele identificar uma letra dentro de uma palavra, queremos que ele pare a execução:



Notem que declaramos uma variável com a palavra LINGUAGEM DE PROGRAMAÇÃO. Em seguida, colocamos ela dentro de uma variável de controle com a função FOR, para que ele faça o serviço de listar essa palavra letra por letra. Só que agora, com uma condição com a função break. Então, enquanto a variável de controle não for igual a letra A, ele continua, caso ele ache uma letra parecida com o que foi estipulado, ele para o ciclo.

FUNÇÃO CONTINUE

A função CONTINUE tem quase a mesma característica da função BREAK, mas em vez de parar o sistema assim que a condição for feita ou achada, ela continua sem a mesma informação:



Então nesse exemplo a acima, ele vai listar todas as letras da palavra dentro da variável, MENOS as vogais A.