Operadores aritméticos:

+ subtração

- Adição

\*multiplicação

/ divisão

[\\](file:///\\\\) divisão inteira (sem números após a virgula)

% resto da divisão

\*\* exponenciação

**Ordem de precedencia:**

**Primeiro: ()**

**Segundo: \*\***  **/** **//** **%** **(executa de acordo com a ordem)**

**Terceiro: +** **-**  **(executa de acordo com a ordem)**

**Manipulando textos**

Toda variavel declarada sem um tipo (inteiro, string, ...) é tratada como um vetor

Cada letra de uma palavra é armazenada em uma posição por exemplo:

Texto de teste

A letra t de texto esta na posição 0 e a letra d esta na posição 5

print('''\n  
##########################################  
# aprendendo #  
# a editar textos em #  
# python #  
##########################################  
''')  
  
frase = 'Roniel Pereira Leite' # a string no python é como um vetor  
# 01234567891111111111  
# 0123456789  
print(frase[0:6])  
print(frase[7:14])  
print(frase[15:20])  
print(frase[0:20:2]) #pula de 2 em 2  
  
print('na frase tem {} posicões'.format(len(frase))) # len(variavel) retorna o tamanho da string  
  
print('a lertra e aparece na frase: ', frase.count('e')) # frase.count('e') conta a quantidade da letras entre aspas  
print('da posição 0 ate a 6 a letra e aparece: ', frase.count('e', 0, 6)) # frase.count('e') conta a quantidade da letras entre aspas da posição 0 ate a 6  
  
print('a palafra Perei comessa na posição {}'.format(frase.find('Perei'))) # frase.find retorna a posição que inicia a palavra entre aspas, se não encontrar a palavra retorna -1  
  
print('existe a palavra Leite na frase? {} '.format('Leite' in frase)) # 'Leite' in frase retorna true se for verdaddeiro e False se for falso  
  
print('trocar Roniel para João Miguel', frase.replace('Roniel','João Miguel'))  
joao = frase.replace('Roniel','João Miguel')  
print('{}\n{}'.format(frase, joao)) # só fica salvo em outra variavel  
  
print('\ntudo maiusculo {} \n ele vouta ao normau sozinho: {}'.format(frase.upper(), frase)) # só fica salvo em outra variavel  
  
print('\ntudo minusculo {} \n ele vouta ao normau sozinho: {}'.format(frase.lower(), frase)) # só fica salvo em outra variavel  
  
print('so a primeira letra da palavra em mauscula: {}'.format(frase.capitalize()))  
  
print('so a primeira letra de cada palavra em mauscula: {}'.format(frase.title()))  
  
frase.strip() # tira espaços e excesso no inicio e no final da frase  
frase.rstrip() # tira espaços e excesso no final da frase  
frase.lstrip() # tira espaços e excesso no inicio da frase  
  
separa = frase.split() # separa a frase por palavras  
# virou um vetor e cada posição possui uma palavra  
print('posição 1 e igual a: {} '.format(separa[1]))  
  
trasso = '->'.join(separa) # posso colocar qualquer coisa entre os espaços  
  
print(trasso)

## Operações booleanas — and, or, not

Esses são as operações booleanas, ordenados por prioridade ascendente:

| **Operação** | **Resultado** | **Notas** |
| --- | --- | --- |
| x or y | se x é falso, então y, do contrário x | (1) |
| x and y | se x é falso, então x, do contrário y | (2) |
| not x | se x é falso, então True, caso contrário False | (3) |

Para ver o conteudo completo:https://docs.python.org/pt-br/3/library/stdtypes.html