=======================================================

***Métodos de Desenvolvimento de Software - Python***

=======================================================

O conteúdo da prova é identificar o que ocorre no código e explanar o que está ocorrendo no código.

=======================================================

# ***If – Elif – Else***

Como já sabemos o comando **If –** **Else** serve para fazer uma verificação no código e se determinada expressão atende à essa condição. A partir do resultado, uma ou mais instruções são executadas.

Já em **Python** temos a opção de usar o comando **Elif** que é utilizado quando queremos realizar a verificação de outra expressão caso a primeira validação seja falsa. Ou seja, adicionamos mais passos de verificação no código.

**Ordem de Execução:**

**If –**  **Elif** (Quantas vezes forem Necessárias)  **–**  **Else** (Por Último)

# ***While***

O laço **While** executa um ou mais comandos até que uma condição seja atendida, onde uma expressão é avaliada a cada **loop** e quando o resultado dessa condição passa a ser falso, a execução do **loop** é interrompida.

# ***Range ()***

Essa função retorna um conjunto de números sequenciais conforme os parâmetros definidos ao utilizarmos um comando.

**Exemplo de Usos:**

|  |
| --- |
| **range(stop)** |
| Corresponde à sintaxe com o parâmetro obrigatório referente ao ponto de parada. |
| **range(start, stop)** |
| Sintaxe com a definição do valor inicial e do ponto de parada. |
| **range(start, stop, step)** |
| Sintaxe com a adição dos parâmetros opcionais start e step, além do stop — que é obrigatório. Step indica qual valor será utilizado para incrementar a sequência numérica, é um parâmetro opcional. |

# ***Random ()***

Puxa os dados da biblioteca **Random**, utiliza a função **Randint** que gera números aleatórios, entre aspas fica o intervalo, e **Range** serve para retornar uma lista de uma lista de números inteiros.

**Ex:**

**import Random** (Importa a Biblioteca Random)

**nList = [random.randint(1, 100) for i in range(10)]**

# ***Len ()***

**Len()** é utilizada para obter o número de itens em um determinado **Objeto, String, Array, Listas, entre outros.**

Quando aplicamos a função **Len()** em uma **String** por exemplo, ela retorna à quantidade de caracteres que possui, ou seja, o comprimento. Uma de suas maiores utilidades é a verificação de itens em uma lista extensa, já que é possível utilizar a função **Len()** para a contagem de caracteres de uma lista.

# ***Upper ()***

**Upper** retorna a **String** com todas as letras em **Maiúsculo.**

**Ex: name\_maiusculo = name.upper()**

# ***Lower ()***

**Lower** retorna uma **String** com as palavras em **Minúsculo.**

**Ex: name\_maiusculo = name.lower()**

# ***Split ()***

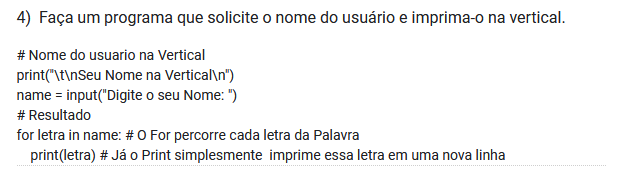
**Split()** transforma uma **String** em um item da lista.

# ***Replace ()***

O método **Replace()** é utilizado para substituir um ou mais trechos em uma string. Ele contém parâmetros para auxiliar a forma de substituição desse conteúdo.

**Ex: cpf = cpf.replace(".", "").replace("-", "")**

# ***Isdigit ()***

* **Isdigit()** retorna **True** se todos os caracteres forem dígitos, caso contrário, **False.**