

GUIA DE ESTUDOS - PYTHON

ILUSTRAÇÕES DAS PRINCIPAIS FUNÇÕES

AO FINAL, SEGUIR O LINK DA VIDEO AULA SOBRE CADA ASSUNTO

DICAS IMPORTANTES

- **(DICAS PARA MELHORAR E FIXAR O CONHECIMENTO):**
- **Sempre veja o (PDF) quando for fazer qualquer exercício**
- **Digite os códigos ao invés de copiar e colar**
- **Reveja as (videos aulas) 2x**
- **Tente explicar para alguém o código**
- **Faça pequenos (resumos)**
- **Python só funciona com (INDENTAÇÃO)**

- **SEMPRE QUANDO OCORRER UM (ERRO) (VERIFIQUE) :**
- **Indentação**
- **Importar (bibliotecas) - FALTANTES**
- **Nome de variaveis – Erradas na lógica**
- **Tipagem de dados – Divergentes**
- **Virgulas ou caracteres – Em locais errados**
- **Funções erradas – Caracteres errados**

IF / ELIF / ELSE - CONDICIONAIS

```
1  idade = 18
2  if idade >= 18:
3      print('maior de idade')
4  else:
5      print('menor de idade')
```

```
1  idade = 18
2  if idade < 12:
3      print('crianca')
4  elif idade < 18:
5      print('adolescente')
6  elif idade < 60:
7      print('adulto')
8  else:
9      print('idoso')
```

WHILE - LOOP


```
while True:

    num = int(input('de (0 ate 5) - Digite o numero 5 '))

    if num >= 0 and num <=5: # (de 0 ate 5)
        if num == 5:
            print('Parabens')
            break # sai do loop
        else:
            print('ainda não é 5')
            continue # volta ao começo do loop

    else:
        print('Errado, digite apenas de (0 ate 5) ')
        continue # volta para o começo do loop
```

```
i = 0
while True:
    print("Hello World")
    i+=1
    if i == 10:
        break
```



Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World

OBS : Sempre precisa ser atendida a condição , se não o loop (NUNCA PARA DE RODAR)

FOR - LOOP

```
lojas = ['Rio de Janeiro', 'São Paulo', 'Belo Horizonte', 'Curitiba']  
  
for loja in lojas:  
    print(loja)  
print('Acabou o FOR')
```

Rio de Janeiro
São Paulo
Belo Horizonte
Curitiba
Acabou o FOR

```
for i in range(4):  
    print(i)  
    print(lojas[i])
```

→ Posição

0
Rio de Janeiro
1
São Paulo
2
Belo Horizonte
3
Curitiba

→ Elemento

```
for x in "Daniel":  
    print(x)
```

D
a
n
i
e
l

LISTAS - I – TRABALHANDO COM ITENS

```
lista_compras = ['banana', 'laranja', 'maçã']  
print(lista_compras)  
['banana', 'laranja', 'maçã']
```

```
[banana , laranja , maçã]  
[ 0      , 1      , 2  ] -> Posições números positivos  
[ -3     , -2     , -1 ] -> Posições números negativos
```

```
print(lista_compras[0])  
banana
```

```
print(lista_compras[1])  
laranja
```

Para adicionar um item a lista:

- `.append()`: adiciona o item ao final da lista;
- `.insert()`: insere um item na lista na posição indicada

Para deletar um item da lista:

- `del`: remove um item da lista baseado na posição indicada;
- `.remove()`: remove um item baseado no seu valor e não na sua posição;
- `.pop()`: remove da **lista_compras** o último item, mas não o exclui.

Inserindo vários itens na lista

Outra forma de criar uma lista é utilizando o `while + input`, entrada seja vazia o programa irá armazenar as informações.

`while` atividade:

```
atividade = input(Insira uma atividade: '  
tarefas.append(atividade)
```

```
Insira uma atividade: Me inscrever no canal da Hashtag  
Insira uma atividade: Fazer o download do minicurso de Python  
Insira uma atividade: Compartilhar o vídeo da Hashtag  
Insira uma atividade
```

```
lista_compras.append('carro')  
print(lista_compras)  
['banana', 'laranja', 'maçã', 'carro']
```

```
lista_compras.insert(1, 'carro')  
print(lista_compras)  
['banana', 'carro', 'laranja', 'maçã']
```

```
del lista_compras[3]  
print(lista_compras)  
['banana', 'laranja', 'maçã']
```

```
item = lista_compras.pop(-1)  
print(item)  
carro
```

LISTA 2 – TRABALHANDO COM ITENS

```
lojas = ['Rio de Janeiro', 'São Paulo', 'Curitiba']  
vendas = [10000, 20000, 50000]
```

```
lojas.sort()  
vendas.sort()
```

```
print(lojas)  
print(vendas)  
['Curitiba', 'Rio de Janeiro', 'São Paulo']  
[10000, 20000, 50000]
```

Unindo listas no Python

Para unir duas listas podemos simplesmente utilizar o símbolo de +:

```
print(lojas+vendas)  
['Curitiba', 'Rio de Janeiro', 'São Paulo', 10000, 20000, 50000]
```

Veja o exemplo com uma lista numérica:

```
numeros = [1, 3, 5, 7, 9]
```

```
print(min(numeros))
```

#Resultado: 1

```
print(max(numeros))
```

#Resultado: 9

```
print(sum(numeros))
```

#Resultado: 25

Verificando a existência de itens nas listas Python (operador in)

Esse operador percorre todos os itens e retorna "True", se houver equivalência, e "False", se não houver o valor. Veja:

```
lista_compras = ['banana', 'laranja', 'maçã']
```

```
'banana' in lista_compras
```

True

```
'mamão' in lista_compras
```

False

DICIONÁRIOS – ITENS COM CHAVE E VALOR



```
emails_gerentes = {  
    "Iguatemi": "iguatemi@gmail.com",  
    "Plaza": "plaza@gmail.com",  
    "Barra": "barra@gmail.com",  
}
```

Se eu quiser descobrir qual o e-mail do shopping "Iguatemi"

```
email = emails_gerentes['Iguatemi']  
print(email)  
  
iguatemi@gmail.com
```

Se eu quiser adicionar um shopping novo

```
emails_gerentes['Leblon'] = "leblon@gmail.com"  
print(emails_gerentes)  
  
{'Iguatemi': 'iguatemi@gmail.com', 'Plaza': 'plaza@gmail.com', 'Barra': 'barra@gmail.com,', 'Leblon': 'leblon@gmail.com'}
```

Se eu quiser descobrir todos os shopping que temos?

```
# forma 1: fazer um for  
for shopping in emails_gerentes:  
    print(shopping)
```

```
Iguatemi  
Plaza  
Barra  
Leblon
```

```
# forma 2: dicionario.keys()  
print(emails_gerentes.keys())
```

```
dict_keys(['Iguatemi', 'Plaza', 'Barra', 'Leblon'])
```

Se eu quiser todos os e-mails?

```
# forma 1: fazer um for  
for shopping in emails_gerentes:  
    print(emails_gerentes[shopping])
```

```
iguatemi@gmail.com  
plaze@gmail.com  
barra@gmail.com,  
leblon@gmail.com
```

```
# forma 2: dicionarios.values  
print(emails_gerentes.values())
```

```
dict_values(['iguatemi@gmail.com', 'plaza@gmail.com', 'barra@gmail.com,', 'leblon@gmail.com'])
```


JSON = DICIONARIOS

```
# Importar o módulo
import json

# String em formato JSON
data_JSON = """
{
    "size": "Medium",
    "price": 15.67,
    "toppings": ["Mushrooms", "Extra Cheese", "Pepperoni", "Basil"],
    "client": {
        "name": "Jane Doe",
        "phone": "455-344-234",
        "email": "janedoe@email.com"
    }
}
"""

# Converter a string em JSON em um dicionário
data_dict = json.loads(data_JSON)
```

```
print(data_dict["size"])
print(data_dict["price"])
print(data_dict["toppings"])
print(data_dict["client"])
```



```
Medium
15.67
['Mushrooms', 'Extra Cheese', 'Pepperoni', 'Basil']
{'name': 'Jane Doe', 'phone': '455-344-234', 'email': 'janedoe@email.com'}
```

MANIPULANDO ARQUIVOS - PYTHON

Criar um arquivo vazio

Certifique-se de que o arquivo não exista, se não ele irá apagar seu conteúdo.

```
arquivo = open('novo-arquivo.txt', 'w')
arquivo.close()
```

Apagar o conteúdo de um arquivo

A mesma que a anterior.

```
arquivo = open('novo-arquivo.txt', 'w')
arquivo.close()
```

Ler todas as linhas em uma lista

```
arquivo = open('arquivo.txt', 'r')
lista = arquivo.readlines() # readlinesssssss
arquivo.close()
```

Ler linha a linha do arquivo

```
arquivo = open('arquivo.txt', 'r')
primeira_linha = arquivo.readline()
segunda_linha = arquivo.readline()
terceira_linha = arquivo.readline()
# etc...
arquivo.close()
```

Escrever em um arquivo

Se o arquivo já existir ele irá sobrescrever todo o conteúdo.

```
arquivo = open('novo-arquivo.txt', 'w')
arquivo.write('nova linha')
arquivo.close()
```

Inserir conteúdo ao já existente (adicionar)

```
arquivo = open('nome.txt', 'r') # Abra o arquivo (leitura)
conteudo = arquivo.readlines()
conteudo.append('Nova linha') # insira seu conteúdo

arquivo = open('nome.txt', 'w') # Abre novamente o arquivo (escrita)
arquivo.writelines(conteudo) # escreva o conteúdo criado anteriormente nele.

arquivo.close()
```

CLASSES

```
class ControleRemoto:  
    def __init__(self, cor, altura, profundidade, largura):  
        self.cor = cor  
        self.altura = altura  
        self.profundidade = profundidade  
        self.largura = largura
```

```
class ControleRemoto:  
    características:  
        - cor  
        - altura  
        - profundidade  
        - largura  
  
    métodos do controle remoto:  
        - passar de canal  
        - mexer no volume  
        - abrir a netflix  
        - desligar a tv
```

vamos criar uma classe para Clientes da Netflix

```
class Cliente:  
    def __init__(self, nome, email, plano):  
        self.nome = nome  
        self.email = email  
        lista_planos = ["basic", "premium"]  
        if plano in lista_planos:  
            self.plano = plano  
        else:  
            raise Exception("Plano inválido")
```

```
cliente = Cliente("Lira", "lira@gmail.com", "blabla")  
print(cliente.nome)
```

**MÉTODO
CONSTRUTOR**

**CONDIÇÕES DO
MÉTODO**

**PREENCHENDO OS
VALORES DAS VARIÁVEIS
DA CLASSE**

REGEX

Assunto Extenso – precisa ver video aulas

- `[]` – conjunto de caracteres;
- `\` – sequência especial de caracteres;
- `^` – buscar elementos no início da string;
- `$` – buscar elementos no final da string;
- `*` – buscar zero ou mais repetições de uma substring;
- `+` – uma ou mais aparições de uma substring;
- `?` – zero ou uma aparição;
- `|` – busca um caractere ou outro.

VIDEO AULAS - REFERENCIAS

- **IF/ELSE ==>** <https://www.youtube.com/watch?v=pfmzkkymZwo>

- **WHILE ==>** <https://www.youtube.com/watch?v=qloaHCfFdM4&t=52s>
- **FOR ==>** <https://www.youtube.com/watch?v=grHpdxUwCHc&t=15s>
- **LISTAS ==>** https://www.youtube.com/watch?v=7h58klJ7tWQ&list=PLOqx8yEV2-I2_QbDc4PNKeaOH49etj2ax&index=13
- **DICIONARIOS ==>** https://www.youtube.com/watch?v=L__xi1V4J0&list=PLOqx8yEV2-I2_QbDc4PNKeaOH49etj2ax&index=14
- **JSON ==>** https://www.youtube.com/watch?v=XXr75lCgzcA&list=PLOqx8yEV2-I2_QbDc4PNKeaOH49etj2ax&index=15
- **ARQUIVOS ==>** https://www.youtube.com/watch?v=5UV3qj64mHE&list=PLOqx8yEV2-I2_QbDc4PNKeaOH49etj2ax&index=16
- **CLASSES ==>** <https://www.youtube.com/watch?v=RhtsCbKyYoA&t=1177s>
- **REGEX ==>** https://www.youtube.com/watch?v=d2uqo6PhdM4&list=PLOqx8yEV2-I2ri3dX_ouZ0sqRU6c78kCB&index=1
- **REGEX ==>** https://www.youtube.com/watch?v=l7l2xxSVSi4&list=PLOqx8yEV2-I2ri3dX_ouZ0sqRU6c78kCB&index=2