



inteiro	int
real	float
caracter	str
logico	boolean

## Operador Significado

+	adição
-	subtração
*	multiplicação
/	divisão
%	resto da divisão ("mod")
**	exponenciação
//	divisão para numeros inteiros
R <sup>2</sup>	math.sqrt ou x**(1/2)

*Deve-se importar a Biblioteca // Função*

## Operador Significado

<	menor
>	maior
<=	menor ou igual
>=	maior ou igual
==	igual
!= ou <>	diferente

*→ Não é muito usado*

## Operador Significado

and e  
or ou  
not não

# abre para por um comentario

print("") Pode usar para escrever em quebra de linha e vai ser  
printado da mesma forma")

Comando no VisualG  
escreva  
nao saltar linha  
leia  
saltar linha

Comando em Python  
print()  
print("mensagem", end="")  
input()  
input("mensagem \n")

< ESQUERDA  
> DIREITA  
^ CENTRALIZADO

import  
from XXXX import xxyy

para importar bibliotecas  
para importar uma  
ferramenta especifica

se if  
se dentro do se elif  
senao else

enquanto while *loop infinito*  
para for n in range (0,5): *Aplicado condição*

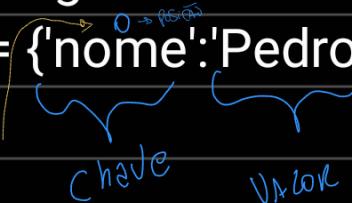
\* (0,5,2) assim ele pula de 2 em 2

## variaveis compostas no python

tuplas usa ( ) sao imutaveis y= tuple()  
listas usa [ ] sao mutaveis x= list() z= dict()  
dicionario usa { } sao mutaveis e podemos  
atribuir nome a cada espaço

#mais info ver programa dicionarios

ex : dados = {'nome':'Pedro', 'Idade':25}



The diagram shows the dictionary 'dados' with two key-value pairs. 'nome' is associated with 'Pedro' and 'Idade' is associated with '25'. Handwritten annotations with arrows point from 'nome' to 'Pedro' and from 'Idade' to '25', both labeled 'Chave' (Key) and 'Valor' (Value).

.upper() faz as letras ficarem todas MAIUSCULAS  
len(x) faz a contagem  
sort() mostra em ordem alfabetica A-Z ou 0-9  
sorted  
index(x) mostra a posição de x  
index(x,1) a partir da posição 1  
lista.append(x) adicionar um item a mais na lista  
lista.insert(0,x) o x assume a posição 0 e joga os  
demais para as posição da frente  
del lanche[3] deleta a pizza que estava la > usa a posição  
#elimina a posição inteira  
lanche.pop(3) deleta a pizza que estava la > usa a posição  
#fazendo com que se a lista  
lanche.remove(pizza) tem que especificar  
#ter 5 itens ela passa a ter 4  
lanche.pop() remove o ultimo item  
enumerate.valores ele retorna INDICE e VALOR (i,v)

filme.values()	ou (posição, valor/palavra) pega o que esta escrito no quadrado ex = 'star wars' 1977 'GeorgeLucas'
filme.keys()	ele pega o nome de cada quadrado ex = Titulo ano diretor
filme.items()	ele pega o valores do keys e items
estado.copy()	para fazer a copia de um dicionario
A[ ]=B[ ]	ela iguala a lista ate o final! se mudar a lista A muda a lista B
B[ ]=A[ : ]	para copiar a lista de A para B sem ter vinculo
sum	comando para soma ex sum(partidas) vai somar todos os valores que estao em partidas
valores.sort()	se valores for uma lista .... ele vai ordenar trocando de lugar
valores.sort(reverse=tTrue)	fazendo isso os valores ficam do maior para o menor
qualquel.isnumeric()	faz com que ele teste se é um numero ou nao
.isalpha()	faz com que ele teste se é alpha numerico
.replace( '.', ',' )	vai trocar todos os . por ,
help(input())	usa o help () para ter mostrar como a função funciona no caso a input
def rotina()	para declarar uma rotina nova #mais info no programa funções que nao existem no python

## COR DE FUNDO

'\33[m', # 0 → sem cor  
'\33[0;30;41m', # 1 → vermelho

'\33[0;30;42m', # 2 → verde  
'\33[0;30;43m', # 3 → amarelo  
'\33[0;30;44m', # 4 → azul  
'\33[0;30;45m', # 5 → roxo  
'\33[0;30m', # → branco

**MODULOS** é dividir o programa

EX → separar as funções DEF do código principal  
chama ele através do → import uteis  
from uteis import dobro

**PACOTES** o Python divide seus modulos em listas

EX → modulos só de números // strings  
datas // cores

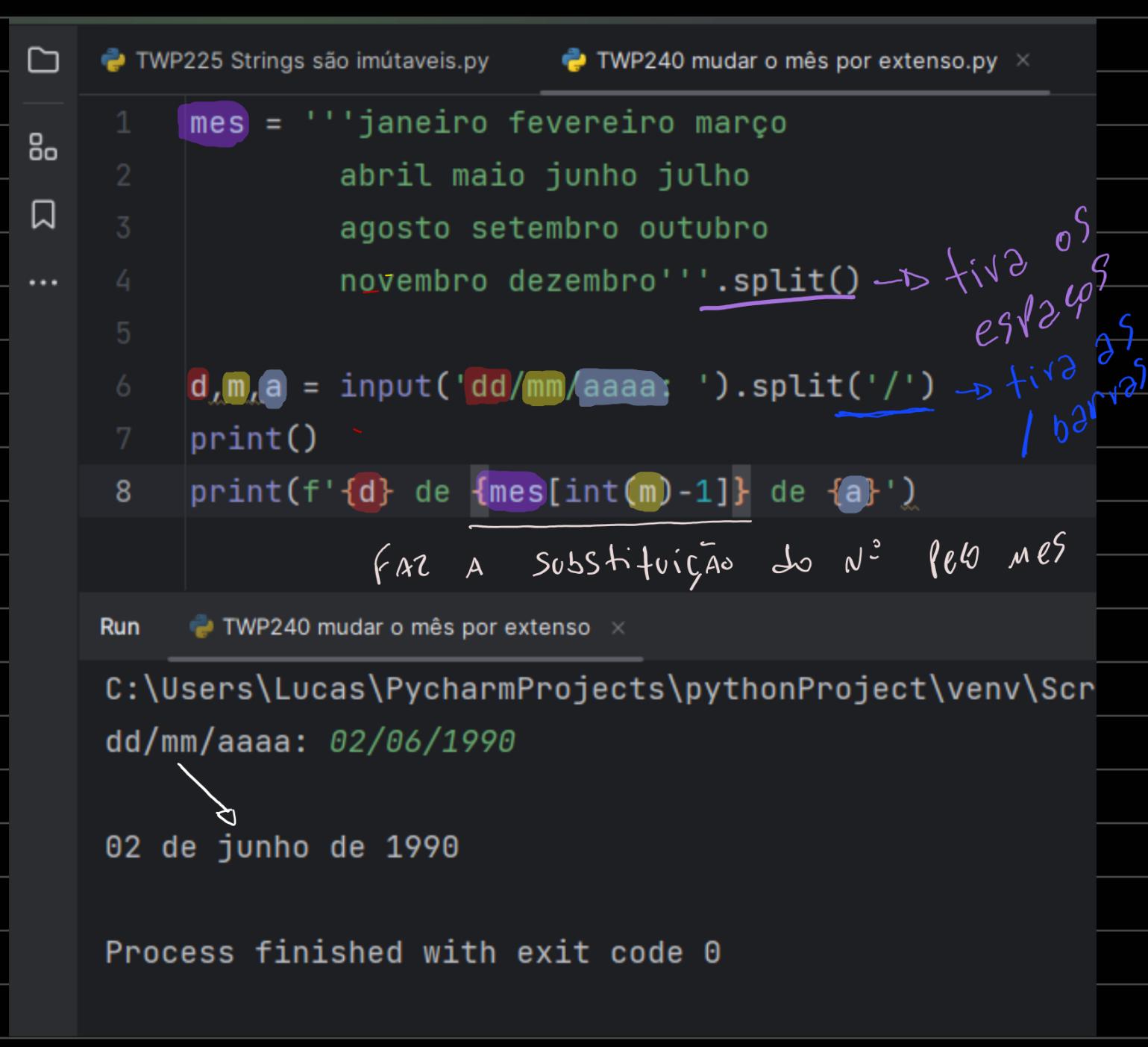
aplica a mesma forma para chamar ou todo o pacote ou  
por modulo

toda PASTA no PYTHON é considerada um **PACOTE** e pode  
colocar pasta sobre pasta

>>>>> OBS → ver solução do exercício 108 a 111 com o  
teste 2 e moeda 2 e moeda3 e teste 3

dir('abacate') me da todas as funções que tem  
para o caractere

help('abacate'.upper) me explica o que é feito com



```
mes = '''janeiro fevereiro março
abril maio junho julho
agosto setembro outubro
novembro dezembro'''.split()
d, m, a = input('dd/mm/aaaa: ').split('/')
print()
print(f'{d} de {mes[int(m)-1]} de {a}')
```

02 de junho de 1990

Process finished with exit code 0

.split()  
.shuffle()  
.choice()  
.sample((100), 10)

Cria uma lista dos intes  
mistura a lista  
faz uma escolha  
faz um sorteio podendo

Range → ↳ Números Sorteados  
esta dentro do def RANDOM

Lista IV exercício 2 outra.py - C:\Users\fmasa\Dropbox\Public\PPZPythonExercícios\Lista IV\Lista IV exercício 2 outra.py (3.8.2)

```
from random import sample
vetor = sample(range(100), 20)
par = [x for x in vetor if x % 2 == 0]
ímpar = [x for x in vetor if x % 2 == 1]
print ('Vetor', vetor)
print ('Pares', par)
print ('Ímpares', ímpar)
↳ forma de fazer um for direto na strin
```

↳ conteúdo um if.

Lista IV exercício 3 outra.py - C:\Users\fmasa\Dropbox\Public\PPZPythonExercícios\Lista IV\Lista IV exercício 3 outra.py (3.8.2)

```
from random import sample
v1 = sample(range(100), 10) ↳ aqui do nº
v2 = sample(range(100), 10)
v3 = []
for x in zip(v1, v2):
    v3.extend(list(x))
print('v1:', v1) ↳ APRENDE 2
print('v2:', v2) ↳ nº OU MAIS
print('v3:', v3)
```

↳ ex ↳

v3	v1	v2
3, 7	3	7
8, 4	8	4
9, 2	9	2

# Arquivos No Python

→ Olhar .py → Arquivos e dicionário

↳ leitura (r) → read  
↳ gravação (w) → write

RAW → Xira 2 funções do \

A

```
TWP225 Strings são imutáveis.py TWP240 mudar o mês por extenso.py Arquivos e Dicionario.py x
1 f = open(r'C:\Users\Lucas\PycharmProjects\pythonProject\aula de arquivos\x.txt', 'w')
2 for linha in range(1, 101):
3     f.write(f'{linha}\n')
4 f.close()
5
```

CRÍACAS DO TXT.

↳ character especial  
↳ tira o character do \n  
Pega linha por linha

NÃO tem  
O W

```
TWP225 Strings são imutáveis.py TWP240 mudar o mês por extenso.py Arquivos e Dicionario gravar arquivos e dicionario ler.py x
1 f = open(r'C:\Users\Lucas\PycharmProjects\pythonProject\aula de arquivos\x.txt')
2 for linha in f.readlines():
3     print(linha.strip())
4 f.close()
5
```

LEITURA DO TXT.

Outra forma de dar o comando é  
with open(r'C:\user.../x.txt') as f:  
print(f.read())

## Classes e Objetos

OBS: \_\_init\_\_ Dunder é o termo utilizado  
para os 2 underline seguidos —

Self. → objeto em si

↳ Sempre que criar um atributo do objeto devemos associá-lo ao Self.

↳ Caso contrário seria apenas uma variável lógica e não um atributo

**class Televisão:**

**def \_\_init\_\_(self):**

→ self.ligada = **False**

→ self.canal = **2**

**def muda\_canal\_para\_baixo(self):**

→ self.canal -= **1**

**def muda\_canal\_para\_cima(self):**

→ self.canal += **1**

**>>> tv = Televisão()**

**>>> tv.muda\_canal\_para\_cima()**

**>>> tv.muda\_canal\_para\_cima()**

**>>> tv.canal**

**4**

**>>> tv.muda\_canal\_para\_baixo()**

**>>> tv.canal**

**3**

**class Cliente:**

**def \_\_init\_\_(self, nome, telefone):**

self.nome = nome

self.telefone = telefone

**class Conta:**

**def \_\_init\_\_(self, clientes, número, saldo = 0):**

→ self.saldo = saldo

→ self.clientes = clientes

→ self.número = número

**def resumo(self):**

print('CC Número: %s Saldo: %10.2f' %  
(self.número, self.saldo))

**def saque(self, valor):**

if self.saldo >= valor:

→ self.saldo -= valor

**def deposito(self, valor):**

self.saldo += valor

*Lenho que Aprofundar Mais*

Existem VARIAS Funções quando For Fazer  
Algo tem que FAZER UMA BUSCA PARA Saber  
Se JA tem UMA FUNÇÃO CRIADA

<https://github.com/calazans10/python-para-zumbis>

↳ Site para acesso aos exercícios do  
Python para zumbis  
↳ Fotos da lista retiradas via Print

uso do **continue** em um laço de repetição faz com que o programa volte ao laço saltando tudo o que tem abaixo

→ While x:  
    X = x - 1  
    if x % 2 != 0: Sendo verdade  
        **continue**

Print (x) → Ignorado



