

Variáveis e tipos de dados

nomes de variáveis: começar em minúsculo e ser significativos
Ex.: nome, nomeUsuario, mediaFinal

tipos de dados:

- int valores inteiros Ex.: 78 -56 0 34567
- float valores reais (com parte fracionária) Ex.: 56.891
- string alfanuméricos Ex.: "este texto tem 23 linhas"
- boolean tem só dois valores: True ou False (verdadeiro ou falso)

variáveis em Python não precisam ser declaradas: o seu tipo é definido a partir da primeira atribuição

Ex.: num = 0.56 → num fica sendo do tipo float
num = 450 → num fica sendo do tipo int
time = "Inter" → time fica sendo do tipo string
result = True → result fica sendo do tipo boolean

Comandos

entrada de dados

```
umTexto = input("Digite um nome:")  
numInt = int(input("Digite um valor inteiro:"))  
numReal = float(input("Digite um valor real:"))
```

processamento (ou computação)

```
num = 10  
num = (nota1 + nota2) / 2  
palavra = ""  
# é uma string "vazia", sem caracteres  
# no processamento podemos ter também comandos condicionais  
# e de repetição, por exemplo.
```

saída de dados

```
print(num) # mostra valor da variável num  
print("num=", num) # mostra o texto "num=" seguido do  
# valor da variável num  
print("soma=", a+b) # mostra o texto "soma=" seguido do  
# resultado da expressão a+b  
print("a=" , a , " b=" , b)  
  
print("valor=%.2f " % valor) # %.2f formata a saída  
# do número para no máximo  
# duas casas fracionárias  
print("valor=%5.2f " % valor) # %5.3f formata a saída  
# do número para no máximo  
# 3 casas fracionárias e um  
# num espaço total de 5 dígitos  
print("valor=", valor, "end=") # não dá nova linha
```

comandos condicionais

```
valor = 13  
valModulo = valor % 2  
if valModulo == 0:  
    print("valor ",valor, "é par.")  
else:  
    print("valor ",valor, "é ímpar.")  
  
valor = int(input("Digite um valor:"))  
if valor == 0:  
    print("valor é zero.")  
elif valor > 0:  
    print("valor ",valor, "é positivo.")  
else:  
    print("valor ",valor, "é negativo.")
```

if expressão condicional :
else:
elif expressão condicional :

identação (tabulação)
determina o trecho que está dentro do comando

elif é else if

Operadores

aritméticos

+ - * / adição, subtração, multiplicação e divisão
** exponenciação
Ex.: num = 3 ** 4 → 81 (3⁴ que é 81)
% módulo (resto da divisão inteira)
Ex.: num = 7 % 4 → 3 (é o resto)
// divisão arredondando para baixo
Ex.: num = 7 // 4 → 1 (é 1.75 mas arredondando fica 1)

de atribuição

= Ex.: num = 3
+= -= *= /= %= //= **= Ex.: num += 2 equivale a num = num + 2

de comparação

> < == != maior que, menor que, igual, diferente
<= >= menor ou igual a, maior ou igual a
Ex.: if (a<10): é True se a for menor que 10

lógicos

and E Ex.: if (x>y) and (z==2) é True se x for maior que y E z for igual a 2
or OU
not negação (inverte o resultado da expressão lógica)
Ex.: if not (a <10): é True se a NÃO for menor que 10

Usos comuns para variáveis

Há algumas maneiras fundamentais de se empregar variáveis. Pode-se tratar uma variável como se ela fosse uma variável:

contadora (usando a variável para contar quantas vezes algo aconteceu)

```
maioresDeldade = 0  
i = 0  
while i<10:  
    idade = int(input("Qual idade?"))  
    if idade >= 18:  
        maioresDeldade += 1  
        i += 1  
print ("total de ", maioresDeldade, " pessoas maiores de idade.")
```

acumuladora (usando a variável para acumular valores)

```
somatorioidades = 0  
i = 0  
while i<10:  
    idade = int(input("Qual idade?"))  
    somatorioidades += idade  
    i += 1  
print ("Media das idades = ", somatorioidades / 10)
```

flag (usando a variável para sinalizar que algum evento aconteceu)

```
idadeinvalida = False  
i = 0  
while i<10:  
    idade = int(input("Qual idade?"))  
    if idade < 0 or idade >120:  
        idadeinvalida = True  
        i += 1  
if idadeinvalida:  
    print ("Foi fornecida idade inválida...")
```

comandos de repetição

while expressão condicional :

```
ij = 0  
while i < 100:  
    print("Valor de i:",i)  
    idade = int(input("Idade?"))  
    if idade >100:  
        break  
    if idade < 0:  
        continue  
    i += 1  
    print("esta é a última linha do laço")  
print("esta linha já está fora do laço")
```

Matemáticas

`import math` *primeiro é necessário importar a biblioteca*

`num = math.pi * 2` *PI é a constante 3.1415926536...*
`num = math.sqrt(25)` *sqrt é a raíz quadrada*
`num = math.sin(1)` *sin é o seno*

Strings *(não precisa importar nada...)*

`s = "internacional"`
`car = 'a'`
`car += 'b'` *concatena 'b' na string car*

`print("comprimento da string s = ", len(s))` *função len() retorna o comprimento da string*

`print("str=%20s"%s)` *insere espaços a esquerda ate completar 20 posicoes*
`print("str=%-20s"%s,".")` *insere espaços a direita ate completar 20 posicoes*

`print("str=%-20s"%s.upper(),".")` *insere espaços a direita ate completar 20 posicoes*
s.upper() retorna string em MAIÚSCULAS

`print("str=%-20s"%s[4],".")` *s[4] mostra o quarto caracter da string*
`print (s[2:5])` *mostra a substring, do índice 2 ao 5 (exclusivo)*

`if nome.lower() == "teste":` *nome.lower() retorna string em minúsculas*
`print("nome é teste.")`

`ss = s + "ee"` *concatena "ee" numa copia da string s e atribui o resultado a string ss*
`ss+="x"` *concatena "x" a string ss*

`a = "oi.sd"`
`x = a.replace(".",",")` *troca (substitui) caracteres . por ,*
`print("x=",x)` *imprime oi,sd*

`valor = 7.3456`
`texto = "valor=" + str(valor)` *a função str converte o parâmetro para string*
`print(texto)`

