

Declaração de variáveis e funções de entrada e saída

COMO DECLARAR VARIÁVEIS, UTILIZANDO A INDENTAÇÃO E OS COMANDOS QUE FAZEM A ENTRADA E A SAÍDA DE DADOS.

AUTOR(A): PROF. DENILSON JOSE SCHAFFER

Olá alunos,

Vamos ver como criar variáveis e como utilizar os comandos input e print.

DECLARAÇÃO DE VARIÁVEIS

As variáveis em Python não precisam necessariamente serem declaradas no início do código, como acontece em C++, e também não é necessário declarar o tipo.

Para nomear as variáveis:

- O primeiro caractere tem que ser um literal ou um underline _.
- Deve ser um nome contínuo, sem espaços em branco.
- Não deve-se utilizar acentuação gráfica, ç, caracteres especiais, exceto o _.

Exemplo:

Idade, idade, _idade, idade2, idade_Aluno

TIPOS DE DADOS EM VARIÁVEIS

int --> Números inteiros. Ex: ra = 6786324

float --> Números com partes fracionárias. Ex: salario = 2345.67

complex --> Números complexos. Ex: x = 5 + 6j

bool --> Booleano, retorna True ou False. Ex: x = True

string --> Caracteres. Ex: nome = "Denilson Schäffer" ou nome = 'Denilson Schäffer'

SINAIS DE ATRIBUIÇÃO

Para atribuir (carregar) um valor em uma variável podemos utilizar o sinal "=".

Exemplo:

X = 5

SINAIS DE COMPARAÇÃO DE IGUALDADE

Para comparar a igualdade entre valores de variáveis ou entre valores e variáveis, utilizando dois sinais de “==”.

Exemplos:

```
X == 5
```

```
X == Y
```

OPERADORES UTILIZADOS EM PYTHON

OPERADORES ARITMÉTICOS:

+	Adição
-	Subtração
*	Multiplicação
/	Divisão
//	Divisão parte inteira
**	Exponencial
%	Retorna resto da divisão

OPERADORES RELACIONAIS:

>	Maior que
<	Menor que
>=	Maior ou Igual
<=	Menor ou Igual
==	Igual
!=	Diferente

OPERADORES LÓGICOS:

and	Retorna verdadeiro se ambas ou mais expressões forem verdadeiras.
or	Retorna verdadeiro se uma das expressões forem verdadeiras.
not	Inverte a expressão se verdadeira passa para falso e vice-versa.

EXEMPLOS:

```
10 // 3 #retorna 3, considerando somente a parte inteira do resultado.
```

```
2 ** 3 #retorna 8, 2 elevado a 3, podemos utilizar também pow(2,3).
```

```
5 % 2 #retorna 1, que é o resto desta divisão.
```

```
>>> nome = 'Denilson'
```

```
>>> nome == 'Denilson'
```

```
True
```

```
>>> nome is not 'Denilson'
```

```
False
```

```
>>> nome != 'Denilson'
```

```
False
```

COMANDOS DE I/O (INPUT/OUTPUT)

Pré-definidos pela linguagem:

LER ==> Receber dados de um dispositivo de entrada como o teclado.

Em Python, utilizamos o comando: `input('Mensagem')`.

Exemplo:

```
ra = input("Digite o número do RA:")
```

OU

```
ra = int(input("Digite o número do RA:"))
```

ESCREVER ==> Envia dados de uma variável de saída para a tela.

Em Python, utilizamos o comando: `print("Mensagem")`.

Exemplo:

```
print("O número do RA é: ", ra)
```

*** A mensagem tipo texto é colocada entre aspas e colocando o nome da variável será impresso o valor da variável ra. A vírgula concatena a mensagem e o valor da variável.

EXEMPLO 1:

Desenvolva um script para receber as notas e calcular a média aritmética simples dos alunos, considerando que a média é composta por três notas:

```
#Cálculo da média simples de 3 notas
media = 0.0
nota1 = float(input('Digite a primeira nota: '))
nota2 = float(input('Digite a segunda nota: '))
nota3 = float(input('Digite a terceira nota: '))
media = (nota1 + nota2 + nota3)/3
print("O valor da média final é: ", media)
input('Pressione Enter para sair...')
```

Analisando o exemplo 1:

- A linha 1 é um comentário.
- Na linha 2 criamos a variável media e atribuímos um valor inicial 0.0 e como não precisamos declarar o tipo, o interpretador já irá determinar que a variável media é do tipo float.
- A partir da linha 3, já criamos três variáveis (nota1, nota2 e nota3), determinamos como float, utilizamos o comando input, exibindo a mensagem ao usuário: "Digite a nota:", o interpretador ficará aguardando o usuário entrar com o valor via teclado e o usuário ao digitar a nota, esta já será atribuída às variáveis nota1, nota2 e nota3.
- Na linha 6 calculamos a média.
- Na linha 7 imprimimos a mensagem: O valor da média final é: e o valor da variável media.
- A linha 8 é opcional, mas ao executarmos o script diretamente com duplo clique do mouse, o script executará e irá fechar. Para que possamos ver o resultado, colocamos esta linha de comando que ficará aguardando o usuário pressionar a tecla ENTER para encerrar a aplicação.

Vamos ver um exemplo, entrando com as notas: 6, 9 e 6 e verificando o resultado na mensagem da linha 4:

```
Digite a primeira nota: 6
Digite a segunda nota: 9
Digite a terceira nota: 6
O valor da média final é: 7.0
Pressione Enter para sair...
```

SAIBA MAIS...

Dê uma olhada nos links abaixo para saber mais sobre a linguagem Python:

<https://www.python.org/doc/> (<https://www.python.org/doc/>)

<https://wiki.python.org/moin/PythonBooks> (<https://wiki.python.org/moin/PythonBooks>)

Neste tópico vimos como declarar e trabalhar com variáveis em Python e também como utilizar os comandos de entrada de dados (input) e o comando de saída de dados (print).

ATIVIDADE FINAL

Quando escrevemos códigos em Python ou em qualquer outra linguagem, é importante escrever comentários para o bom entendimento dos blocos de instruções. Estas linhas de comentários serão ignoradas pelo interpretador. Qual das linhas abaixo corresponde a uma linha de comentário em Python:

- A. – isto é um comentário em Python.
- B. # isto é um comentário em Python.
- C. // isto é um comentário em Python.

Quais dos comandos abaixo representa a criação de uma variável determinando números inteiros como tipo:

- A. RA = str(input('Digite o RA: '))
- B. RA = int(input('Digite o RA: '))
- C. RA = input('Digite o RA: ')

Quais dos comandos abaixo representam a entrada e a saída de dados, respectivamente:

- A. print () e input ().
- B. input () e print ().
- C. enter () e print |().

REFERÊNCIA

SUMMERFIELD, M. *Programação em Python 3: Uma introdução completa à linguagem Python*. Rio de Janeiro Alta Books, 2012. 495 p.

MENEZES, N. N. C. *Introdução à programação com Python: algoritmos e lógica de programação para iniciantes*. 2. ed. São Paulo: Novatec, 2014. 328 p.

SWEIGART, AL. *Automatize tarefas maçantes com Python: programação prática para verdadeiros iniciantes*. São Paulo: Novatec, 2015. 568 p.

PYTHON, doc. Disponível em: <<https://www.python.org/doc/>>. Acesso em: Junho/2018.

PYTHON, books. Disponível em: <<https://wiki.python.org/moin/PythonBooks>>. Acesso em: Junho/2018.

