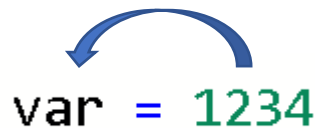


# Aula 01 – Operações Básicas, Variáveis e Tipos de Dados

## Atribuição

Em Python, as variáveis são criadas a partir de uma operação de **atribuição** que usa o sinal de igual (=). Na atribuição, a variável que está à esquerda do sinal de igual recebe o elemento que está à direita, alocando-o em uma porção de memória.



```
var = 1234
```

**Obs.** Não confundir atribuição com a operação relacional de igualdade ('==')

## Criação de Variáveis

Para a criação de uma variável, existem algumas regras importantes a serem seguidas.

- a) O nome de uma variável não pode conter espaços
- b) O nome de uma variável não pode começar com números
- c) O nome de uma variável não pode conter acentos
- d) É aconselhável padronizar o nome das variáveis

Uma variável comporta vários tipos de dados, podem ser textos, números, expressões matemáticas, equações, etc. Veremos esses tipos a seguir:

## Tipos de Variáveis

### 1. String

Todo tipo de dado textual em Python é chamado de string, seja esse dado uma letra, uma palavra, uma frase ou um texto inteiro. Isso é um pouquinho diferente de outras linguagens de programação, como o C, que chamam uma única letra de char.

Uma string pode ser declarada entre aspas simples ou aspas duplas. A declaração entre aspas simples exige o cuidado de adicionarmos um caractere de escape ('\') caso haja aspas simples dentro da frase. Para converter um dado em string utilizamos a função str().

```
print("tomar um copo d'água")  
print('tomar um copo d\'água')
```

## 2. Integer

Armazenam números inteiros (sem casas decimais). Para converter um número em tipo inteiro, usamos a função `int()`.

## 3. Float

Armazenam números quebrados, chamados de ponto flutuante. Para converter um número em tipo inteiro, usamos a função `float()`.

## 4. Boolean

Armazena valores booleanos (True ou False). Todo resultado de uma operação lógica em Python é um boolean. Para converter uma variável em bool usamos `bool()`

**Importante:** a função `input()` sempre considerará a entrada uma string. Então, caso queiramos fazer manipulações numéricas, precisamos converter esse número no tipo adequado. A função `print()` sempre converte os dados para string a fim de mostra-los na tela, por isso, não precisamos nos preocupar com o tipo de variável que passamos para ela.

### Operações Aritméticas

O Python pode funcionar como uma calculadora. As operações aritméticas nativas da linguagem (sem importar nenhuma biblioteca, estão indicadas abaixo).

Operação	Operador
adição	+
subtração	-
multiplicação	*
divisão	/

Operação	Operador
exponenciação	<code>**</code>
parte inteira	<code>//</code>
módulo	<code>%</code>

### Operações Lógicas e Relacionais

Operação Relacional	Operador
Igualdade	<code>==</code>
Diferença	<code>!=</code>
Maior ou Igual	<code>&gt;=</code>
Maior	<code>&gt;</code>
Menor ou igual	<code>&lt;=</code>
Menor	<code>&lt;</code>

Operação Lógica	Operador
Conjunção (E)	<code>and</code>
Disjunção (OU)	<code>or</code>
Negação (NÃO)	<code>not</code>

## Comentários

Para que o interpretador do Python (quem processa efetivamente o nosso código) ignore uma linha inteira. Basta adicionarmos o sinal de hashtag (#) na frente da linha. As linhas ignoradas são chamadas de 'comentário', pois elas geralmente servem para explicar alguma coisa no código para outros programadores. Para comentar várias linhas, usamos o marcador multiline que são as aspas 'tripas'.

## Estruturas de Dados em Python

Ao concatenar múltiplas variáveis em uma única variável mais ampla, estamos utilizando estruturas de dados. As estruturas de dados mais comuns em Python são:

1. **Listas:** conjunto de dados mutáveis declaradas com []
2. **Tuplas:** conjunto de dados imutáveis declaradas com ()
3. **Dicionários:** conjunto de chave-valor declarados com {:}
4. **Sets:** conjunto de dados únicos declarados com {}.

Veremos cada uma dessas em mais detalhes nas aulas futuras (após estruturas de decisão e repetição).

## Exercícios

1. Faça um programa que solicite a um estudante universitário o valor de suas quatro notas bimestrais e retorne ao mesmo a sua média de notas anual.
2. Faça um programa que calcule o aumento de salário de uma pessoa. Ele deve solicitar ao usuário o valor do salário e a porcentagem do aumento, retornando o valor do novo salário e o valor do aumento.
3. Escreva um programa que pergunte a quantidade de km percorridos por um carro alugado pelo usuário, assim como a quantidade de dias que o mesmo carro foi alugado. Calcule o preço final a ser pago, sabendo que o aluguel do carro é de R\$60,00 por dia e R\$0,15 por km rodado.