**AULA 3**

* **CRIAÇÃO DE VARIÁVEIS  
  -** Não podemos dar nomes para variáveis:  
  \* Que comecem com número  
  \* Que tenham espaço  
  \* Que tenham caracteres especiais (exceto “ \_ “)   
  \* Que sejam palavras reservadas da linguagem
* **SINTAXE  
  -**  Está ligado à FORMA  
  - Conjunto de regras da linguagem  
  - Por exemplo:  
  **\*** Criar variáveis com nome errado  
  \* Não abrir ou fechar um parêntesis  
  \* Esquecer de um determinado sinal ou símbolo  
  \* Escrever comandos de forma errada  
  \* Etc.
* **SEMÂNTICA  
  -** Está ligado ao SIGNIFICADO  
  - Por exemplo: O gato latiu muito ontem.  
  - Na programação, problemas de semântica podem ser:  
   **\*** Criar um inteiro e tentar atribuir outro tipo de dado  
   \* Dividir um número por algo que não é número  
   \* Usar um operador no contexto errado (comum com =)  
   \* Divisão por zero  
   \* Etc.

**\n 🡪**  Quebra de Linha  
**\t 🡪** Texto com tabulação no fim  
**%.2f % num 3 🡪** Formatação do número, irá virar String

* **BOOLEANOS**- VERDADEIRO E FALSO (True e False)  
  - São valores lógicos  
  - Ajudam na programação a tomar decisões!

# Em Python é sempre com a primeira letra maiúscula

**EXPRESSÃO RELACIONAL**

* Expressões relacionais permitem comparar dois valores através de operadores relacionais, resultando em **True** ou **False**
* **Exemplos:**

**-** x > 13 será **True** se o valor da variável *x* for superior a 13

**-** aux + valor == 10 será **True** se o valor da soma de *aux* com *valor* for 10

O resultado das operações relacionais **SEMPRE** será um **BOOLEANO**

**EXPRESSÃO LÓGICA**

* OPERADORES LÓGICOS EM PYTHON

|  |  |
| --- | --- |
| **OPERADOR** | **SIGNIFICADO** |
| **not** | Negação lógica (inverte o valor da expressão (exemplo: not op) |
| **and** | **e** lógico: será True se ambos os lados forem True |
| **or** | **ou** lógico: será True se pelo menos um dos lados for True |

**OPERADOR**

* **O operador % é chamado de MOD**
* **Retorna o resto da divisão inteira entre dois valores**
* **Por exemplo:**

**-** 4 % 2 🡪 0  
- 5 % 2 🡪 1  
- 10 % 3 🡪 1  
- 15 % 4 🡪 3  
- 22 % 9 🡪 4

* **Exemplos de uso:**

Como saber se um valor é par ou ímpar?

Como saber se um valor é divisível por outro?

**INSTRUÇÃO IF (SE)**

|  |
| --- |
| **CONDIÇÃO:** Qualquer valor ou expressão booleana **COMANDO:** Qualquer instrução válida da linguagem |
| Na instrução IF, somente se a condição for verdadeira (ou seja, com valor **True**), o(s) comando(s) serão executados. |
| Quais comandos estão dentro do IF no Python? Todos os que estiverem com a **tabulação** indicativa. |
| Quais comandos estão dentro do ELSE no Python? Todos os que estiverem com a **tabulação** indicativa. |

**Sintaxe no Python:**

If condição:  
 comando1  
 comando2  
 comando3  
 ...  
 comandoN

Else:  
 comando1  
 comando2  
 comando3  
 ...  
 comando

**INSTRUÇÃO ELIF**

**elif** significa “senão, se...” # Condição do Else