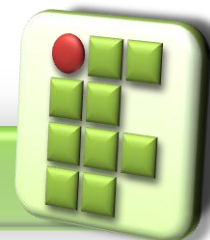


# Técnico Subsequente em Redes de Computadores

## Programação para Redes

### Aula 03 - Python (Parte 01)



# Por que estudar Python?

- Linguagem de Programação de Altíssimo Nível;
  - VHLL (Very High Level Language);
- Criada por Guido Van Rossum
  - “Programação de Computadores para Todos”
    - Liberdade (gratuita, código aberto);
    - Disponibilidade (disponível em diversos sistemas);
    - Clareza de sintaxe (resulta em alta produtividade);
- Permite orientação a objetos;
- Altamente modular ;
  - Módulos disponíveis para aproveitamento;



# Usando Interpretador Python

- IDLE
  - Ambiente Windows e Linux;
- Interpretador nativo
  - Linha de comando;



# Operadores Algébricos e Booleanos

- Operadores algébricos;
  - Adição (+)
  - Subtração (-)
  - Multiplicação (\*)
  - Divisão (/)
  - Exponenciação (\*\*)



# Operadores Algébricos e Booleanos

- Operadores Booleanos;
  - Igualdade (==)
  - Diferente (!=)
  - Maior (>)
  - Menor (<)
- Comentários
  - Precedido de #



# Usando Interpretador Python

```
>>> a=1
>>> b=2
>>> a == b
False
>>> a > b
False
>>> a < b
True
>>> a != b
True
>>> 2*a == b
True
>>> a == b/2
True
```



# Variáveis

- Linguagem dinamicamente tipada;
  - Saber o tipo de determinada variável;
    - `type(x)`, retorna o tipo do objetos `x`;

```
>>> valor=100
>>> valor
100
>>> type(valor)
<class 'int'>
>>> valor= "bola"
>>> valor
'bola'
>>> type(valor)
<class 'str'>
```



# Variáveis

- Principais Tipos de Objetos
  - Inteiros
  - Floats
  - Strings
  - Listas
  - Tuplas
  - Dicionários





# Variáveis

```
>>> numero = 10.45
```

```
>>> type(numero)
```

```
<class 'float'>
```

```
>>> int(numero)
```

```
10
```

```
>>> numero
```

```
10.45
```

```
>>> str(numero)
```

```
'10.45'
```

```
>>> type(str(numero))
```

```
<class 'str'>
```



# Tipos de Objetos

## ● Strings

- Trabalhar com texto;
- São tratadas como sequências;
- Para requisitar um valor atribuído a um endereço basta
- $S[x]$ , onde  $S$  é a sequência e  $x$  a posição

```
>>> palavra="dinâmica"
```

```
>>> palavra
```

```
'dinâmica'
```

```
>>> word='dinâmica'
```

```
>>> word
```

```
'dinâmica'
```

```
>>> print(word)
```

```
dinâmica
```

```
>>> word[2]
```

```
'n'
```

```
>>> word[0]
```

```
'd'
```

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |    |   |    |   |    |   |    |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----|---|----|---|----|---|----|
|   | t |   | e |   | r |   | m |   | o |   | d |   | i |   | n |   | â |   | m |    | i |    | c |    | a |    |
| 0 |   | 1 |   | 2 |   | 3 |   | 4 |   | 5 |   | 6 |   | 7 |   | 8 |   | 9 |   | 10 |   | 11 |   | 12 |   | 13 |

Como retornar 'modin'?

\*É possível omitir o início e o fim do intervalo, o que ocorre?

