



**INSTITUTO  
FEDERAL**

Paraíba

---

Campus  
Cajazeiras

# PROGRAMAÇÃO ORIENTADA À OBJETO

PROFESSOR MICHEL  
CAJAZEIRAS / IFPB

## Encapsulamento - Métodos REPR e STR

# OBJETIVOS

- ▶ Continuação dos estudos da estrutura de uma classe, seus métodos e atributos.
- ▶ Estudar os métodos especiais `__repr__` e `__str__`
- ▶ Estudar a entradas de parâmetros indefinidos

# ROTEIRO

- ▶ Introdução ao POO
  - ▶ Estrutura de uma classe
  - ▶ parâmetros indefinidos
  - ▶ métodos especiais `__repr__` e `__str__`

# Passando vários argumentos para um método

▶ Quando precisamos passar uma indeterminada quantidade de parâmetros para uma classe, método ou função, temos basicamente duas opções de uma coleção de forma posicional ou por chaves.

▶ Para isso usamos o operador de descompactação ( `*` ) para argumentos posicional e ( `**` ) para uso com chaves.

▶ fazer um exemplo com o posicional ( `*` ) tuplas

# métodos especiais `__repr__` e `__str__`

- ▶ Existe dois métodos especiais para saída, o `str` e o `repr`.
  - ▶ `str` - Cria um novo objeto de string a partir do objeto dado.
  - ▶ `repr` - Retorna a representação da string canônica do objeto.
- ▶ O que é representação canônica:
  - ▶ • (a "representação lexical" do booleano pode ser um dos seguintes: `{true, false, 1, 0}` enquanto
  - ▶ • a "representação canônica" só pode ser uma das `{true, false}`
- ▶ Isso, em essência, significa que
  - ▶ • `"true"` e `"1"` seja mapeado para a repr. canônica. `"true"` e
  - ▶ • `"false"` e `"0"` sejam mapeados para o repr. canonical. `"false"`
- ▶ fazer um exemplo

# Exercícios

Em Python existem modificadores de acesso como public e private? Justifique.

Não é possível criar uma instância de uma classe sem chamar o método `__init__`? justifique.