

LINGUAGEM DE PROGRAMAÇÃO II

AULA 02
CLASSES E OBJETOS

LUIS GUSTAVO ARAUJO



03

APRIMORANDO NOSSAS
CLASSES

USANDO PARÂMETROS NO CONSTRUTOR:

Nenhuma calculadora usar dois números fixo. Esses números são variados. Para fazer isso mais real, precisamos usar parâmetros no **construtor**.

NomedaClasse(a, b):

O QUE SÃO CONSTRUTORES:

Construtor é um método especial que constrói o nosso objeto. Quando chamamos `Calculadora()`, na verdade não estamos chamando o nome da classe e sim o construtor dela.

NomedaClasse():

USANDO PARÂMETROS NO CONSTRUTOR:

No Python a criação do construtor obedece algumas regras:

```
def __init__(self):
```

USANDO PARÂMETROS NO CONSTRUTOR:

No Python a criação do construtor obedece algumas regras:

```
def __init__(self ):
```



O **def** já é conhecido! Usamos ele para criar os métodos. O construtor é uma espécie de método especial da nossa classe.

USANDO PARÂMETROS NO CONSTRUTOR:

No Python a criação do construtor obedece algumas regras:

```
def __init__(self):
```



`__init__` é o nome usado para o construtor. Assim o Python sabe que aquele método é um construtor e não um método “normal”.

USANDO PARÂMETROS NO CONSTRUTOR:

No Python a criação do construtor obedece algumas regras:

```
def __init__(self):
```



self em português quer dizer: próprio. É no Python um palavra especial (reservada) para referencia a nossa própria classe. É usada como nos métodos.

PARÂMETROS NO CONSTRUTOR:

No Python a criação do construtor obedece algumas regras:

```
def __init__(self  ):
```

↑
Após o **self**, usamos quantos
parâmetros queremos...

USANDO PARÂMETROS NO CONSTRUTOR:

```
def __init__(self, num1, num2):
```

USANDO PARÂMETROS NO CONSTRUTOR:

Mas apenas o uso dos parâmetros não garante que a sua classe terá aqueles valores. É preciso usar **atributos** e copiar esses valores:

```
def __init__(self, num1, num2):  
    self.numero1 = num1  
    self.numero2 = num2
```

ATIVIDADE:

Vamos adicionar o construtor às nossas classes Calculadora e Boletim, usando os parâmetros necessários para cada caso.

DESAFIO:

Poderíamos ter uma classe que tenha como atributo uma imagem (Picture) e ter um método para aplicar efeito?

LINGUAGEM DE PROGRAMAÇÃO I

AULA 02
CLASSES E OBJETOS

LUIS GUSTAVO ARAUJO