



**UnB**

**Algoritmos e Estrutura de Dados**

# **Introdução à Orientação a Objetos**

Prof. Vinícius Pereira Gonçalves

vpgvinicius@unb.br

The background of the slide is a blurred LEGO scene. In the upper right, a white and black LEGO car is shown in a crash, with its wheels and body parts scattered. In the lower left, a LEGO minifigure of a person in a white shirt and black pants is walking away from the viewer, carrying a white suitcase. The ground is made of grey LEGO bricks with white dashed lines and a crosswalk pattern.

# MÉTODOS ESPECIAIS

# **Programação Orientada a Objetos?**

- **GETTERS (Métodos Acessores)**
- **SETTERS (Métodos Modificadores)**

# Programação Orientada a Objetos?

## GET

Nomeamos um método acessor com **get** toda vez que este método for verificar algum campo ou atributo de uma classe.

Como este método irá verificar um valor, ele sempre terá um retorno como String, int, float, etc.

# Programação Orientada a Objetos?

```
mesa = new Mesa  
pessoa=mesa.TotalProva
```

```
mesa = new Mesa  
pessoa=mesa.getTotalProva()
```

# Programação Orientada a Objetos?

## SET

Nomeamos um método acessor com **set** toda vez que este método for modificar algum campo ou atributo de uma classe, ou seja, se não criarmos um método acessor **set** para algum atributo, isso quer dizer que este atributo não deve ser modificado.

Portanto, como o valor de um atributo da classe será modificado, não é necessário que este método retorne nenhum valor, por isso, os métodos setters são **void**. Porém, obrigatoriamente, eles tem que receber um argumento que será o novo valor do campo.

# Programação Orientada a Objetos?

```
mesa = new Mesa
```

```
mesa.TotalProva=mesa.TotalProva +1
```

```
mesa = new Mesa
```

```
mesa.setTotalProva(prova)
```

<b>ClasseCaneta</b>
+ modelo + cor - ponta # carga # Tampada
+ escrever() + rabiscar() + pintar() -tampar() -destampar()

<b>ClasseCaneta</b>
+ modelo - ponta
+ getModelo() + setModelo(m) + getPonta() + setPonta(p)



ClasseCaneta
+ modelo - ponta
+ getModelo() + setModelo(m) + getPonta() + setPonta(p)

```

Classe Caneta  modelo:
publico  Caractere  ponta: Real
privado  Metodo getModelo()
publico  retorne modelo
          FimMetodo
          Metodo setModelo()
          Publico  (m:Caracter)
                  modelo=m
                  FimMetodo  Metodo
                  getPonta()
          publico  retorne ponta
                  FimMetodo  Metodo
                  setPonta()
          publico  (p:Real)  ponta=p
                  FimMetodo  FimClasse

```

Classe Caneta modelo:

publico Caractere ponta: Real

privado Metodo getModelo()

publico retorne modelo

FimMetodo

Metodo setModelo()

Publico (m:Caracter)

modelo=m

FimMetodo Metodo

getPonta()

publico retorne ponta

FimMetodo Metodo

setPonta()

publico (p:Real) ponta=p

FimMetodo FimClasse

C1=nova Caneta

C1.setModelo("MasterBoard")

C1.setPonta(2.0)

ESCREVA(C1.getModelo())

ESCREVA(C1.getPonta())



**UnB**

**Algoritmos e Estrutura de Dados**

# **Introdução à Orientação a Objetos**

Prof. Vinícius Pereira Gonçalves

[vpgvinicius@unb.br](mailto:vpgvinicius@unb.br)