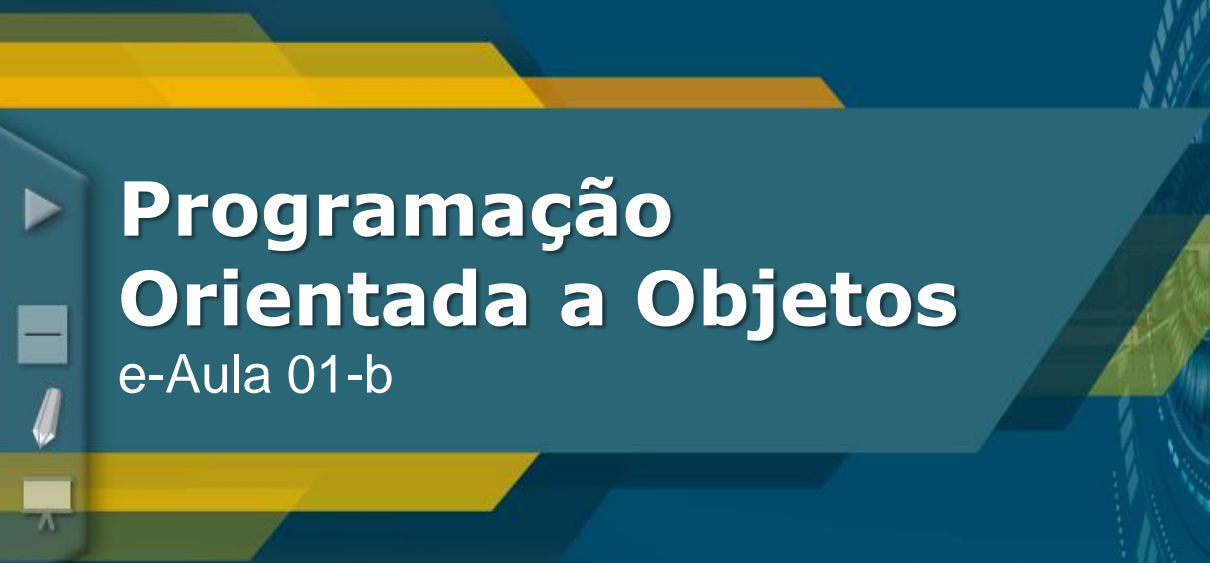




**AULA e1-b**



# **Programação Orientada a Objetos**

e-Aula 01-b

**Prof. Maristela Weinfurter**

**Vamos falar sobre  
VISIBILIDADE!**



**Ao observarmos nossa Classe  
VEÍCULO inicial, percebemos  
que não há nenhuma  
referência à **Visibilidade**.**

**Visibilidade** (ou acessibilidade) é a capacidade que uma classe tem de **alterar quem pode**, ou não, **ter acesso** aos seus **atributos e métodos**.

**Atributos** também podem ser públicos, mas não é normal.

O normal é termos os atributos privados e os métodos de acesso a eles como públicos!!!

## Públicos

Quando expõem os atributos ou métodos que eles modificam para outros objetos, sendo possível sua visualização e alteração.

## Privados

Quando não permitem que os métodos e atributos por eles modificados sejam visíveis e alteráveis pelos outros objetos que compõe o sistema.

## Protegidos

Que funciona de forma semelhante ao modificador privado, mas permite que os atributos e métodos sejam visíveis às subclasses.



## Nome da Classe

+ atributo1: String  
# atributo2: Integer  
- atributo3: String

+ método1()  
# método2()  
- método3()

# e-Aula 01b - POO

## Visibilidade

+ **public**  
# **protected**  
- **private**

### Nome da Classe

+ atributo1: String  
# atributo2: Integer  
- atributo3: String

+ método1()  
# método2()  
- método3()

# e-Aula 01b - POO

**Nome da Classe**

**Nome da Classe**

+ atributo1: String  
# atributo2: Integer  
- atributo3: String

**Atributos**

+ método1()  
# método2()  
- método3()

**Métodos**



**Vamos ampliar nossos  
conhecimentos sobre MÉTODOS!**

## Método

- Toda classe tem um **CONSTRUTOR**
- Se não criarmos, o compilador Java criará

## Método

- Cada novo OBJETO criado invoca pelo menos um CONSTRUTOR.
- Uma classe pode ter mais de um CONSTRUTOR.

## Método

```
public class Fornecedor{  
    public Fornecedor(){  
    }  
  
    public Fornecedor(String nome){  
        // nome é parâmetro do construtor  
    }  
}
```

## Construtor

- Cada novo OBJETO criado invoca pelo menos um CONSTRUTOR.
- Uma classe pode ter mais de um CONSTRUTOR.



## Construtor

- É o primeiro MÉTODO a ser executado quando uma CLASSE é INSTANCIADA!
- Método de ENTRADA no OBJETO.

## Getters & Setters

### Método GET:

- Quando **ACESSAMOS**, “**PEGAMOS**” alguns atributos da classe, devemos utilizar os métodos **GET**.

## Getters & Setters

### Método GET:

- Esse método sempre retornará um valor, seja ele String, int, double, ...

## Getters & Setters

### Método GET:

- Sintaxe do uso do método get

**Tipo do Atributo GET + Nome \_do  
\_Atributo()**

- **String getNome()**

## Getters & Setters

### Método SET:

- Para **MODIFICARMOS** os valores de um atributo da classe de maneira **PROTEGIDA**, utilizamos o método **SET**.

## Getters & Setters

### Método SET:

- **Esse método não tem retorno, pois o atributo é só modificado.**

## Getters & Setters

### Método SET:

- **VOID** cria um método SEM RETORNO.
- Mas recebe um argumento para alteração.

## Getters & Setters

### Método SET:

- Protegendo e trafegando corretamente com os atributos.



## Getters & Setters

### Método SET:

- Sintaxe do uso do método set

**VOID SET + Nome \_do  
\_Atributo(Tipo\_Atributo Nome\_Parametro)**

- **String setNome(String nome)**


## Vamos retomar nossa Classe VEÍCULO!



## Veículo

- modeloVeiculo : String
- marcaVeiculo : String
- tipoVeiculo : String
- qtdadeRodas : Int
- qtdadePassageiros : Int

- + incluiVeiculo()
- + mostraVeiculo()



**Agora, vamos alterar nossa classe Veículo no Eclipse em Java e depois trabalharemos a Classe FiguraGeométrica!**