# ORIENTAÇÃO A OBJETOS MÉTODOS

# ALGUMAS CLASSES DA AULA ANTERIOR...

## FUNCIONÁRIO

Desenvolva uma classe Principal que crie um Funcionário, e em um loop que represente um <u>simulador</u>, com possíveis ações do funcionário, tais como: trabalhar, tomar café, fofocar, solicitar aumento, etc...

Estas ações devem manipular os atributos da classe



#### Funcionario

Nome Cpf Data

Data de nasc.

Salario

Data de admissão

Cargo

Energia

credibilidade

Trabalhar()

tomarCafe()

fofocar()

SolicitarAumento()

dormir()

verificarStatus()



### CLASSE FUNCIONÁRIO

Modifique a classe Funcionário. Agora, o método trabalhar deve ler o número de horas trabalhadas e o método dormir deve ler o tempo, em horas de sono.

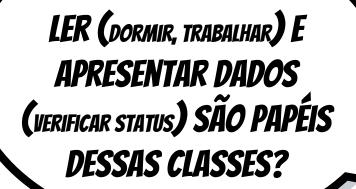
A lógica dos métodos devem ser adaptadas para que funcionem corretamente

## RECAPITULANDO UM CONCEITO DA AULA ANTERIOR...



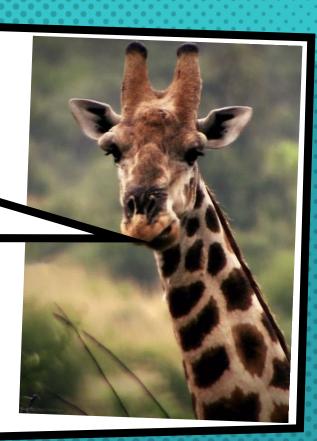
## ORIENTAÇÃO A OBJETOS

Em uma boa orientação a objetos, as responsabilidades e validações que dizem respeito a uma classe X, estão **na classe X**.





QUANDO FAÇO A LEITURA E SAÍDA NA PRÓPRIA CLASSE ESTOU ENGESSANDO QUE MINHA CLASSE SÓ PODERÁ SER UTILIZADA NO TERMINAL



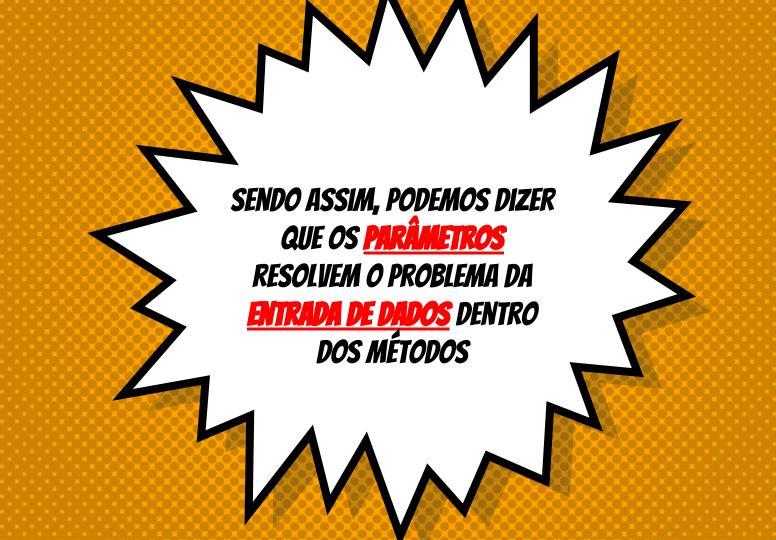


## MÉTODOS COM PARÂMETROS



#### PARÂMETRO

Os **parâmetros** especificam dados que são necessários para o funcionamento do método Esses parâmetros podem ser de qualquer tipo de dado, seja ele primitivo ou referência.



## MÉTODO COM RETORNO



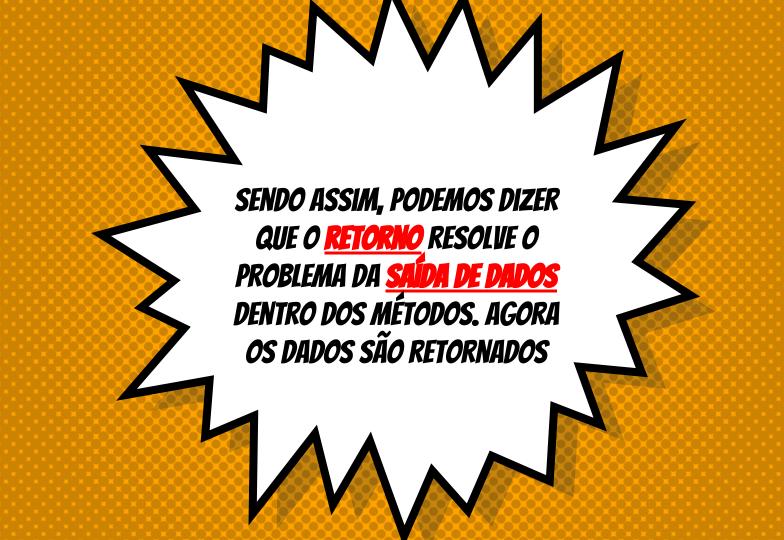
### ROTORNO DO MÉTODO

O retorno do método devolve a quem o invocou um dado que é resultado do processamento do mesmo

Este dado pode ser o resultado de um processamento ou uma mensagem a ser impressa, quando necessário

## ROTORNO DO MÉTODO

Quando é especificado que o método possui um tipo de **retorno** que não seja void, obrigatoriamente ele **deve** retornar um valor.



## ROTORNO DO MÉTODO

Assim como os parâmetros, o retorno pode ser de qualquer tipo de dado. Pode ser um primitivo ou referência. Ou seja, o retorno pode ser um int, double, Funcionario, Cliente, Veiculo, String, etc.

## Atividade - Criar e ser <u>criativo</u>! As classes são apenas um guia de exemplo









Modelo
Marca
Ano
placa
Potencia
estado(lig/des)
Num. de marchas
Marcha atual

Veiculo

Tam. do tanque Nível do combustível Disponível valor

Ligar()
Acelerar()
Parar()
aumentarMarcha()
diminuirMarcha()
Buzinar()
verificarDados()

