

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
ESCOLA DE CIÊNCIAS E TECNOLOGIA  
GRUPO DE INTEGRAÇÃO MAKER

## QUARTA LISTA DE EXERCÍCIOS

Luiz Eduardo Barros e Victor Matheus Castro

Natal, RN  
2016

# ORIENTAÇÃO A OBJETOS

**1. Classe Funcionário.** Implemente a classe Funcionário. Um empregado tem um nome (um string) e um salário(um double). Escreva um construtor com dois parâmetros (nome e salário) e métodos para devolver nome e salário, além disso crie um método aumentarSalario(percentualDeAumento) que aumente o salário do funcionário em uma certa porcentagem. Escreva um pequeno programa que teste sua classe.

**2. Classe Carro:** Implemente uma classe chamada Carro com as seguintes propriedades:

1. Um veículo tem um certo consumo de combustível (medidos em km / litro) e uma certa quantidade de combustível no tanque.
2. O consumo é especificado no construtor e o nível de combustível inicial é 0.
3. Forneça um método andar( ) que simule o ato de dirigir o veículo por uma certa distância, reduzindo o nível de combustível no tanque de gasolina.
4. Forneça um método obterGasolina( ), que retorna o nível atual de combustível.
5. Forneça um método adicionarGasolina( ), para abastecer o tanque.

**3. Classe Conta Corrente:** Crie uma classe para implementar uma conta corrente. A classe deve possuir os seguintes atributos: número da conta, nome do correntista e saldo. Os métodos são os seguintes: alterarNome, depósito e saque; No construtor, saldo é opcional, com valor default zero e os demais atributos são obrigatórios.

**4. Classe Bichinho Virtual:** Crie uma classe que modele um Tamagushi (Bichinho Eletrônico):

- a. Atributos: Nome, Fome, Saúde e Idade
- b. Métodos: Alterar Nome, Fome, Saúde e Idade; Retornar Nome, Fome, Saúde e Idade

**Obs:** Existe mais uma informação que devemos levar em consideração, o Humor do nosso tamagushi, este humor é uma combinação entre os atributos Fome e Saúde, ou seja, um campo calculado, então não devemos criar um atributo para armazenar esta informação por que ela pode ser calculada a qualquer momento.

**c.** Permita que o usuário especifique quanto de comida ele fornece ao bichinho e por quanto tempo ele brinca com o bichinho. Faça com que estes valores afetem quão rapidamente os níveis de fome e tédio caem.

**5.** Mostre a declaração de uma classe “Quadrado” que deriva de “Retângulo”, que por sua vez deriva de “Forma”, utilize o conceito de Polimorfismo e Herança.

**6.** Crie uma classe Veiculo. Com base nessa classe, defina duas classes herdeiras denominadas Carro e Caminhão com as seguintes características:

| Função (classe Carro)    | Descrição   |
|--------------------------|---|
| Construtor               | Inicializado com o número de passageiros, número de rodas e carga de passageiros suportada. |
| getPassageiros()         | Retorna o número atual de passageiros no veículo.   |
| setPassageiros(variavel) | Define o número atual de passageiros no veículo.  |
| Função (classe Caminhão) | Descrição   |
| Construtor               | Inicializado com o número de número de rodas e carga máxima suportada.                      |
| setCarga(variavel)       | Define a carga transportada.  |
| eficiencia()             | Retorna a relação entre a carga suportada e massa total do veículo carregado.               |
| cargaPorRoda()           | Retorna a massa suportada por cada uma das rodas do caminhão, supondo distribuição igual.   |