



Instituto Federal de Santa Catarina – IFSC  
Campus São José

# Programação Orientada a Objetos

## Associação entre Classes: Agregação e Composição

Prof. Francisco de Assis S. Santos, Dr.

São José, 2015.

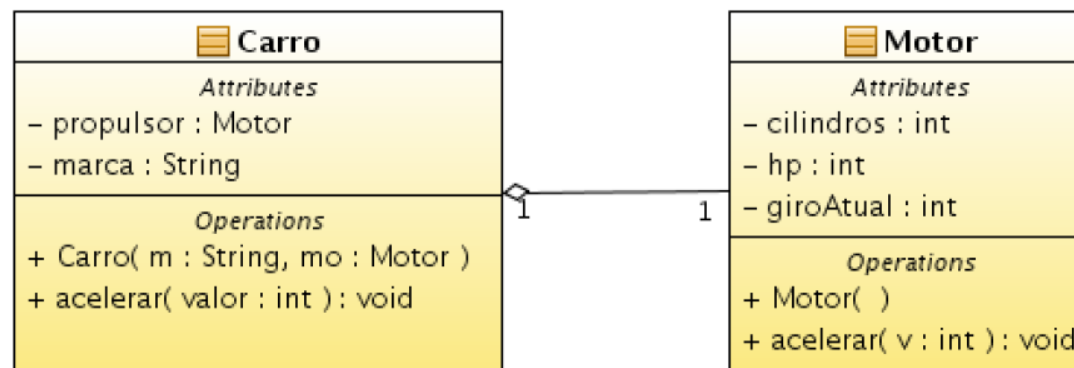
## Associação entre Classes

- Na modelagem orientada a objetos a associação entre classes representa um tipo de relacionamento entre os objetos destas classes
  - Por exemplo, a classe Carro possui um relacionamento com a classe Motor, pois um objeto da classe Carro contém 1 objeto da classe Motor

```
public class Carro{  
    private String marca;  
    private Motor propulsor;  
  
    public Carro(String m, Motor mo){  
        this.marca = m;  
        this.propulsor = mo;  
    }  
}
```

## Classificação das Associações: Agregação

- Trata-se de um relacionamento “é parte de”
- Quando o objeto contido faz sentido mesmo sem ser parte do objeto que o contém
  - Exemplo: Um carro possui um motor. Se o carro deixar de existir, o motor poderia ser colocado em outro carro

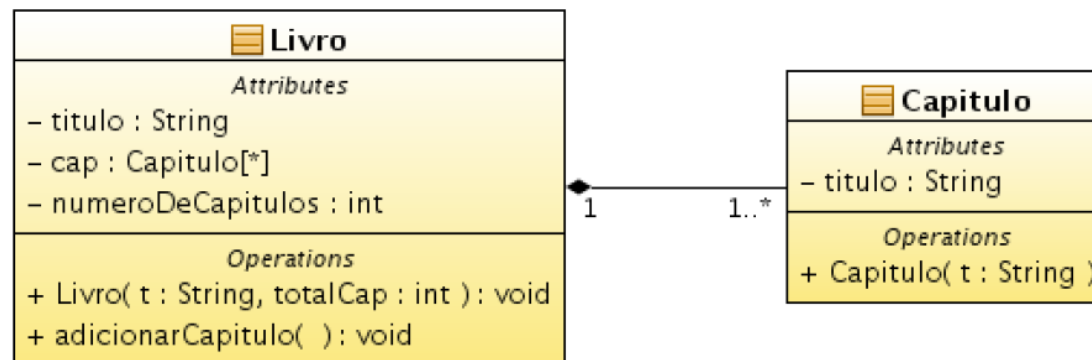


## Agregação: Carro e Motor

```
public class Carro{  
    private String marca;  
    private Motor propulsor;  
  
    public Carro(String m, Motor mo){  
        this.marca = m;  
        this.propulsor = mo;  
    }  
  
    public void acelerar(int valor){  
        this.propulsor.acelerar(valor);  
    }  
}
```

## Classificação das Associações: Composição

- Trata-se de um relacionamento “faz parte de” (mais restritivo que a Agregação)
- Quando o objeto contido não faz sentido sem o objeto que o contém
  - Exemplo: Um livro é composto por diversos capítulos. Se destruirmos um livro, não faria mais sentido os capítulos existirem



## Composição: Livro e Capítulos

```
import java.util.Scanner;
public class Livro{
    private String titulo;
    private Capitulo[] cap;
    private int numeroDeCapitulos;

    public Livro(String t, int totalDeCapitulos){
        this.titulo = t;
        this.cap = new Capitulo[totalDeCapitulos];
        this.numeroDeCapitulos = 0;
    }
    public void adicionarCapitulo(){
        Scanner ler = new Scanner(System.in);
        String t = ler.nextLine();
        this.cap[numeroDeCapitulos] = new Capitulo(t);
        numeroDeCapitulos++;
    }
}
```

## Exercícios

- 1) Modele um diagrama de classes para representar funcionários, salários, clientes e produtos em um sistema de gestão de pessoas e vendas.
  - É necessário que nesta modelagem seja feito uso de associação e agregação entre as classes, quando possível.
- 2) Implemente em Java as classes definidas no diagrama de classes.

# Referências

Notas de aula do Prof. Emerson Ribeiro de Mello