

Monitoria P00

Agregação e Composição

Monitor: Johnes Thomas



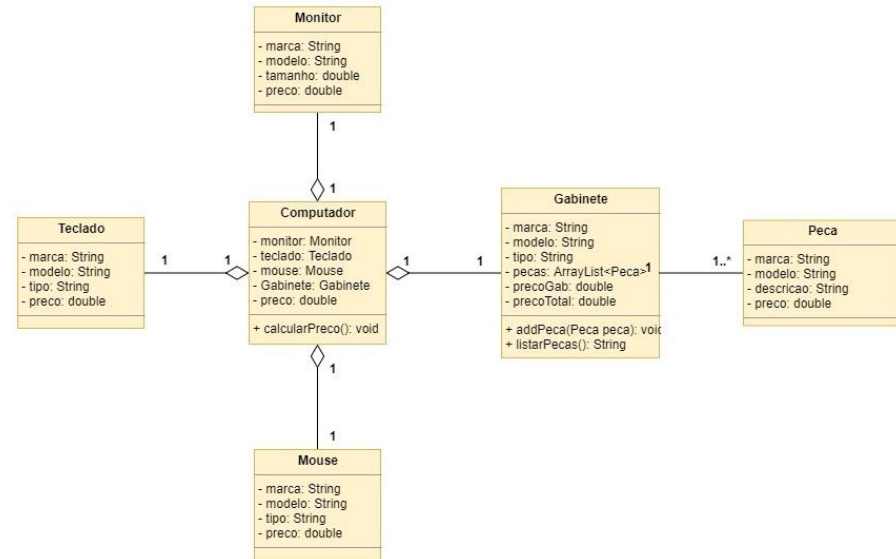
/johnesthomas



/johnesthomas

Agregação

Associação por agregação é quando uma classe usa outras classes para realizar suas operações. No exemplo ao lado, Computador é o objeto definido como sendo o **todo**. E este objeto só pode existir se os demais existirem. A classe principal (agregadora) não possui as classes secundárias (agregadas) com sua parte.



Implementação Agregação

A classe **Peca** será a primeira a ser implementada, em seguida a classe **Gabinete** que contém como um de seus atributos uma lista de peças.

```
public class Gabinete {  
  
    private String marca;  
    private String modelo;  
    private String tipo;  
    private ArrayList<Peca> pecas = new ArrayList<>();  
    private double precoGab; // preco do gabinete  
    private double precoTotal; // preco do gabinete + pecas
```

```
public class Peca {  
  
    private String marca;  
    private String modelo;  
    private String descricao;  
    private double preco;
```

Implementação Agregação

As classes componentes podem ser implementadas sem uma ordem definida, já que nenhuma classe precisa da outra. Depois de implementar as classes partes, a classe principal **Computador** será criada, ela é a última pois seus atributos dependem de outras classes, ou seja, o computador só faz sentido se as demais classes existirem.

```
public class Computador {  
  
    private Monitor monitor;  
    private Teclado teclado;  
    private Mouse mouse;  
    private Gabinete gabinete;  
    private double preco; // valor total do computador  
}
```

```
public class Teclado {  
  
    private String marca;  
    private String modelo;  
    private String tipo;  
    private double preco;  
}
```

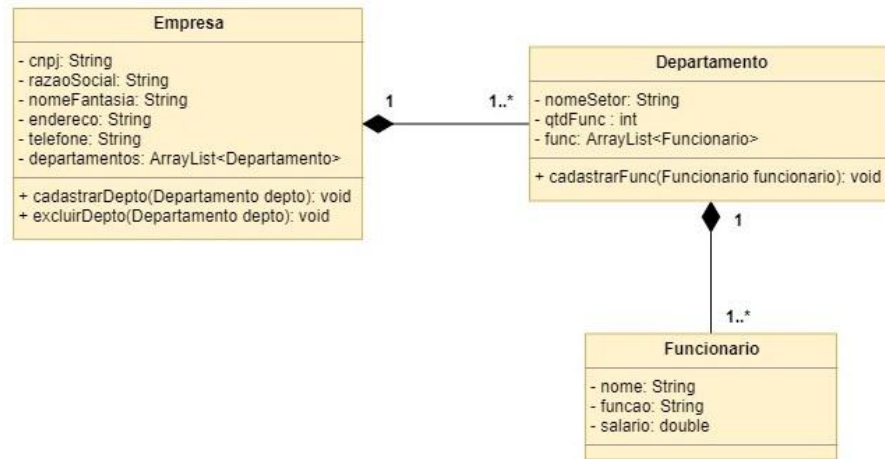
```
public class Mouse {  
  
    private String marca;  
    private String modelo;  
    private String tipo;  
    private double preco;  
}
```

```
public class Monitor {  
  
    private String marca;  
    private String modelo;  
    private double tamanho;  
    private double preco;  
}
```

Bora praticar !

Composição

Associação por composição é quando uma classe definida como o **todo** é composta por classes definidas como **partes**. Se o objeto principal deixa de existir os objetos secundários deixarão de existir. O objeto principal pode manipular os objetos secundários.



Implementação Composição

A classe **Funcionario** será a primeira a ser implementada pois quando a classe **Departamento** for implementada será necessário criar uma lista de funcionários, e por final a classe **Empresa** será implementada e um dos seus atributos é uma lista de departamentos.

```
public class Empresa {  
    private String cnpj;  
    private String razaoSocial;  
    private String nomeFantasia;  
    private String endereco;  
    private String telefone;  
    private ArrayList<Departamento> departamentos = new ArrayList<Departamento>();  
}
```

```
public class Departamento {  
    private String nomeSetor;  
    private int qtdFunc; // quantidade de funcionarios  
    private ArrayList<Funcionario> funcionarios = new ArrayList<Funcionario>();  
}
```

```
public class Funcionario {  
    private String nome;  
    private String funcao;  
    private double salario;  
}
```

OBS.: Podemos perceber a associação por composição, se uma empresa deixa de existir, obrigatoriamente os departamento deixam de existir.

Bora praticar !

Referências

<https://pt.slideshare.net/armandodaniel777/java-orientao-a-objetos-associacao-composicao-agregacao>