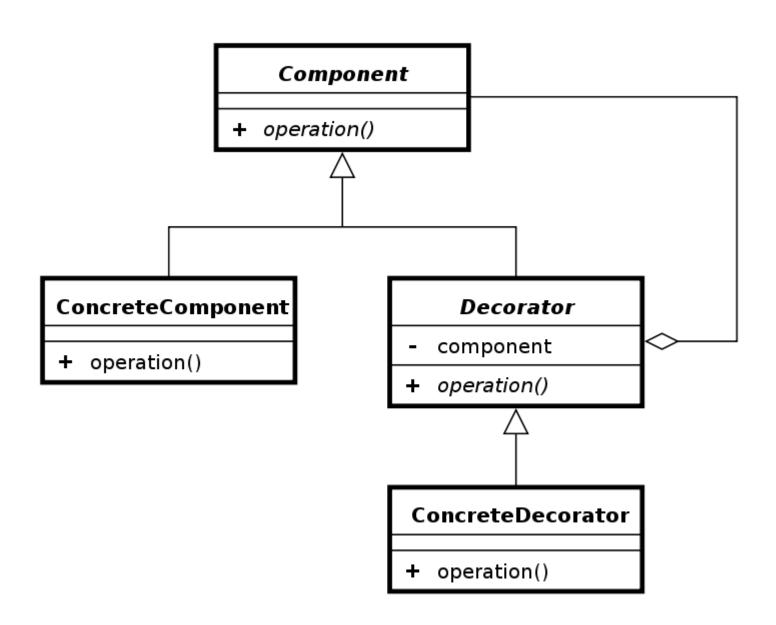
https://www.vw.com.br/pt/configurador.html

Como ficaria a UML?



«interface» Carro + getCusto(): double + getDescricao() : string + getProdutos() : string CrossFox CarroDecorador # carroDecorado : Carro descricao : string + getCusto() : double produtos : string + getDescricao() : string + getCusto() : double + getProdutos() : string + getDescricao() : string + getProdutos() : string TravaEletrica custo : double produtos : string

+ getCusto() : double

+ getDescricao(): string

+ getProdutos() : string

custo : double

ArCondicionado

- custo : double
- produtos : string
- + getCusto() : double
- + getDescricao() : string
- + getProdutos() : string

 Lembre: associação é melhor que herança!

 Princípio: as classes devem estar abertas para extensão mas fechadas para modificação

- Definição: anexa responsabilidades adicionais a um objeto dinamicamente.
- Fornecem uma alternativa flexível de subclasse para estender a funcionalidade

Desafio

- Adicionar outros itens opcionais:
 - Air bag
 - Kit esportivo
 - Direção hidráulica

 http://www.patternizando.com.br/2011/01/ padrao-de-projeto-decorator-uma-aplicaca o-real-em-java/