**Decorator**



**abstract** **public** **class** Cafe {

**public** **abstract** **double** calcPreco();

}

**abstract** **public** **class** CafeDecorator **extends** Cafe{

//Se eu quiser Expresso com Canela

//Devo guardar a instancia de Expresso

//aqui neste atributo

**protected** Cafe cafe; //prox

//Seto no teste qual o proximo

//da cadeia

**public** **void** setAdicional(Cafe cafe) {

**this**.cafe = cafe;

}

}

**public** **class** Canela **extends** CafeDecorator{

@Override

**public** **double** calcPreco() {

**return** 0.70+cafe.calcPreco();

}

}

**public** **class** Capuccino **extends** Cafe{

@Override

**public** **double** calcPreco() {

**return** 4.00;

}

}

**public** **class** Chocolate **extends** CafeDecorator{

@Override

**public** **double** calcPreco() {

**return** 2.30 + cafe.calcPreco();

}

}

**public** **class** Expresso **extends** Cafe{

@Override

**public** **double** calcPreco() {

**return** 2.00;

}

}

**public** **class** Teste {

**public** **static** **void** main(String[] args) {

Expresso e = **new** Expresso();

Chocolate ch = **new** Chocolate();

Chocolate ch1 = **new** Chocolate();

Canela cn = **new** Canela();

ch.setAdicional(e);

ch1.setAdicional(ch);

cn.setAdicional(ch1);

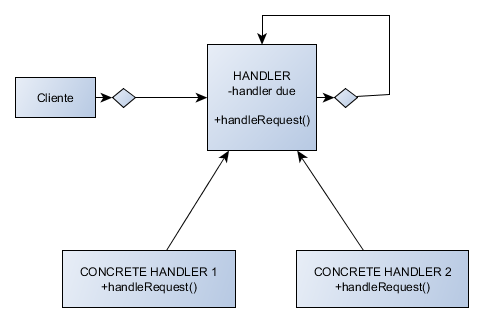
System.***out***.println(cn.calcPreco());

}

}

**Chain of Responsibility**

**Definição:** evita o acoplamento entre o emissor e o receptor de uma requição dando a chance de tratamento para mais de um **objeto**. forma uma estrutura de **objeto** que passam a requisição até esta ser tomada.



**abstract** **public** **class** Empresa {

**private** Verba verba;

**private** String classNome;

**protected** Empresa cargos;

**public** **void** adicionarCargos(Empresa c){

cargos = c;

}

**abstract** **public** **void** aprovarVerba(Verba v);

**public** **void** mensagem(){

Class c = **this**.getClass();

System.***out***.println("Verba Aprovado pelo " + c.getSimpleName() );

}

}

**public** **class** Verba {

**private** **double** valor;

**public** Verba(**double** v){

valor = v;

}

**public** **double** getValor(){

**return** valor;

}

}

**public** **class** Analista **extends** Empresa{

@Override

**public** **void** aprovarVerba(Verba v) {

**if**( v.getValor() < 10000 ){

mensagem();

}**else**{

**this**.cargos.aprovarVerba(v);

}

}

}

**public** **class** Diretor **extends** Empresa {

@Override

**public** **void** aprovarVerba(Verba v) {

**if**( v.getValor() < 50000 ){

mensagem();

}**else**{

**this**.cargos.aprovarVerba(v);

}

}

}

**public** **class** Gerente **extends** Empresa{

@Override

**public** **void** aprovarVerba(Verba v) {

**if**( v.getValor() < 25000 ){

mensagem();

}**else**{

**this**.cargos.aprovarVerba(v);

}

}

}

**public** **class** CEO **extends** Empresa {

@Override

**public** **void** aprovarVerba(Verba v) {

mensagem();

}

}