装饰者模式

问题：咖啡店系统需要升级，原来的设计为一个超类Beverage

class Beverage{

String description=“unknow Beverage”;

public String getDescripition(){

return description;

}

public abstract double cost();

}

店内的所有种类咖啡(如HouseBlend,DarRoast,Decaf,Espresso)都要继承之，并实现description属性及cost方法(返回所需费用)，并且每种咖啡都可以加入各种调料，如蒸奶，豆浆，魔卡，奶泡之类相应会收取不同费用

解决：

拿到这个问题，初步的想法如下：

结合真实的情形，应如此，某种类型咖啡继承Beverage，然后各种调料之类的单独独立出来，需要则add到具体类型咖啡上，然后价格按照某种机制加在一起

class DarRoast extends Beverage{

double costs;

DarRoast(){

description=”DarRoast ”;

costs=xxx;

}

public double cost(){

return costs;

}

}

//调料也可以继承Beverage超类

class milk extends Beverage {

double costs;

milk (){

description=”add milk ”;

costs=xxx;

}

public double cost(){

return costs;

}

}

class Mocha extends Beverage {

double costs;

Mocha (){

description=”add Mocha ”;

costs=xxx;

}

public double cost(){

return costs;

}

}

怎样将各种调料的价格加到咖啡上去呢？可以在具体的咖啡中增加一个add\_[flavour](file:///C:\Users\wangchaoqun-pc\AppData\Local\youdao\dict\Application\7.0.1.0227\resultui\dict\?keyword=flavour)(Object [flavour](file:///C:\Users\wangchaoqun-pc\AppData\Local\youdao\dict\Application\7.0.1.0227\resultui\dict\?keyword=flavour))方法，

class DarRoast extends Beverage{

double costs;

DarRoast(){

description=”DarRoast ”;

costs=xxx;

}

public double cost(){

printf(“cost all … %d”,costs);

return costs;

}

public void add\_[flavour](file:///C:\Users\wangchaoqun-pc\AppData\Local\youdao\dict\Application\7.0.1.0227\resultui\dict\?keyword=flavour)(Object [flavour](file:///C:\Users\wangchaoqun-pc\AppData\Local\youdao\dict\Application\7.0.1.0227\resultui\dict\?keyword=flavour)){

printf(“add flavour name %s”,flavour.description);

costs +=flavour.cost()

}

}