1. 以对象取代数据值

如果一个数据值需要和一些行为绑定在一起，那么提炼出一个新类吧

// 重构前

class 房子{

    ...

    int 门的长度;

    int 门的宽度;

    int 获取门的面积() { ... }

    int 获取门的占地面积() { ... }

}

// 重构后

class 房子{

    门 门;

    ...

}

class 门{

    int 门的长度;

    int 门的宽度;

    int 获取门的面积() { ... }

    int 获取门的占地面积() { ... }

}

1. 将值对象改为引用对象

如果一个是对象可修改，那么它应该被当作一个引用对象

值对象的属性在对象生成之后就不能修改，修改值对象的唯一方法就是生成新的值对象替换原来的值对象

值对象与引用对象代码上没有区别，但引用对象允许引用对象自身的方法修改引用对象自身的属性

// 重构前

class 门{

    // 假设这类是值对象

    int 门的高度 { get; }

    int 门的宽度 { get; }

    int 获取门的面积() { ... }

    int 获取门的占地面积() { ... }

}

// 重构后

class 门{

    // id 是应用对象的标识

    int id;

    int 门的高度;

    int 门的宽度;

    int 获取门的面积() { ... }

    int 获取门的占地面积() { ... }

    // 该方法会修改自身的属性

    void 根据给定的房间高度自动设置门的高度(){ ... }

}

值对象容易管理，上述功能其实我们可以不转为引用对象

// 重构前

class 门{

    // 假设这类是值对象

    int 门的高度 { get; }

    int 门的宽度 { get; }

    int 获取门的面积() { ... }

    int 获取门的占地面积() { ... }

}

// 重构后

class 门{

    // 还是值对象

    int 门的高度;

    int 门的宽度;

    int 获取门的面积() { ... }

    int 获取门的占地面积() { ... }

    // 该方法会计算并返回一个新的值对象

    门 根据给定的房间高度自动设置门的高度(){ ... }

}

1. 以对象取代数组

不要以数组索引或某个数值去代表某种意思，

即，不用用arr[0]代表长，arr[1]代表宽这种方式

1. 封装集合

如果不想在类外直接修改类的集合，那应该返回集合的只读副本，然后提供接口

1. 以类（值对象）取代类型码

类有一个数值类型吗，起标识作用，但不影响类的行为，如下是重构后的代码（这里我更倾向于使用枚举类型）

class Person

{

    public person(Group group){

    }

    public Group Group{ get; protected set; }

}

public class Group{

public int Code{ get; protected set; }

    protect Group(int code){

        Code = code;

}

public static Group A { get{ return new Group(0); } }

public static Group B { get{ return new Group(1); } }

    // 在这里重写比较符 ==

    ...

}

重构前后代码对比

// 重构前

var person = new Person(0);

if(person.Code == 0){

    ...

}

// 重构后

var person = new Person(Group.A);

// 注意，我们在值对象中重写比较符

if(person.Group == Group.A){

    ...

}

1. 以子类取代类型码

类有一个不可变数值类型码，起标识作用，影响类的行为

使用继承关系：

Class 动物

Class 人：动物

Class 猴子：动物

这里我们会移除掉类型码字段，因为“人”，“猴”这两个类就可以区分了，不需要通过类型码

1. 状态模式取代类型码

类有一个可变数值类型码，起标识作用，影响类的行为

Class 人{

StateBase state;

}

Class 小：StateBase

Class 大：StateBase

这里我们会移除掉类型码字段

注：状态码类是值对象，不是实体，也就是说状态码类不应该有对应的数据库表