

Java私塾-最专业的Java就业培训专家，因为专业，所以出色！值得你的信赖！



# 私塾在线 《研磨设计模式》 ——跟着CC学设计系列精品教程

10101010101010101010101010101

私塾在线<http://sishuok.com?frombook> 独家提供配套教学视频，更有大量免费在线学习视频独家大放送

## 本节课程概览

### n 学习适配器模式

#### 一：初识适配器模式

包括：定义、结构、参考实现

#### 二：体会适配器模式

包括：场景问题、不用模式的解决方案、使用模式的解决方案

#### 三：理解适配器模式

包括：认识适配器模式、适配器模式的实现、双向适配器、  
对象适配器和类适配器、适配器模式的优缺点

#### 四：思考适配器模式

包括：适配器模式的本质、何时选用

**做最好的在线学习社区**

网 址：<http://sishuok.com>

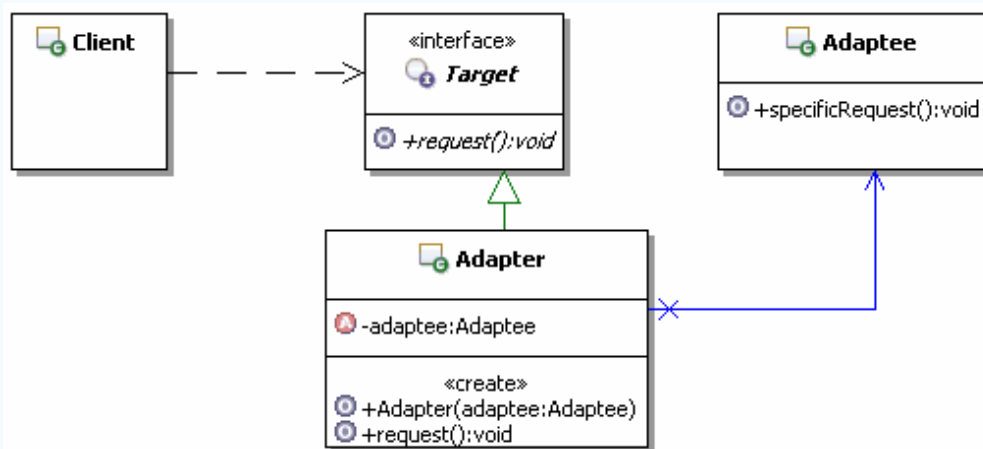
咨询QQ：2371651507

## 初识适配器模式

### n 定义

将一个类的接口转换成客户希望的另外一个接口。适配器模式使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。

### n 结构和说明



Client: 客户端，调用自己需要的领域接口Target。

Target: 定义客户端需要的跟特定领域相关的接口。

Adaptee: 已经存在的接口，但与客户端要求的特定领域接口不一致，需要被适配。

Adapter: 适配器，把Adaptee适配成为Client需要的Target

**做最好的在线学习社区**

网 址: <http://sishuok.com>

咨询QQ: 2371651507

私塾在线<http://sishuok.com?frombook> 独家提供配套教学视频，更有大量免费在线学习视频独家大放送

## 体会适配器模式

### n 装配电脑的例子

#### 1: 旧的硬盘和电源



#### 2: 加入新的硬盘



做最好的在线学习社区

网 址: <http://sishuok.com>

咨询QQ: 2371651507

私塾在线<http://sishuok.com?frombook> 独家提供配套教学视频，更有大量免费在线学习视频独家大放送

## 体会适配器模式

### 3: 有何问题

如果把上面的问题抽象一下，用对象来描述，那就是：有一个电源类和旧的硬盘类配合工作得很好，现在又有了一个新的硬盘类，现在想让新的硬盘类和电源类也配合使用，但是发现它们的接口无法匹配，问题就产生了：如何让原有的电源类的接口能够适应新的硬盘类的电源接口的需要呢？

### 4: 如何解决

解决方法是采用一个转接线类，转接线可以把电源的接口适配成为新的硬盘所需要的接口，那么这个转接线类就类似本节的主角——适配器。



## 体会适配器模式

n 同时支持数据库和文件的日志管理

1: 日志管理第一版

在第一版的时候，用户要求日志以文件的形式记录。

2: 日志管理第二版

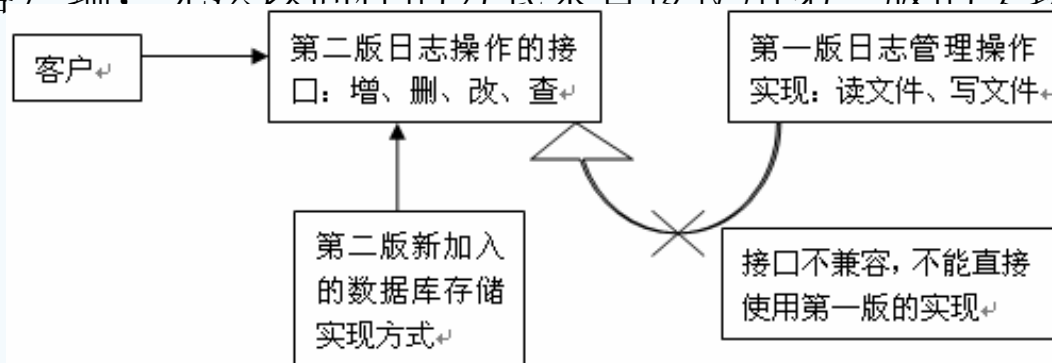
要采用数据库来管理日志

n 不用模式的解决方案

直接参看代码示例

n 存在的问题

现在的客户端，无法以同样的方式来直接使用第一版的实现



做最好的在线学习社区

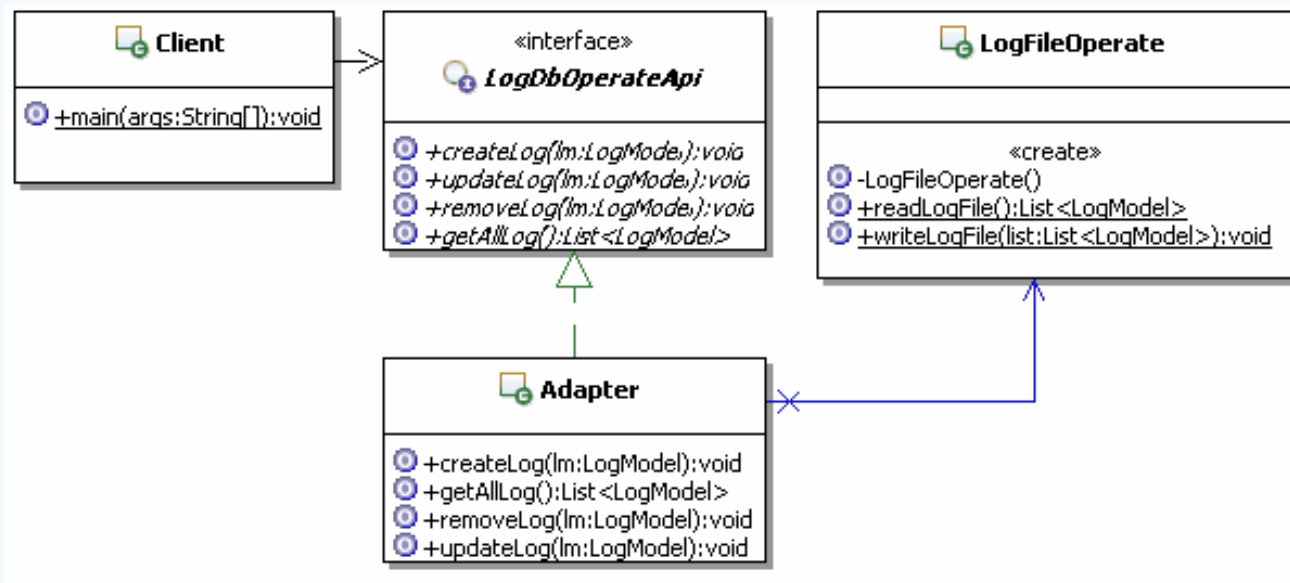
网 址: <http://sishuok.com>

咨询QQ: 2371651507

私塾在线<http://sishuok.com?frombook> 独家提供配套教学视频，更有大量免费在线学习视频独家大放送

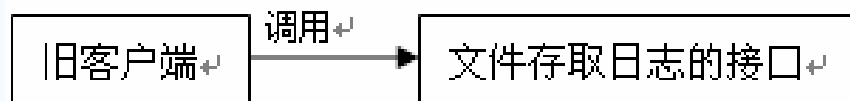
## 体会适配器模式

### n 使用模式的解决方案



### n 小结一下思路

1: 原有文件存取日志的方式，运行得很好



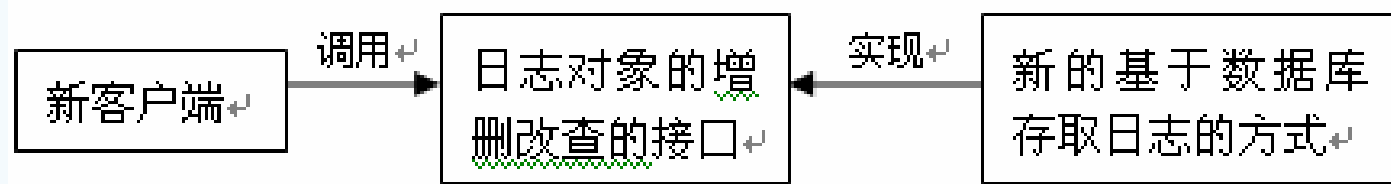
做最好的在线学习社区

网 址: <http://sishuok.com>

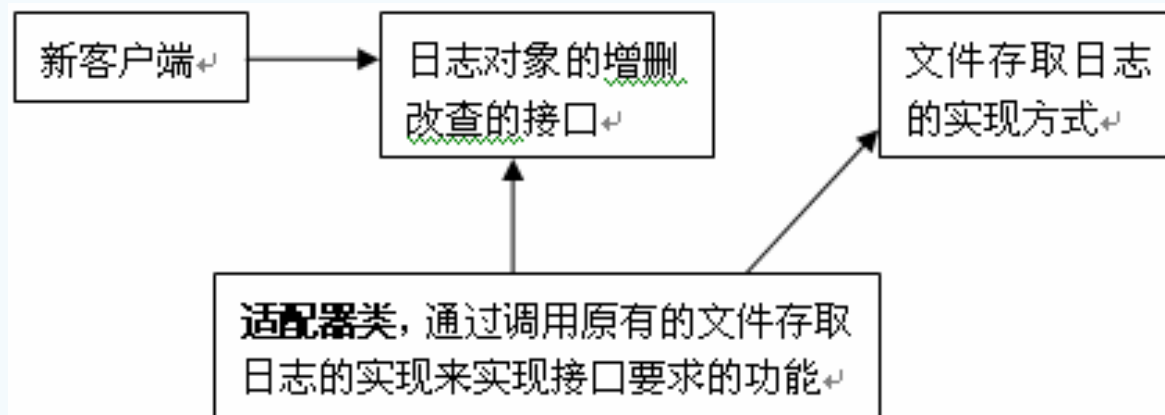
咨询QQ: 2371651507

## 体会适配器模式

2: 现在有了新的基于数据库的实现，新的实现有自己的接口



3: 现在想要在第二版的实现里面，能够同时兼容第一版的功能，那么就应有一个类来实现第二版的接口，然后在这个类里面去调用已有的功能实现，这个类就是适配器



做最好的在线学习社区

网 址: <http://sishuok.com>

咨询QQ: 2371651507

私塾在线<http://sishuok.com?frombook> 独家提供配套教学视频，更有大量免费在线学习视频独家大放送



## 理解适配器模式

### n 认识适配器模式

#### 1: 适配器模式的功能

适配器模式的主要功能是进行转换匹配，目的是复用已有的功能，而不是来实现新的接口。

适配器里面也可以实现功能，称这种适配器为智能适配器。

#### 2: Adaptee和Target的关系

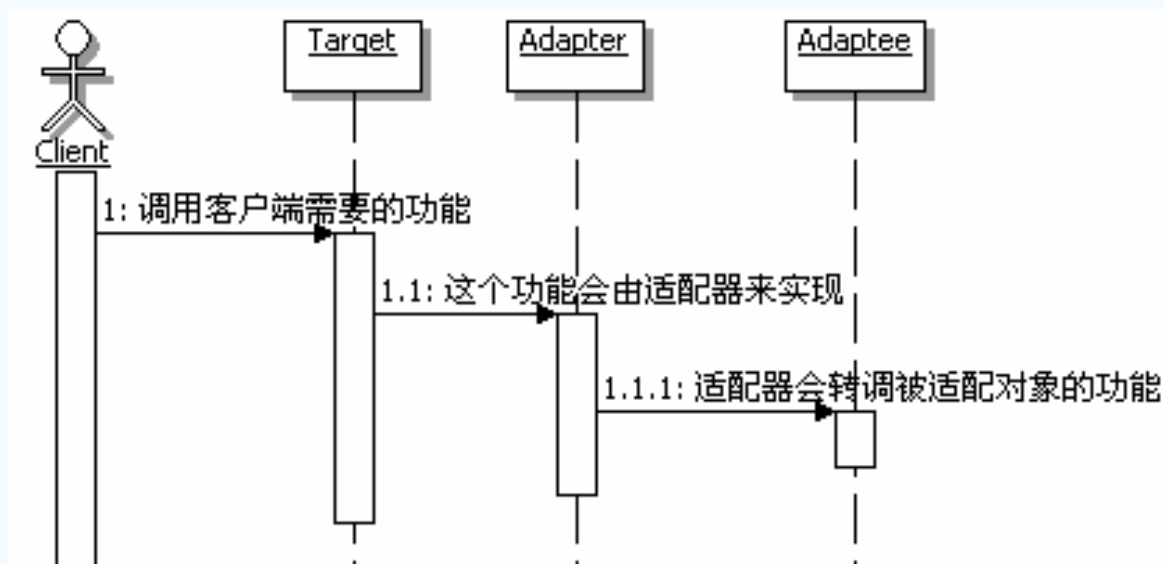
适配器模式中被适配的接口Adaptee和适配成为的接口Target是没有关联的

#### 3: 对象组合

适配器的实现方式其实是依靠对象组合的方式。

## 理解适配器模式

### 5: 适配器模式的调用顺序示意图



做最好的在线学习社区

网 址: <http://sishuok.com>

咨询QQ: 2371651507

私塾在线<http://sishuok.com?frombook> 独家提供配套教学视频，更有大量免费在线学习视频独家大放送

## 理解适配器模式

### n 适配器模式的实现

#### 1: 适配器的常见实现

适配器通常是一个类，一般会让适配器类去实现Target接口，然后在适配器的具体实现里面调用Adaptee。

#### 2: 智能适配器

可以在适配器的实现里面，加入新功能的实现，这种适配器被称为智能适配器。

#### 3: 适配多个Adaptee

#### 4: 适配器Adapter实现的复杂程度

适配器Adapter实现的复杂程度，取决于Target和Adaptee的相似程度。

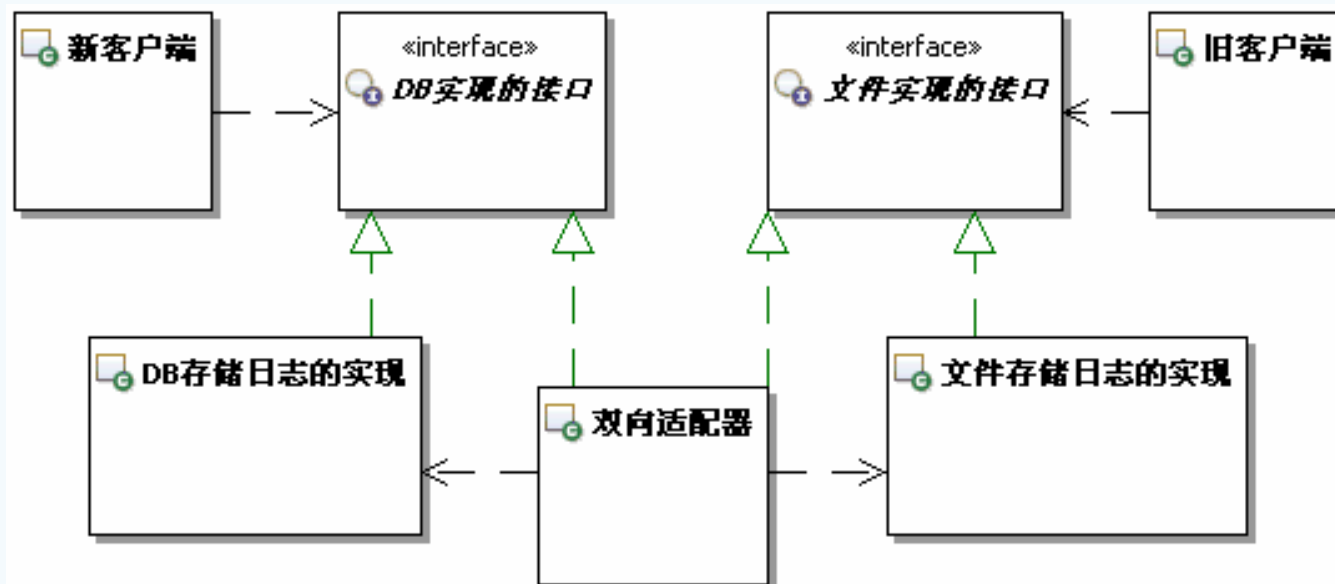
#### 5: 缺省适配

缺省适配的意思是：为一个接口提供缺省实现。

## 理解适配器模式

### n 双向适配器

适配器也可以实现双向的适配，前面我们讲的都是把Adaptee适配成为Target，其实也可以把Target适配成为Adaptee，也就是说这个适配器可以同时当作Target和Adaptee来使用。

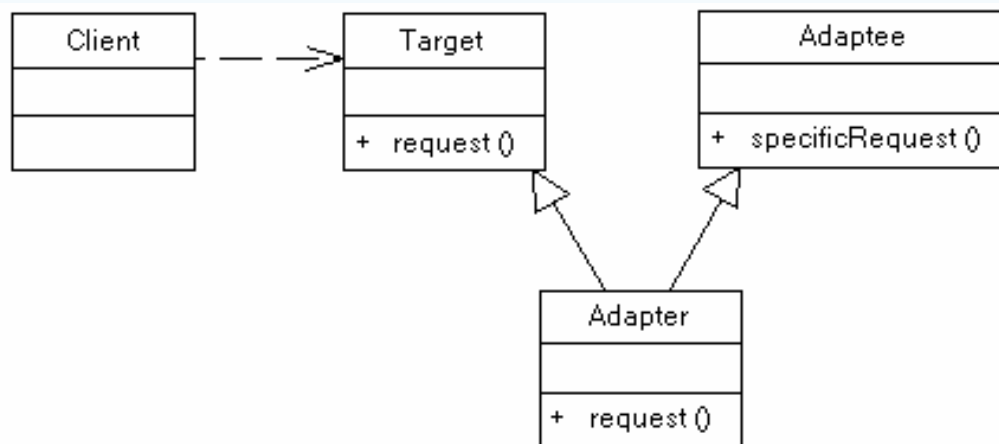


## 理解适配器模式

### n 对象适配器和类适配器

- 1: 对象适配器的实现：依赖于对象组合。就如同前面的实现示例，都是采用的对象组合的方式，也就是前面讲述的都是对象适配器实现的方式。
- 2: 类适配器的实现：采用多重继承对一个接口与另一个接口进行匹配。由于Java不支持多重继承，所以到目前为止，还没有涉及到。

类适配器的结构图如下：



做最好的在线学习社区

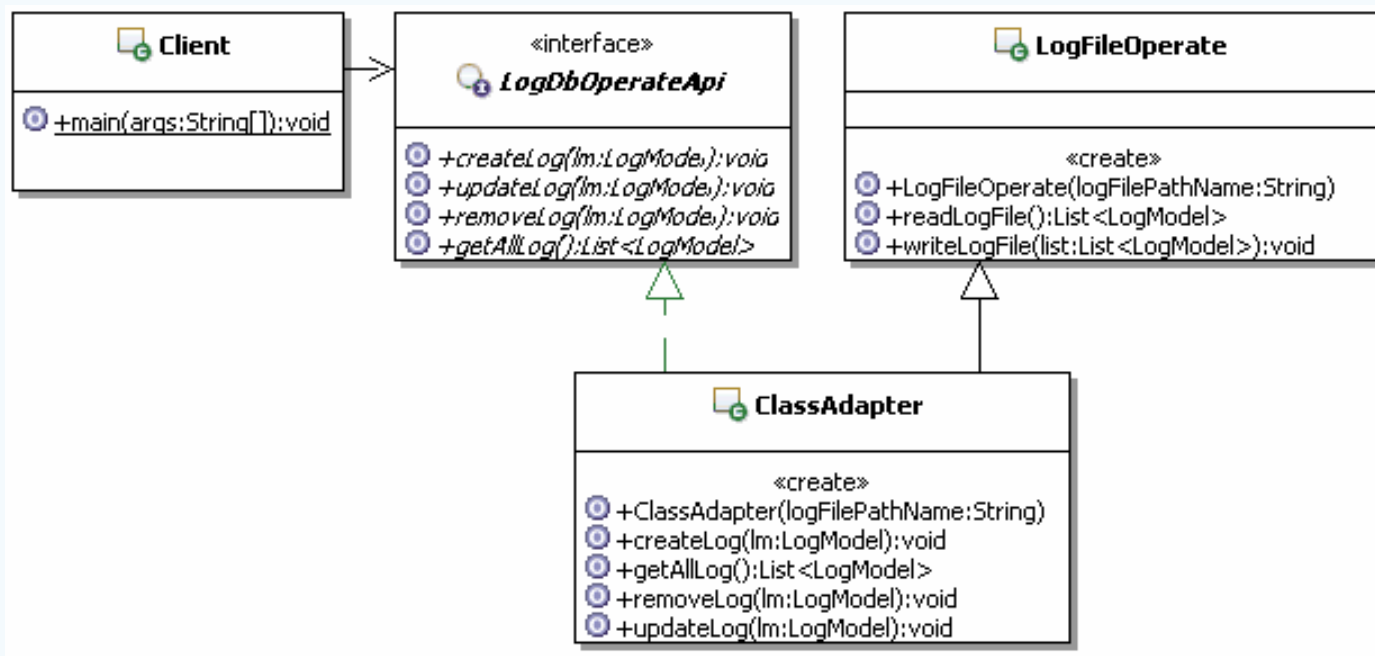
网 址：<http://sishuok.com>

咨询QQ：2371651507

私塾在线<http://sishuok.com?frombook> 独家提供配套教学视频，更有大量免费在线学习视频独家大放送

## 理解适配器模式

### n Java中类似实现类适配器





## 理解适配器模式

### n 类适配器和对象适配器的权衡

- 1: 从实现上：类适配器使用对象继承的方式，是静态的定义方式；而对象适配器使用对象组合的方式，是动态组合的方式
- 2: 对于类适配器：由于适配器直接继承了Adaptee，使得适配器不能和Adaptee的子类一起工作，因为继承是个静态的关系，当适配器继承了Adaptee过后，就不可能再去处理Adaptee的子类了。

对于对象适配器：允许一个Adapter和多个Adaptee，包括Adaptee和它所有的子类一起工作。因为对象适配器采用的是对象组合的关系，只要对象类型正确，是不是子类都无所谓。

- 3: 对于类适配器：适配器可以重定义Adaptee的部分行为，相当于子类覆盖父类的部分实现方法。

对于对象适配器：要重定义Adaptee的行为比较困难，这种情况下，需要定义Adaptee的子类来实现重定义，然后让适配器组合子类。

- 4: 对于类适配器：仅仅引入了一个对象，并不需要额外的引用来间接得到Adaptee。

对于对象适配器：需要额外的引用来间接得到Adaptee。

做最好的在线学习社区

网 址：<http://sishuok.com>

咨询QQ：2371651507

私塾在线<http://sishuok.com?frombook> 独家提供配套教学视频，更有大量免费在线学习视频独家大放送

## 理解适配器模式

### n 适配器模式的优缺点

- 1: 更好的复用性
- 2: 更好的可扩展性
- 3: 过多的使用适配器，会让系统非常零乱，不容易整体进行把握

## 思考适配器模式

### n 适配器模式的本质

适配器模式的本质是：转换匹配，复用功能

### n 何时选用适配器模式

- 1: 如果你想要使用一个已经存在的类，但是它的接口不符合你的需求，这种情况可以使用适配器模式，来把已有的实现转换成你需要的接口
- 2: 如果你想创建一个可以复用的类，这个类可能和一些不兼容的类一起工作，这种情况可以使用适配器模式，到时候需要什么就适配什么
- 3: 如果你想使用一些已经存在的子类，但是不可能对每一个子类都进行适配，这种情况可以选用对象适配器，直接适配这些子类的父类就可以了