

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 学习原型模式

一：初识原型模式

包括：定义、结构、参考实现

二：体会原型模式

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：理解原型模式

包括：认识原型模式、Java中的克隆方法、浅度克隆和深度克隆、
原型管理器、原型模式的优缺点

四：思考原型模式

包括：原型模式的本质、何时选用

做最好的在线学习社区

网 址：<http://sishuok.com>

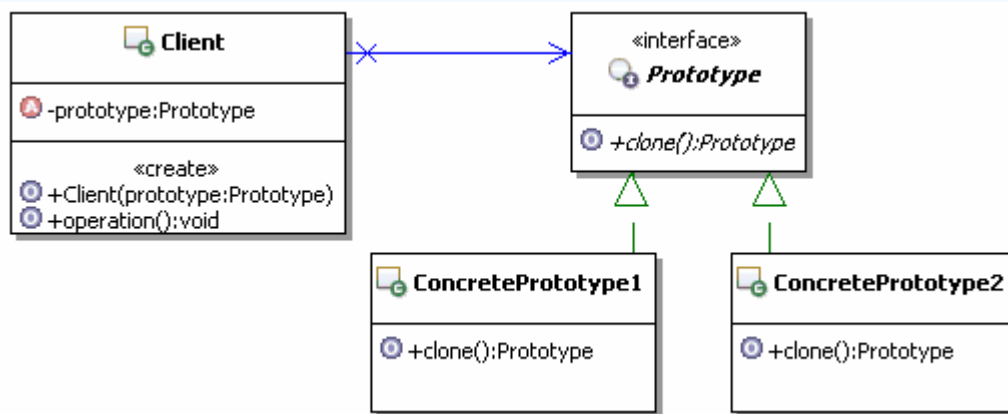
咨询QQ：2371651507

初识原型模式

n 定义

用原型实例指定创建对象的种类，并通过拷贝这些原型创建新的对象。

n 结构和说明



Prototype: 声明一个克隆自身的接口，用来约束想要克隆自己的类，要求它们都要实现这里定义的克隆方法。

ConcretePrototype: 实现Prototype接口的类，这些类真正实现克隆自身的功能

Client: 使用原型的客户端，首先要获取到原型实例对象，然后通过原型实例克隆自身来创建新的对象实例。

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

体会原型模式

n 订单处理系统

考虑这样一个实际应用：订单处理系统。

现在有一个订单处理的系统，里面有个保存订单的业务功能，在这个业务功能里面，客户有这么一个需求：每当订单的预定产品数量超过1000的时候，就需要把订单拆成两份订单来保存，如果拆成两份订单后，还是超过1000，那就继续拆分，直到每份订单的预定产品数量不超过1000。至于为什么要拆分，原因是好进行订单的后续处理，后续是由人工来处理，每个人工工作小组的处理能力上限是1000。

根据业务，目前的订单类型被分成两种：一种是个人订单，一种是公司订单。现在想要实现一个通用的订单处理系统，也就是说，不管具体是什么类型的订单，都要能够正常的处理。

该怎么实现呢？

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

体会原型模式

n 不用模式的解决方案

n 存在的问题

仔细想想，真的没有关心订单的类型和具体实现吗？答案是“否定的”。

事实上在实现订单处理的时候，上面的实现是按照订单的类型和具体实现来处理的，就是instanceof的那一段。有朋友可能会问，这样实现有何不可吗？

这样的实现有如下几个问题：

- (1)既然想要实现通用的订单处理，那么对于订单处理的实现对象，是不应该知道订单的具体实现的，更不应该依赖订单的具体实现。但是上面的实现中，很明显订单处理的对象依赖了订单的具体实现对象。
- (2)这种实现方式另外一个问题就是：难以扩展新的订单类型。假如现在要加入一个大客户专用订单的类型，那么就需要修改订单处理的对象，要在里面添加对新的订单类型的支持，这算哪门子的通用处理。

因此，上面的实现是不太好的，把上面的问题再抽象描述一下：已经有了某个对象实例后，如何能够快速简单地创建出更多的这种对象？

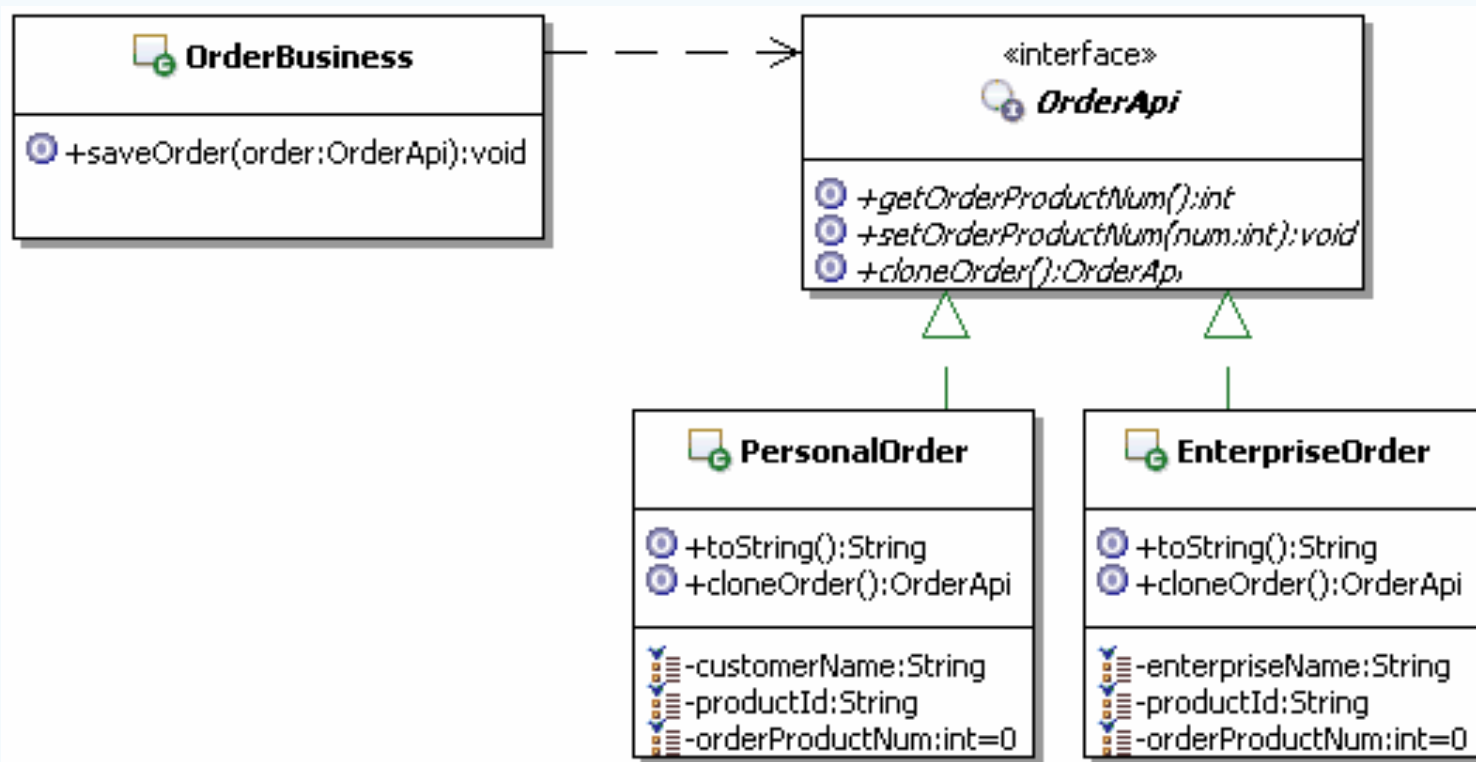
做最好的在线学习社区

网 址：<http://sishuok.com>

咨询QQ：2371651507

体会原型模式

n 使用模式的解决方案



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

理解原型模式

n 认识原型模式

1: 原型模式的功能

原型模式的功能实际上包含两个方面：

(1)一个是通过克隆来创建新的对象实例

(2)另一个是为克隆出来的新的对象实例复制原型实例属性的值

原型模式要实现的主要功能就是：通过克隆来创建新的对象实例。一般来讲，新创建出来的实例的数据是和原型实例一样的。但是具体如何实现克隆，需要由程序自行实现，原型模式并没有统一的要求和实现算法。

2: 原型与new

原型模式从某种意义上说，就像是new操作，在前面的例子实现中，克隆方法就是使用new来实现的，但请注意，只是“类似于new”而不是“就是new”。

克隆方法和new操作最明显的不同就在于：new一个对象实例，一般属性是没有值的，或者是只有默认值；如果是克隆得到的一个实例，通常属性是有值的，属性的值就是原型对象实例在克隆的时候，原型对象实例的属性的值。

做最好的在线学习社区

网 址：<http://sishuok.com>

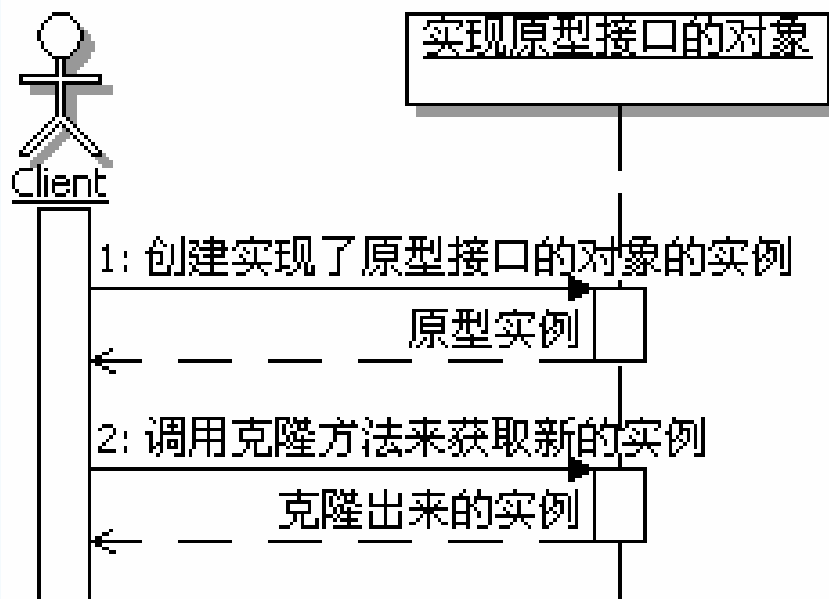
咨询QQ：2371651507

理解原型模式

3: 原型实例和克隆的实例

原型实例和克隆出来的实例，本质上是不同的实例，克隆完成后，它们之间是没有关联的，如果克隆完成后，克隆出来的实例的属性的值发生了改变，是不会影响到原型实例的。

4: 原型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

理解原型模式

n Java中的克隆方法

在Java语言中已经提供了clone方法，定义在Object类中。需要克隆功能的类，只需要实现java.lang.Cloneable接口，这个接口没有需要实现的方法，是一个标识接口。

n 浅度克隆和深度克隆

什么是浅度克隆？什么是深度克隆呢？简单地解释一下：

- (1)浅度克隆：只负责克隆按值传递的数据（比如：基本数据类型、String类型）
- (2)深度克隆：除了浅度克隆要克隆的值外，还负责克隆引用类型的数据，基本上就是被克隆实例所有的属性的数据都会被克隆出来。

深度克隆还有一个特点，如果被克隆的对象里面的属性数据是引用类型，也就是属性的类型也是对象，那么需要一直递归的克隆下去。这也意味着，要想深度克隆成功，必须要整个克隆所涉及的对象都要正确实现克隆方法，如果其中有一个没有正确实现克隆，那么就会导致克隆失败。

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

理解原型模式

n 原型管理器

如果一个系统中原型的数目不固定，比如系统中的原型可以被动态的创建和销毁，那么就需要在系统中维护一个当前可用的原型的注册表，这个注册表就被称为原型管理器。

其实如果把原型当成一个资源的话，原型管理器就相当于一个资源管理器，在系统开始运行的时候初始化，然后运行期间可以动态的添加资源和销毁资源。从这个角度看，原型管理器就可以相当于一个缓存资源的实现，只不过里面缓存和管理的是原型实例而已。

有了原型管理器过后，一般情况下，除了向原型管理器里面添加原型对象的时候是通过new来创造的对象，其余时候都是通过向原型管理器来请求原型实例，然后通过克隆方法来获取新的对象实例，这就可以实现动态管理、或者动态切换具体的实现对象实例。

理解原型模式

n 原型模式的优缺点

- 1: 对客户端隐藏具体的实现类型
- 2: 在运行时动态改变具体的实现类型
- 3: 深度克隆方法实现会比较困难

思考原型模式

n 原型模式的本质

原型模式的本质是：克隆生成对象

n 何时选用原型模式

- 1: 如果一个系统想要独立于它想要使用的对象时，可以使用原型模式，让系统只面向接口编程，在系统需要新的对象的时候，可以通过克隆原型来得到
- 2: 如果需要实例化的类是在运行时刻动态指定时，可以使用原型模式，通过克隆原型来得到需要的实例