

在方法签名中使用控制反转（IoC）[2] PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/167/2021_2022__E5_9C_A8_E6_96_B9_E6_B3_95_E7_c104_167025.htm 依赖项方法将方法参数

定义成接口。调用组件将实现这些接口的对象（故意不称之为值对象）实例化，并在这两个方法调用中将该对象用作方法参数。以下例子突出显示了该方法的一些优势。例子：定价和交货期 再次假设 method1 决定订单的交货期

，method2 决定价格。清单 1 是这些组件和方法的简单定义：

清单 1. 使用接口作为方法参数的组件样例 interface

```
LeadtimeComponent { void getLeadtimes(List items) throws  
LeadtimeException.}interface LeadtimeItem { Long getArticleId().  
BigDecimal getQuantity(). String getQuantityUnit(). void  
setLeadtimeInDays(Integer leadtime).}interface PricingComponent  
{ void getPrices(List items) throws PricingException.}interface  
PriceItem { Long getArticleId(). BigDecimal getQuantity(). String  
getQuantityUnit(). void setPrice(BigDecimal price). void  
setPriceUnit(String currency).} 请注意这两个接口为检索商品数据的方法定义了极为相似的方法签名：getArticleId()
```

、getQuantity() 和 getQuantityUnit()。还要注意组件方法没有返回值；它们通过对参数对象（即接口）调用 setter 方法来修改提供的“现成”对象，以设置价格和交货期。这个方法简化了管道模式（参见参考资料）的实现，在该模式下，数据通过“管道”从一个组件输送到下一个组件，管道的一个阶段（组件）使用该管道在前面的步骤中提供的数据。图 3 显示了使用管道模式的一个序列图：图 3. 使用管道模式准备订

单的序列图 在本例中，订单准备过程首先从购物车中读取商品 ID。然后从分类数据库添加更多信息、检索交货期和价格（其中的价格依赖于交货期）并存储购物车中的其他信息，以便使用这些信息决定最终价格。如果读取购物车所返回的条目对象实现了分类、交货期和价格方法所要求的接口，则该过程不需要进行任何复制。工厂方法 您可能想了解图 3 中的 OrderDB 组件及其 readCart() 方法。确实，这是一个特殊情况。在前面的例子中，依赖项方法（如 getPrices(...)）修改过的所有对象均已作为方法参数进行传递。当组件正在从数据库中读取数据时这是不可能的，因为在本例中，购物车中的条目数量在读取前是未知的。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com