接口的 默认方法的优势  
  
默认方法的主要优势是提供一种拓展接口的方法，而不破坏现有代码。加入我们有一个已经投入使用接口需要拓展一个新的方法，在JDK8以前，如果为一个使用的接口增加一个新方法，则我们必须在所有实现类中添加该方法的实现，否则编译会出现异常。如果实现类数量少并且我们有权限修改，可能会工作量相对较少。如果实现类比较多或者我们没有权限修改实现类源代码，这样可能就比较麻烦。而默认方法则解决了这个问题，它提供了一个实现，当没有显示提供其他实现时就采用这个实现。这样新添加的方法将不会破坏现有代码。  
默认方法的另一个优势是该方法是可选的，子类可以根据不同的需求Override默认实现。例如，我们定义一个集合接口，其中有增、删、改等操作。如果我们的实现类90%都是以数组保存数据，那么我们可以定义针对这些方法给出默认实现，而对于其他非数组集合或者有其他类似业务，可以选择性复写接口中默认方法。(由于接口不允许有成员变量，所以本示例旨在说明默认方法的优势，并不具有生产可能性)具体参照如下代码：

问题：1.当继承的父类和实现的接口中有相同签名的方法时，优先使用父类的方法。（“类优先原则”）

2.当接口的父接口中也有同样的默认方法时，就近原则调用子接口的方法。

3.当实现的多个接口中有相同签名的方法时，必须在实现类中通过重写方法解决冲突问题，否者无法通过编译，在重写的方法中可以通过：

接口名.super.方法名(); 的方式显式调用需要的方法