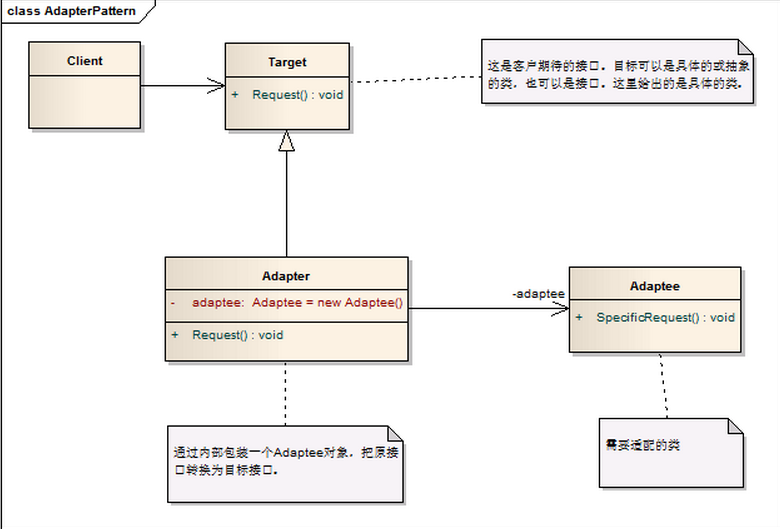
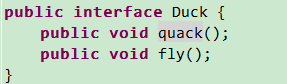
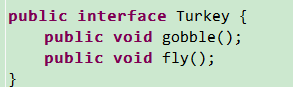
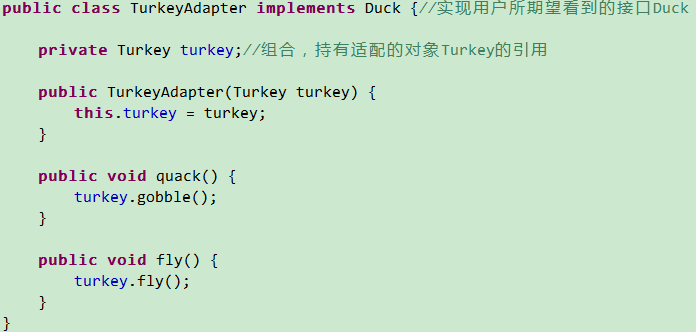
1. 适配器模式：将一个类的接口转换成客户期望的另一个接口，让原本不兼容的类可以合作无间(类似于插头转换)。(使原本由于接口不兼容不能一起工作的类可以一起工作)
2. 对象适配器类图(类适配器需要多重继承，java中不能实现)

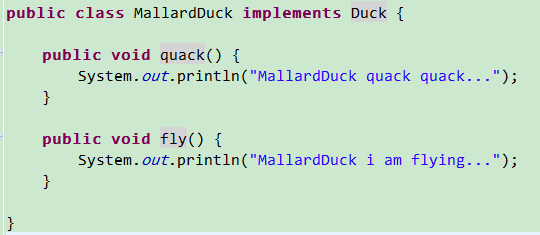


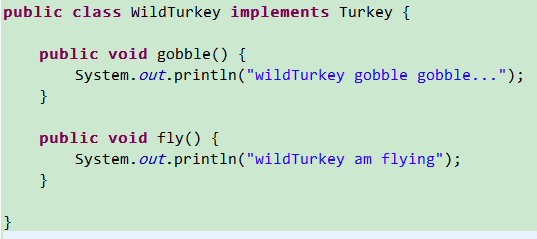
1. 实例, 用户需要Duck接口, 我们已有Turkey能实现类似功能, 但是与Duck接口不兼容, 创建一个TurkeyAdapter作为适配器来满足用户需要

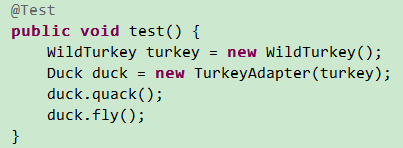












1. 适配器模式的三个特点
   1. 适配器对象实现原有接口(Duck)
   2. 适配器对象组合一个实现新接口(Turkey)的对象（这个对象也可以不实现一个接口，只是一个单纯的对象）
   3. 对适配器原有接口方法的调用被委托给新接口的实例的特定方法
2. 常见场景:
   1. java.io.InputStreamReader#InputStreamReader(java.io.InputStream)
   2. java.io.OutputStreamWriter#OutputStreamWriter(java.io.OutputStream)