主要用来做方法的增强，让你可以在不修改源码的情况下，增强一些方法，在方法执行前后做任何你想做的事情（甚至根本不去执行这个方法），因为在***InvocationHandler***的invoke方法中，你可以直接获取正在调用方法对应的***Method***对象，具体应用的话，比如可以添加调用日志，做事务控制等。  
  
还有一个有趣的作用是可以用作远程调用，比如现在有Java接口，这个接口的实现部署在其它服务器上，在编写客户端代码的时候，没办法直接调用接口方法，因为接口是不能直接生成对象的，这个时候就可以考虑代理模式（动态代理）了，通过***Proxy.newProxyInstance***代理一个该接口对应的**InvocationHandler**对象，然后在**InvocationHandler**的invoke方法内封装通讯细节就可以了。具体的应用，最经典的当然是Java标准库的RMI，其它比如hessian，各种webservice框架中的远程调用，大致都是这么实现的。

作者：KevinShn  
链接：https://www.zhihu.com/question/20794107/answer/23334315  
来源：知乎  
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

补充一下。  
动态代理是JAVA的一大特性。  
动态代理的优势就是实现无侵入式的代码扩展。  
目前动态代理主要分为JAVA自己提供的动态代理和CGLIB类似框架。  
JAVA自带的动态代理是需要接口的。CGLIB这种则是直接修改字节码