interface MailService{  
 void send(String msg);  
}  
class DefalutMailService implements MailService{  
  
 @Override  
 public void send(String msg) {  
 System.*out*.println(msg);  
 }  
}  
*//基于OCP原则通过继承(is a)实现功能扩展*class MyMailService extends DefalutMailService{  
 @Override  
 public void send(String msg) {  
 System.*out*.println("start："+System.*currentTimeMillis*());  
 super.send(msg);  
 System.*out*.println("end："+System.*currentTimeMillis*());  
 }  
}  
*//基于OCP(开闭原则)通过组合方式实现对DefaultMailService对象的功能扩展*class LogMailService{  
 *//has a* private MailService service;  
 public LogMailService(MailService service){  
 this.service = service;  
 }  
 public void send(String msg){  
 System.*out*.println("start："+System.*currentTimeMillis*());  
 service.send(msg);  
 System.*out*.println("end："+System.*currentTimeMillis*());  
 }  
}  
public class TestCompose {  
 public static void main(String[] args) {  
 MailService service = new MyMailService();  
 service.send("MyMailService");  
 LogMailService service1 = new LogMailService(new DefalutMailService());  
 service1.send("LogMailService");  
 }  
}