// 模板方法，make，模板方法，可以做成final，不让子类去覆盖  
// 但是我想问下，子类可以使用 抽象类 定义的 final修饰的普通方法吗？？？  
// 这是 不行的，所以，这个子类对象 的 类型【存在向上转型的特点】  
// 你要明确一点，抽象类 不存在 对象，对象只能由其子类 创建  
// 其实本身在是属于 抽象类类型【静态绑定】，此时可以调用抽象类的普通方法，只是说其子类没有覆盖，故没有【动态绑定】  
// 综上：子类 只是 使用到 向上转型的特点 来 调用 抽象类 的 final修饰的方法  
// 【所以，我还是将其归类为多态的形式，具体见下面的注释。】  
  
*/\*\*  
 \*  
 \* 对于多态，可以总结它为：  
 \* 　　一、使用父类类型的引用指向子类的对象；  
 \* 　　二、该引用只能调用父类中定义的方法和变量；【静态绑定】  
 \* 　　三、如果子类中重写了父类中的一个方法，那么在调用这个方法的时候，将会调用子类中的这个方法；（动态连接、动态调用）【动态绑定】  
 \* 　　四、变量不能被重写（覆盖），”重写“的概念只针对方法，如果在子类中”重写“了父类中的变量，那么在编译时会报错。  
 \*/*