德米特法则的分析，体现依赖倒置设计实例的分析。

德米特法则实现的特点:德米特法则的好处在于使用了组合类的客户代码只需要知道组合本身，不需要知道组合的构件。德米特法则其根本思想，是强调了类之间的松耦合， 我们在程序设计时，类之间的耦合越弱，越有利于复用， 一个处于弱耦合的类被修改， 不会对有关系的类造成波及。即信息的隐藏促进了软件的复用。

依赖倒置设计的特点：如果类和类直接依赖于细节，那么它们之间就有直接的耦合，当具体实现需要变化时，意味着要同时修改依赖着的代码，这限制了系统的可扩展性。依赖倒置指代了一种特定的解耦形式，使得高层次的模块不依赖于低层次的模块的实现的细节的目的，依赖模块被颠倒了。高层模块不依赖低层模块，两者都应该依赖其抽象；抽象不应该依赖细节；细节应该依赖抽象。优点是提高了扩展性；使代码层次更加清晰；降低了耦合度。