（一）单一职责原则（Single responsibility Principle SRP）

注意职责扩散

（二）里氏替换原则（Liskov Substitution Principle LSP）

（1）任何基类可以出现的地方，子类一定可以出现

（2）子类可以扩展父类的功能，但不能改变父类原有的功能。

（三）依赖倒置原则（Dependency Inverse Principle DIP）

（1）高层模块不应该依赖于低层模块，高层模块和低层模块都应该依赖于抽象。关联一个抽象接口

（2）抽象不应该依赖于细节，细节应该依赖于抽象。具体实现抽象接口

（四）接口隔离原则（Interface Segregation Principle ISP）

建立单一接口，不要建立庞大臃肿的接口，尽量细化接口，接口中的方法尽量少。也就是说，我们要为各个类建立专用的接口，而不要试图去建立一个很庞大的接口供所有依赖它的类去调用

（五）迪米特法则（Least Knowledge Principle 简写LKP）

只与直接的朋友通信。首先来解释一下什么是直接的朋友：每个对象都会与其他对象有耦合关系，只要两个对象之间有耦合关系，我们就说这两个对象之间是朋友关系。耦合的方式很多，依赖、关联、组合、聚合等。其中，我们称出现成员变量、方法参数、方法返回值中的类为直接的朋友，而出现在局部变量中的类则不是直接的朋友。

（六）开闭原则（Open Close Principle OCP）

（1）（2）不错的设计

（3）（4）有设计缺陷

（5）设计严重缺陷

（6）过度设计