# 面向对象心得体会

在我看来：面向对象三大特征、六大原则主要是为了解决两个实际问题：需求变更、分工合作。

程序源于需求，当拿到需求首先会将需求拆分成功能点，然后再按照一定的技术架构实现业务逻辑。功能点可以用封装的技术实现，要讲究“**功能单一原则**”和“**开闭原则**”，尽量减少功能间耦合，不要修改原有代码，以减少影响的扩散。方便日后的需求变更处理和协同合作。

设计程序时，需引入“抽象”的概念而不要过分依赖现实，这样更符合人的思考方式。因此面向对象便有了继承这个特征，将多个同一概念的类抽象出一个父类，将具体变化封装为子类。这样使用“**依赖倒置**”原则，实现用与做的隔离。从分工合作角度上讲，继承可以提供一个统一的子类规范便于多人合作。

子类应符合“**里式替换原则**”，即在父类的方法上扩展，而不能改变父类原有方法。子类的这种方法上的特性即“多态”。多态可以实现共性中的特性，完成程序的业务实现细节。

程序如果想更好的支持需求变更，还要尽可能的低耦合。

从依赖关系上讲：优先考虑依赖(实例对象作为方法参数)，其次考虑关联(实例对象作为构造函数参数)，最后考虑泛化(即继承)。客户端依赖的接口，应传尽量少的数据，即“**组合复用原则**”和“**迪米特原则**”。

六大原则可以看作是用来修饰三大特征的，属于理想化的状态，但是现实中的开发项目一定是会受到成本、进度、团队能力等综合因素的影响。因此工作中要考虑实际情况，不可完全理想化。要做一个“活在现实中，胸怀理想的程序员”。

最后，除了三特征六原则外，还想说两个很重要的题外话：

1. 写代码要养成良好的编写规范：注释、命名规则、逻辑复杂度等等

要符合标准，能够“让别人能看明白”。

1. 一定要在正确的时候，敢于怼产品经理！！ ~