

面向对象设计需要注意的问题 PDF转换可能丢失图片或格式
，建议阅读原文

https://www.100test.com/kao_ti2020/141/2021_2022__E9_9D_A2_E5_90_91_E5_AF_B9_E8_c29_141419.htm

摘要：本文根据个人经验和实践，详细地讨论和分析在面向对象的设计过程中，需要注意的问题。

1．面向对象技术 思维方式决定解决问题的方式，传统软件开发采用自顶向下的思想指导程序设计，即将目标划分为若干子目标，子目标再进一步划分下去，直到目标能被编程实现为止。面向对象技术给软件设计领域带来极大的变化，它利用软件对象来进行程序开发，所谓对象是包含数据和对数据操作的代码实体，或者说是在传统的数据结构中加入一些被称为成员函数的过程，因而赋予对象以动作。而在程序设计中，对象具有与现实世界的某种对应关系，我们正是利用这种关系对问题进行分解。从程序语言角度来看，在一个对象中代码和（或）数据可以是这个对象私有的，不能被对象外的部分直接访问。因而对象提供了一种高级保护以防止程序被无关部分错误修改或错误地使用了对象的私有部分。当从对象外部试图直接对受保护的内部数据进行修改时，将被程序拒绝，只有通过对象所提供的对外服务函数才能够对其内部数据进行必要的加工，从而保证了数据加工的合法性。从这一意义上讲，把这种代码和数据的联系称为"封装"。换句话说，封装是将对象封闭保护起来，是将内部细节隐蔽起来的能力。

2．面向对象设计要点 下面是根据我个人的设计经验，总结的在面向对象设计时需要注意的问题。（1）为实际工作设计：软件是一种工具，一种实现目标的方法，所以我们的软件设计一定要建立在软件需求

基础上，根据需求确定软件的工作范围，再根据确定的工作范围来进行软件设计，另外，我们不要迷失在各种技术的诱惑之中，不能为了尝试某些新技术而试图解决实际不存在的问题，这样会使软件设计处于一片混乱之中。（2）理解要实现的东西：我们应该在建立模型上花大量的时间，偶尔写一些源代码来验证设计过程中所遇到的问题，这样做可以使我们的设计方案更加可行。如果我们不了解我们的软件设计要实现什么目标，这个设计注定是失败的。（3）需求的重要性：留出时间努力去充分收集相关信息和分析软件用户的需求。让用户参与到这个过程中，但不要盲目接受用户或客户提出的功能需求。保证在开发过程中引入一种有效的方法去收集、组织、验证支撑工作的信息，要学习问题领域的描述语言和掌握相关工作的流程。如果没有需求，就不要动手做任何事情，成功的软件取决于时间、预算和是否满足用户的需求，如果不能确切地知道用户需要的是什麼，那么所做的软件必定是失败的。（4）在现有任务中应用多个模型：当收集需求的时候，应该考虑使用用例模型、用户界面模型和领域级的类模型。当设计软件时，应该考虑制作类模型、顺序图、状态图、协作图和最终的软件实际物理模型。仅仅使用一个模型而实现的软件要么不能满足用户的需求，要么很难扩展。（5）用例的重要性：在一个软件系统中，用例可大可小，但它必须是对一个具体的用户目标实现的完整描述。通过用例图把系统和外界的交互描述出来，以便来鉴别和划分系统功能。（6）文档的重要性：拥有准确的技术文档不仅对于整个开发小组非常有益，而且也能让客户从中受益。由于软件开发在某种程度上要依赖技术文档来进行说明

，因此文档必须十分准确可靠。在编写文档时，要明确该文档的目的和对象，同时，要保证文档之间的前后一致性。

(7) 证明软件的设计在实践中是可行的：在设计的时候应先建立一个技术模型，或者称为"点到点"的原型，用此来证明我们的设计是可行的。如果软件的设计方案是不可行的，在编码时采用任何策略都是毫无用处的。(8) 应用已知的模式：一般来说，应该避免重新设计已经成熟的并被广泛使用的设计方法或者设计模式。(9) 类的内聚性：一个类应该有且仅有一个职责。所谓一个类的职责是指引起该类变化的原因，如果一个类具有一个以上的职责，那么就会有多个不同的原因引起该类变化，其实就是耦合了多个互不相关的职责，降低这个类的内聚性。(10) 充分考虑软件的可移植性：当使用了某个操作系统的特性，或者利用某个数据库专用语言写了存储过程，这个软件和特定产品的结合度就已经很高了。所以，为了增强软件的可移植性，应该把这些特有的实现细节封装在一个类中，这样，当它们改变的时候，只需要更改那个类的代码就可以了。(11) 建立对象数据辞典：为了便于内部重用和共享，应该建立电子化的对象数据辞典，以便对对象进行统一归类管理。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com