

java认证:Java开发中类隐藏及应用场景分析Java认证考试 PDF  
转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/584/2021\\_2022\\_java\\_E8\\_AE\\_A4\\_E8\\_AF\\_81\\_c104\\_584694.htm](https://www.100test.com/kao_ti2020/584/2021_2022_java_E8_AE_A4_E8_AF_81_c104_584694.htm) 隐藏是JAVA语言平台中提出来的一个新名词。其次在其他语言平台上，也有类似的说法，只是称呼不一样而已。但是隐藏的功能对于应用程序开发来说，特别是大型应用程序设计，非常的有用。作为一名合格的JAVA平台程序开发人员，应该结合JAVA语言提供的三个关键字，来实现类的隐藏。

### 一、类隐藏的基本建议

虽然在JAVA语言中要实现类的隐藏非常简单，只需要通过三个关键字即可以完成。但是在大部分情况下，让JAVA程序员感到疑惑的是，在什么时候该把什么类隐藏。确实，在JAVA开发语言平台中，对于什么类需要隐藏，什么类不需要隐藏，往往没有很严格的要求。为此程序开发人员需要根据自己的工作经验来进行判断。而即使有一定程序开发经验的员工，也不一定能够在这个问题上给出很好的答案。公说公有理，婆说婆有理，为此很难有一个统一的标准。笔者下面几个建议，也是对自己这几年来JAVA开发工作的总结。希望这些建议能够帮助大家解惑。

#### 1、不同模块之间的类要相互隐藏。

现在不少应用系统，为了提高开发效率，都提出了模块化的设计理念。如最近比较流行的ERP系统，如果按模块来进行设计的话，则在开发时不同的人员可以负责各自的模块，而互不干涉。在ERP系统出售的时候，也可以按模块来进行出售，因为各个模块的功能是相对独立的。当然这只是从管理层面来说模块化设计的重要性，不过这不是我们今天要谈论内容的重点。今天要谈的是，如果应用系统按模块化来进行设

计，那么不同模块之间的内如何实现隐藏呢?其实很简单。如供应商基本信息维护、供应商进货单、应付帐款分属于三个不同的模块，分别为采购管理、库存管理与应收应付管理。现在在设计应收应付这个模块时其中有一个生成应付帐款的类，其需要用到采购管理模块中供应商的信息(付款条件等等)、仓库管理模块中的供应商进货单据(产品编号、数量、到货日期等等)，然后才能够生成准确的应付帐款。那么在应用程序设计的时候该如何实现呢?是在那个应付帐款作业对应的类中直接从数据库中去查询这些信息吗，又或者去修改采购模块中对应的类来实现?这虽然都可以达到预定的目的，但是都不是上上之策。个人认为，最好的方法就是程序开发人员先设计一个类(如Vender类)，这个类会返回供应商编号、付款条件等信息。然后应付帐款作业中的类就直接调用这个类获取需要的信息。一方面可能很多地方都需要用到这些信息，如果每次需要的时候都利用SQL代码来获取这些信息的话，那么代码量也会很大。相反直接调用类来实现这些信息反而更加的方面。另一方面，这个Vender类对于其他类是隐藏的。也就是说，其他类只能够通过预定义的接口向这个类传递参数并直接引用其返回的结果，而不能够修改类内部的元素。如此即使其他有十几个类或者应用程序需要调用这个类的时候，其只要输入的参数相同，则其返回的结果也就相同。那么其他类在引用返回结果的时候，就不用担心不兼容了。所以，应用程序的模块化设计促使JAVA程序要实现类与类之间的隐藏.同时JAVA类相互之间隐藏的特性也方便了应用程序的模块化设计。

## 2、数据类型类与其他类之间需要实现隐藏。

在任何一个应用程序开发过程中，都需要根据企业的

需要设计一些比较有针对性的数据类型。因为光靠应用程序所提供的那些数据类型往往不能够满足程序开发人员的需要。如需要设计一些数组、记录等等数据类型。但是在设计这些数据类型的时候，程序人员就需要了一个难题。如现在在开发ERP系统的过程中，多个模块(每个模块由不同的程序员负责)都需要用到同一个记录数据类型。那么是每个程序员都设计一个记录类型的数据类型，还是共用同一个数据类型呢？如果各管各的，就需要建立多个记录类型的数据类型，管理起来比较麻烦，对后续升级也会遇到障碍。而如果大家共用一个记录类型的数据类型，大家又担心其他程序人员不小心修改了这个数据类型而给他们造成了麻烦。这之间难道就没有和平的共存方式了呢？其实，在应用程序开发的过程中可以认为的将应用程序开发人员分为数据类型创者者与业务逻辑开发者。然后数据类型开发者可以根据其他开发人员的需要创建一些定义数据类型的类，这些类的话对于其他程序开发人员开发的类是隐藏的。也就是说，其他类不能够对这个数据类型进行修改。那么即可以提高这些数据类型的利用率，而且也可以保证其他类可以以固定的格式引用这些类。

### 3、业务层与表示层之间的类要相互隐藏。

利用过J2EE开发平台的人肯定知道，在开发应用程序的时候，业务逻辑层与表示层之间是相互独立的。业务逻辑层主要实现一些负责的运算，如物料需求计划运算。而表示层的话只负责一些具体的显示。如通过浏览器客户端来显示结果还是通过软件窗口来显示运算结果。虽然客户端所采用的方式不同，但是其内部的运算是相同的。此时最好把实现业务逻辑层的类与实现表示层的类相互隐藏起来。因为在实际开发过程中，他们往往是

由不同的程序开发人员完成的。为了避免不同程序开发人员之间的相互干扰，避免表示层的开发人员无意中修改了业务逻辑层的JAVA类，就必须要把业务逻辑层的类隐藏起来。也就是说，表示层开发人员只需要知道业务逻辑层的类需要哪些参数、能够以什么样的形式返回结果即可。然后表示层的类在调用业务逻辑层中的类的时候，只需要按照要求输入参数、按照规则获取结果即可，而不需要关注其业务逻辑层类内部的实现机制，从而减轻了表示层开发人员的工作量。同时也避免了业务层开发人员对这个类进行修改，从而其他的应用程序开发人员可以以同一种方式来引用这些类。

4、如果某个业务逻辑还不怎么完善时，要执行隐藏。在应用程序开发的过程中，可能某些业务逻辑暂时没有很好的解决方案，但是需要的参数或者返回的结果是确定的。此时可以把这个业务逻辑建立一个的类，并对其他类进行隐藏。如此的话，当出现更好的解决方案之后，就可以直接更改这个类中的业务逻辑。由于类中成员对于其他程序人员是隐藏的，为此对类中的处理机制进行更改并不会影响到其他人员开发的应用程序。他们只关心你能够在最短时间内获得准确的结果。为此在遇到这种情况时，最好把类与其他的类执行隐藏。可见，哪些类需要相互隐藏可能判断起来比较复杂。但是有一个普遍适用的原则，就是如果一个类有不同的程序人员需要引用。如用户A设计了一个统计应收帐款余额的类，用户B与C，包括用户A都需要调用这个类。那么这个类就最好进行隐藏。防止其他类在调用这个类的时候，不小心修改了其内部的实现方式。这会对其他用户调用这个类产生不利影响。

二、实现JAVA类隐藏的三个关键字 之所有要实现JAVA类隐

藏，主要是让任何程序开发人员都无法触及他们不应该初期的部分（如业务逻辑或者数据类型的内部操作）。虽然项目管理员可能在程序开发过程中多次告诫团队成员不要直接操作其他开发人员所设计类中的成员。但是如果缺乏一定的控制手段，这些事情往往会在有意无意中发生。为此需要采取一些强制的手段来防止这些情况的发生。而且这些实现方式必须简单，不能够太过于复杂。在JAVA语言平台中，就提供了Public、private、protected三个关键字来实现对类的隐藏与控制。这些关键字决定了类或者类中的成员是否对其他类是隐藏的，能否被其他类进行操作与修改。如public关键字表示紧随其后的元素对任何人都是可用的，可以被任何人所修改。而关键字Private则表示除了类创建者和这个类型的内部方法之外的任何人都不能够进行访问。而关键Protected的作用跟Private类似，只是其是作用在继承中的。可见在JAVA程序开发中，要实现类的隐藏是很简单的，也是JAVA程序员必须要掌握的一项基本功。而其难度就在于在适当的时候实现类和类中成员的隐藏。这就要程序员在程序开发过程中好好琢磨了。上面提到的一些建议可供大家做参考。更多优质资料尽在百考试题论坛 百考试题在线题库 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](http://www.100test.com)