

java认证:访问控制符、修饰符和其它关键字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\\_ti2020/564/2021\\_2022\\_java\\_E8\\_AE\\_A4\\_E8\\_AF\\_81\\_c104\\_564723.htm](https://www.100test.com/kao_ti2020/564/2021_2022_java_E8_AE_A4_E8_AF_81_c104_564723.htm) 访问控制符的作用是说明被声明的内容（类、属性、方法和构造方法）的访问权限，就像发布的文件一样，在文件中标注机密，就是说明该文件可以被那些人阅读。访问控制在面向对象技术中处于很重要的地位，合理的使用访问控制符，可以通过降低类和类之间的耦合性（关联性）来降低整个项目的复杂度，也便于整个项目的开发和维护。具体的实现就是通过访问控制符将类中会被其它类调用的内容开放出来，而把不希望别人调用的内容隐藏起来，这样一个类开放的信息变得比较有限，从而降低了整个项目开放的信息，另外因为不被别人调用的功能被隐藏起来，在修改类内部隐藏的内容时，只要最终的功能没有改变，即使改变功能的实现方式，项目中其它的类不需要更改，这样可以提高了代码的可维护性，便于项目代码的修改。在Java语言中访问控制权限有4种，使用三个关键字进行表达，依次如下：  
gt.| gt.public公共的 gt.| gt.protected受保护的 gt.|  
gt.无访问控制符默认的 gt.| gt.private私有的 其中无访问控制符是指不书写任何的关键字，也代表一种访问权限，访问控制符的使用示例如下所示：  

```
public class AccessControl { int n.  
public AccessControl(){ init(). }private void init(){ } protected void  
test(int k){ } }
```

该示例代码中演示了各个访问控制符的实际使用示例，其中属性n的访问控制符就是默认的。在实际使用时，类声明的访问控制符只有2个：public和无访问控制符，属性声明、构造方法声明和方法声明的访问控制符可以是以上4

种中的任何一个。这4个访问控制符的权限作用如下表所示

：访问控制符同一个类内部同一个包内部不同包中的子类不同包中的非子类  
public Yes Yes Yes Yes protected Yes Yes Yes No 无访问控制符 Yes Yes No No private Yes No No No 说明：在该表中

，Yes代表具备对应的权限，No代表不具备对应的权限。在4种访问控制中，public一般称作公共权限，其限制最小，也可以说没有限制，使用public修饰的内容可以在其它所有位置访问，只要能访问到对应的类，就可以访问到类内部public修饰的内容，一般在项目中开放的方法和构造方法使用public修饰，开放给项目使用的类也使用public修饰。protected一般称作继承权限，使用protected修饰的内容可以被同一个包中的类访问也可以在不同包内部的子类中访问，一般用于修饰只开放给子类的属性、方法和构造方法。无访问控制符一般称作包权限，无访问控制符修饰的内容可以被同一个包中的类访问，一般用于修饰项目中一个包内部的功能类，这些类的功能只是辅助其它的类实现，而为包外部的类提供功能

。private一般称作私有权限，其限制最大，类似于文件中的绝密，使用private修饰的内容只能在当前类中访问，而不能被类外部的任何内容访问，一般修饰不开放给外部使用的内容，修改private的内容一般对外部的实现没有影响。下面以两个基本的示例来说明访问控制符在实际项目中的使用方式。第一个使用示例：在项目中，一般不会将类的属性开放给其它的类，也就是不允许外部的类直接访问属性，而是使用对应的存取方法来进行访问。例如在学校的学员管理系统中，需要实现的学生类，按照访问控制符的一般使用规则，实现的代码如下：/\*\* \* 学员类，演示访问控制符的使用 \*/

```
public class Student { /**年龄*/ private int age. /**学员ID*/ private
int id. public int getAge() { return age. } public void setAge(int age) {
if(age < 0){ //处理代码，未实现 } this.age = age. } public int
getId() { return id. } public void setId(int id) { //校验id是否合法的
代码，未实现 this.id = id. }} 通过将属性的访问权限设定
为private，限制所有类外部对属性的访问，而为了让外部可
以访问这些属性，专门声明对应的get/set方法来读取/存储数
据，这样在设置属性值的set方法中，可以对于参数做出基本
的校验，在上面的示例代码中，留出了校验参数的位置，具
体的代码未在示例代码中实现。 小技巧：在eclipse中，get
和set方法可以在选中对应类的代码以后，使用“Source”菜
单中的“Generate Getters and Setters...”菜单实现。 第二个使
用示例：在项目中，一般为了设计的需要实现一些特定的功
能，下面介绍一下使用访问控制符实现的一个功能使一个类
既不能创建对象也不能被继承。实现的方法如下：该类中只
实现一个构造方法，而且将该构造方法的访问权限设置为私
有的。具体实现代码如下：/** * 不能创建对象且不能被继承
的子类 */ public class PrivateDemo { private PrivateDemo(){} } 在
该示例中，PrivateDemo类只有一个构造方法，且该构造方法
为私有。按照以前的介绍，创建对象时需要调用构造方法，
而private修饰的构造方法无法在类的外部进行访问，所以无
法创建对象。另外，在子类的构造方法中也需要调用父类的
构造方法，由于private的构造方法无法得到调用，所以该类
也不可以有对应的子类。 这里说明的只是两个基本的用途，
在实际的项目中，可以根据需要灵活的使用访问控制符实现
对应的功能。 总之，访问控制符通过控制声明的内容的访问
```

权限，实现对于内容的隐藏，从而降低使代码的耦合性降低，降低项目的复杂度，并且方便实际项目中代码的维护。更多优质资料尽在百考试题论坛 百考试题在线题库 java认证更多详细资料 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)