**Java权限修饰符private、default（一般省略）、public、protected**

权限修饰符概念：

1. private：Java语言中对访问权限限制的最窄的修饰符，一般称之为“私有的”。被其修饰的属性以及方法只能被该类的对象访问，其子类不能访问，更不能允许跨包访问。
2. default：即不加任何访问修饰符，通常称为“默认访问权限“或者“包访问权限”。该模式下，只允许在同一个包中进行访问。
3. protected：介于public 和 private 之间的一种访问修饰符，一般称之为“保护访问权限”。被其修饰的属性以及方法只能被类本身的方法及子类访问，即使子类在不同的包中也可以访问。
4. public： Java语言中访问限制最宽的修饰符，一般称之为“公共的”。被其修饰的类、属性以及方法不仅可以跨类访问，而且允许跨包访问。

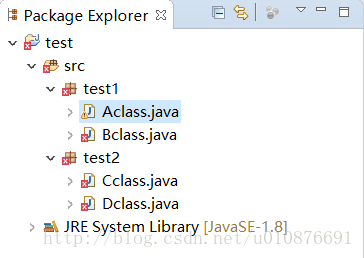
注意点：

这里需要注意的是，所谓的访问，可以分为两种不同方式：第一，通过对象实例访问；第二，直接访问。   
***比如说，***某父类protected权限的成员，子类是可以直接访问的，换一种说话是子类其实继承了父类的除了private成员外的所有成员，包括protected成员，所以与其说是子类访问了父类的protected成员，不如说子类访问了自己的从父类继承来的protected成员。另一方面，如果该子类与父类不在同一个包里，那么通过父类的对象实例是不能访问父类的protected成员的。

## 二、下面用表格来展示四种修饰符的访问权限范围：

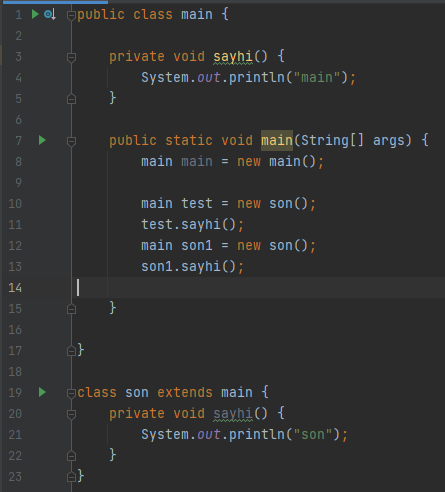
| **权限** | **类内** | **同包** | **不同包子类** | **不同包非子类** |
| --- | --- | --- | --- | --- |
| private | √ | × | × | × |
| default | √ | √ | × | × |
| protected | √ | √ | √ | × |
| public | √ | √ | √ | √ |

## 三、举个例子

在Eclipse中新建个类,四个类的结构如下，Aclass 和Bclass 是同包的类，Cclass 和Dclass是同包的类，Cclass 继承了Aclass。   


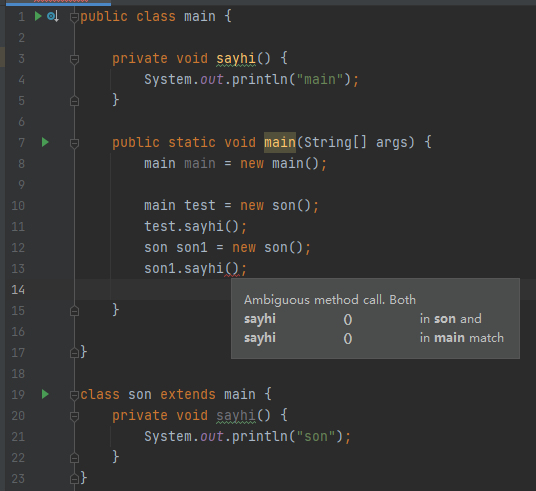
最近在学习到Java的继承和多态时遇到了这样的一个问题：关于继承链中子类对父类的私有方法是否可以覆盖的问题，在此记录一下自己对这个问题解决以后的一些心得。

首先要明确：子类是不能够覆盖（重写）父类的私有方法的。比较直观的理由是，子类要能够重写父类的方法的前提条件是继承，子类对父类的私有方法并不继承，也没有访问权限，自然也是不能重写的。



son 根本没继承main的sayhi方法，所以没有重写，这里son只是调用了main的私有方法





这里在son的外部调用了son的私有方法，是不被允许的

调用私有方法的正确的做法



