## 基本内部类结构

class Outer{  
 private String msg = "hello,world";  
 class Inner{ //内部类,这里的this表示内部类的实例,外部类.this表示外部类的实例  
 private String info = "世界,你好";  
 public void print(){  
 System.*out*.print(msg); //内部类直接访问外部属性和方法(不受防控属性影响)  
 //System.out.print(Outer.this.msg); 正规写法  
 }  
 }  
 public void fun(){  
 Inner in = new Inner();

//Inner in = new Outer.Inner();也可以这么写  
 System.*out*.print(in.info); //外部类通过实例化内部类对象直接访问属性和方法(不受防控属性的影响)  
 }  
}

**注意点:**

1. **内部类里的this表示内部类的实例,内部类能访问到**

**外部类的实例:外部类.this**

**内部类能通过 外部类.this 能访问所有外部类的方法和属性(不受防控属性的影响),就一个类的this一样,可以访问这个类的所有方法和属性**

1. **外部类的方法里能直接实例化内部类对象,并且这个内部类对象能访问内部类的所有方法和属性(不受防控属性的影响)**

## 访问内部类

1. public class test{  
    public static void main(String args[]){  
    //由于内部类可能会用到外部类的方法和属性,所以必须先实例化外部  
    //类,再实例化内部类  
    Outer.Inner in = new Outer().new Inner();  
    in.print();  
    //Outer out = new Outer();  
    //Outer.Inner in = out.new Inner();  
    }  
   }  
   class Outer{  
    private String msg = "hello,world";  
    class Inner{  
    public void print(){  
    System.*out*.print(msg);  
    }  
    }  
   }

外部的程序访问内部类受到防控属性的影响,内部类的防控属性可以为public,private,default,protecetd,和方法,属性的防控属性一样

**如果是内部类是非static,则外部程序实例化内部类，需要先实例化外部类,所以格式为:**

外部类名称.内部类名称 实例 = new 外部类名称().new 内部类名称()

**如果是内部类是static,则不受外部实例影响,并且成了一个外部类，则格式为:**

外部类名称.内部类名称 实例 = new 外部类名称.内部类名称()

范例如下:

public class test{  
 public static void main(String args[]){  
 Outer.Inner in = new Outer.Inner();  
 in.print();  
 }  
}  
class Outer{  
 private String msg = "hello,world";  
 static class Inner{  
 public void print(){  
 System.*out*.print("hahahha");  
 }  
 }  
}

## 在方法中定义内部类

内部类可以在任意位置定义:类中,代码块,方法里(常用)

d