带两个类型参数的泛型类

在泛型中，可以声明一个以上的类型参数，只需要在这些类型参数之间用逗号隔开。下面看一个简单的例子。

//本类带有两个类型参数  
public class twoGen<T,V>{  
 T ob1;  
 V ob2;  
 //构造方法也可以使用这两个类型参数  
 twoGen(T o1, V o2){  
 ob1 = o1;  
 ob2 = o2;  
 }  
 //显示T和V的类型  
 void showTypes(){  
 System.out.println("Type of T is "+ob1.getClass().getName());  
 System.out.println("Type of V is "+ob2.getClass().getName());  
 }  
 T getOb1(){  
 return ob1;  
 }  
 V getOb2(){  
 return ob2;  
 }  
}

下面这个程序演示流如何使用上面这个泛型类。

public class simpGen{  
 public static void main(String args[]){  
 twoGen<Integer, String> tgObj;   //指定类型参数的实际类型  
 //构造方法中需要再次指定类型参数，同时还要传递实际参数  
 tgObj = new twoGen<Integer, String>(100,"Hello");  
 tgObj.showTypes();  
 int v = tgObj.getOb1();  
 System.out.println("value: "+v);  
 String  str = tgObj.getOb2();  
 System.out.println("value: "+str);  
 }  
}

程序的输出结果如下：

Type of T is java.lang.Integer  
Type of V is java.lang.String  
value: 100  
value: Hello

与只有一个类型参数的泛型相比，本例并没有什么难于理解的地方。Java并没有规定这两个类型参数是否要相同，比如，下面这样来创建对象也是可以的：

twoGen<String, String> tgObj = new twoGen<Integer, String> ("Hello","World");

这样T和V都是String类型，这个例子并没有错。但如果所有的实例都是如此，就没有必要用两个参数。