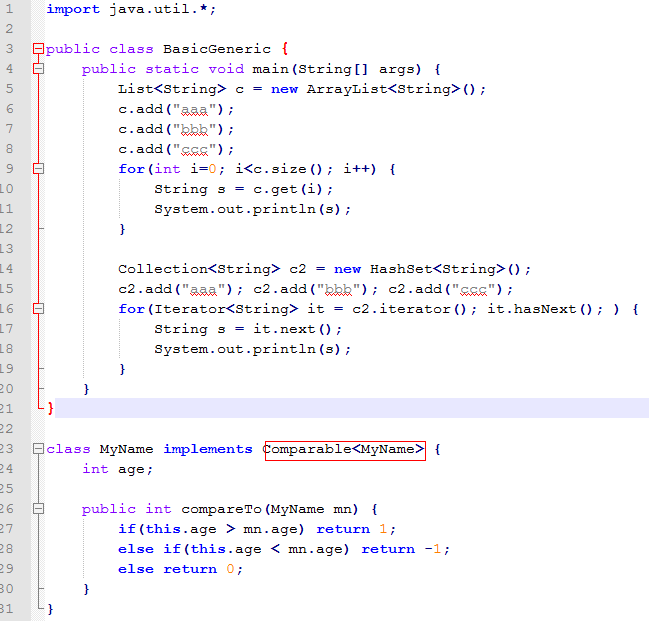
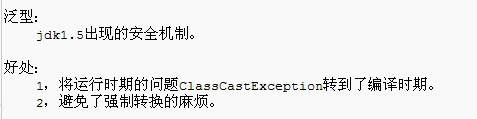
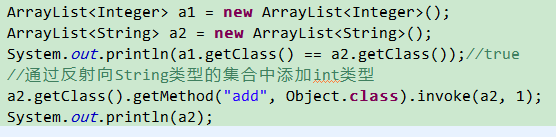
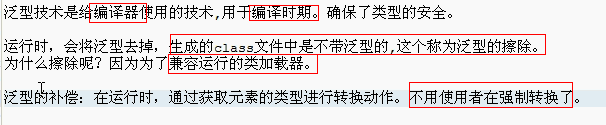
1. JDK1.5泛型：
   1. 起因：JDK1.4以前类型不明确
      1. 装入集合的类型都被当作Object对待，从而失去自己的实际类型。
      2. 从集合取出时往往需要转型，效率低，容易产生错误
   2. 解决办法：在定义集合的时候同时定义集合中对象的类型。
   3. 好处：增强程序的可读性和稳定性。





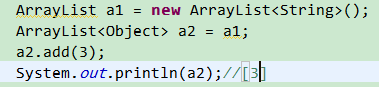
1. 泛型是提供给Javac编译器使用的，可以限定集合中的输入类型，让编译器挡住源程序中的非法输入，编译器编译完带类型说明的集合时会**去除掉**“类型”信息，使程序运行效率不受影响。对于参数化的泛型类型，getClass()方法的返回值和原始类型完全一样。由于编译生成的字节码会去除掉泛型的类型信息，只要能跳过编译器，就可以往某个泛型集合中加入其他类型的数据，例如，用反射得到集合，再调用其add方法即可。

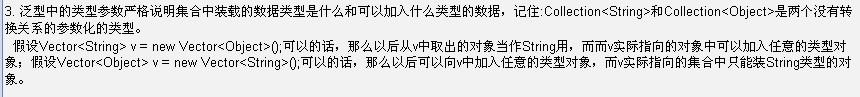




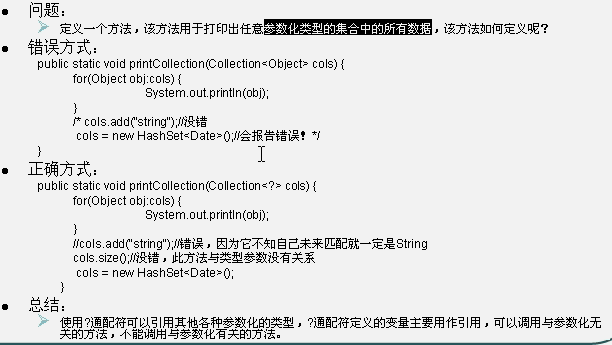
1. 了解泛型

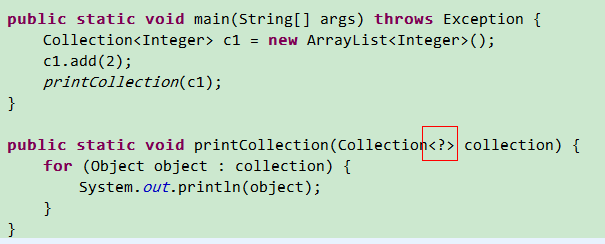






1. 泛型中的?通配符

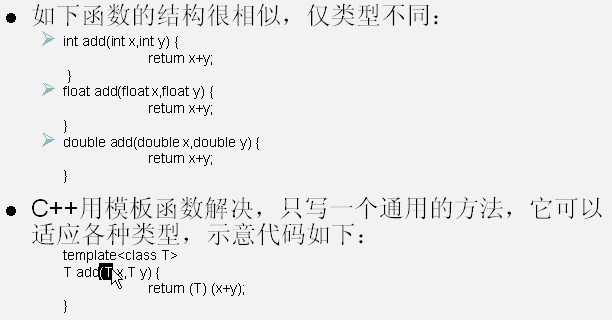


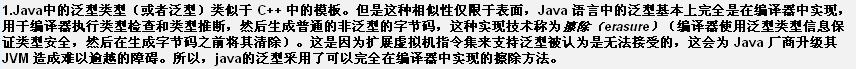


1. 泛型中的?通配符的扩展

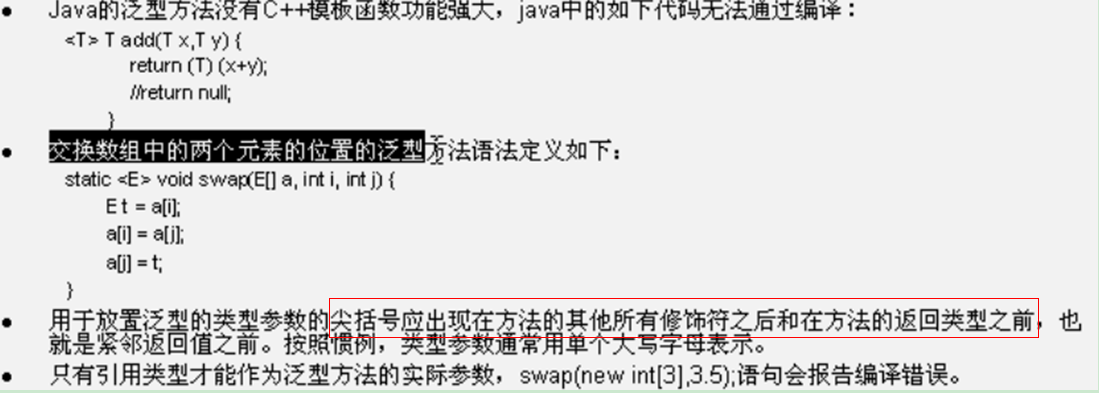


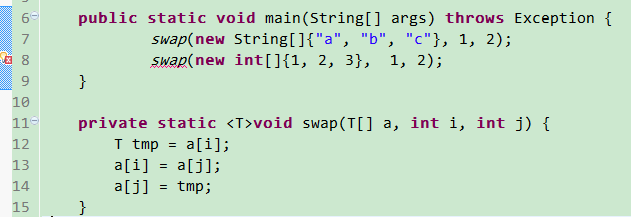
1. 自定义泛型方法以及应用

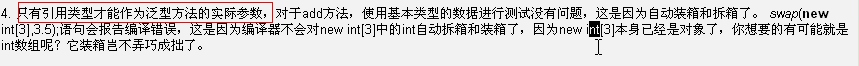




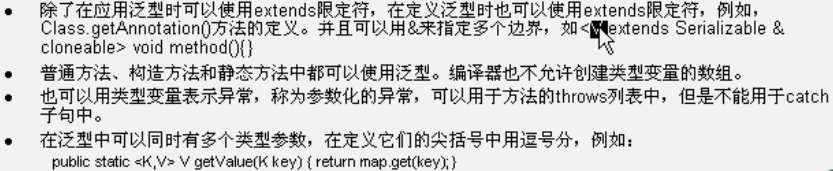
1. 定义泛型方法

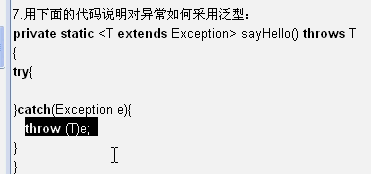




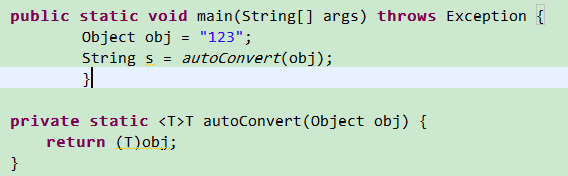


T不能被int等基本数据类型替换，只能被引用类型替换。

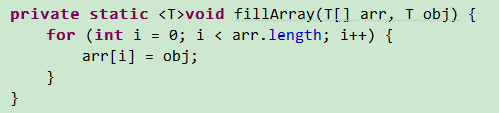




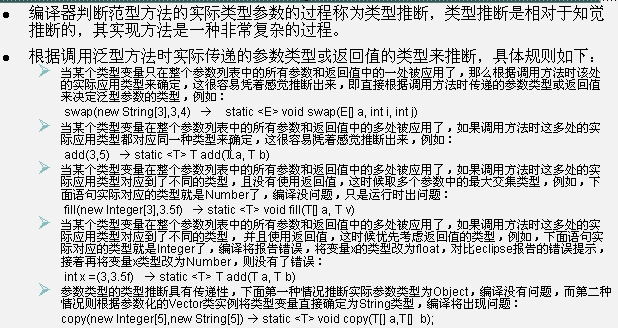
1. 泛型方法的练习题
   1. 



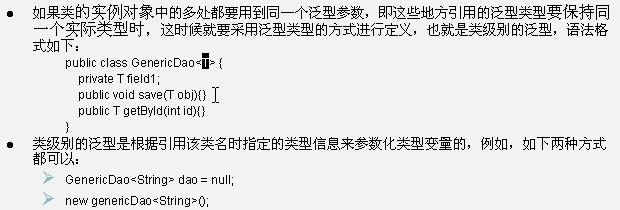
* 1. 

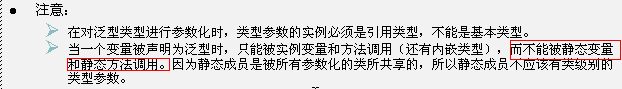


1. 类型参数的类型推断



1. 定义泛型类





1. 通过反射获得泛型的实际类型参数

