**实验报告**

计63 国雨晴 2016050024

**第一题** 1.1: 没有正确使用封装概念.

Fred的成员变量x是public的,这意味着外部类可以任意地访问修改它.应该改为private,并且提供方便的更改x的public方法.

**第一题** 1.2: interface和abstract class都不可以被实例化,只能被其他类或接口实现或继承.interface中不能包含实例域,只能有静态方法和常量,以及一些default方法(在JAVA SE 8后),在设计框架中往往是”请求类提供一些特定的功能”,如Comparator接口使类提供”比较”的功能;abstract class中可以有实例域,在框架上往往是子类的”抽象”等,与之功能上密切相关.

class中是不可以包含未定义的函数的,并且也必须对所有implement的interface和expand的父类中未定义的函数给出定义.否则,这个类也必须是abstract的.interface也可以用expand关键字合并别的多个interface,并且不用实现它们.class只能expand一个父类,但可以implement多个interface.

**第二题**: 代码见文件夹2.

各对象可以调用的方法:

man可以使用changeSomething(返回--count),getName(返回”man”),getDescription(返回”nothing”)和move(输出I am moving…).

superman可以使用changeSomething(返回++count),getName(返回”superman”),getDescription(返回”nothing”),move(输出I am flying…)和fly(输出” fly,I am a SuperMan”).

pman可以使用changeSomething(返回--count),getName(返回”pman”),getDescription(返回”nothing”).

psman可以使用changeSomething(返回--count),getName(返回”psman”),getDescription(返回”nothing”).

msMan可以使用changeSomething(返回++count),getName(返回”msMan”),getDescription(返回”nothing”),move(输出I am flying…).

代码分析:

(1)

|  |
| --- |
| Man man=new Man("man","nothing");  SuperMan sman=(SuperMan)man; |

不可以.让SuperMan类的引用sman来指向Man类对象,会导致SuperMan可以调用子类扩展的方法和实例域,编译器会检查出这种错误.

(2)

|  |
| --- |
| Man man=new SuperMan("superman","nothing");  SuperMan sman=(SuperMan)man;  Man man2=(Man)sman; |

可以.第一行和第三行用Man类型引用SuperMan对象,是允许的;第二行强制转换,用SuperMan类型引用替换原来Man类型的引用,但是由于对象还是SuperMan型的,这种做法也是允许的.

(3)

|  |
| --- |
| Person man=new Man("man","nothing");  SuperMan sman=(SuperMan)man; |

不可以.第一行用Person引用指向Man对象,由于Person是Man实现的接口,虽然不能实例化Person类,但是可以用Person的引用指向其子类的实例,是可以的;而第二行试图用SuperMan类指向Man类型的实例,这是非法的(同问题1).

**第三题**:代码见文件夹3.

思路:为达到”只创建一个实例”的效果,需要有一个静态变量记录”是否创造过实例”;同时,为了以后能方便地引用这个实例,可以把这个实例的引用以静态变量的方式存在这个类中.