**上机实践5 求方程的根**

**1. 相关知识点**

**包是Java语言中有效地管理类的一个机制**。通过关键字package声明包语句，package语句作为Java源文件的第一条语句，指明该源文件定义的类所在的包。

使用import语句可以引入包中的类。在编写源文件时，除了自己编写类外，经常需要使用Java提供的许多类，这些类可能在不同的包中。在学习Java语言时，使用己经存在的类，避免一切从头做起，这是面向对象编程的一个重要方面。**为了能使用Java提供的类，可以使用import语句来引入包中类。**在一个Java源程序中可以有多个import语句，它们必须写在package语句（假如有package语句的话）和源文件中类的定义之间。

**2. 实验目的**

本实验的目的是让学生掌握使用package和import语句。

**3. 实验要求**

按照实验要求使用package语句将方程的属性即计算根的方法封装在一个有包名的类中，包名是tom.jiafei，类的名字是SquareEquation。编写一个SunRise的主类，该主类使用 import 语句引入 tom.jiafei 包中的 SquareEquation 类。

**4.程序模板**

**模板1**

将模板1给出的Java源文件命名为SquareEqtiationjava,将编译后得到的字节码文件 复制到 C:/1000/tom/jiafei 目录中。

**SquareEquation. Java**

package tom.jiafei;

public class SquareEquation {

double a,b,c;

double root1,root2;

boolean boo;

public SquareEquation(double a,double b,double c) {

this.a=a;

this.b=b;

this.c=c;

if(a!=0)

boo=true;

else

boo=false;

}

public void getRoots() {

if(boo) {

System.out.println("是一元2次方程");

double disk=b\*b-4\*a\*c;

if(disk>=0) {

root1=(-b+Math.sqrt(disk))/(2\*a);

root2=(-b-Math.sqrt(disk))/(2\*a);

System.out.printf("方程的根:%f,%f\n",root1,root2);

}

else {

System.out.printf("方程没有实根\n");

}

}

else {

System.out.println("不是一元2次方程");

}

}

public void setCoefficient(double a,double b,double c) {

this.a=a;

this.b=b;

this.c=c;

if(a!=0)

boo=true;

else

boo=false;

}

}

**如果文件名写成：SquareEquation. Java**

**(后缀java大写了)**

**会报如下错误**

**错误: 仅当显式请求注释处理时才接受类名称 'SquareEquation.Java'**

**1 个错误**

**模板2**

将模板2给出的Java源程序SunRise.java保存到D:\2000中。在编译模板2给出的Java 源文件之前，要重新设置classpath。假设本地机JDK的安装目录是D:\jdkl.6。

在命令行执行如下命令：

set classpath=D:\jdk1.6\jre\lib\rt.jar;.;c:\1000

然后编译模板2给出的Java源程序。或用鼠标右击“我的电脑”，弹出菜单，然后选择“属性”，弹出“系统特性”对话框，再单击该对话框中的“高级选项”，然后单击“环境变量” 按钮，将classpath的值修改为：

set classpath=D:\jdk1.6\jre\lib\rt.jar;.;c:\1000

然后重新打开一个命令行窗口，编译模板2给出的Java源程序.

**SunRise.java**

import tom.jiafei.\*;

public class SunRise {

public static void main(String args[]) {

SquareEquation equation=new SquareEquation(4,5,1);

equation.getRoots();

equation.setCoefficient(-3,4,5);

equation.getRoots();

}

}

**我在笔记本上的执行：**

**D:\Java\test5>set classpath=C:\Program Files (x86)\Java\jdk-13\lib\rt.jar;.;D:\Java\test5**

**D:\Java\test5>cd tom**

**D:\Java\test5\tom>cd jiafei**

**D:\Java\test5\tom\jiafei>javac -encoding UTF-8 SquareEquation.Java**

**错误: 仅当显式请求注释处理时才接受类名称 'SquareEquation.Java'**

**1 个错误**

**D:\Java\test5\tom\jiafei>javac -encoding UTF-8 SquareEquation.java**

**D:\Java\test5\tom\jiafei>cd..**

**D:\Java\test5\tom>cd..**

**D:\Java\test5>cd..**

**D:\Java>cd 200**

**系统找不到指定的路径。**

**D:\Java>cd 2000**

**D:\Java\2000>javac -encoding UTF-8 SunRise.java**

**D:\Java\2000>java SunRise**

**是一元2次方程**

**方程的根:-0.250000,-1.000000**

**是一元2次方程**

**方程的根:-0.786300,2.119633**

**5.实验指导**

如果使用import语句引入了整个包中的类，那么可能会增加编译时间。但绝对不会影响程序运行的性能。Java运行平台由所需要的Java类库和虚拟机组成，这些类库被包含在一个jre\lib中的压缩文件rt.jar中，当程序执行时，Java运行平台从类库中加载程序真正使用的类字节码到内存。

可以使用import语句引入自定义包中的类，但必须在classpath中指明包的位置。

**6.实验后的练习**

假设JDK的安装目录是D:\jdk1.6，那么Java运行系统默认classpath的值是：

D:\jdk1.6\jre\lib\rt.jar;.;

其中的“.;”表示应用程序可以使用所在当前目录中的无名包类以及当前目录下的子目录中的类，子目录中的类必须有包名，而且包名要和子目录结构相对应。

因此，**如果将模板2应用程序SimRise.java的字节码文件存放到D:\5000中，）并将SquareEquation.java的字节码文件存放在D:\5000\tom\jiafei中，那么就不需要修改dasspath。**

要特别注意的是，因为SquareEquation.java有包名，切不可将SquareEquationjava以及它的字节码文件存放在D:\5000中，**即不可以和SunRise.java的字节码存放在一起**。请进行如下的练习：

(1)将 SquareEquation.java 存放在 D:\5000\tom\jiafei中，编译：

D: \5000\tom\ jiafei> javac SquareEquation .java

**SunRise.java:4: 错误: 无法访问SquareEquation**

**SquareEquation equation=new SquareEquation(4,5,1);**

**^**

**错误的类文件: .\SquareEquation.class**

**类文件包含错误的类: tom.jiafei.SquareEquation**

**请删除该文件或确保该文件位于正确的类路径子目录中。**

**1 个错误**

(2) 将 SunRise.java 存放到 D:\5000 中，编译：

D:\5000> javac SunRise.java

(3) 运行：

D:\5000> java SunRise

**D:\Java\test5>javac -encoding UTF-8 SunRise.java**

**D:\Java\test5>java SunRise**

**是一元2次方程**

**方程的根:-0.250000,-1.000000**

**是一元2次方程**

**方程的根:-0.786300,2.119633**

**7.填写实验报告**