**面向对象程序设计实验**

**实验项目名称: 实验4 类的继承与多态**

**学生姓名: 范少帅 学号： 20181101047 班级： 计科3班**

**提交时间: 2019/10/9 指导教师: 司建辉**

格式说明: 正文 1.25倍行距 小四 宋体

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**（一）实验目的**

1、掌握类的继承的原则、抽象类的声明、多态性的体现；

2、熟练掌握子类的构造方法、子类成员的定义；

3、掌握方法的继承、重载与覆盖。

**（二）实验内容和实验原理**

**1、类的继承**

**交通工具的类继承。**

**父类：Vehicles类，包括name 名称， color 颜色 seats座位等属性，方法showDetail()（输出汽车名称、颜色信息）。**

**子类：car类**

**思路：Car类对象分别调用vehicle类（父类）方法showDetail()（输出汽车名称、颜色信息）和本类方法show()（输出汽车交通工具的名称、交通工具的颜色、座位的数量信息）输出汽车的属性值。**

**2、类的多态**

**请编码实现如下需求：**

**（1）乐器（Instrument）分为：钢琴(Piano)、小提琴(Violin)。**

**（2）各种乐器的弹奏（ play ）方法各不相同。**

**编写一个测试类InstrumentTest，要求：**

**（1）编写方法testPlay，对各种乐器进行弹奏测试。要依据乐器的不同，进行相应的弹奏。**

**（2）在main方法中进行测试。**

**思路：**

**（1）乐器Instrument为父类（抽象类），钢琴(Piano)、小提琴(Violin)为子类；**

**（2）乐器类只有play()方法，钢琴(Piano) 子类和小提琴(Violin)子类有同名方法play()即重写父类乐器的play()方法。**

**（3）测试类InstrumentTest，定义testPlay(Instrument ins)方法，参数类型为Instrument 。**

**（三）实验步骤**

1．在MyEclipse环境中，新建工作区和类；

2．编辑并编译源程序；

3、运行。

**（四）实验报告要求**

1、提交实验源程序；

类的继承：

**public** **class** Car **extends** Vehicles

{

String unique;

**public** Car(String name,String color,**int** seat,String unique){

**super**.name=name;

**super**.color=color;

**super**.seat=seat;

**this**.unique=unique;

}

**public** **void** showDetail(){

**super**.showDetail();

System.out.println(name+"特有的行为有："+unique);

}

}

**public** **class** Vehicles

{

String name;

String color;

**int** seat;

**public** **void** showDetail(){

System.***out***.println("名称："+name+"颜色："+color+"座位数："+seat);

}

}

//class Train extends Vehicles{

// String unique;

// public Train(String name,String color,int seat,String unique)

// {

// super.name = name;

// super.color = color;

// super.seat = seat;

// this.unique = unique;

// }

// public void showDetail()

// {

// super.showDetail();

// System.out.println(name+"特有的行为有："+unique);

// }

//}

**public** **class** Ch2{

**public** **static** **void** main(String[] args){

Car c=**new** Car("客车","绿色",32,"不能拉货");

c.showDetail();

// Train a= new Train("火车","白色",600,"需要固定铁轨");

// a.showDetail();

}

}

**类的多态源代码：**

**public** **abstract** **class** Instrument {

**private** String name;

**public** String getName(){

**return** name;

}

**public** **void** setName(String name){

**this**.name = name;

}

**public** **void** play(){

System.***out***.println(name+"音乐好听！！！");

}

}

**public** **class** Piano **extends** Instrument{

**public** Piano(String name){

**this**.setName(name);

}

**public** **void** play(){

System.***out***.println(getName()+"是弹得！！！");

}

}

**public** **class** Violin **extends** Instrument{

**public** Violin(String name){

**this**.setName(name);

}

**public** **void** play(){

System.***out***.println(getName()+"是拉的！！！");

}

}

**public** **class** InstrumentTest {

**public** String testPlay(Instrument a){

System.***out***.println("test Play:");

a.play();

**return** a.getName();

}

**public** **static** **void** main(String[]args){

Piano piano = **new** Piano("钢琴");

Violin violin = **new** Violin("小提琴");

InstrumentTest b= **new** InstrumentTest();

b.testPlay(violin);

System.***out***.println("测试2：");

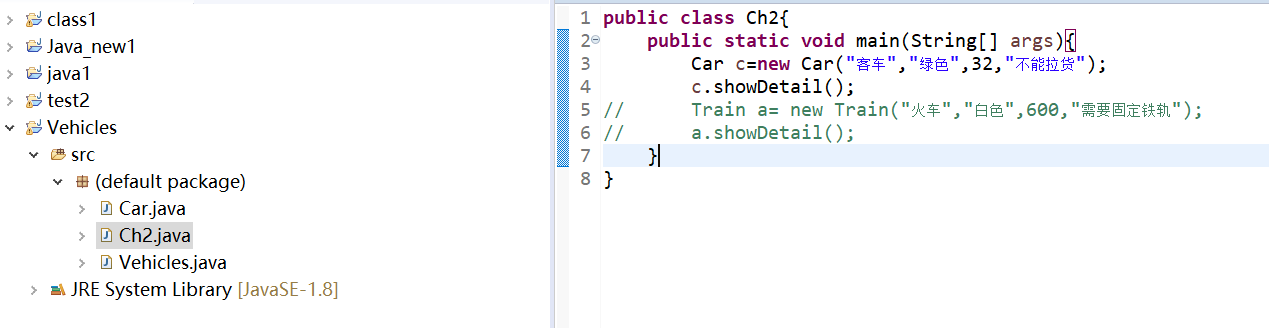
b.testPlay(piano);

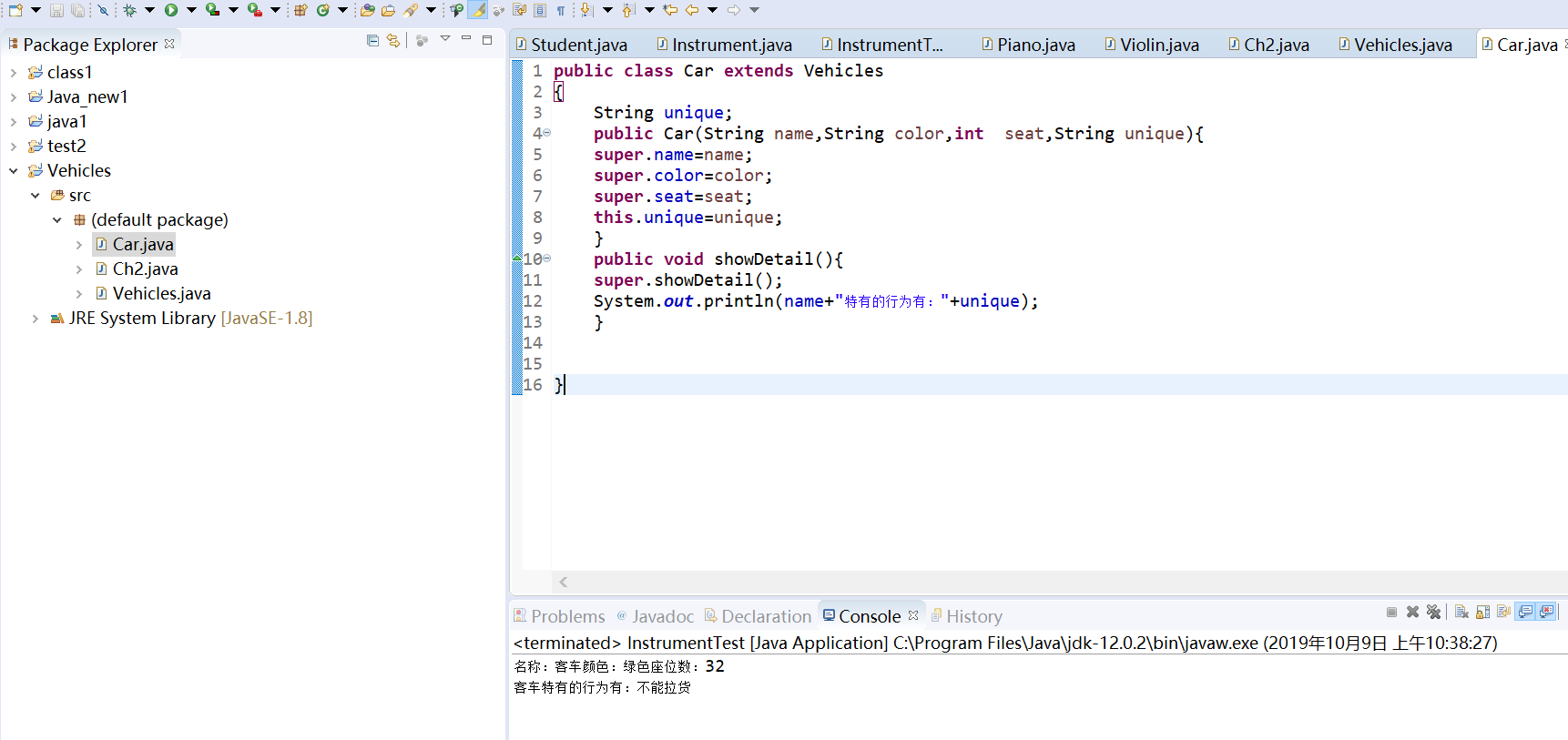
}

}

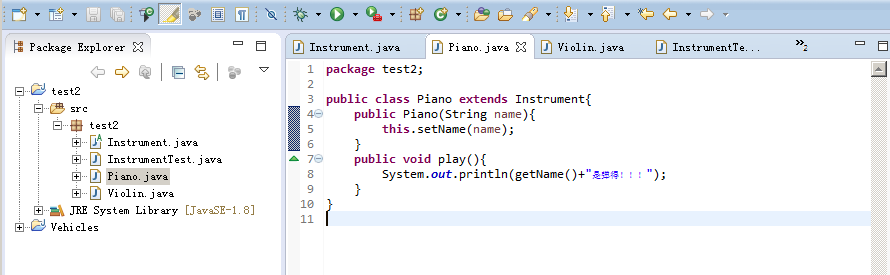
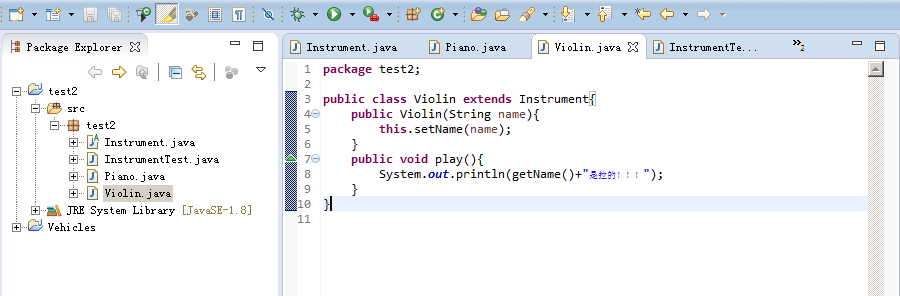
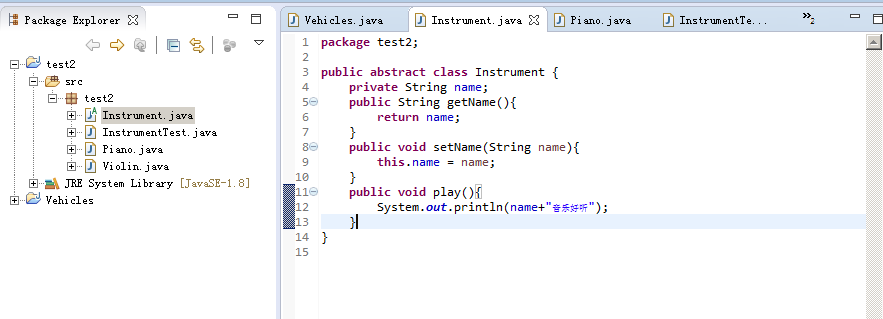
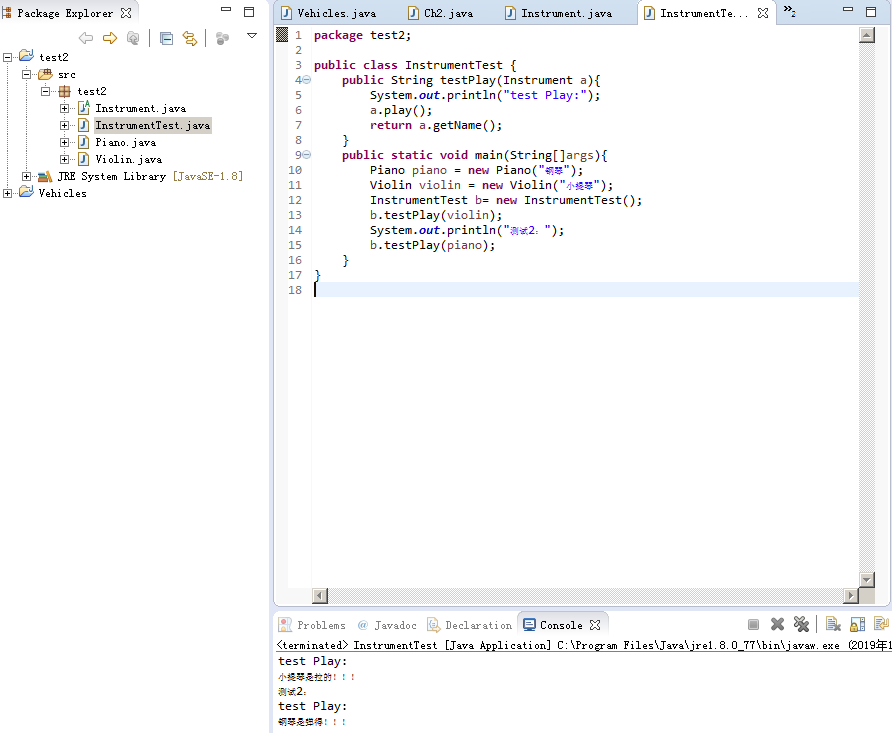
2、提交实验过程的截图；

类的继承：





类的多态截图：



3、撰写心得体会。

1、一个类文件里面一般只有一个public class类，所以在继承的时候应该在新建一个class类内编写。

2、抽象类不能被具体实例化，即不能创建对象。但其可以创建构造函数，所创建的构造函数可以被子类实例化的对象所调用。