《面向对象程序设计与实践》实验报告

实验名称： 《面向对象程序设计与实践》实验五

学 期： 2022春季学期 任课教师： 陆歌皓 学院 软件学院

专业： 计算机 学号： 20211120138 姓名： 薛凡豪 成绩：

# 一、实验源代码及运行结果：

## 中国人、北京人和美国人

### （1）问题分析：

1.本实验主要需要运用的是子类的继承性和子类的写法及应用

2.第一个类要写出原始的父类（即People类）

3.第二三个写出中国人和美国人class作为People的子类

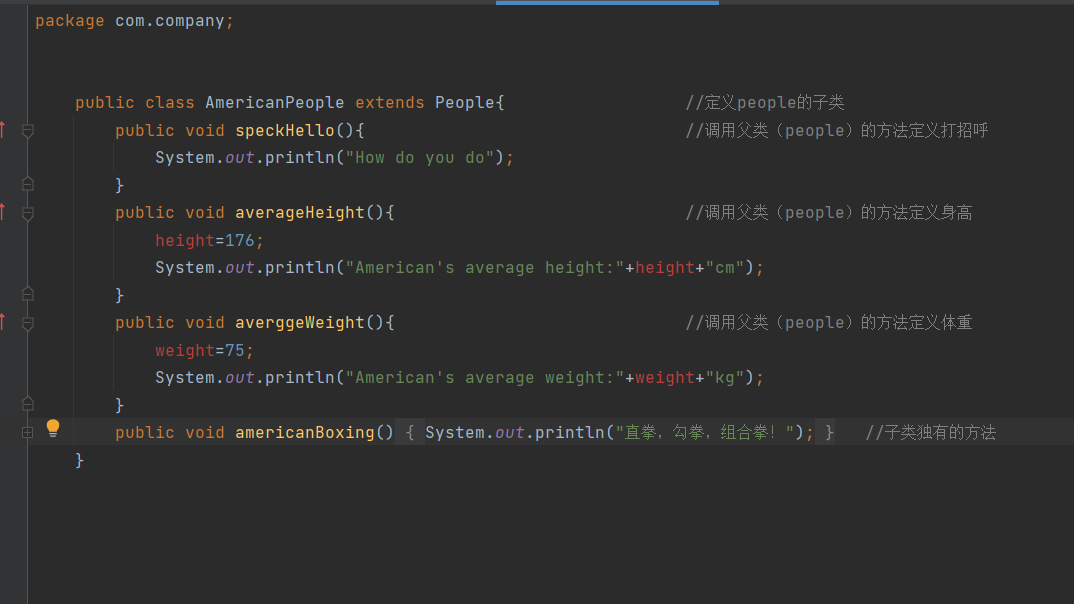
4.最后一个类是北京人class，作为ChinaPeople的子类，即People的孙子类

5.最后的运行类调用方法来输出

### （2）程序的运行和结果





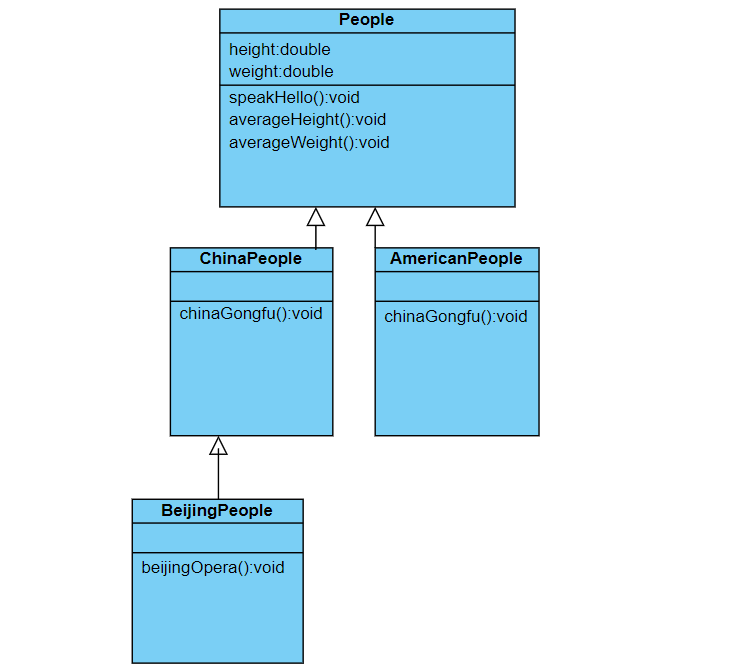




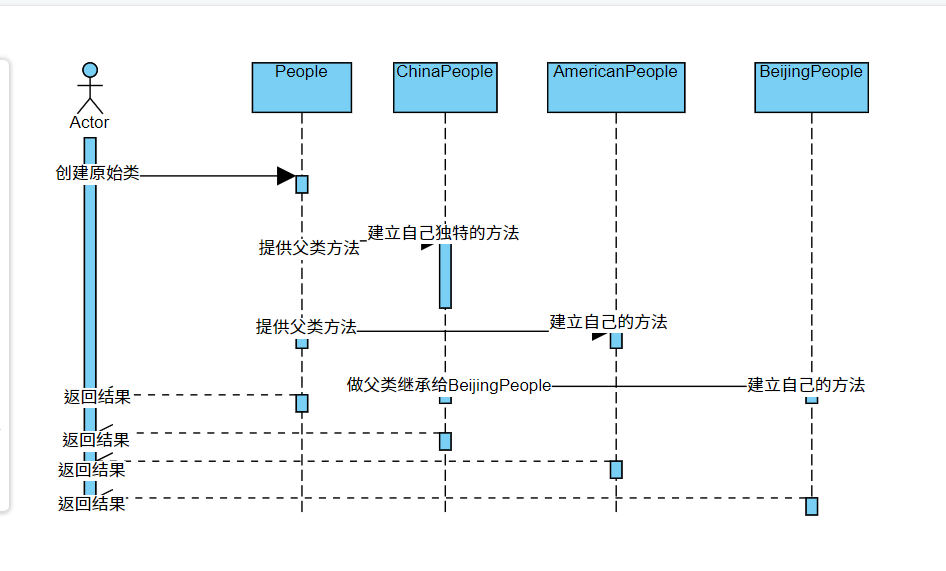


### （3）UML图

类图：



时序图：



### （4）源代码：

**一、People.java**

package com.company;  
  
public class People {  
  
    protected double weight;                  //定义保护型变量，可以被同包的其他类调用  
    protected double height;  
  
    public void speckHello() {                //public方法定义人类打招呼的方式，可以被调用  
        System.out.println("yayaya");  
    }  
  
    public void averageHeight() {             //同上  
        height = 173;  
        System.out.println("average height:" + height);  
    }  
  
    public void averggeWeight() {              //同上  
        weight = 70;  
        System.out.println("average weight:" + weight);  
    }  
}

————————————————————————————————

**二、ChinaPeople.java**

package com.company;  
  
public class ChinaPeople extends People {                 //定义成People的子类  
    public void speckHello(){  
        System.out.println("您好");                        //调用父类（people）的方法定义打招呼  
    }  
    public void averageHeight(){                          //调用父类（people）的方法定义身高  
        height=168.78;  
        System.out.println("中国人的平均身高:"+height+"厘米");  
    }  
    public void averggeWeight(){                           //调用父类（people）的方法定义体重  
        weight=65;  
        System.out.println("中国人的平均体重:"+weight+"千克");  
    }  
    public void chinaGongfu(){                              //子类独有的方法（父类没有）  
        System.out.println("坐如钟，站如松，睡如弓！/n20211120138薛凡豪");  
    }

}

**三、AmericanPeople**

package com.company;  
  
  
    public class AmericanPeople extends People{                  //定义people的子类  
        public void speckHello(){                                //调用父类（people）的方法定义打招呼  
            System.out.println("How do you do");  
        }  
        public void averageHeight(){                             //调用父类（people）的方法定义身高  
            height=176;  
            System.out.println("American&apos;s average height:"+height+"cm");  
        }  
        public void averggeWeight(){                             //调用父类（people）的方法定义体重  
            weight=75;  
            System.out.println("American&apos;s average weight:"+weight+"kg");  
        }  
        public void americanBoxing(){  
            System.out.println("直拳，勾拳，组合拳！");  
        }   //子类独有的方法  
    }

**四、BeijingPeople.java**

package com.company;  
  
    public class BeijingPeople extends ChinaPeople{            //定义ChinaPeople的子类（即People的孙子类）  
        public void averageHeight(){                           //调用父类（people）的方法定义身高  
            height=172.5;  
            System.out.println("北京人的平均身高:"+height+"厘米");  
        }  
        public void averggeWeight(){                           //调用父类（people）的方法定义体重  
            weight=70;  
            System.out.println("北京人的平均体重:"+weight+"千克");  
        }  
        public void0 beijingOpera(){  
            System.out.println("花脸，青衣，花旦和老生！");          //子类独有的方法  
        }  
    }

**五、Main.java**

package com.company;  
  
public class Main {  
  
    public static void main(String[] args) {  
  
        ChinaPeople chinapeople;  
        chinapeople = new ChinaPeople();  
        AmericanPeople americanpeople;  
        americanpeople = new AmericanPeople();  
        BeijingPeople beijingpeople;  
        beijingpeople = new BeijingPeople();  
        chinapeople.speckHello();  
        americanpeople.speckHello();  
        beijingpeople.speckHello();  
        chinapeople.averageHeight();  
        americanpeople.averageHeight();  
        beijingpeople.averageHeight();  
        chinapeople.averggeWeight();  
        americanpeople.averggeWeight();  
        beijingpeople.averggeWeight();  
        chinapeople.chinaGongfu();  
        americanpeople.americanBoxing();  
        beijingpeople.beijingOpera();  
        beijingpeople.chinaGongfu();  
        System.out.printf("20211120138薛凡豪");  
        }  
    }

2.银行利息计算

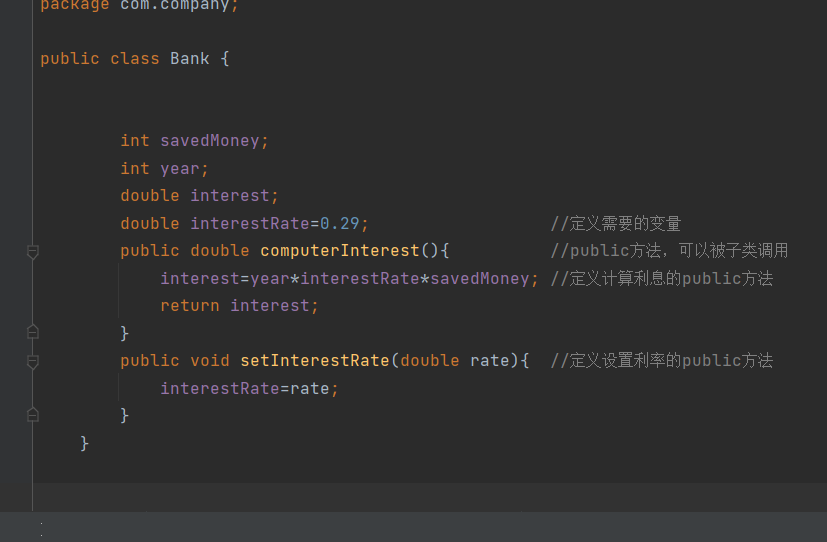
### （1）问题分析：

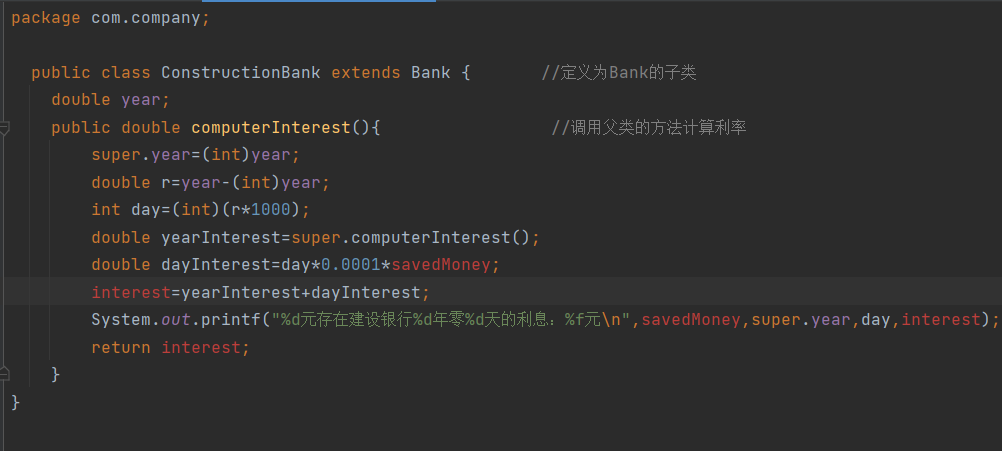
1.本次实验主要练习类的组合和super方法。

2.本次实验有四个Java文件:Bank.java(创建银行对象和方法) ConstructionBank.java（继承Bank类）BankOfDalian.java（继承Bank类），SaveMoneyjava（运行类）

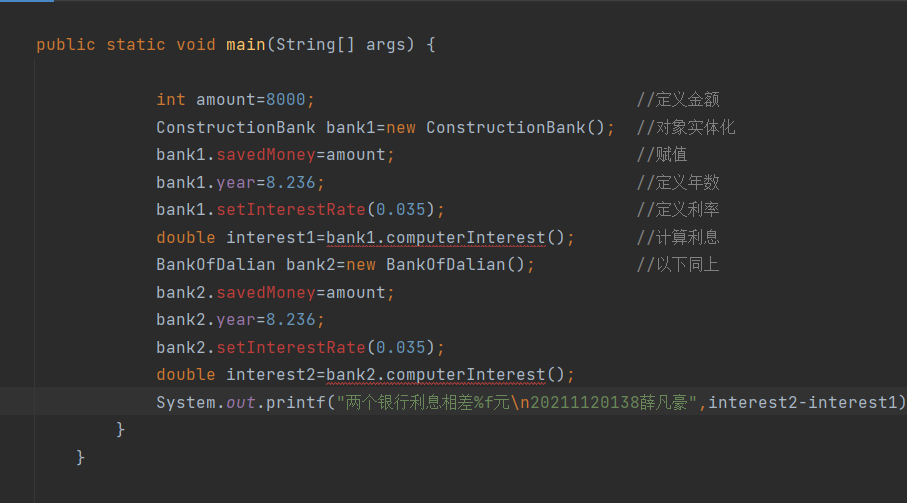
3.实验运用了类的组合以及super方法来调用隐藏的方法

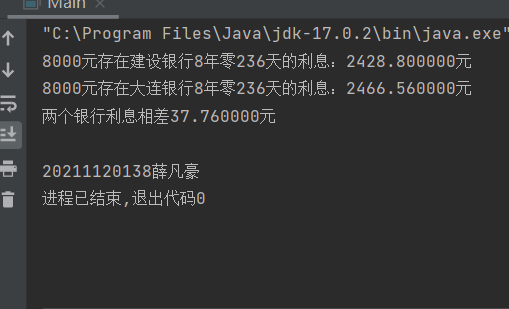
### （2）程序的运行和结果





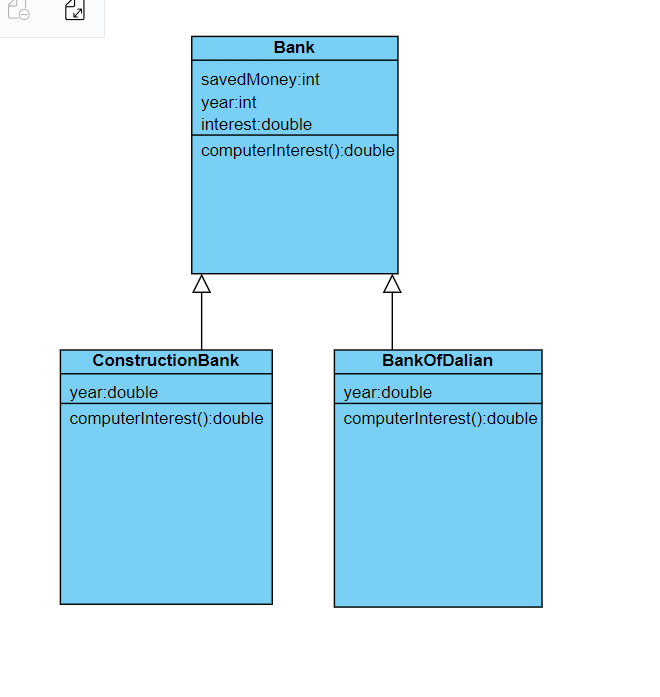




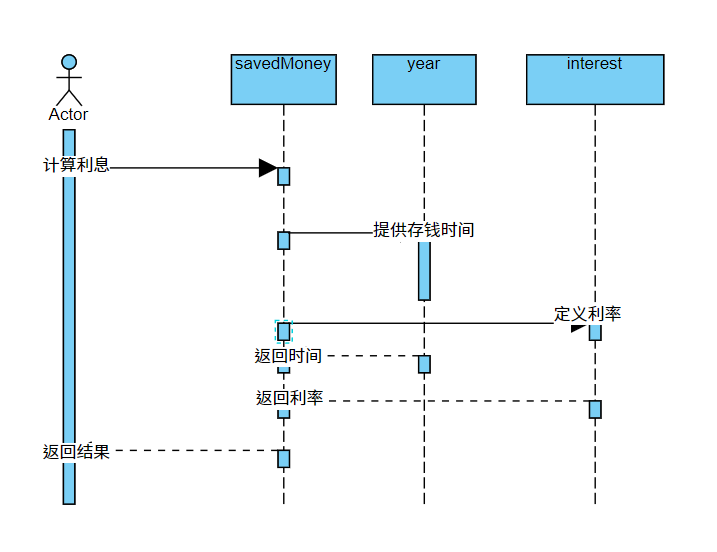


### （3）UML图

类图：



时序图：



### （4）源代码

一、Bank.java

 package com.company;  
  
public class Bank {  
  
  
        int savedMoney;  
        int year;  
        double interest;  
        double interestRate=0.29;                  //定义需要的变量  
        public double computerInterest(){          //public方法，可以被子类调用  
            interest=year\*interestRate\*savedMoney; //定义计算利息的public方法  
            return interest;  
        }  
        public void setInterestRate(double rate){  //定义设置利率的public方法  
            interestRate=rate;  
        }  
    }  
  
  
.二、ConstructionBank.java

package com.company;  
  
  public class ConstructionBank extends Bank {       //定义为Bank的子类  
    double year;  
    public double computerInterest(){                 //调用父类的方法计算利率  
        super.year=(int)year;  
        double r=year-(int)year;  
        int day=(int)(r\*1000);  
        double yearInterest=super.computerInterest();  
        double dayInterest=day\*0.0001\*savedMoney;  
        interest=yearInterest+dayInterest;  
        System.out.printf("%d元存在建设银行%d年零%d天的利息：%f元\n",savedMoney,super.year,day,interest);  
        return interest;  
    }

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
}三、BankOfDalian.java

package com.company;  
  
public class BankOfDalian extends Bank {          //定义为Bankde子类  
    double year;  
    public double computerInterest(){              //依然调用父类Bank的方法计算利率  
        super.year=(int)year;  
        double r=year-(int)year;  
        int day=(int)(r\*1000);  
        double yearInterest=super.computerInterest();  
        double dayInterest=day\*0.00012\*savedMoney;  
        interest=yearInterest+dayInterest;  
        System.out.printf("%d元存在大连银行%d年零%d天的利息：%f元\n",savedMoney,super.year,day,interest);  
        return interest;  
    }  
}

**四、Main.java**

package com.company;  
  
public class Main {  
  
    public static void main(String[] args) {  
  
                int amount=8000;                                //定义金额  
                ConstructionBank bank1=new ConstructionBank();  //对象实体化  
                bank1.savedMoney=amount;                        //赋值  
                bank1.year=8.236;                               //定义年数  
                bank1.setInterestRate(0.035);                   //定义利率  
                double interest1=bank1.computerInterest();      //计算利息  
                BankOfDalian bank2=new BankOfDalian();          //以下同上  
                bank2.savedMoney=amount;  
                bank2.year=8.236;  
                bank2.setInterestRate(0.035);  
                double interest2=bank2.computerInterest();  
                System.out.printf("两个银行利息相差%f元\n20211120138薛凡豪",interest2-interest1);  
            }  
        }

## 3.公司支出的总薪水

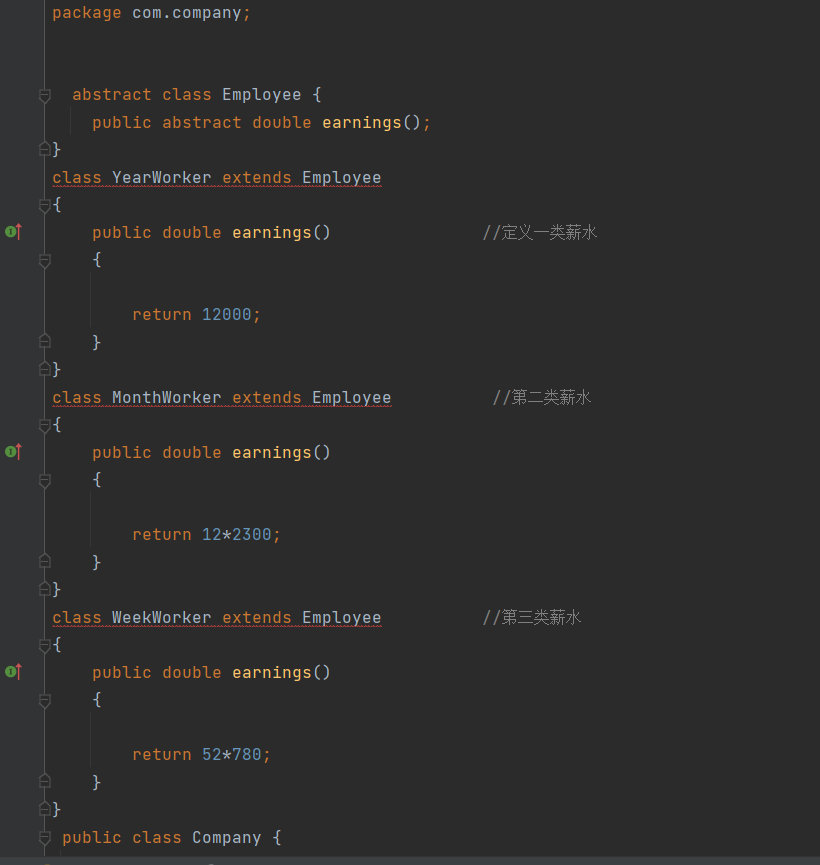
### (1)问题分析：

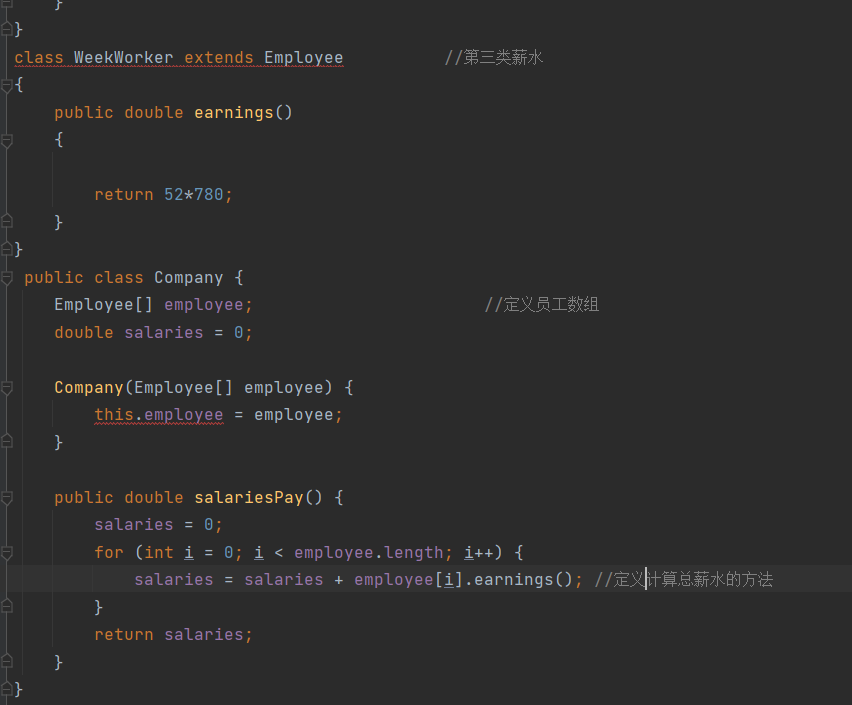
1.掌握上转型变量的使用，体会多态性。

2.本次实验有两个需要写的点，即薪水的计算方法以及不同员工的不同薪水来计算薪水的多少并求和。

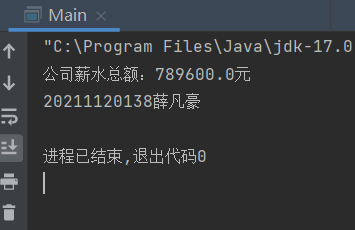
3.本次实验利用了对象数组实现员工分类（注意对象数组每个都需要实体化！！！）

### (2)程序的运行和结果



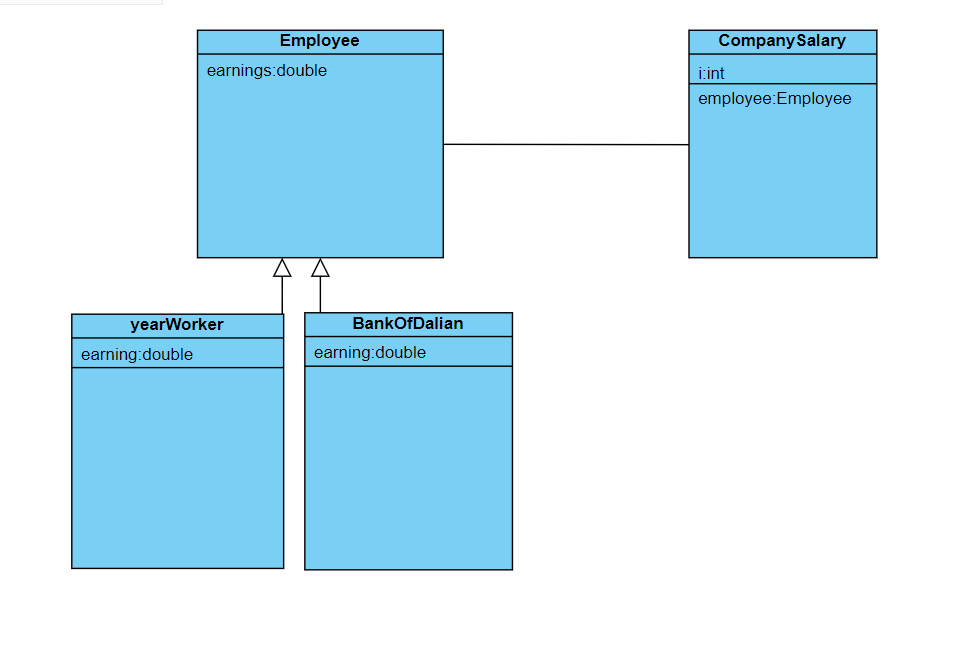




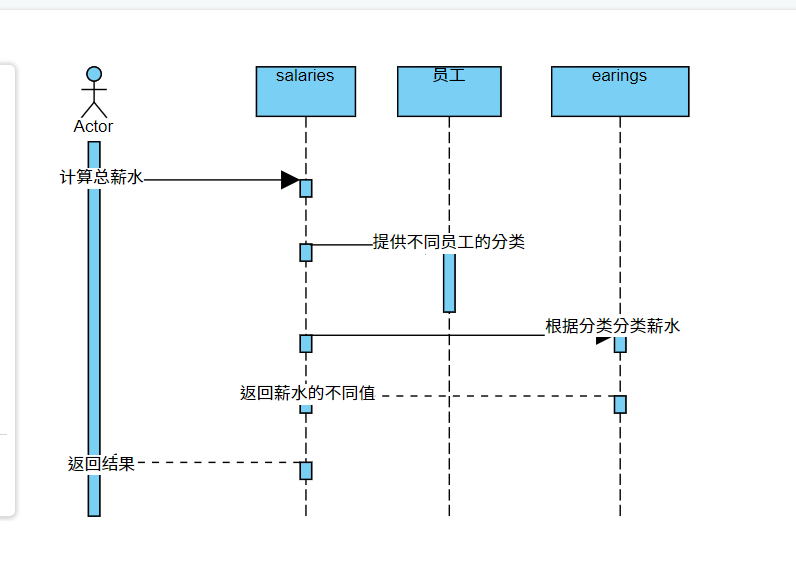


### （3）UML图

类图：



时序图：



### （4）源代码

package com.company;  
  
  
  abstract class Employee {  
    public abstract double earnings();  
}  
class YearWorker extends Employee  
{  
    public double earnings()               //定义一类薪水  
    {  
  
        return 12000;  
    }  
}  
class MonthWorker extends Employee          //第二类薪水  
{  
    public double earnings()  
    {  
  
        return 12\*2300;  
    }  
}  
class WeekWorker extends Employee          //第三类薪水  
{  
    public double earnings()  
    {  
  
        return 52\*780;  
    }  
}  
 public class Company {  
    Employee[] employee;                       //定义员工数组  
    double salaries = 0;  
  
    Company(Employee[] employee) {  
        this.employee = employee;  
    }  
  
    public double salariesPay() {  
        salaries = 0;  
        for (int i = 0; i < employee.length; i++) {  
            salaries = salaries + employee[i].earnings(); //定义计算总薪水的方法  
        }  
        return salaries;  
    }  
}

# 二、总结

1.本次实验较前四次更为综合。运用了类的组合和方法的调用以及super方法还有对象数组以及上转型对象来解决问题。

2.主要使用了set赋值方法和改变方法以及参数的传递和封装性问题，另外使用了super方法和对象数组使用。

3. 主要的思想仍然是类与对象的综合使用和相互调用。

4.仍然需要强调的是定义一个对象（包括对象数组中的每一个对象），需要给予他真实的空间，即实体化，注意new的使用。

5.难点仍然是类与类的关系和调用以及子类继承。