

Java程序设计实验报告

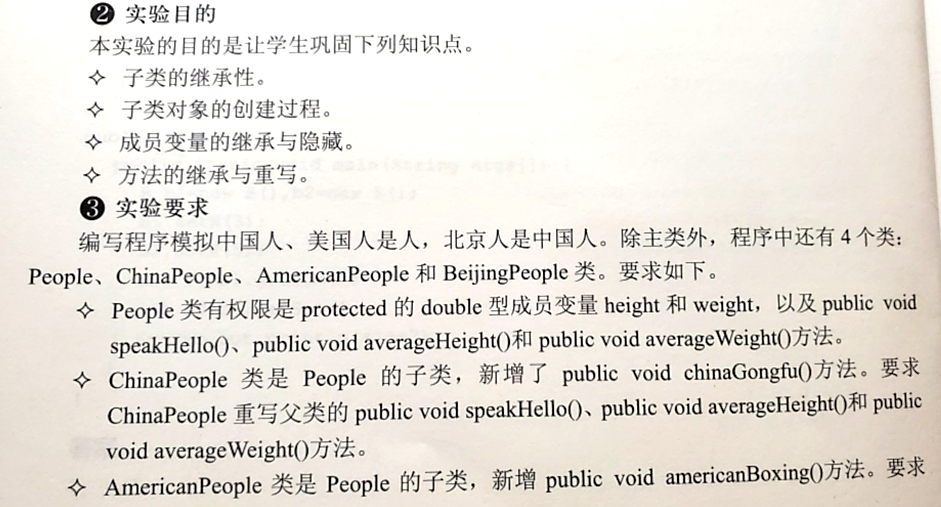
实验报告

【实验题目】 Java实验4

#### 【实验内容】

##### 1. 练习1

###### **实验目的**

**实验要求**

###### 代码：

**People.java**

package People;

public class People {//父类

    protected double height = 0, weight = 0;//protect修饰只可以子类调用

    public void speakHello() {

    }

    public void averageHeight() {

    }

    public void averageWeight() {

    }

}

**AmericanPeople.java**

package People;

public class AmericanPeople extends People {//子类

    public void americanBoxing() {//新方法

        System.out.println("直拳，勾拳，组合拳");

    }

    public void speakHello() {//重写父类的speakHello方法

        System.out.println("How do you do");

    }

    public void averageHeight() {//重写父类的averageHeight方法

        height = 176.0;//继承父类成员变量

        System.out.println("American's average height:" + height + "cm");

    }

    public void averageWeight() {//重写父类的averageWeight方法

        weight = 75.0;//继承父类成员变量

        System.out.println("American's average weight:" + weight + "kg");

    }

}

**BeijingPeople.java**

package People;

public class BeijingPeople extends ChinaPeople {//继承ChinaPeople

    public void speakHello() {//重写父类的speakHello方法

        System.out.println("您好");

    }

    public void averageHeight() {//重写父类的averageHeight方法

        height = 172.5;//对People类的成员变量赋值

        System.out.println("北京人的平均身高为:" + height + "厘米");

    }

    public void averageWeight() {//重写父类的averageWeight方法

        weight = 70.0;//对People类的成员变量赋值

        System.out.println("北京人的平均体重为:" + weight + "千克");

    }

    public void BeijingOpera() {//新方法

        System.out.println("花脸、青衣、花旦和老生");

    }

}

**ChinaPeople.java**

package People;

public class ChinaPeople extends People {//子类

    public void speakHello() {//重写父类的speakHello方法

        System.out.println("您好");

    }

    public void averageHeight() {//重写父类的averageHeight方法

        height = 168.78;//继承父类成员变量

        System.out.println("中国人的平均身高为:" + height + "厘米");

    }

    public void averageWeight() {//重写父类的averageWeight方法

        weight = 65.0;//继承父类成员变量

        System.out.println("中国人的平均体重为:" + weight + "千克");

    }

    public void chinaGongfu() {//新方法

        System.out.println("坐如钟，站如松，睡如弓");

    }

}

**Test1.java**

package People;

public class Test1 {

    public static void main(String[] args) {

        ChinaPeople cp = new ChinaPeople();//new一个对象

        AmericanPeople ap = new AmericanPeople();//new一个对象

        BeijingPeople bp = new BeijingPeople();//new一个对象

        //调用各个类里面的speakHello方法

        cp.speakHello();

        ap.speakHello();

        bp.speakHello();

        //调用各个类里面的averageHeight方法

        cp.averageHeight();

        ap.averageHeight();

        bp.averageHeight();

        //调用各个类里面的averageWeight方法

        cp.averageWeight();

        ap.averageWeight();

        bp.averageWeight();

        cp.chinaGongfu();//特有方法

        ap.americanBoxing();//特有方法

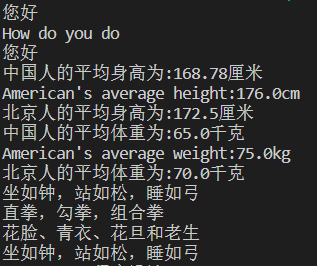
        bp.BeijingOpera();//特有方法

        bp.chinaGongfu();//特有方法

    }

}

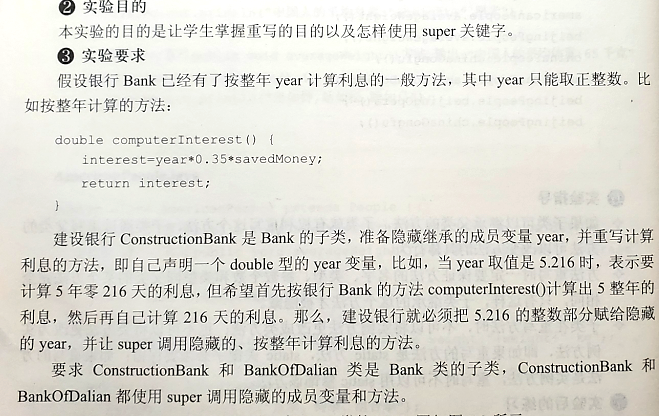
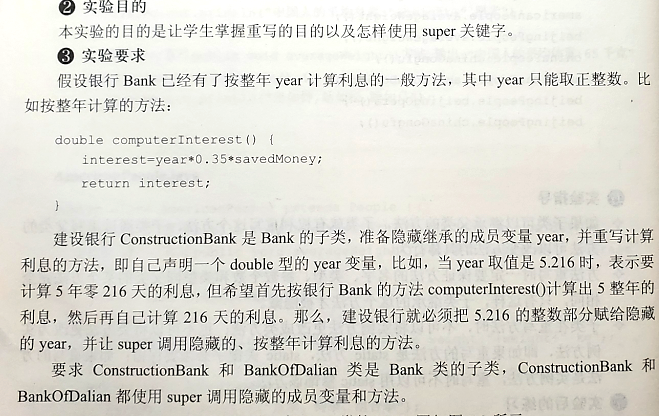
###### 输出结果：



###### 分析：

##### ChinaPeople和AmericanPeople两个类都是继承父类People类，所以可以调用父类中的protected修饰的变量和成员方法，BeijingPeople则是继承ChinaPeople类的，所以也可以使用People的变量和方法，当然也可以使用ChianPeople的方法

##### 2.练习2

**实验目的****实验要求**

###### 代码：

**Bank.java**

package Bank;

public class Bank {

    int year, savedMoney;//成员变量：存的年数，存的钱数目

    double interest;//利息

    public double computerInterest() {

        interest = year \* 0.035 \* savedMoney;//算年利率

        return interest;

    }

}

**ConstructionBank.java**

package Bank;

public class ConstructionBank extends Bank {

    double year;//隐藏父类year重新定义

    int day;//存储的天数

    public double computerInterest() {

        super.year = (int) year;//这样就可以使父类的year存储整数部分

        double y = -super.year + year;//原来的数-整数部分就是小数部分

        day = (int) (y \* 1000);//计算实际的天数

        double interest1 = super.computerInterest();//调用父类方法计算年利

        double interest2 = day \* 0.00032 \* savedMoney;//计算日利

        return interest1 + interest2;//二者相加就是总共的钱

    }

}

**BankOfDalian.java**

package Bank;

public class BankOfDalian extends Bank {

    double year;//隐藏父类year重新定义

    int day;//存储的天数

    public double computerInterest() {

        super.year = (int) year;//这样就可以使父类的year存储整数部分

        double y = -super.year + year;//原来的数-整数部分就是小数部分

        day = (int) (y \* 1000);//计算实际的天数

        double interest1 = super.computerInterest();//调用父类方法计算年利

        double interest2 = day \* 0.00037 \* savedMoney;//计算日利

        return interest1 + interest2;//二者相加就是总共的钱

    }

}

**Test2.java**

package Bank;

public class Test2 {

    public static void main(String[] args) {

        int money = 10000;//存钱1w

        ConstructionBank cb = new ConstructionBank();

        BankOfDalian db = new BankOfDalian();

        cb.savedMoney = money;//给父类变量值

        cb.year = 5.216;//子类变量值

        double interest1 = cb.computerInterest();

        db.savedMoney = money;

        db.year = 5.216;

        double interest2 = db.computerInterest();

        double delta = Math.abs(interest1 - interest2);//为了防止出现负数用绝对值

        System.out.println(String.format("%d元存在建设银行%d年零%d天的利息：%f", money, (int) cb.year, cb.day, interest1));

        System.out.println(String.format("%d元存在大连银行%d年零%d天的利息：%f", money, (int) db.year, db.day, interest2));

        System.out.println("两个银行利息相差" + delta + "元");

    }

}

###### 输出结果：

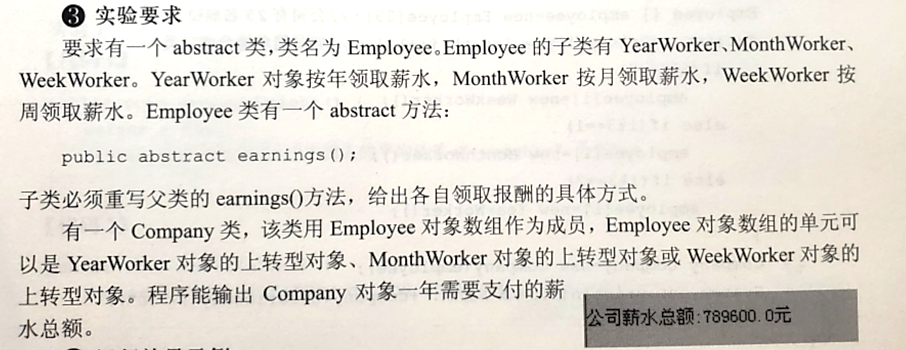
###### 

###### 分析：

父类的成员方法仅仅是计算整年的利息，在子类中用super调用父类方法，之后在计算小数部分（即日利息），二者相加即可得出最终的利息

##### 练习3

###### 实验目的 学会使用抽象类的使用和调用以及方法重写

**实验要求**

###### 代码：

**Employee.java**

package Company;

public abstract class Employee {

    public abstract double earnings();// 抽象类

}

**Company.java**

package Company;

public class Company {

    Employee[] e;// Employee类型的数组

    Company(Employee[] people) {

        e = people;

    }

}

**MonthWorker.java**

package Company;

public class MonthWorker extends Employee {

    int month = 40;//初始化

    double monthSalary = 1025.01;//初始化

    public double earnings() {//重写方法

        return month \* monthSalary;

    }

}

**WeekWorker.java**

package Company;

public class WeekWorker extends Employee {

    int week = 30;// 初始化

    double weekSalary = 1000.25;// 初始化

    public double earnings() {// 重写方法

        return week \* weekSalary;

    }

}

**YearWorker.java**

package Company;

public class YearWorker extends Employee {

    int year = 1;// 初始化

    double yearSalary = 2000.56;// 初始化

    public double earnings() {// 重写方法

        return year \* yearSalary;

    }

}

**Test3.java**

package Company;

public class Test3 {

    public static void main(String[] args) {

        Employee e[] = new Employee[3];//Employee变量

        e[0] = new YearWorker();

        e[1] = new MonthWorker();

        e[2] = new WeekWorker();

        Company com = new Company(e);

        double sum;

        sum = e[0].earnings() + e[1].earnings() + e[2].earnings();//三个相加就是总和

        System.out.println("总工资为:" + sum);

    }

}

###### 输出结果：



###### 分析：

抽象类的方法在子类重写，分开计算周、月、年的工人的薪水，在加起来就可以求出总和

##### 练习4

###### 实验目的

熟练掌握接口的使用

###### 实验要求

###### 

###### 代码：

**Animal.java**

package Animal;

public interface Animal {

    public abstract void cry();// 抽象方法cry

    public abstract void getAnimalName();// 抽象方法getAnimalName

}

**Simulator.java**

package Animal;

public class Simulator {

    Animal animal;// 成员变量

    public void playSound(Animal animal) {

        this.animal = animal;

        animal.cry();// 调用cry方法

        animal.getAnimalName();// 调用getAnimalName方法

    }

}

**Dog.java**

package Animal;

public class Dog implements Animal {// Animal的子类Dog

    public void cry() {// 重写方法cry

        System.out.println("wang wang wang!");

    }

    public void getAnimalName() {// 重写方法getAnimalName

        System.out.println("这是狗");

    }

}

**Cat.java**

package Animal;

public class Cat implements Animal {// Animal的子类Cat

    public void cry() { // 重写方法cry

        System.out.println("miao miao miao!");

    }

    public void getAnimalName() {// 重写方法getAnimalName

        System.out.println("这是猫");

    }

}

**Test4.java**

package Animal;

public class Test4 {

    public static void main(String[] args) {

        Simulator simulator = new Simulator();// 定义新对象

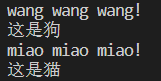
        simulator.playSound(new Dog());// 调用方法

        simulator.playSound(new Cat());// 调用方法

    }

}

###### 输出结果：



###### 分析：

构造接口Animal，含有cry和getAnimalName两个成员方法，Dog类和Cat类都是Animal类的子类，所以他们也都含有这两个成员方法，我们只需要在子类中对父类的方法进行重写即可，Simulator类只是调用相应的成员方法。

#### 【小结与讨论】

1.接口和类类似，在继承的时候需要用implements

2.用super可以调用父类的变量和方法

3.protected保护的变量或者方法只能在子类中使用