

## 【案例 4-5】研究生薪资管理系统

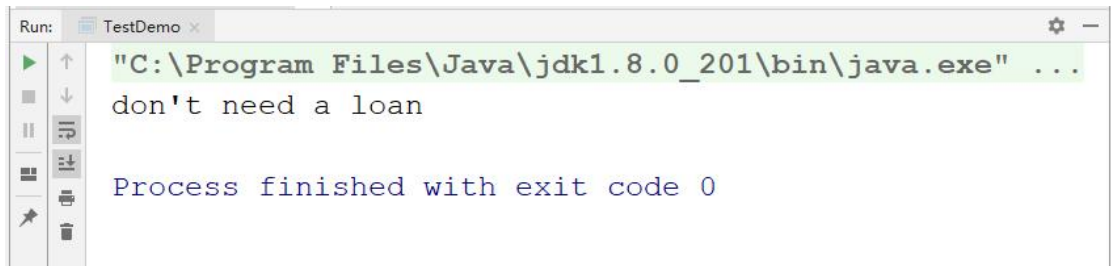
### 【案例介绍】

#### 1. 案例描述

在学校中，学生每个月需要交相应的生活费，老师每个月有相应的工资，而在职研究生既是老师又是学生，所以在在职研究生既需要交学费又会有工资。下面要求编写一个程序来统计在职研究生的收入与学费，如果收入减去学费不足 2000 元，则输出“provide a loan”(需要贷款)信息。

本案例要求使用接口实现该程序。

#### 2. 运行结果



```
Run: TestDemo x
"C:\Program Files\Java\jdk1.8.0_201\bin\java.exe" ...
don't need a loan
Process finished with exit code 0
```

### 【案例目标】

- 学会“抽象类实现不同图形的打印程序设计”的实现思路。
- 独立完成“抽象类实现不同图形的打印程序设计”的源代码编写、编译及运行。
- 掌握方法的重写。
- 掌握接口的应用。

### 【案例分析】

(1) 定义两个接口，在 `StudentManageInterface` 接口中声明两个方法：`getFree()`和 `setFree()`，用于设置和获取学生的学费；在 `TeacherManageInterface` 接口中声明两个方法：`gePay()`和 `setPaye()`，用于设置和获取教师的工资。

(2) 定义主类 `Graduate`，分别实现 `StudentManageInterface` 和 `TeacherManageInterface` 接口。

(3) 定义 `Graduate` 类的成员变量，和构造方法。

(4) 给出四个接口方法的实现。

(5) 给出一个计算是否需要贷款的方法，在里面统计年收入和学费，并输出是否需要贷款的信息。

(6) 在 `main()` 方法中创建一个姓名为“zhangsan”的研究生，调用计算是否需要贷款的方法。

### 【案例实现】

Graduate.java

```

1 class Graduate implements StudentManagerInterface,
2 TeacherManagerInterface {
3     private String name, sex;
4     private int age;
5     private double fee, pay;
6     Graduate() {
7     }
8     Graduate(String name, String sex, int age, double fee, double pay) {
9         this.name = name;
10        this.sex = sex;
11        this.age = age;
12        this.fee = fee;
13        this.pay = pay;
14    }
15    public String getName() {
16        return name;
17    }
18    public String getSex() {
19        return sex;
20    }
21    public int getAge() {
22        return age;
23    }
24    public void setFree(double fee) {
25        this.fee = fee;
26    }
27    public double getFree() {
28        return fee;
29    }
30    public void setPay(double pay) {
31        this.pay = pay;
32    }
33    public double getPay() { //对接口的方法必须重写
34        return pay;
35    }
36 }

```

上述代码中，Graduate 类实现了 StudentManagerInterface 和 TeacherManagerInterface 接口，并重写了两个接口中的方法。在 main()方法中，计算了名为“zhangsan”的研究生是否需要贷款。

#### StudentManagerInterface.java

```

1 interface StudentManagerInterface { //interface 接口名
2     public abstract void setFree(double fee); //抽象方法
3     public abstract double getFree();
4 }

```

上述代码中，创建了一个 `StudentManagerInterface` 接口，定义了 `setFree()`和 `get Free ()` 方法，用于设置和获取学生学费。

#### TeacherManagerInterface.java

```
37 interface TeacherManageInterface{
38     public abstract double getPay();
39     public abstract void setPay(double pay);
40 }
```

上述代码中，创建了一个 `TeacherManagerInterface` 接口，定义了 `setPay()`和 `getPay()`方法，用于设置和获取教师工资。

#### TestDemo.java

```
1 public class TestDemo {
2     public static void main(String[] args) {
3         Graduate gr = new Graduate("zhangsan", "男", 5, 8000, 3000);
4         judgeLoan(gr);
5     }
6     public static void judgeLoan(Graduate gr) { //对象作形参
7         if (gr.getPay() * 12 - gr.getFree() * 2 < 2000) {
8             System.out.println("provide a loan");
9         } else
10            System.out.println("don't need a loan");
11     }
12 }
```