

大 連 工 業 大 学

设 计 性 实 验 报 告

题 目： 实验五 继承

专业： 计算机科学与技术

班级学号： 计算机 182-07

学生姓名： 梁宇龙

指导教师： 苏波

实验室名称： 综 B322

所属学院： 信息科学与工程学院

实验日期： 2020 年 4 月 19 日

实验五 继承

本实验 4 学时

1 实验目的

通过对具体类的继承(扩充)，为程序增加新的功能。

2 技术路线

通过在已有程序的基础上增加新的方法，实现程序功能的扩充

3 实验环境与条件

JDK1.8, NetBeans 8

4 实验内容要求

在实验 3 或实验 4 的基础上，利用继承，增加新的功能

5 实验内容设计

本实验基于实验四基础上进行。

场景描述：模拟教务处系统实现成绩查询和成绩录入、修改功能

功能描述：在实验四功能基础上新增功能：在查询学生成绩信息同时，显示该生平均成绩加权绩点（GPA），计算方法如下：

平均学分绩点= $\Sigma(\text{课程学分} \times \text{成绩绩点}) / \Sigma \text{课程学分}$

使用过程描述：

进入服务界面提示，用户选择服务

1、进入成绩查询系统，用户输入学生姓名，输出用户所学课程名称及对应成绩及平均成绩加权绩点。

2、进入成绩修改系统，用户输入学生姓名，修改课程课程名，平时成绩与期末成绩，系统算出总成绩和绩点并存回文件。

6 程序记录

本处只写新增功能，原有功能源代码将会以附录形式在报告末尾呈现。

```
//通过继承 Manager 类中相关方法及变量完成计算学生 GPA 的操作
package oop_experiment;
import java.math.*;
public class GPAManager extends Manager{
    static Student stu1 = new Student();
    static double gpa;
    public static void getGPA(Student stu1){
        gpa = figureGPA(stu1.getStuNum());
        //实现绩点保留小数点后 2 位数
        BigDecimal b = new BigDecimal(gpa);
        double f1 = b.setScale(2, BigDecimal.ROUND_HALF_UP).doubleValue();
        System.out.println("该生平均成绩绩点为: "+ f1);
    }
    //计算出该学生 GPA
    public static double figureGPA(String StuNum){
        double sum = 0,ave = 0;
        String content = FileTool.read("SC-lab.txt");
        String[] lines = content.split("\r\n");
        scs = new SC[lines.length];
        for(int i=0;i<lines.length;i++){
            String[] items = lines[i].split(",");
            scs[i] = new SC(items[0],items[1],Integer.parseInt(items[2]),Double.valueOf(items[3]));
            if(scs[i].getSNo().equals(StuNum)){
                sum += scs[i].getGredit() * getCourseCredit(scs[i].getCNo());
                ave += getCourseCredit(scs[i].getCNo());
            }
        }
        sum = sum/ave;
        return sum;
    }
    //获取课程所对应学分
```

```

public static double getCourseCredit(String CourseNum){
    String content = FileTool.read("Course-lab.txt");
    String[] lines = content.split("\r\n");
    courses = new Course[lines.length];
    for(int i=0;i<courses.length;i++){
        String[] items = lines[i].split(",");
        courses[i] = new Course(items[0],items[1],Double.valueOf(
items[2]));
        if(courses[i].getCourseNum().equals(CourseNum)){
            return courses[i].getCredit();
        }
    }
    return 0;
}
}

```

7 程序结果记录

run:

请选择服务项目

1.成绩查询

2.成绩修改

0.退出系统

1

请输入学生姓名

Yulong

该学生成绩为:

大学物理 2 78

电工学 90

离散数学 83

数据结构 92

该生平均成绩绩点为: 3.36

查询结束, 请选择操作:

1.成绩查询

2.成绩修改

0.退出系统

1

请输入学生姓名

Xianyu

该学生成绩为:

大学物理 2 54

电工学 52

离散数学 46

数据结构 61

该生平均成绩绩点为: 0.29

查询结束, 请选择操作:

1.成绩查询

2.成绩修改

0.退出系统

0

正在退出系统 ...

成功构建 (总时间: 16 秒)

8 结论

编写了在已有类 Manager 的基础上扩充出具有核算学生平均成绩加权绩点 GPA 功能的类 GPAManager。

运行是 GPAManager 可以实现查询学生成绩信息, 修改成绩信息的功能, 并且可以在查询成绩功能中实现计算学生 GPA 的功能。

结果表明可以通过继承实现程序的扩充, 即重复利用已有的类, 扩充出具有更多功能的类。

附录：基于实验四修改的源代码

```
//主类
package oop_experiment;

public class OOP_Experiment {

    public static void main(String[] args) {
        System.out.println("请选择服务项目");
        Service.serviseTip();
        int service = Console.inputInt();
        Service.servise(service);
    }
}
```

```
//服务部分的封装类 Service.java
package oop_experiment;

public class Service {
    public static Student SearchStu(){
        System.out.println("请输入学生姓名");
        String StuName = Console.inputString();
        Student stu = Manager.stuManager(StuName);
        return stu;
    }
    public static void servise(int service){
        while(service!=0){
            switch (service) {
                case 1:
                    Student stu1 = SearchStu();
                    if(stu1==null){
                        System.out.println("\n 无该学生成绩! 请重新选择操作: ");
                        serviseTip();
                        service = Console.inputInt();
                    }
                    else{
                        System.out.println("\n 该学生成绩为: ");

```

```

        Manager.stuGrade(stu1.getStuNum());
        GPAManager.getGPA(stu1);
        System.out.println("\n 查询结束，请选择操作：");
        serviseTip();
        service = Console.inputInt();
    }
    break;
case 2:
    Student stu2 = SearchStu();
    System.out.println("\n 请输入课程号");
    String CourseNum = Console.inputString();
    System.out.println("\n 请输入平时成绩");
    int grade1 = Console.inputInt();
    System.out.println("\n 请输入期末成绩");
    int grade2 = Console.inputInt();
    int grade = (int)(grade1 * 0.4 + grade2 * 0.6);
    double Gredit;
    if(grade>=90){
        Gredit = 4;
    }
    else if(grade>=80&&grade<90){
        Gredit = 3;
    }
    else if(grade>=70&&grade<80){
        Gredit = 2;
    }
    else if(grade>=60&&grade<70){
        Gredit = 1;
    }
    else{
        Gredit = 0;
    }
    int tf = Manager.ChangeGrade(stu2.getStuNum(),CourseNum,grade,Gredit);
    if(tf == 1){
        System.out.println("\n 更改成功，请选择操作：");
        serviseTip();
        service = Console.inputInt();
    }

```

```

        }
        else{
            System.out.println("\n 更改失败，请选择操作：");
            serviseTip();
            service = Console.inputInt();
        }
        break;
    default:
        System.out.println("\n 非法输入，请重新选择");
        serviseTip();
        service = Console.inputInt();
        break;
    }
}
System.out.println("正在退出系统...");
}
public static void serviseTip(){
    System.out.println("1.成绩查询");
    System.out.println("2.成绩修改");
    System.out.println("0.退出系统");
}
}
}

```

```

//对查询信息功能进行封装 Manager.java
package oop_experiment;

public class Manager {
    static Student[] students;
    static SC[] scs;
    static Course[] courses;
    public static Student stuManager(String StuName){
        String content = FileTool.read("stu-lab.txt");
        String[] lines = content.split("\r\n");
        students = new Student[lines.length];
        for(int i=0;i<lines.length;i++){
            String[] items = lines[i].split(",");
            students[i] = new Student(items[0],items[1]);
            if(students[i].getStuName().equals(StuName)){

```



```

        return students[i];
    }
}
return null;
}
public static void stuGrade(String StuNum){
    String content = FileTool.read("SC-lab.txt");
    String[] lines = content.split("\r\n");
    scs = new SC[lines.length];
    for(int i=0;i<lines.length;i++){
        String[] items = lines[i].split(",");
        scs[i] = new SC(items[0],items[1],Integer.parseInt(items[
2]),Double.valueOf(items[3]));
        if(scs[i].getSNo().equals(StuNum)){
            System.out.println(getCourseName(scs[i].getCNo())+"\t
"+scs[i].getGrade());
        }
    }
}
public static String getCourseName(String CourseNum){
    String content = FileTool.read("Course-lab.txt");
    String[] lines = content.split("\r\n");
    courses = new Course[lines.length];
    for(int i=0;i<courses.length;i++){
        String[] items = lines[i].split(",");
        courses[i] = new Course(items[0],items[1],Double.valueOf(
items[2]));
        if(courses[i].getCourseNum().equals(CourseNum)){
            return courses[i].getCourseName();
        }
    }
    return null;
}
public static int ChangeGrade(String StuNum,String CourseNum,int
Grade,double Gredit){
    int tf = 0;
    String content = FileTool.read("SC-lab.txt");
    String[] lines = content.split("\r\n");

```

```

        scs = new SC[lines.length];
        for(int i=0;i<lines.length;i++){
            String[] items = lines[i].split(",");
            scs[i] = new SC(items[0],items[1],Integer.parseInt(items[
2]),Double.valueOf(items[3]));
            if(scs[i].getSNo().equals(StuNum)){
                if (scs[i].getCNo().equals(CourseNum)) {
                    scs[i].setGrade(Grade);
                    scs[i].setGredit(Gredit);
                    tf = 1;
                }
            }
        }
        String contents = "";
        for (SC sc : scs) {
            contents += sc.getSNo() + "," + sc.getCNo() + "," + sc.ge
tGrade() + "," + sc.getGredit() + "\r\n";
        }
        if(tf==0){
            contents += StuNum+","+CourseNum+","+Grade+","+Gredit+"\r
\n";

            tf=1;
        }
        FileTool.write("SC-lab.txt", contents);
        return tf;
    }
}

```

//对学生信息进行封装 Student.java

```

package oop_experiment;
public class Student {
    private String stuNum;           //学号
    private String stuName;          //学生姓名
    //构造方法
    public Student(){

    }
    public Student(String stuNum){

```

```

        this.stuNum = stuNum;
    }
    public Student(String stuNum,String stuName){
        this.stuNum = stuNum;
        this.stuName = stuName;
    }
    //getter 和 setter 方法
    public String getStuNum(){
        return stuNum;
    }
    public String getStuName(){
        return stuName;
    }
    public void setStuNum(String stuNum){
        this.stuNum = stuNum;
    }
    public void setStuName(String stuName){
        this.stuName = stuName;
    }
}

```

```

//对课程信息进行封装 Course.java
package oop_experiment;

public class Course {
    private String courseNum;
    private String courseName;
    private double credit;
    //构造方法
    public Course(String courseNum){
        this.courseNum = courseNum;
    }
    public Course(String courseNum,String courseName,double credit){
        this.courseNum = courseNum;
        this.courseName = courseName;
        this.credit = credit;
    }
    //各个变量的 getter 和 setter

```

```

    public String getCourseNum(){
        return courseNum;
    }
    public String getCourseName(){
        return courseName;
    }
    public double getCredit(){
        return credit;
    }
    public void setCourseNum(String courseNum){
        this.courseNum = courseNum;
    }
    public void setCourseName(String courseName){
        this.courseName = courseName;
    }
    public void setCredit(double credit){
        this.credit = credit;
    }
}

```

```

//对学生课程成绩信息进行封装 SC.java
package oop_experiment;

/**
 *
 * @author yulon
 */
public class SC {
    private String SNo;
    private String CNo;
    private int Grade;
    private double Gcredit;
    //构造方法
    public SC (String SNo,String CNo){
        this.SNo = SNo;
        this.CNo = CNo;
    }
}

```

```

public SC (String SNo,String CNo,int Grade,double Gredit){
    this.SNo = SNo;
    this.CNo = CNo;
    this.Grade = Grade;
    this.Gredit = Gredit;
}
//getter 和 setter 方法
public String getSNo(){
    return SNo;
}
public String getCNo(){
    return CNo;
}
public int getGrade(){
    return Grade;
}
public double getGredit(){
    return Gredit;
}
public void setSNo(String SNo){
    this.SNo = SNo;
}
public void setCNo(String CNo){
    this.CNo = CNo;
}
public void setGrade(int Grade){
    this.Grade = Grade;
}
public void setGredit(double Gredit){
    this.Gredit = Gredit;
}
}

```

```

//对输入方式函数进行封装 Console.java
package oop_experiment;

import java.util.Scanner;
public class Console {

```

```

    public static String inputString(){
        Scanner scanner = new Scanner(System.in);
        String str = scanner.next();
        return str;
    }
    public static int inputInt(){
        Scanner scanner = new Scanner(System.in);
        int num = scanner.nextInt();
        return num;
    }
}

```

```

package oop_experiment;
//文件读写方法 FileTool.java
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;

public class FileTool {

    public static String read(String fileName) {
        StringBuilder content = new StringBuilder();
        try (FileInputStream fileInput = new FileInputStream(fileName)
;
                InputStreamReader inputStreamReader = new InputStreaMR
eader(fileInput);) {
            while (inputStreamReader.ready()) {
                content.append((char) inputStreamReader.read());
            }
        } catch (IOException ex) {
        }
        return content.toString();
    }

    public static boolean write(String fileName, String content) {
        try (FileOutputStream fileOutput = new FileOutputStream(fileNa
me);) {

```

```
        fileOutput.write(content.getBytes());  
    } catch (IOException ex) {  
        return false;  
    }  
    return true;  
}  
}
```