**JAVA课程上机报告**

**学院：计算机科学与技术学院 学号：20009200601 姓名：杜新鹏**

**题目：**

在 Java 学习班中有一个老师，有许多学生。老师是观察者模式中的主题 （Subject）,学生是观察者模式中的观察者（Observer）。老师为学生们讲课， 传授知识；学生可以一字不差的吸收老师讲的内容（以 String 形式存在），并可 以一字不差的复述老师在课堂上讲的内容。 具体要求如下：

(1). 实现老师类（Teacher）、学生类（Student）、主题接口（Subject）、观 察者接口（Observer）、测试类 Test；

(2). 老师讲授的 Java 课程可以被学生选课(registerObserver())、被学生 退课(removeObserver())、课程内容通过 ontifyObservers()方法传授给所有选 课的学生；第 2 / 2 页

(3). 老师讲授的内容通过从文本文件中读取，在 Test 类中操作老师实例读 取文本文件中的知识。老师将读取到的知识传授给选课的学生；

(4). 学生可以将学到的知识记下来(保存到对应学生名的文本文件中)，在 需要的时候（考试、提问）进行展示；

(5). 在 Test 类中可以创建需要的参数，提供相应的操作菜单：上课 (classTime())、考试(examine())、提问(classroomQuestioning)、学生选课 (registerCourse())、学生退课(dropCourse());

(6). 考试方法中学生展学到的全部知识，可以通过传递一些知识掌握情况、 考试时状态等参数，返回相应的考试成绩；

(7). 提问方法中选定学生，传入关键词，展示该学生学到的相关知识；

(8). 考试应当在程序的下一次运行时进行（在程序结束运行后，应当能够 保存学生信息、学生学到的知识、学生与老师的关系；在下一次运行时能够顺利 恢复上次运行时建立的对象实体，以及他们之间的关系，然后进行相应的考试操 作）；

(9). 可以自定义异常类，并在合适的位置使用异常处理机制。

**问题分析：**

该问题需要用到一定的接口知识去解决，接口虽然没有方法体，却也需要包含类所具有的方法，接口可以使被编写的多个程序重复调用，可以节省电脑的存储空间，并且使用接口可以减少程序运行时出现错误，在运行程序时往往更有效率，使用接口来解决这一问题还是比较合适的。但是我的接口还不是很熟悉，所以在代码中并没有很多的用到接口，所以还是会去找时间练习一下基础的。

具体的问题分析，问题中已经写得足够详细了，在此不再过多赘述了。

**代码实现：**

**1.Teacher类**

**package 第三次上机作业;**

**import java.util.ArrayList;**

**import java.io.BufferedReader;**

**import java.io.ByteArrayInputStream;**

**import java.io.IOException;**

**import java.io.InputStreamReader;**

**import java.nio.charset.Charset;**

**public class Teacher implements Subject {**

**String name;**

**String knowledge;**

**int time;**

**ArrayList<String> courses = new ArrayList<String>();**

**ArrayList<String> students = new ArrayList<String>();**

**public Teacher(String name,String course) {**

**this.name=name;**

**this.courses.add(course);**

**}**

**public String toString() {**

**return this.name+" "+this.courses;**

**}**

**public void registerObserver(Student stu) {**

**if(students.contains(stu.name)) {**

**System.out.println("您已经选过该课了!");**

**}**

**else {**

**students.add(stu.name);**

**}**

**}**

**public void removeObserver(Student stu) {**

**if(!students.contains(stu.name)) {**

**System.out.println("您的课表中没有该课程!");**

**}**

**else {**

**students.remove(stu.name);**

**}**

**}**

**public void notifyObservers(Student stu) {**

**stu.knowledge.add(knowledge);**

**}**

**public void getKnowledge(String text) {**

**BufferedReader textIn = new BufferedReader(new InputStreamReader**

**(new ByteArrayInputStream(text.getBytes(Charset.forName("utf8"))), Charset.forName("utf8")));**

**String line;**

**//下面是异常处理机制，防止程序在出错时卡住**

**try {**

**while ( (line = textIn.readLine()) != null ) {**

**if(!line.equals("")){**

**knowledge = line;**

**System.out.println("读取的知识点如下:");**

**System.out.println(line);**

**}**

**}**

**}catch (IOException e) {**

**e.printStackTrace();**

**}**

**}**

**}**

**2.Subject接口：**

**package** 第三次上机作业;

**public** **interface** Subject {

**public** **void** registerObserver(Student stu);

**public** **void** removeObserver(Student stu);

**public** **void** notifyObservers(Student stu);

**public** **void** getKnowledge(String text);

}

**3.Student类：**

package 第三次上机作业;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.io.RandomAccessFile;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.ArrayList;

public class Student {

String name;

ArrayList<String> teacherName = new ArrayList<String>();

ArrayList<String> knowledge=new ArrayList<String>();

ArrayList<String> course = new ArrayList<String>();

public Student(String name,String teacherName) {

this.name=name;

this.teacherName.add(teacherName);

}

public void addKnowledge(String knowledge) {

if(this.knowledge.contains(knowledge)) {

System.out.println("该知识点已经掌握!");

}

else {

this.knowledge.add(knowledge);

}

}

public boolean study() {

// 将老师主题发送的 knowledge 存储到学生名对应的文本文件中

try {

RandomAccessFile studentTxt = new RandomAccessFile(this.name+".txt","rwd");

try {

long sTLength = studentTxt.length();

studentTxt.seek(sTLength);

Calendar calendar = Calendar.getInstance();

SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

byte[] b = new String(formatter.format(calendar.getTime())+" "+knowledge+"\n").getBytes("utf-8");

studentTxt.write(b);

studentTxt.close();

} catch (IOException e) {

e.printStackTrace();

}

} catch (FileNotFoundException e) {

e.printStackTrace();

}

return false;

}

}

**4.Observer接口：**

**package** 第三次上机作业;

**public** **interface** Obsever {

**public** **void** addKnowledge(String knowledge);

**public** **boolean** study();

}

**5.Test类：**

package 第三次上机作业;

import java.util.Random;

import java.util.Scanner;

public class Test {

public static void main(String args[]) {

Scanner in=new Scanner(System.in);

Scanner charin=new Scanner(System.in);

Teacher[] teachers=new Teacher[100];

Student[] students=new Student[100];

while(true) {

System.out.println("\n操作菜单：\n1.创建老师\n2.创建学生\n3.教师列表\n4.学生列表\n5.学生选课\n"

+ "6.学生退课\n7.上课\n8.课堂提问\n9.考试\n10.退出\n请选择操作：");

int flag=in.nextInt();

if(flag==1) {

System.out.println("请输入要新建的老师的个数:");

int number=in.nextInt();

for(int i=0;i<number;i++) {

System.out.println("请输入新建老师的姓名:");

String name = charin.nextLine();

System.out.println("请输入老师讲授的课程:");

String course = charin.nextLine();

System.out.println("请输入老师开课的时间（时间点采用24小时制,所有课程都是持续一小时）:");

int time=in.nextInt();

teachers[i].name=name;

teachers[i].courses.add(course);

teachers[i].time=time;

}

}

else if(flag==2) {

System.out.println("请输入要创建的学生的个数:");

int number=in.nextInt();

for(int i=0;i<number;i++) {

System.out.println("请输入学生的姓名:");

String name = charin.nextLine();

}

}

else if(flag==3) {

System.out.println("老师列表:\n");

for(int i=0;teachers[i]!=null;i++) {

System.out.println("老师姓名\t老师课程\t老师开课时间");

System.out.println(teachers[i].name+"\t"+teachers[i].courses+"\t"+teachers[i].time);

}

}

else if(flag==4) {

System.out.println("学生列表:\n");

for(int i=0;students[i]!=null;i++) {

System.out.println("学生姓名\t所选老师\t所选课程");

System.out.println(students[i].name+"\t"+students[i].teacherName+"\t"+students[i].course);

}

}

else if(flag==5) {

int i;

System.out.println("请输入要选课的学生的姓名:");

String name=charin.nextLine();

for(i=0;students[i]!=null;i++) {

if(students[i].name.equals(name)) break;

}

System.out.println("请输入要选课程的老师姓名:");

String teacherName=charin.nextLine();

System.out.println("请输入要选课程:");

String course=charin.nextLine();

if(students[i].teacherName.contains(teacherName)) {

System.out.println("该学生已经选了该老师的课了!");

}

else {

students[i].teacherName.add(teacherName);

students[i].course.add(course);

}

}

else if(flag==6) {

int i;

System.out.println("请输入要退课的学生的姓名:");

String name=charin.nextLine();

for(i=0;students[i]!=null;i++) {

if(students[i].name.equals(name)) break;

}

System.out.println("请输入要退选课程的老师姓名:");

String teacherName=charin.nextLine();

System.out.println("请输入要退选课程:");

String course=charin.nextLine();

if(!students[i].teacherName.contains(teacherName)) {

System.out.println("该学生的老师列表没有该老师!");

}

else {

students[i].teacherName.remove(teacherName);

students[i].course.remove(course);

}

}

else if(flag==7) {

System.out.println("请输入当前时间（例：此时时间点位于下午两点到三点，则输入14）:");

int t=in.nextInt();

int i;

for(i=0;teachers[i]!=null;i++) {

if(teachers[i].time==t) break;

}

System.out.println(teachers[i].name+"正在上课");

System.out.println("请输入txt文本路径");

String text=charin.nextLine();

teachers[i].getKnowledge(text);

for(int j=0;students[j]!=null;j++) {

if(students[j].teacherName.contains(teachers[i].name)) {

students[i].knowledge.add(teachers[i].knowledge);

}

}

}

else if(flag==8) {

System.out.println("请输入提问的老师姓名:");

String teacherName=charin.nextLine();

System.out.println("请输入提问的知识点:");

String knowledge=charin.nextLine();

while(true) {

Random r = new Random(1);

int ran1=r.nextInt(100);

if(students[ran1]!=null&&students[ran1].teacherName.contains(teacherName)) {

if(students[ran1].knowledge.contains(knowledge)) {

System.out.println("该同学掌握该知识点!");

}

else {

System.out.println("该同学没有掌握该知识点!");

}

break;

}

else continue;

}

}

else if(flag==9) {

System.out.println("请输入考试老师姓名:");

String teacherName=charin.nextLine();

System.out.println("请输入考核知识点:");

String knowledge=charin.nextLine();

for(int i=0;students[i]!=null;i++) {

if(students[i].teacherName.contains(teacherName)) {

if(students[i].knowledge.contains(knowledge)) {

System.out.println("该班同学掌握该知识点!");

}

else {

System.out.println("该班同学未掌握该知识点!");

}

break;

}

else continue;

}

}

else if(flag==10) break;

}

}

}

**遇到的问题：**

这个问题还是蛮复杂的，一开始看到的时候确实是无从下手，但是最后还是慢慢的开始做了，果然，只要开始了，一切就都好解决了。开始的时候由于问题过于宏大，很容易让人头疼，于是我把问题分解成了很多个小问题，逐渐解决后就最终完成了代码。主要问题还是在接口以及文件读取和生成方面，这方面自己比较陌生，最终是搜集了一些资料才最终完成的。

**思考总结：**

自己的水平还是有限，平时应该多练习一些代码，这样在做到大题目时才会有迹可循，不至于头疼到无法下手，多练习多练习多练习。