

# 浙江大学 实验报告

课程名称: Java 应用技术 指导老师: 翁恺 学生姓名: 尹嘉权  
实验名称: Grading System 实验类型: 编程实现 学生学号: 3120000419

## 一、实验目的和要求

目的:

了解 Java 基本语法, 熟悉文件输入输出流和容器的使用。

要求:

Write a program that maintains a "database" of names and scores of students.

A student has a name consists of several words and records of scores of varies courses. Each record has a name of the course and the marks the student gets.

User can input or import information and scores for students and retrieve them later. To input a score, the user may key in the name of the student, the name of the course and the marks. Your program then try to find the student in the database and add or replace the score for the course. If the student can not be found in the database, a new student should be created and be put into the database. If the course can not be found in the records of the student, a new record for the course should be created, otherwise, the score should be replaced by the newly inputed one.

The user can provide name of a .CSV file which contains records of students information to import the data into the database. The columns in the .CSV files are:

`name, course name, score`

User can input the name of a student to retrieve the scores of all the courses he/she gets, along with the total and average marks. User can input name of a course to retrieve a list of every student who has marks for this course, along with the number of the students and the average marks.

Your program is able to store the database into a file and read them back at the beginning of the execution.

The whole system is a Unix style shell system, which means you are going to develop one program for one function of your system, not a concrete one big program, no menus, no GUIs.

## 二、实验内容和原理

内容:

编写一个成绩管理系统，要求是一个 Unix 形式下的 shell 系统，分别实现导入/更新学生成绩，查询学生/课程情况，主要这里面要求并不是写一个整合好的系统，而是一个模块功能对应一个 shell 命令，其中

**Input** 函数，输入 `name,course,mark`，优先查找数据库，如果数据库有同人名同课程的数据，则更新成绩 `mark`，如果其中一项或者两项都没有，则加入新的学生课程数据。

**Import** 函数，输入一个.csv 文件，把文件中的数据导入到数据库中，注意这时候做的是更新，而非把原来的数据库删除并且替换成当前的这个，更新原则同 **Input**。

**SearchStudent** 函数，输入一个学生名，如果在数据库中有任何关于这名学生的信息，则进行统计，打印该学生每一门课程和对应成绩，最后打印总课程数量和平均分；如果没有，应给与相关提示语。

**SearchCourse** 函数，输入一个课程名，如果在数据库中有任何关于这个课

程的任何相关信息，则进行统计，打印参与该课程的所有学生和他们对应的成绩，最后打印总学生数量和平均分；如果没有，应给与相关提示语。

PrintAll 函数，打印数据库所有的学生课程信息。

原理：

定义 Base.java，里面包含两个类，一个是数据信息的类，另一个类是管理数据库的，实现对数据库的基本读写操作，以及对输入的字符串转化为数据的类的基本功能。

自定义 Input, Import, SearchStudent, SearchCourse, PrintAll 函数，主要是通过数据库文件的输入输出来访问数据库，更新和打印相关的内容。其中，对数据库文件的输入使用的是 `BufferedReader`<sup>1</sup>；对数据库文件的输出使用的是 `FileWriter`<sup>2</sup>。

### 三、主要仪器设备

MacBook Pro, MacOSX 10.10, 2.6GHz Intel Core i5, 8G RAM

JSE: JavaSE-1.8

JDK: oracle-java8-jdk

### 四、操作方法和实验步骤

分析题目，然后打开 Eclipse 建立相对应的工程文件，输入代码，进行初步的调试运行之后把所有的.java 文件整合到一个文件夹中，打开 terminal，找到相对应的程序目录，用 javac 通过 JDK 编译得到相对应的 class 文件，然后用 java 命令运行调试。注意一开始在这个整合文件里面要 touch database.csv 来建立空的数据库（如果一开始没有数据库文件的话）

具体的代码可以参见/src 文件夹下的各个.java 文件，其中 Base.java 为其他各个函数所共用的公共文件。

另外，对于输入给出约定，使用 Input 函数的时候，输入不是 name course mark，而是 name,course,mark，加入中间的两个 comma 是为了和 csv 数

---

<sup>1</sup> Reference: <http://www.cnblogs.com/lovebread/archive/2009/11/23/1609122.html>

<sup>2</sup> Reference: <http://www.cnblogs.com/yezhenhan/archive/2012/09/10/2678690.html>

据文件的输入统一；另外，对于数据的分数，要求是整数，而不能是浮点数。  
具体使用方法如下：

Input function: java Input [name,course,mark] [回车]

或者    java Input [回车]  
         [name,course,mark] [回车]

Import function: java Import [\*\*\*.csv] [回车]

或者    java Import [回车]  
         \*\*\*.csv [回车]

SearchStudent function: java SearchStudent [name] [回车]

或者    java SearchStudent [回车]  
         [name] [回车]

SearchCourse function: java SearchCourse [course] [回车]

或者    java SearchCourse [回车]  
         [course] [回车]

PrintAll function: java PrintAll [回车]

五、实验数据记录和处理

```
Last login: Sun Nov 30 19:04:59 on ttys000
172-13-0-196:~ SmilENow$ cd Desktop/GradingSystem
172-13-0-196:GradingSystem SmilENow$ ls
Base.java          Input.java          SearchCourse.java   test.csv
Import.java        PrintAll.java       SearchStudent.java
172-13-0-196:GradingSystem SmilENow$ touch database.csv
172-13-0-196:GradingSystem SmilENow$ javac *.java
172-13-0-196:GradingSystem SmilENow$ java PrintAll
There is no data item in current database!
172-13-0-196:GradingSystem SmilENow$ java Input aa,math,95
aa's math has been added, and the mark is 95
172-13-0-196:GradingSystem SmilENow$ java Input bb,math,70
bb's math has been added, and the mark is 70
172-13-0-196:GradingSystem SmilENow$ java SearchCourse
Math
Cannot find course called :Math
172-13-0-196:GradingSystem SmilENow$ java SearchCourse
math
The gradebook of math :
+-----+-----+
| Student | + | Mark |
+-----+-----+
| aa      | + | 95   |
| bb      | + | 70   |
+-----+-----+
Total student(s): 2
Average marks: 82.50
172-13-0-196:GradingSystem SmilENow$
```

用 touch database.csv 建立空的数据库文件，然后编译 java 文件，之后先打印当前数据库，给出合理的应答，随后加入两个学生的关于 math 这门课的信息，用 SearchCourse 来搜索 Math，给出合理的反馈信息，又搜索 math，打印这门课的所有学生信息的数据，并给出学生总数和平均分，符合要求。

下面测试导入(Import)功能：

首先 test.csv 文件内容如下：

	A	B	C
1	aa	math	100
2	aa	ENG	78
3	aa	PHY	88
4	aa	c0	93
5	aa	c1	41
6			

可以看到，新导入的 aa 同学的 math 成绩有变动，同时多加了几门课程，我们先通过导入然后 SearchStudent aa 看看结果：

```

172-13-0-196:GradingSystem SmileNow$ java Import test.csv
aa's math has been changed to 100
aa's ENG has been added, and the mark is 78
aa's PHY has been added, and the mark is 88
aa's c0 has been added, and the mark is 93
aa's c1 has been added, and the mark is 41
172-13-0-196:GradingSystem SmileNow$ java SearchStudent
aa
The gradebook of aa :
+-----+-----+
| Course | Mark |
+-----+-----+
| math   | 100  |
| ENG    | 78   |
| PHY    | 88   |
| c0     | 93   |
| c1     | 41   |
+-----+-----+
Total course(s): 5
Average marks: 80.00
172-13-0-196:GradingSystem SmileNow$ █

```

符合要求，同时被修改（replace）的数据也有不同的输出来提示。

最后打印当前数据库的所有信息：

```

172-13-0-196:GradingSystem SmileNow$ java PrintAll
The gradebook :
+-----+-----+-----+
| Student | Course | Mark |
+-----+-----+-----+
| aa      | math   | 100  |
| bb      | math   | 70   |
| aa      | ENG    | 78   |
| aa      | PHY    | 88   |
| aa      | c0     | 93   |
| aa      | c1     | 41   |
+-----+-----+-----+
172-13-0-196:GradingSystem SmileNow$ █

```

实验正确！

## 六、实验结果与分析

实验结果正确，给出友好的用户交互界面。支持实现导入/更新学生成绩，查询学生/课程情况，对于没有该课程、该学生、当前数据库没有数据给出

合理的反馈，同时在 **Input** 和 **Import** 的时候支持更新当前内容，而不是直接的添加，均符合实验预期要求。

## 七、讨论心得

在整个实验中，对于 **Java** 的关于文件 **IO** 有了进一步的了解，类比了几种不同的写文件的方法，采取了相对来说效率较高的 **FileWriter**。另外在文件 **IO** 中，通过使用 **try catch** 的异常抛出让程序显得更加 **OOP**，能有效避免退栈造成的错误，万一发生异常，利用 **printStackTrace()**来更加有效地让程序员和用户得知具体错误来源，方便调试和用户使用。