Assignment 2

Introduction

This assignment asks you to implement the function of administrating and analyzing student information tables by *JavaFX* and *.csv* files. You may refer to the tutorial provided in lab 6 and 7.

The format and column names of the *csv* file remain the same, but the order of the columns may be changed. That is, there will be exactly seven columns in the csv file, and each column name will appear only once, but the order of the columns may be different.

```
"ID","Name","Gender","Department","GPA","Credit Earned","Birthday"
"10004325","S2","FEMALE","D2","0.0","0","1999-12-31"
"10001235","S10086","MALE","D1","4.0","62","2000-01-01"
"10001234","S1","MALE","D1","4.0","63","2000-01-01"
```

To simplify your work, there will only be two kinds of values in *Gender* column - `MALE` and `FEMALE`. The birthday format should be like `yyyy-MM-dd`.

The format of *csv* files in this assignment is as same as the format of the output file in Assignment 1. Package `apache.commons.commons-Csv` will help you read or write *csv* files in this format.

Requirements

1. Use `FileChooser` to open an existing *csv* file, and then show the following five columns `ID, Name, Department, GPA, Credit Earned` in the main part of the user interface. The FileChooser should **only** show files with an extension of ***.csv***. See the 2nd and the 3rd items in *Sample Program* part.

2. The program should allow users to select a row (i.e. a specific student). When a row is selected, all seven attributes should be shown in the detail box on the right. See the 4th item in *Sample Program* part.

3. The program should allow users to create a new table without any other rows except the first row (header), i.e., it should contain **all seven columns**. See the 5th item in *Sample Program* part.

4. The program should allow users to add a student (i.e., a row), where the information cannot be empty. Note that `GPA` should belong to the interval [0.0, 4.0], and `Credit Earned` should be an integer greater than or equal 0. Besides, users should select `Gender` by `ChoiceBox`, and users should select `Birthday` by `DatePicker`. See the 6th and the 7th items in *Sample Program* part.

5. The program should allow users to delete a selected student (i.e., a row). See the 10th item in *Sample Program* part.

6. The program should allow users to edit a selected student (i.e., a row), where the requirement is as same as the requirement of adding a new student. See the 7th item in *Sample Program* part.

7. The program should allow users to save the information by *csv* format. The format should be RFC4180, and the content should be enclosed by double quotes. If you use `commons-Csv` package, you can just

use `CSVFormat.RFC4180.withQuoteMode(QuoteMode.ALL)`. The *csv* file should contain the header (column name), while the order of the columns will not be restricted. See the 8th and the 9th items in *Sample Program* part.

   - For files that were newly created, the program should use `FileChooser` to ask for a saving path. Users can type the file name in the textbox of the FileChooser.
   - For files that were *opened* (i.e., existing files), the program has to write to the original file directly.

8. The program should allow *Save As* operation, i.e. to save a copy of the table. The path should be asked by a `FileChooser`, and the format is the same as the **7th** item of the `Requirements`.

9. The program should allow opening several tables at the same time. `TabPane` and `Tab` in *JavaFX* are recommended.

10. (bonus +10 points)

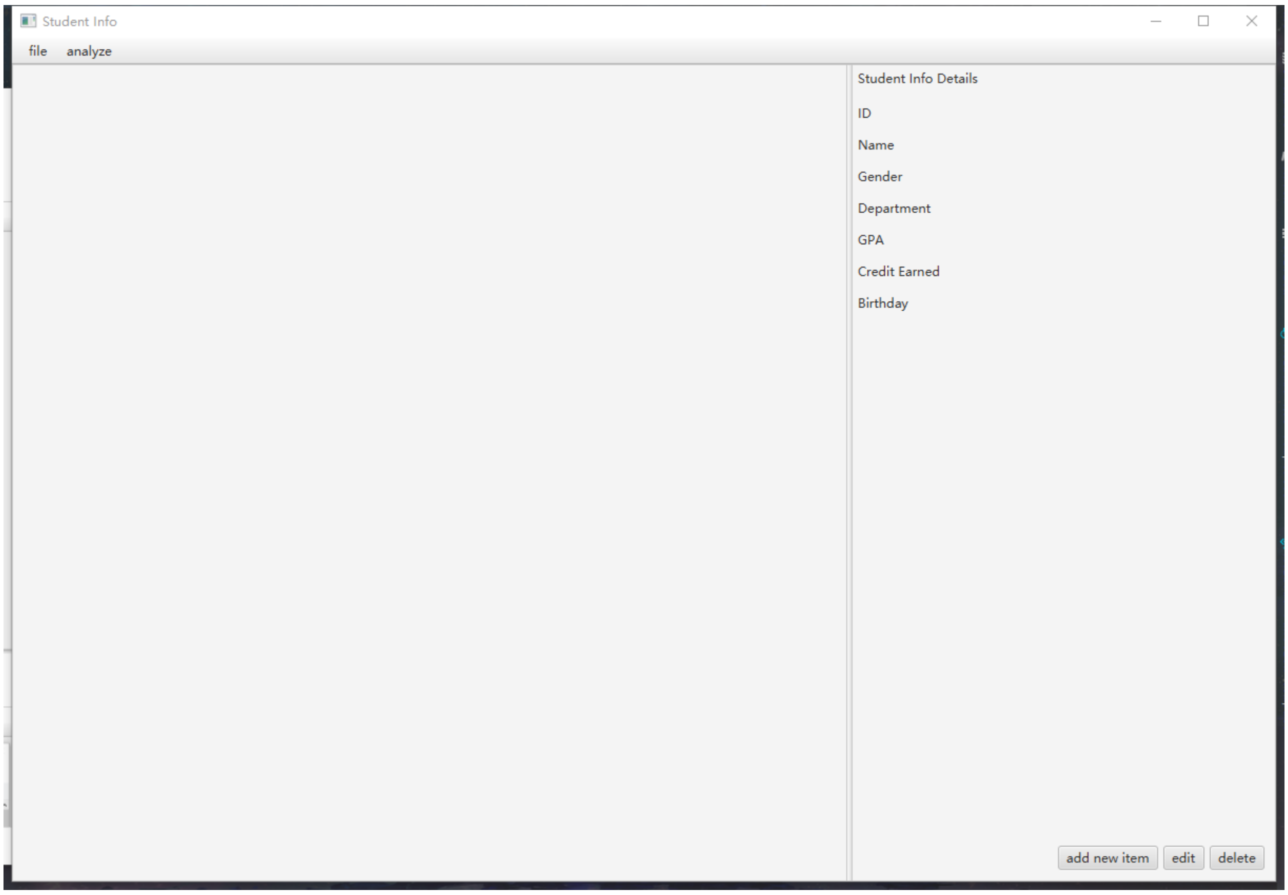
    - When the user closes the table, the program is able to ask whether users want to save the table or not. The format is still the same as the 7th item in the `Requirements`. (5 points)
    - When the user closes the application (i.e. close the window), the program is able to ask whether users want to save the tables **one by one**. Note that if there are no changes applied to the table, the program needn't to ask. (5 points)

11. (bonus +10 points) search for some special contents

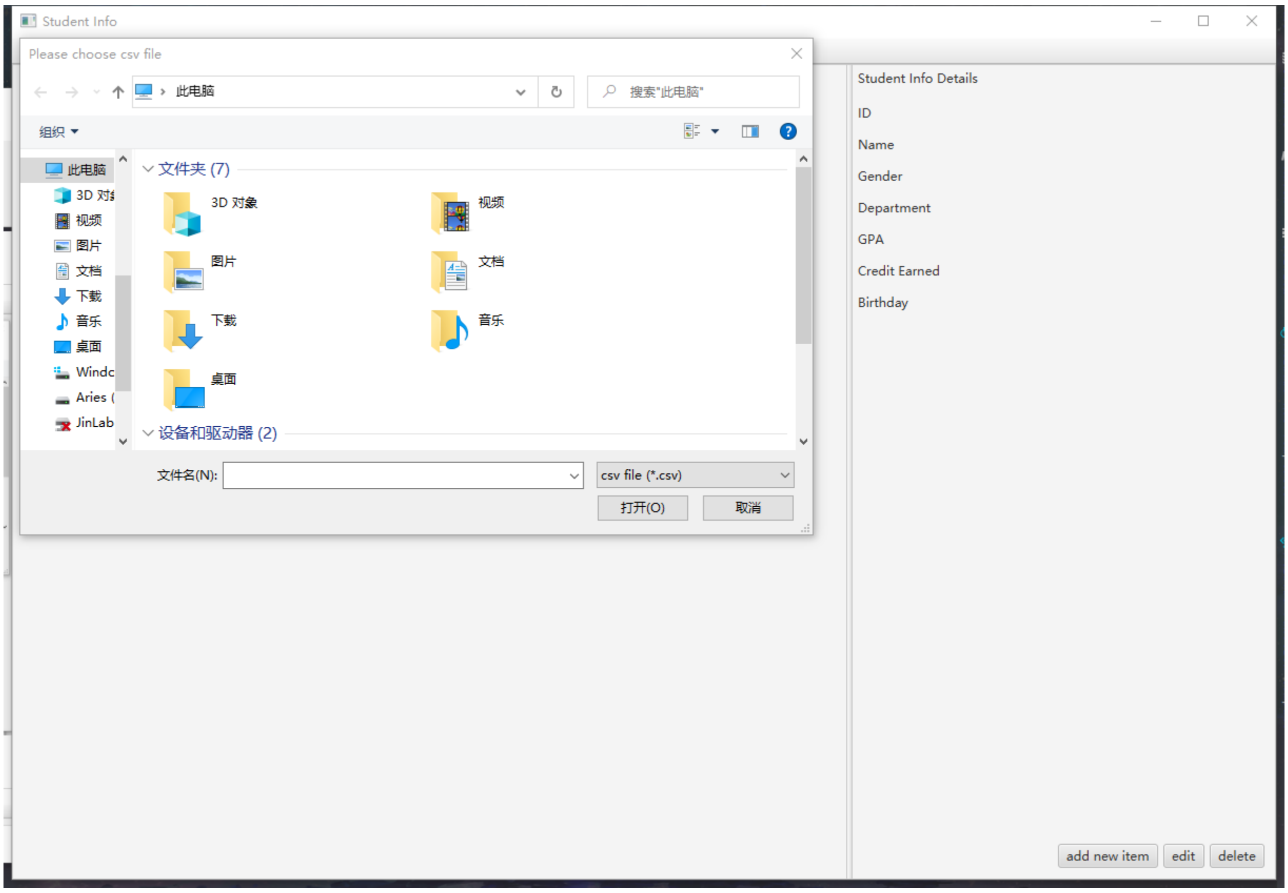
    - You should use `contains` to implement the basic fuzzy search function.
    - The function can be implemented as either kind of situations:
      - Search one single result at a time, and use a button like *next* to move on to the next result. The student (i.e. row) should be selected and all seven attributes should be displayed in the detail box on the right.
      - Search all results at the same time. All students (i.e. rows) should be selected, and the program should display all attributes of any one of the selected students in the detail box on the right.
    - The search function can be applied on all text. You do not have to apply it in a particular column.

Sample Program

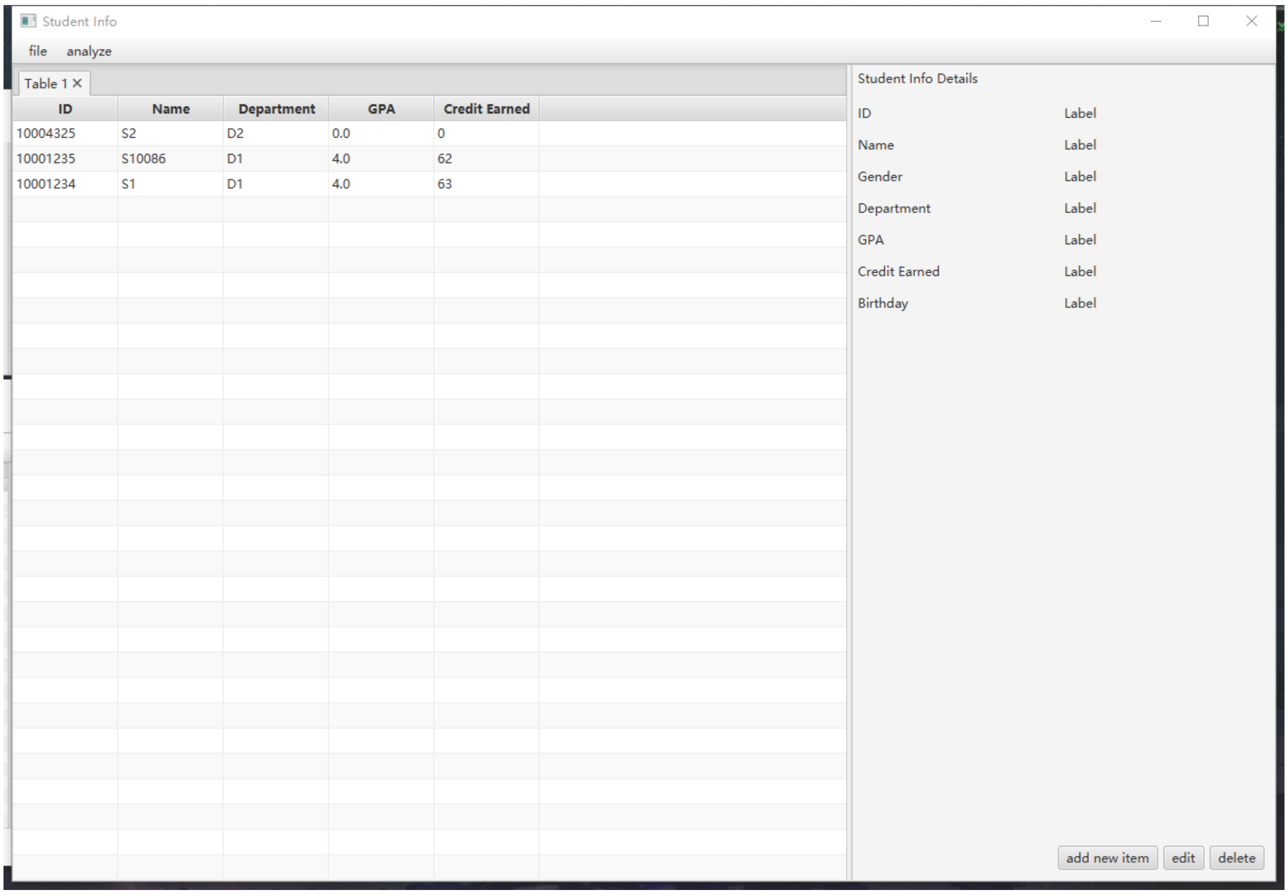
1. When you start the application:



2. Click on `file->open`, display the `FileChooser`:



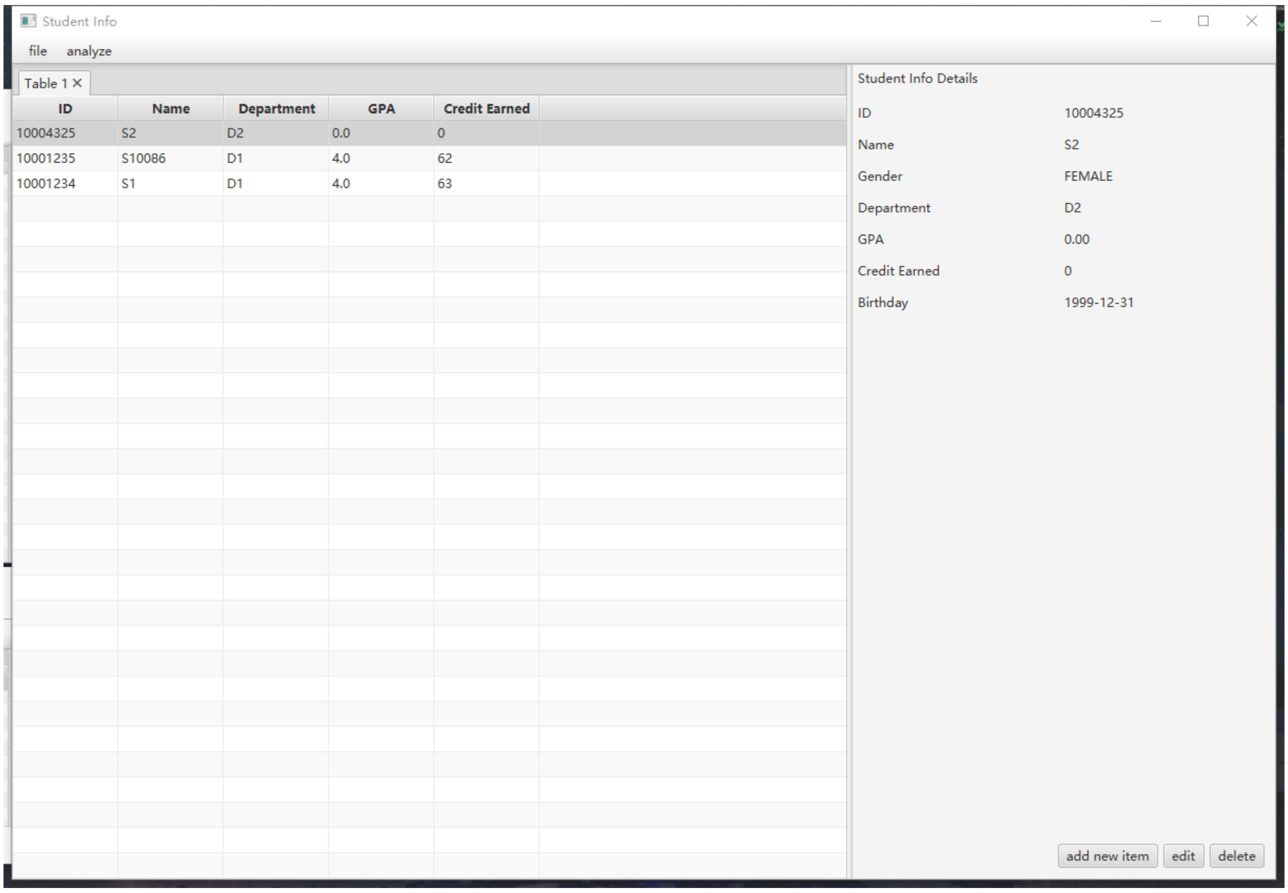
3. Open `example.csv`:



ID	Name	Department	GPA	Credit Earned
10004325	S2	D2	0.0	0
10001235	S10086	D1	4.0	62
10001234	S1	D1	4.0	63

ID	Label
Name	Label
Gender	Label
Department	Label
GPA	Label
Credit Earned	Label
Birthday	Label

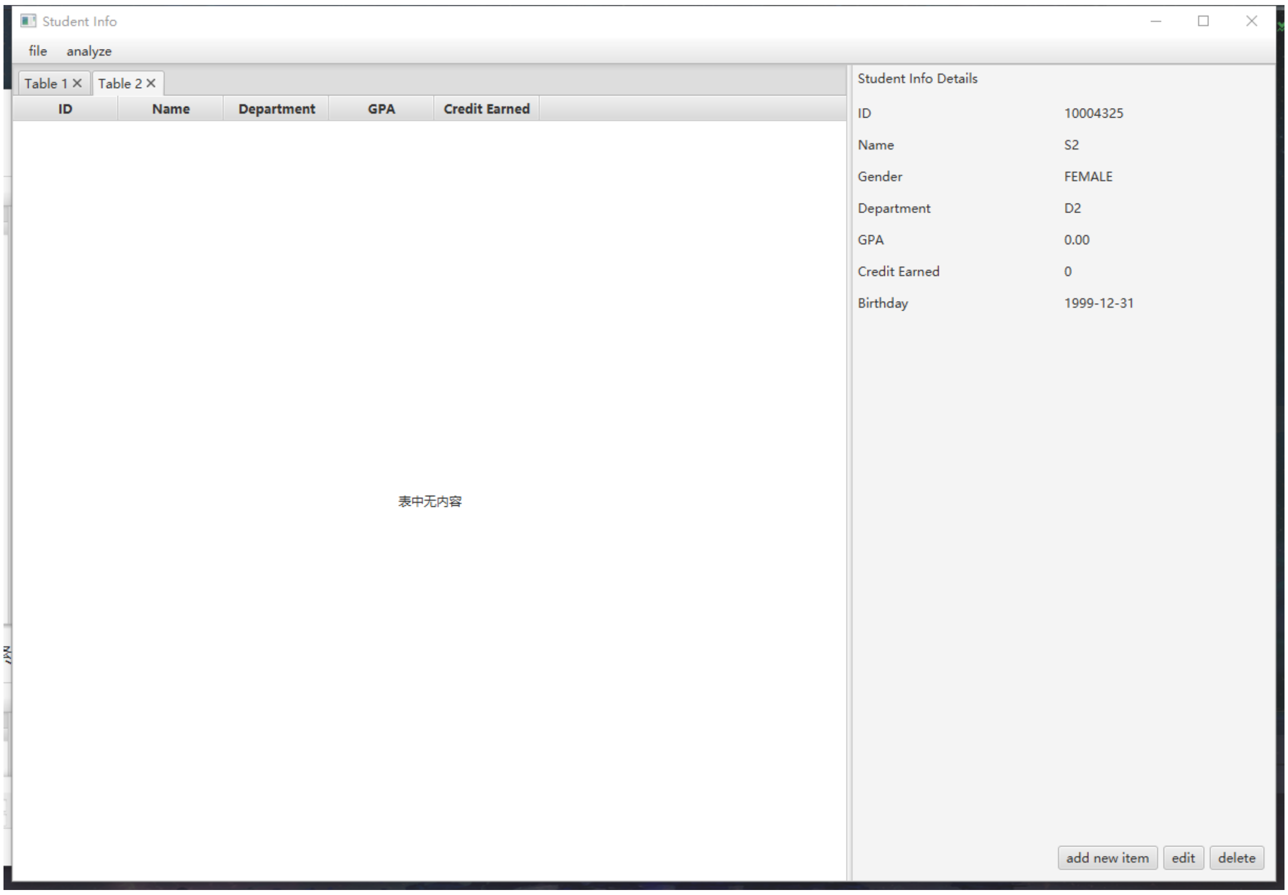
4. Select the first student:



ID	Name	Department	GPA	Credit Earned
10004325	S2	D2	0.0	0
10001235	S10086	D1	4.0	62
10001234	S1	D1	4.0	63

ID	10004325
Name	S2
Gender	FEMALE
Department	D2
GPA	0.00
Credit Earned	0
Birthday	1999-12-31

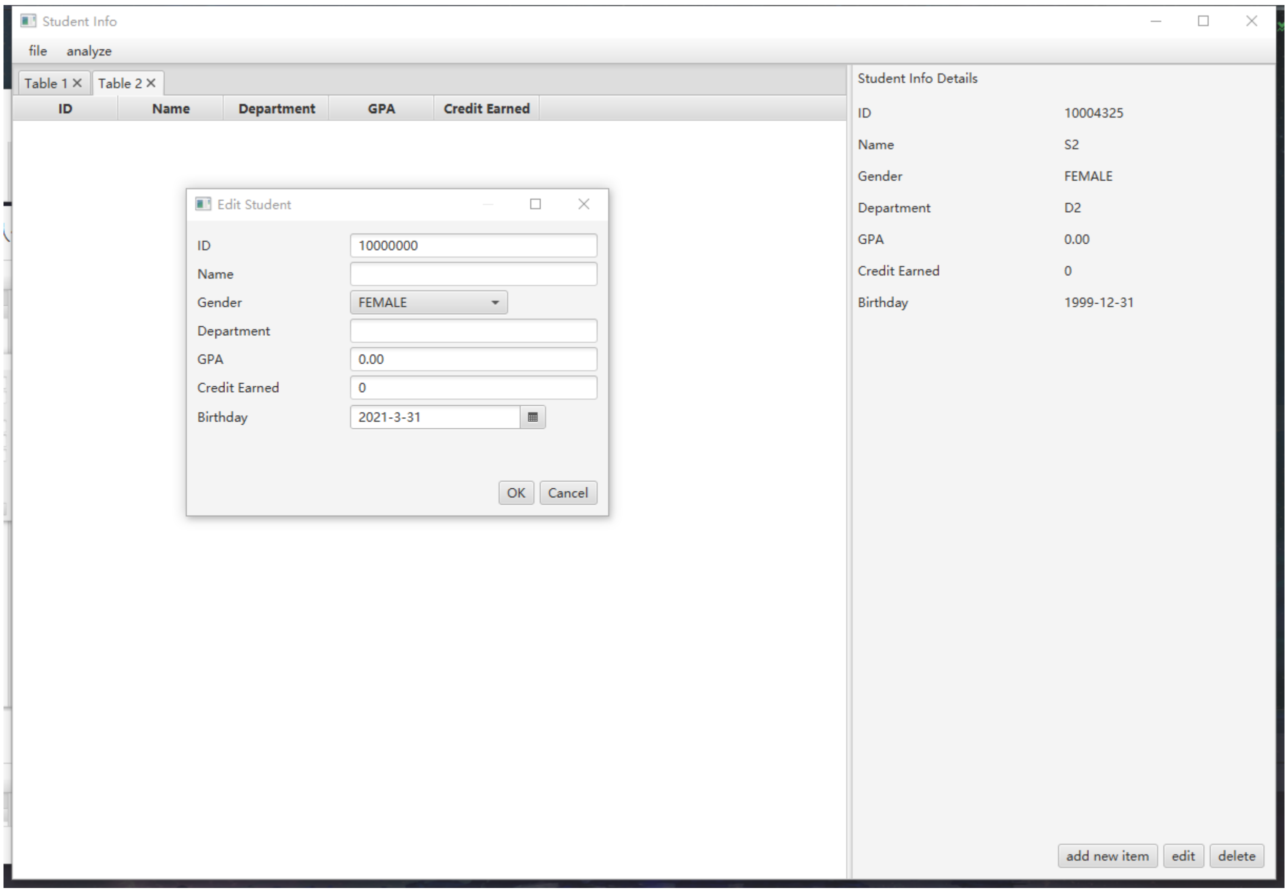
5. Click on `file->new` and create a new table:



Student Info Details

ID	10004325
Name	S2
Gender	FEMALE
Department	D2
GPA	0.00
Credit Earned	0
Birthday	1999-12-31

6. Click on `add new item`, display the `Edit Student` window (note that the default values here are just samples):

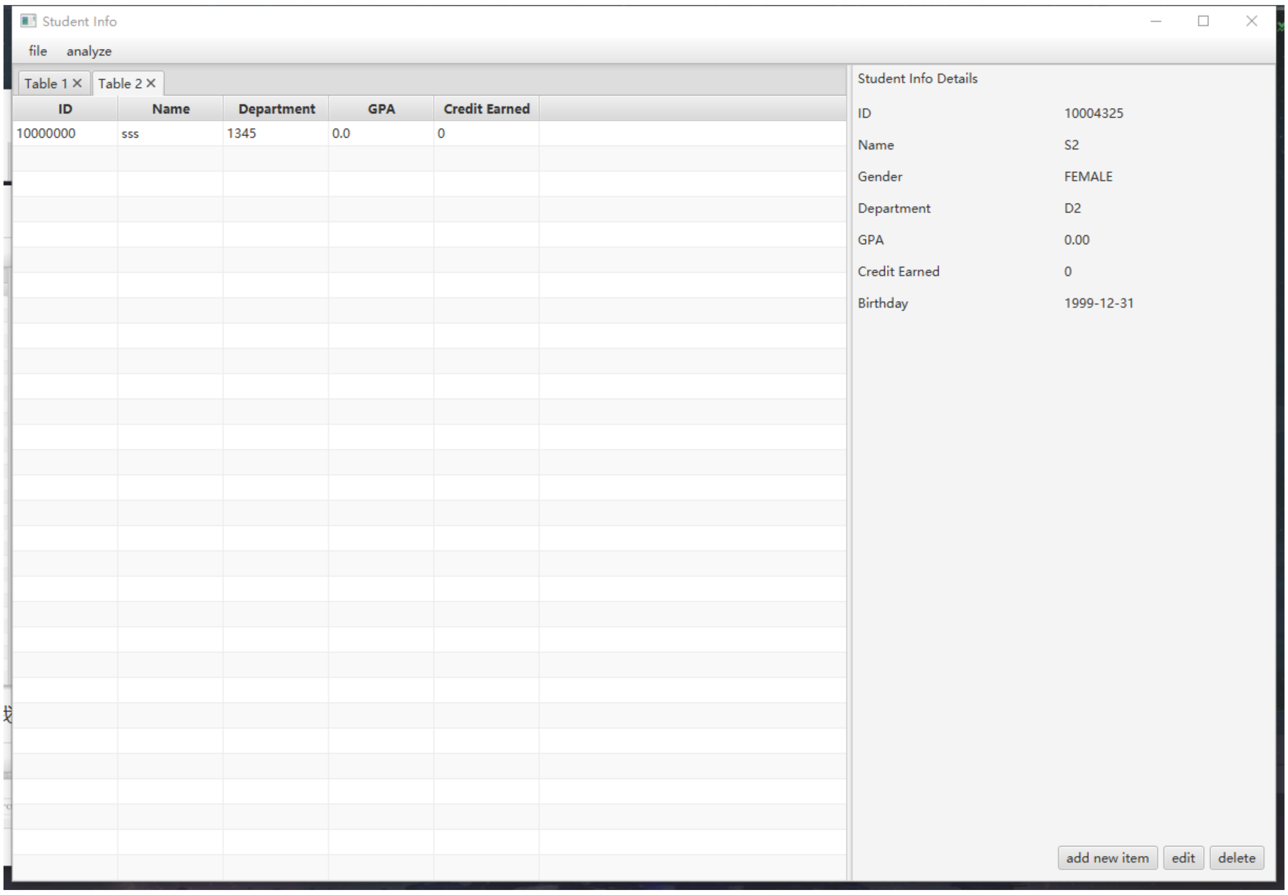


ID	10000000
Name	
Gender	FEMALE
Department	
GPA	0.00
Credit Earned	0
Birthday	2021-3-31

Student Info Details

ID	10004325
Name	S2
Gender	FEMALE
Department	D2
GPA	0.00
Credit Earned	0
Birthday	1999-12-31

7. After finishing input, display the table as follows:

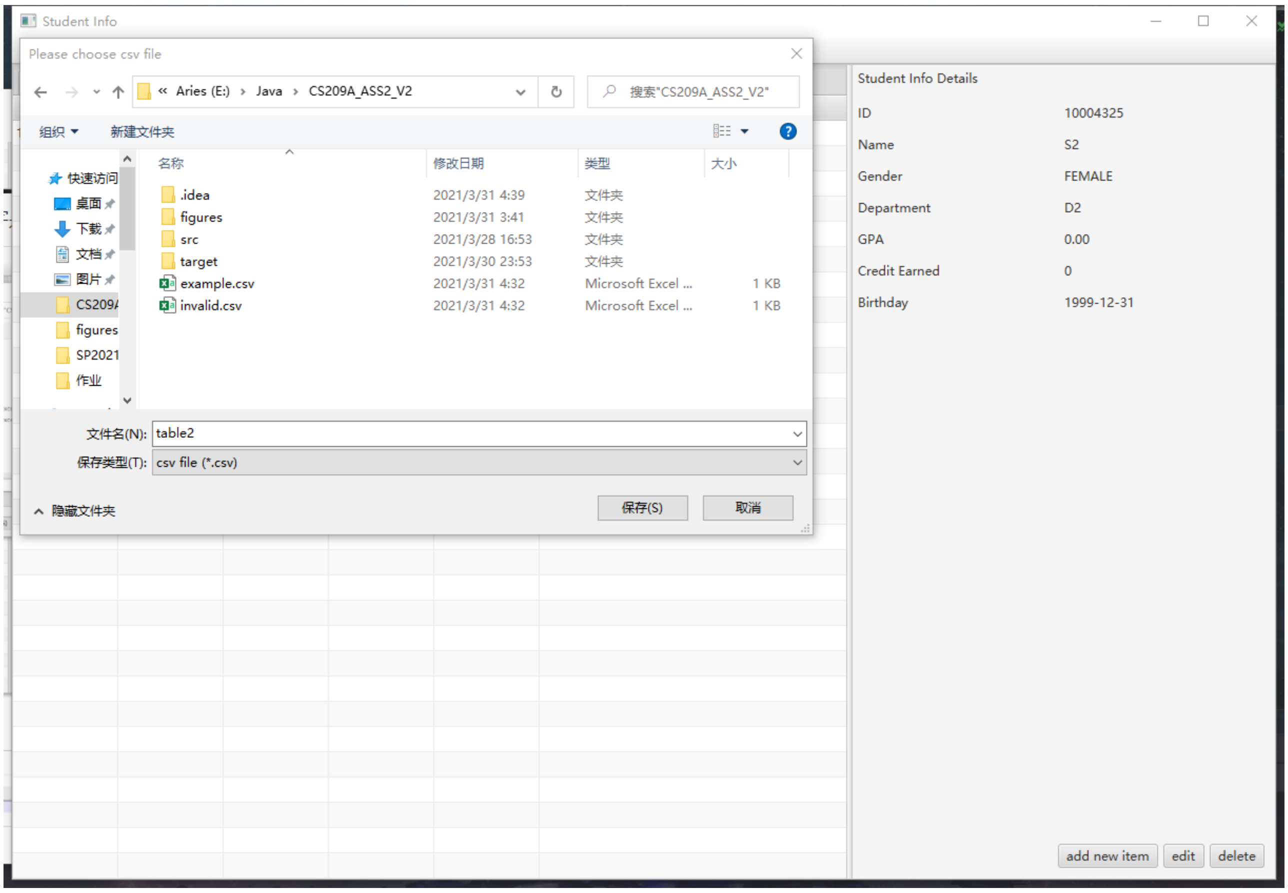


ID	Name	Department	GPA	Credit Earned
10000000	sss	1345	0.0	0

Student Info Details

ID	10004325
Name	S2
Gender	FEMALE
Department	D2
GPA	0.00
Credit Earned	0
Birthday	1999-12-31

8. Click on `file->save`, display the `FileChooser` (this "table2" was manually input):



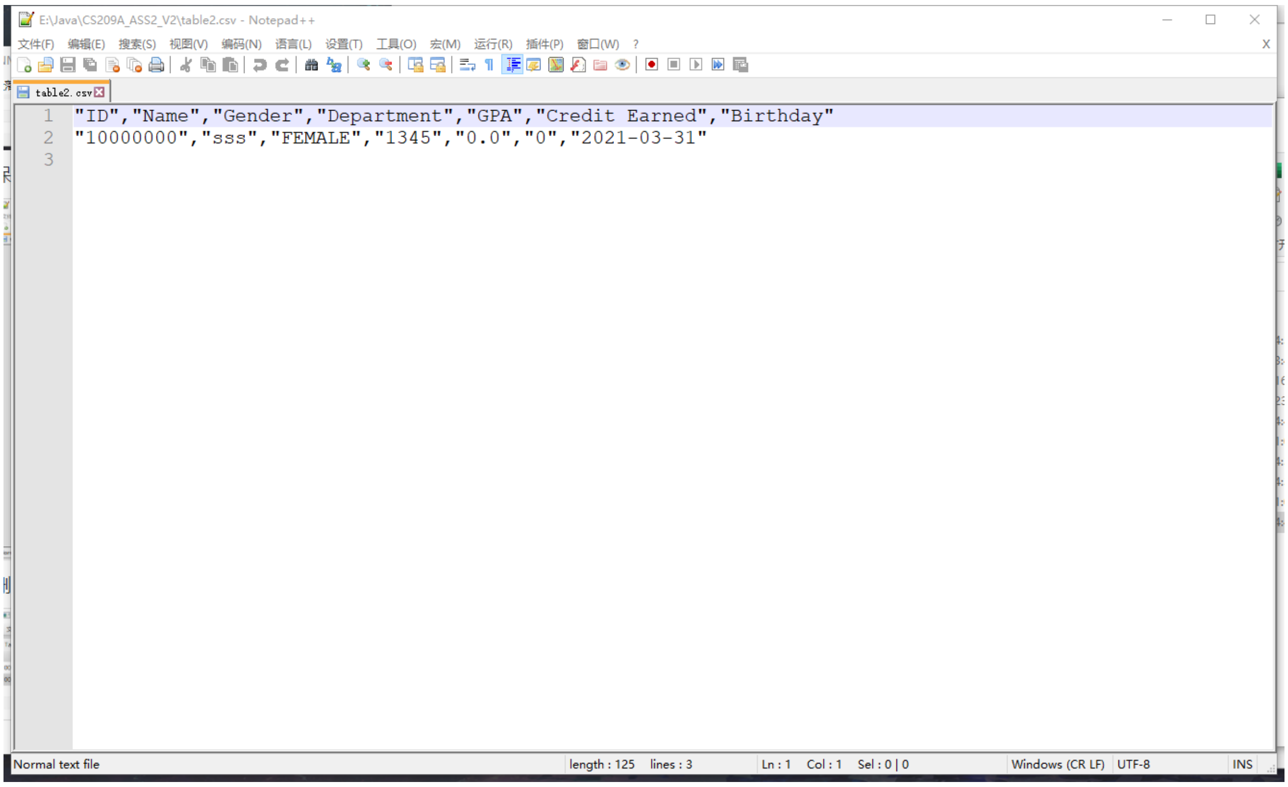
名称	修改日期	类型	大小
.idea	2021/3/31 4:39	文件夹	
figures	2021/3/31 3:41	文件夹	
src	2021/3/28 16:53	文件夹	
target	2021/3/30 23:53	文件夹	
example.csv	2021/3/31 4:32	Microsoft Excel ...	1 KB
invalid.csv	2021/3/31 4:32	Microsoft Excel ...	1 KB

文件名(N): table2
保存类型(T): csv file (*.csv)

Student Info Details

ID	10004325
Name	S2
Gender	FEMALE
Department	D2
GPA	0.00
Credit Earned	0
Birthday	1999-12-31

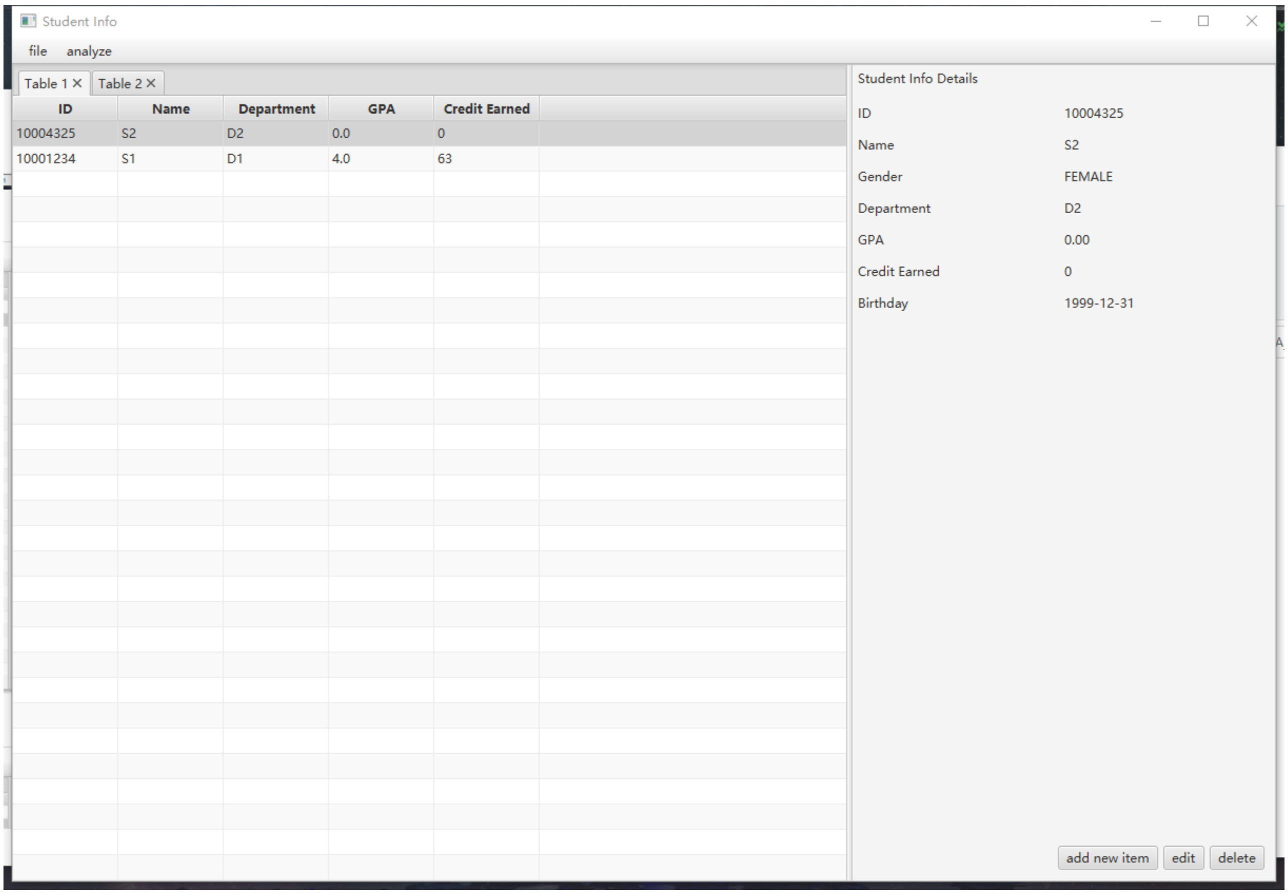
9. Save `table2.csv` successfully:


E:\Java\CS209A_ASS2_V2\table2.csv - Notepad++
文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?
table2.csv

```
"ID","Name","Gender","Department","GPA","Credit Earned","Birthday"
"10000000","sss","FEMALE","1345","0.0","0","2021-03-31"
```

Normal text file
length : 125 lines : 3
Ln : 1 Col : 1 Sel : 0 | 0
Windows (CR LF) UTF-8
INS

10. Delete S10086 from *table1* (`example.csv`):

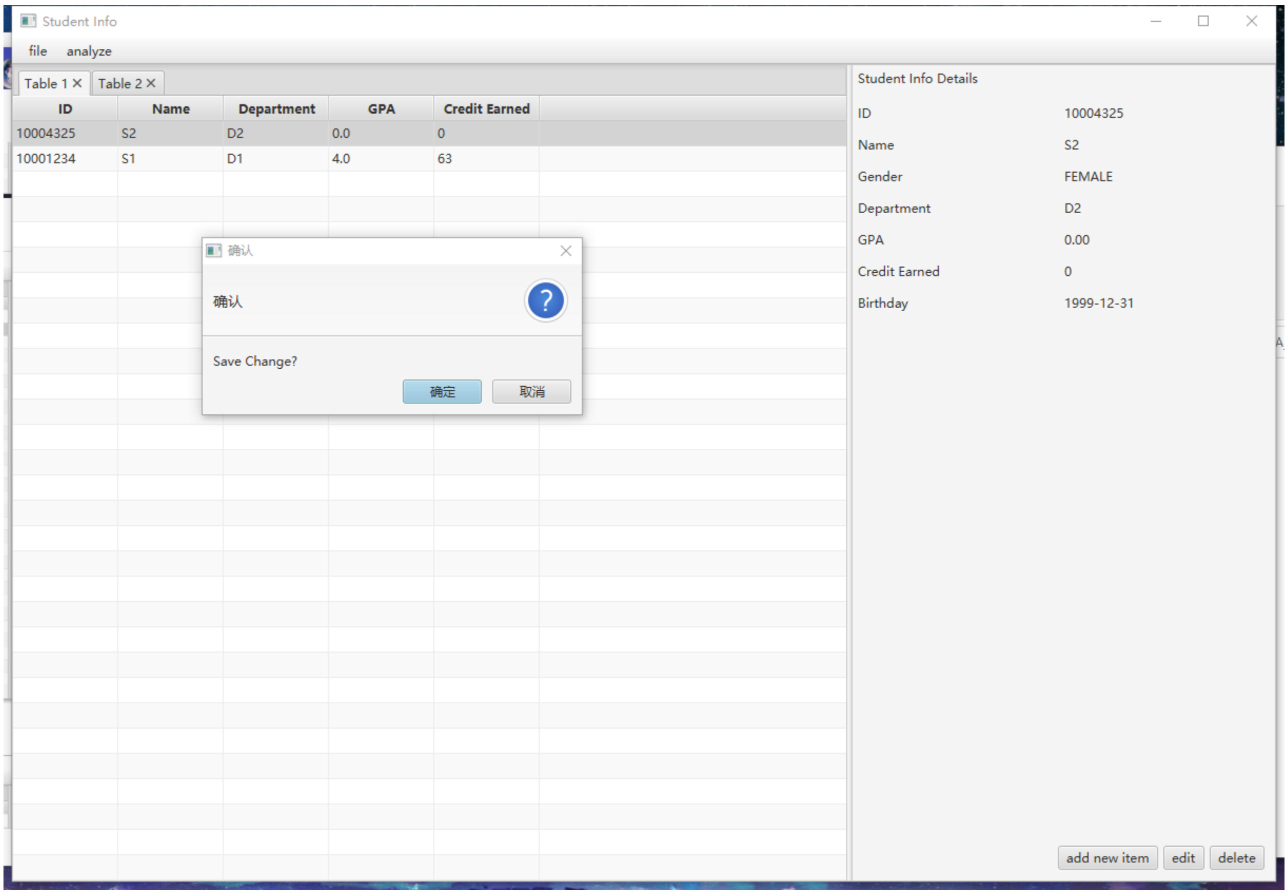


ID	Name	Department	GPA	Credit Earned
10004325	S2	D2	0.0	0
10001234	S1	D1	4.0	63

Student Info Details

ID	10004325
Name	S2
Gender	FEMALE
Department	D2
GPA	0.00
Credit Earned	0
Birthday	1999-12-31

11. Close *table1* (not saved):



ID	Name	Department	GPA	Credit Earned
10004325	S2	D2	0.0	0
10001234	S1	D1	4.0	63

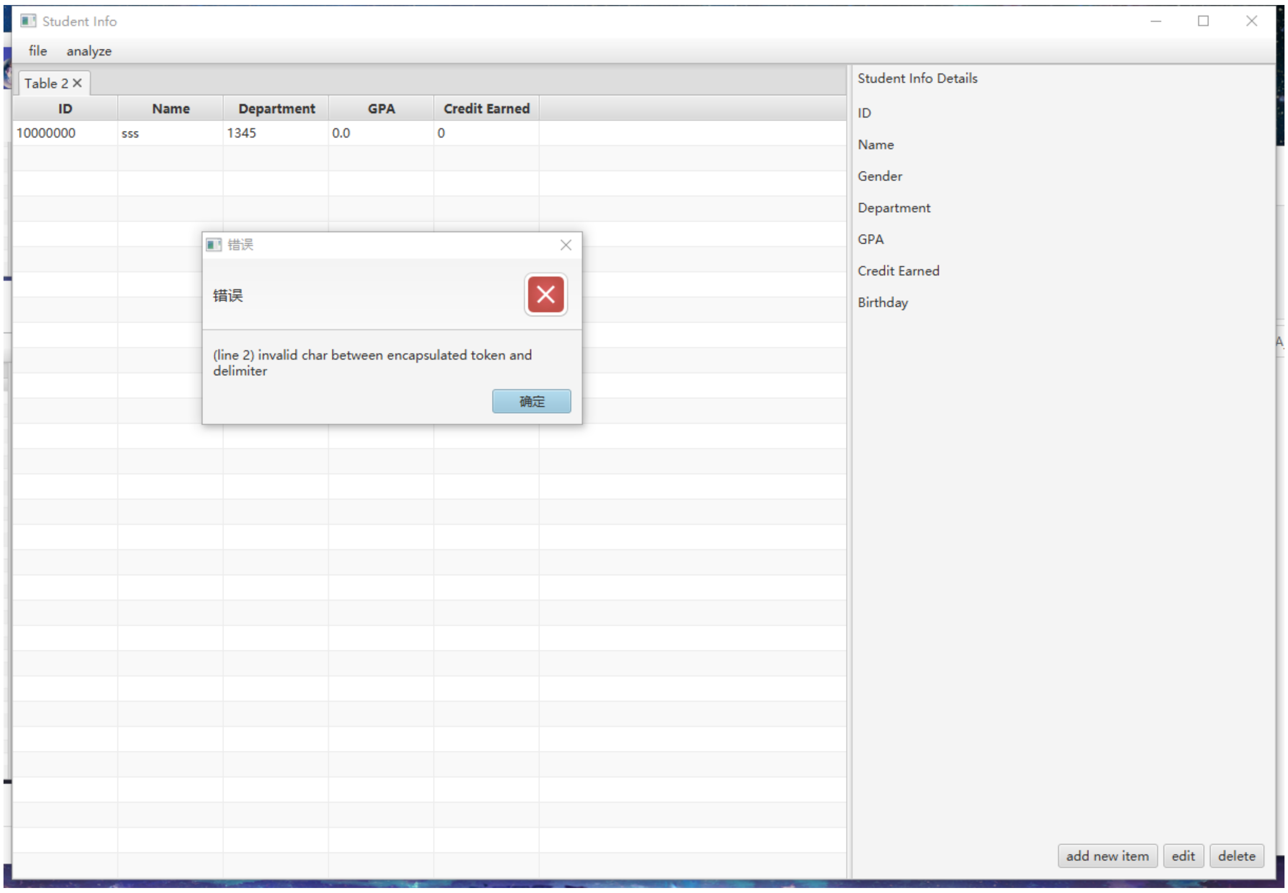
确认

Save Change?

Student Info Details

ID	10004325
Name	S2
Gender	FEMALE
Department	D2
GPA	0.00
Credit Earned	0
Birthday	1999-12-31

12. Open an invalid file (`invalid.csv`):



ID	Name	Department	GPA	Credit Earned
10000000	sss	1345	0.0	0

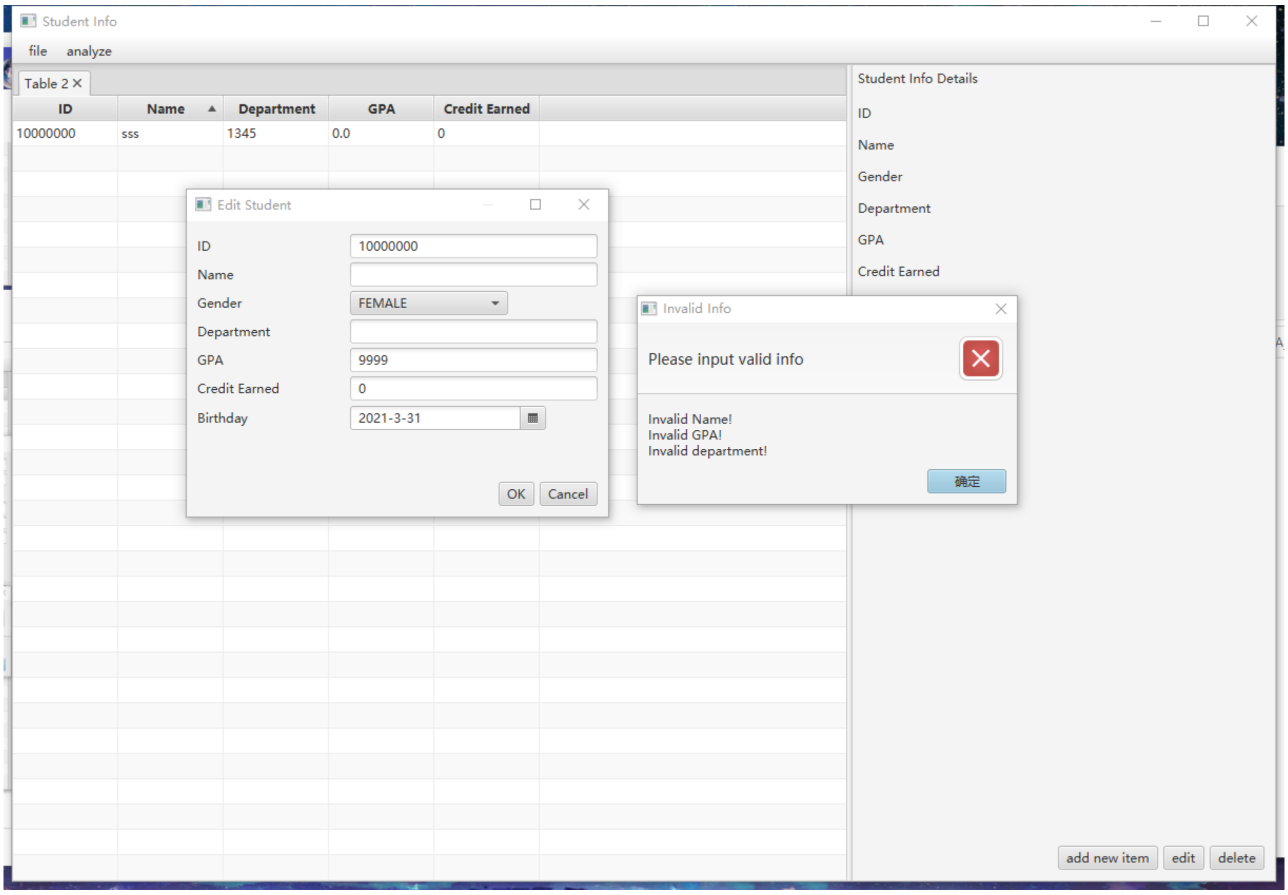
错误

(line 2) invalid char between encapsulated token and delimiter

Student Info Details

ID
Name
Gender
Department
GPA
Credit Earned
Birthday

13. Add an invalid item:



ID	Name	Department	GPA	Credit Earned
10000000	sss	1345	0.0	0

Edit Student

ID	10000000
Name	
Gender	FEMALE
Department	
GPA	9999
Credit Earned	0
Birthday	2021-3-31

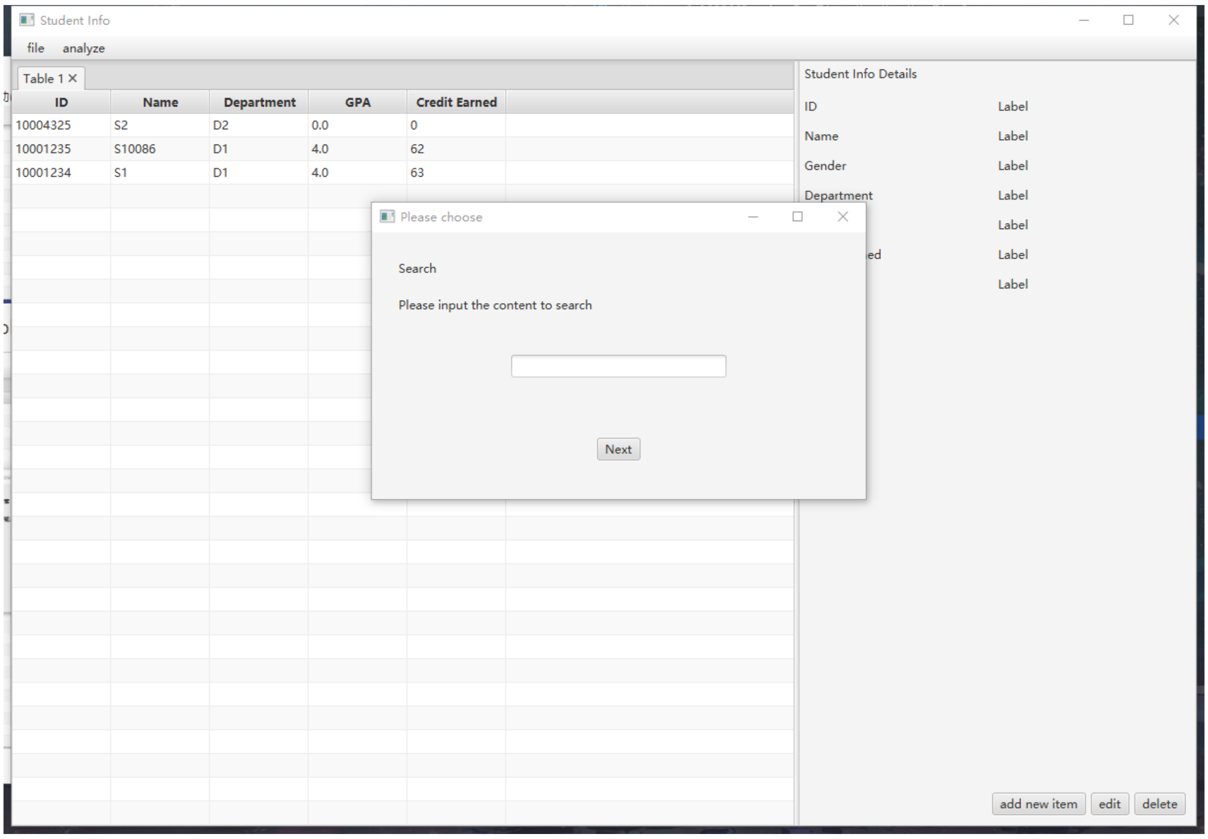
OK Cancel

Please input valid info

Invalid Name!
Invalid GPA!
Invalid department!

14. Search (based on the original `example.csv` file):

    1. A sample GUI:



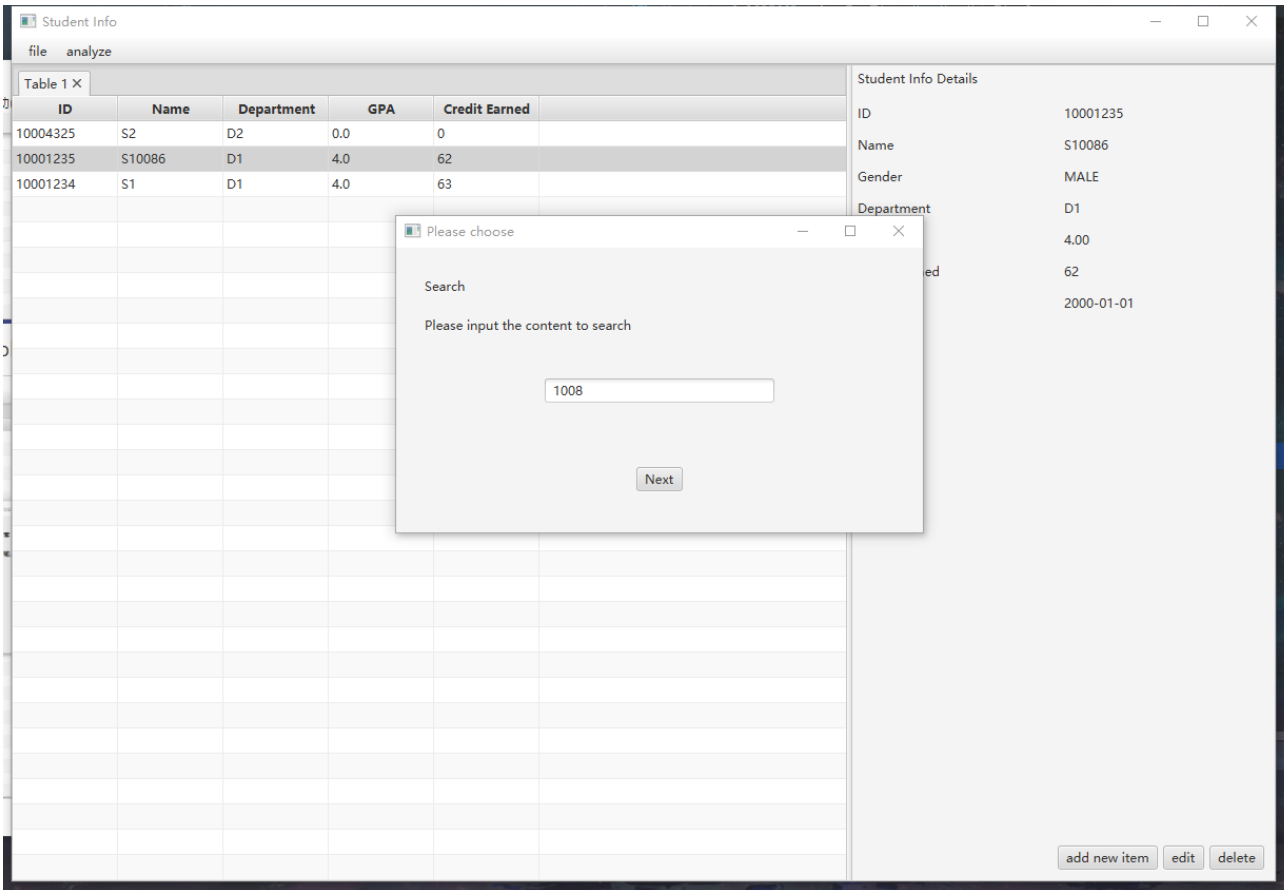
ID	Name	Department	GPA	Credit Earned
10004325	S2	D2	0.0	0
10001235	S10086	D1	4.0	62
10001234	S1	D1	4.0	63

ID	Label
Name	Label
Gender	Label
Department	Label
	Label
ed	Label
	Label

Search

Please input the content to search

    2. Try to find `1008`:



ID	Name	Department	GPA	Credit Earned
10004325	S2	D2	0.0	0
10001235	S10086	D1	4.0	62
10001234	S1	D1	4.0	63

Student Info Details

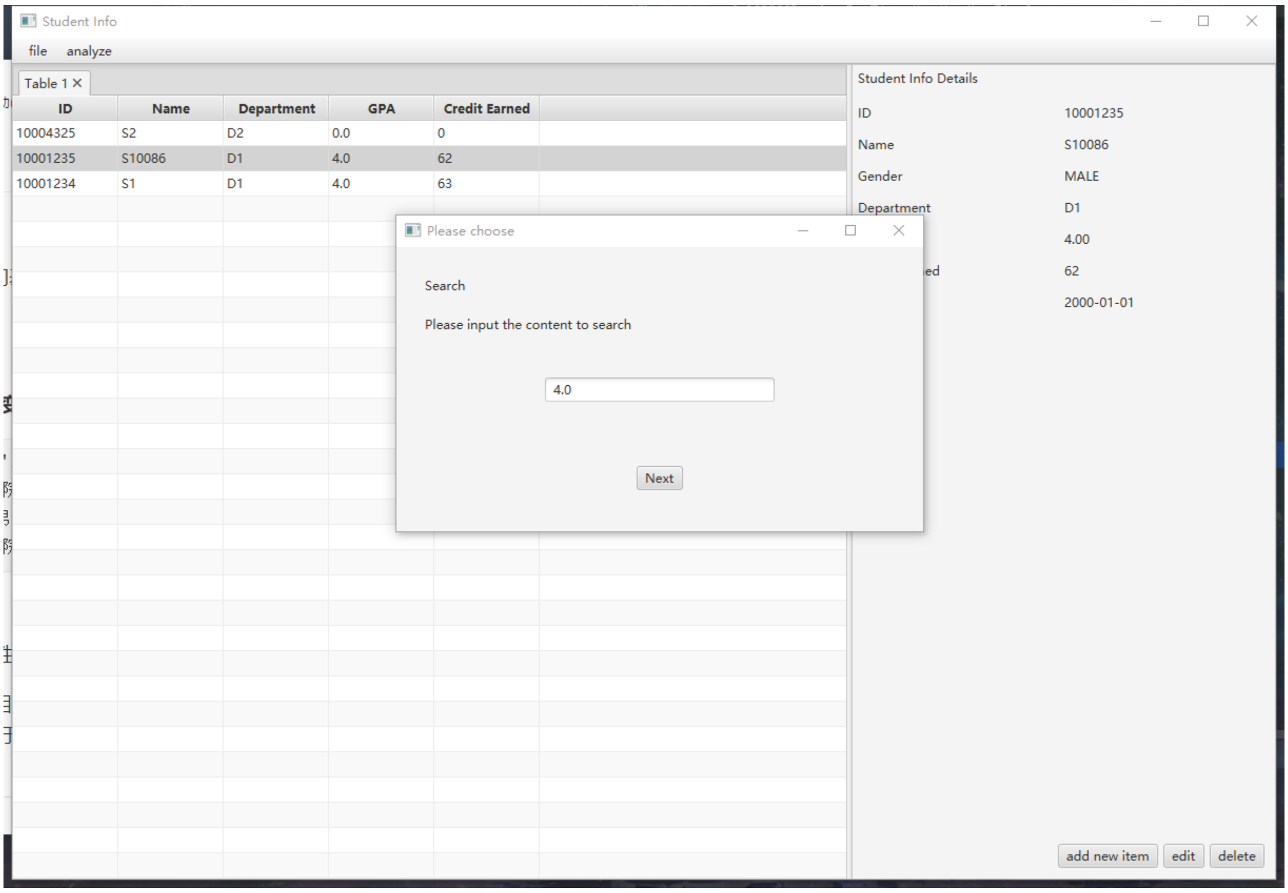
ID	10001235
Name	S10086
Gender	MALE
Department	D1
	4.00
ed	62
	2000-01-01

Search

Please input the content to search

1008

3. Try to find `4.0`:



ID	Name	Department	GPA	Credit Earned
10004325	S2	D2	0.0	0
10001235	S10086	D1	4.0	62
10001234	S1	D1	4.0	63

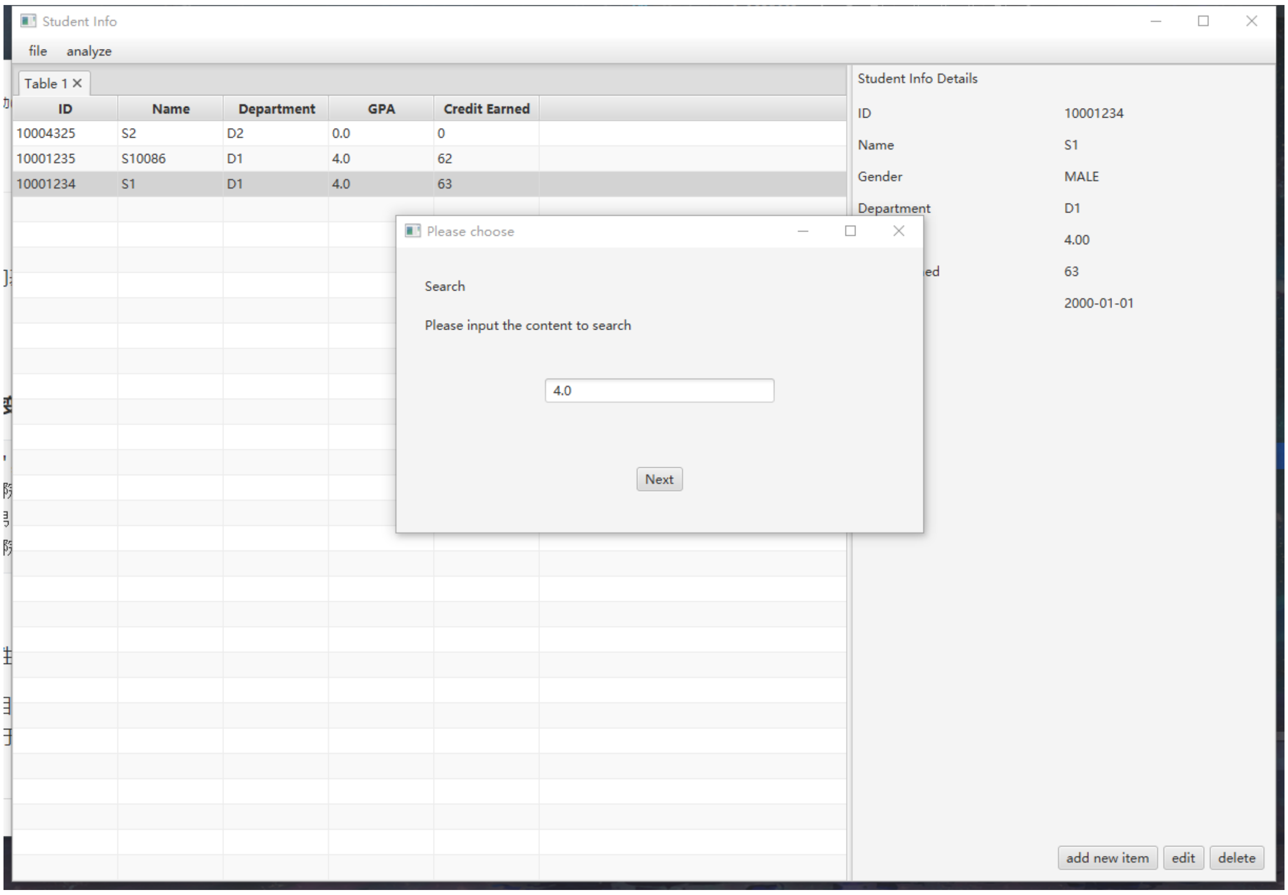
Student Info Details

ID	10001235
Name	S10086
Gender	MALE
Department	D1
	4.00
ed	62
	2000-01-01

Search

Please input the content to search

4.0



ID	Name	Department	GPA	Credit Earned
10004325	S2	D2	0.0	0
10001235	S10086	D1	4.0	62
10001234	S1	D1	4.0	63

Student Info Details

ID	10001234
Name	S1
Gender	MALE
Department	D1
	4.00
ed	63
	2000-01-01

Search

Please input the content to search

4.0